



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

**Simulación Numérica de un Medio Poroso  
Fracturado Mediante el Uso de  
Inteligencia Artificial Entrenada con  
Física**

**TESIS**

Que para obtener el título de  
**Ingeniero Petrolero**

**P R E S E N T A**

Andrés Barranco Tamayo

**DIRECTOR DE TESIS**

Dr. Erick Emanuel Luna Rojero



**Ciudad Universitaria, Cd. Mx., 2019**

*Esta página fue dejada intencionalmente en blanco*

*A mis padres María Dolores Tamayo Flores y Federico Barranco Cicilia, todos mis logros se los debo a ustedes. Infinitas gracias por todo.*

*A Rodrigo y Santiago Federico, van por buen camino.*

*A Itzel Daniela y Mateo Sebastián, al éxito de ustedes.*

*A la memoria de Gloria Cicilia Cortez y Elia Flores Jaimes, mis padres son producto de su esfuerzo y dedicación.*

# Agradecimientos

Agradezco al Instituto Mexicano del Petróleo, especialmente a la Gerencia de Ingeniería de Yacimientos por abrirme las puertas desde la realización del servicio social. Muestro mi gratitud por sus enseñanzas y consejos a mi asesor el Dr. Erick Emanuel Luna Rojero.

A los miembros del jurado: Dr. Fernando Rodríguez de la Garza, Dr. Rodolfo Camacho Velázquez, Dr. Teodoro Iván Guerrero Sarabia y a la Dra. Ana Paulina Gómora Figueroa por sus valiosos comentarios y aportaciones al presente trabajo.

A todos los amigos por la buena compañía y apoyo a lo largo de la licenciatura.

A mi *alma mater*, nuestra UNAM.

# Resumen

La Ciencia de Datos juega un papel determinante en la Revolución Industrial 4.0, por lo que en el sector energético nacional es necesario incursionar a la brevedad en estas nuevas tecnologías. Existen trabajos en donde la Ciencia de Datos se ha utilizado para el modelado de yacimientos de hidrocarburos o para la perforación de pozos a partir del uso de redes neuronales, así como datos históricos estructurados y no estructurados. Sin embargo, existen pocos trabajos de dichas metodologías aplicadas directamente a la simulación numérica de yacimientos.

Actualmente, la simulación numérica de yacimientos no contempla todos los fenómenos fisico-químicos involucrados y consume enormes recursos computacionales y tiempo, por lo que se requieren simuladores numéricos de alta precisión y confiabilidad. Estos requerimientos computacionales se acentúan para el caso de los yacimientos naturalmente fracturados, y debido a su grado de complejidad, se requieren esfuerzos adicionales en investigación y desarrollo tecnológico.

El objetivo de este trabajo de tesis es simular numéricamente un sistema poroso fracturado con una inteligencia artificial (red neuronal) entrenada con física para representar el comportamiento de un bloque de matriz. Lo anterior bajo la hipótesis de la posibilidad de emplear inteligencia artificial entrenada a partir de principios físicos, para optimizar el tiempo de cálculo y recursos computacionales al simular un medio poroso fracturado.

Para los propósitos de esta tesis fue generado un modelo asintótico de transporte de masa entre una matriz porosa rectangular y las fracturas que la rodean. El problema asintótico en la matriz porosa, a cualquier orden, es una ecuación de Laplace con condiciones de Dirichlet lineales. Para la solución de éste se usó una red neuronal, la cual se entrenó con una solución exacta. Posteriormente, se realizaron comparaciones del método de solución con la red neuronal y los métodos de diferencias finitas y de iteración con sobre relajación. Entre los resultados más importantes, destacan 1) los tiempos de cálculo con la red neuronal de hasta  $4 \times 10^6$  veces más rápidos, 2) la cantidad de memoria RAM utilizada es la mitad comparada contra un método iterativo y de hasta  $116 \times 10^3$  veces menor con un método de diferencias finitas, y 3) los errores de entrenamiento y validación para la red neuronal que fueron obtenidos en el proceso de educación son en promedio del orden de  $10^{-3}$  y  $10^{-4}$ , similares a lo que se usa en la práctica.

Con la estrategia propuesta basada en el uso de inteligencia artificial, se demostró que es

posible la integración de modelos físicos en la simulación numérica de yacimientos y a su vez, reducir el tiempo de cálculo y los recursos computacionales asociados. Esto abre una ventana de oportunidades en la simulación numérica de yacimientos.

# Abstract

Data Science has a decisive role in the Industrial Revolution 4.0, consequently, it is necessary that the national energy industry would apply be involved in these new technologies as soon as possible. There are some studies in which Data Science has been used to model hydrocarbon reservoirs or to drilling wells using neural networks, as well as structured and unstructured historical data. However, there are few references to these methodologies which are directly applied to numerical reservoir simulation.

Nowadays, numerical reservoir simulation does not contemplate all the physicochemical phenomena that are involved and it requires enormous computational resources and time, therefore high precision and reliability numerical simulators are required. These computational requirements are emphasized in the case of naturally fractured reservoirs, and due to their degree of complexity, additional efforts in research and technological development are needed.

The objective of this work is to simulate numerically a fractured porous system by coupling the fracture network to trained artificial intelligence (neural network) with physics to represent the behavior of the matrix blocks. The foregoing, under the hypothesis of the use of trained artificial intelligence from physical principles, to optimize the calculation time and the computational resources by simulating a fractured porous medium.

For the purpose of this work, an asymptotic model of mass transport between a rectangular porous matrix and the surrounding fractures was developed. The asymptotic problem in the porous matrix, at any order, is a Laplace equation with linear Dirichlet conditions. A neural network was used for the solution of this problem, which was trained with an exact solution. Subsequently, comparisons were made between the solution method with the neural network and the methods of finite differences and iteration with over-relaxation. The most notable results of this study are; 1) calculation times with the neural network up to  $4 \times 10^6$  times faster, 2) the RAM used is half compared to an iterative method and up to  $116 \times 10^3$  times less with a finite difference method, and 3) the training and validation errors for the neural network that were obtained in the training process are in the order of  $10^{-3}$  and  $10^{-4}$ , which are similar to those used in practice.

Finally, it was demonstrated that it is possible to integrate physical models in the numerical simulation of reservoirs and, at the same time, reduce the calculation time and the associat-

ed computational resources. This opens a wide range of opportunities in numerical reservoir simulation.



# Índice general

<b>1. Conceptos fundamentales de inteligencia artificial</b>	<b>1</b>
1.1. Redes Neuronales . . . . .	1
1.1.1. Arquitectura de una red neuronal . . . . .	2
1.1.2. Funciones de activación . . . . .	6
1.1.3. Proceso de entrenamiento . . . . .	7
1.2. Machine learning . . . . .	8
1.3. Deep learning . . . . .	9
<b>2. Modelado de Yacimientos Naturalmente Fracturados</b>	<b>11</b>
2.1. Modelo de doble porosidad . . . . .	12
<b>3. Definición del problema</b>	<b>17</b>
3.1. Modelo conceptual . . . . .	17
3.2. Objetivo . . . . .	18
3.3. Hipótesis . . . . .	19
3.4. Alcances . . . . .	19
<b>4. Modelado analítico de un bloque de matriz poroso rodeado de fracturas</b>	<b>20</b>
4.1. Solución asintótica . . . . .	25
4.1.1. Solución a orden cero . . . . .	27
4.1.2. Solución a orden uno . . . . .	32

4.2. Solución general de la ecuación de Laplace en dos dimensiones . . . . .	38
<b>5. Definición y entrenamiento de la red neuronal</b>	<b>44</b>
5.1. Entrenamiento de redes neuronales . . . . .	46
<b>6. Comparación entre modelos tradicionales y la red neuronal</b>	<b>51</b>
6.1. Validación de la red neuronal . . . . .	51
<b>7. Resultados de modelos tradicionales y red neuronal</b>	<b>55</b>
<b>8. Conclusiones</b>	<b>61</b>
<b>A. Ecuación de Karode</b>	<b>66</b>

# Índice de figuras

1-1. Diagrama de una neurona (Modificado de Haykin, 2019) . . . . .	3
1-2. Diagrama de una neurona considerando bias como entrada (Modificado de Haykin, 2019) . . . . .	4
1-3. Red neuronal de una capa . . . . .	5
1-4. Red neuronal de tres capas ocultas . . . . .	6
1-5. a) Función de umbral binario . . . . .	6
1-6. b) Función tangente hiperbólica . . . . .	7
1-7. c) Función logística . . . . .	7
2-1. Modelo conceptual de Warren y Root (Modificado de Warren <i>et al.</i> , 2019) . . . . .	13
2-2. Comportamiento de la caída de presión para un yacimiento infinito de doble porosidad (Warren <i>et al.</i> , 1963) . . . . .	16
3-1. Modelo de un sistema poroso fracturado. . . . .	18
4-1. Bloque de matriz poroso rodeado por cuatro fracturas. . . . .	22
4-2. Gráfica de la solución particular a la ecuación de Laplace en dos dimensiones. . . . .	42
5-1. Entradas y salidas de la red neuronal. . . . .	45
5-2. Diagrama de flujo general de algoritmo . . . . .	46

5-3. Arquitectura de red neuronal 4-12-8-4. . . . .	47
5-4. Gráfica de error de entrenamiento y validación de la red neuronal. . . . .	48
5-5. Ajuste de datos para la salida 1 de la red neuronal entrenada. . . . .	49
5-6. Ajuste de datos para la salida 2 de la red neuronal entrenada. . . . .	49
5-7. Ajuste de datos para la salida 3 de la red neuronal entrenada. . . . .	50
5-8. Ajuste de datos para la salida 4 de la red neuronal entrenada. . . . .	50
6-1. Gráfico en 2D de la solución del problema propuesto. . . . .	54
7-1. Diagramas de flujo de procesos a comparar, a)Modelo analítico, b)Modelo de redes neuronales . . . . .	56
7-2. Comparación de tiempo de cómputo con respecto a la red neuronal. . . . .	59
A-1. Balance del gasto en un diferencial de $z$ . (Modificado de Luna Barrera, 2019) . .	67

# Índice de tablas

5.1. Valores de regresión lineal para cada una de las salidas . . . . .	48
6.1. Resultados obtenidos para el cálculo de flujos, comparando red neuronal con modelos tradicionales. . . . .	52
6.2. Resultados obtenidos de las integrales requeridas por la solución asintótica, comparando red neuronal con modelos tradicionales. . . . .	52
6.3. Resultados obtenidos con otras condiciones de frontera para el cálculo de flujos, comparando red neuronal con modelos tradicionales. . . . .	53
6.4. Resultados obtenidos con otras condiciones de frontera de las integrales requeridas por la solución asintótica, comparando red neuronal con modelos tradicionales.	53
7.1. Datos de entrada de programa . . . . .	55
7.2. Datos adimensionales de entrada de programa para un caso simétrico . . . . .	57
7.3. Presiones que resultan de los cálculos en un caso de simetría . . . . .	57
7.4. Gastos totales que resultan de los cálculos en un caso de simetría . . . . .	57
7.5. Datos adimensionales de entrada de programa para un caso general . . . . .	58
7.6. Presiones que resultan de los cálculos de un caso general . . . . .	58
7.7. Gastos totales que resultan de los cálculos de un caso general . . . . .	58
7.8. Tiempo de cómputo para los distintos métodos de solución . . . . .	59
7.9. Memoria RAM usada por los métodos numéricos y la red neuronal. . . . .	60

# Introducción

Las revoluciones industriales ocurridas en los siglos XVIII, XIX y XX, liberaron al hombre de la dependencia de los animales para realizar trabajos, hicieron posible la producción en masa y también aportaron capacidades digitales a millones de personas. En el presente, la Revolución Industrial 4.0 es fundamentalmente diferente; se caracteriza por una serie de tecnologías novedosas que fusionan los mundos físico, digital y biológico, impactando a múltiples disciplinas del conocimiento. La tecnología también está cada vez más conectada a medida que avanza la Revolución 4.0, las innovaciones son más rápidas, más eficientes y más accesibles que antes. Una herramienta fundamental de dicho movimiento es el uso de la Inteligencia Artificial.

La Inteligencia Artificial (IA) tuvo sus inicios formales en 1956 cuando John McCarthy (1955) realizó trabajos en el Proyecto de Investigación de Verano de Dartmouth, cuyo objetivo era analizar las formas en que se podrían fabricar máquinas para simular aspectos de la inteligencia humana, lo cual es la esencia impulsando el desarrollo de la IA. La idea de crear físicamente una máquina para ejecutar secuencias de instrucciones, que habían conceptualizado pioneros como Charles Babbage, había madurado en la década de 1950, y dio lugar a la construcción de las primeras computadoras electrónicas y los primeros robots.

Alan Turing (1950), quien propuso un modelo formal de computación en su ensayo *Computing Machinery and Intelligence*, imaginó computadoras creadas para simular inteligencia y aprender automáticamente. Aunque estas ideas inspiraron a Turing, sin embargo, en la década de los años 40, no tuvo acceso a los recursos computacionales que hoy se tienen. En cuanto a las redes neuronales, el primer trabajo fue el *Perceptron* de Rosenblatt (1957), el cual es un modelo computacional basado en neuronas biológicas que se convirtió en la base del campo de las redes neuronales artificiales.

Un mayor uso de la IA está teniendo un impacto significativo en la sociedad, cambiando la forma en que se trabaja, vive y se interactúa. Las IAs ya están cambiando la vida diaria, en formas que mejoran la salud, la seguridad y la productividad. Las principales universidades alrededor del mundo dedican esfuerzos a los estudios de IA. Las grandes empresas de tecnología como Apple, Facebook, Google, IBM y Microsoft realizan investigaciones para explorar las aplicaciones de IA que se consideran fundamentales para el procesamiento de información y toma de decisiones.

La industria de los hidrocarburos es una de las más rezagadas en la implementación de tecnologías de inteligencia artificial, teniendo un amplio campo de aplicación en toda la cadena de valor, desde aguas arriba (*upstream*) hasta aguas abajo (*downstream*). Algunas de las aplicaciones futuras de la IA en la industria energética son la interpretación automática de los registros de pozos, simulación numérica de yacimientos, automatización total de los equipos de perforación, identificación del mantenimiento de las instalaciones en tierra o costa afuera utilizando drones y visión computarizada; además de la administración de yacimientos que consideren al sistema integral de producción, monitoreo, análisis de datos y toma de decisiones en tiempo real; cabe mencionar que los simuladores comerciales están incursionando en la implementación de IA, por ejemplo Computer Modelling Group añadió una nueva herramienta llamada CMOST AI, en la cual se combina el uso de machine learning con análisis estadístico avanzado así como la interpretación de datos.

La mayor producción de hidrocarburos en México proviene de los yacimientos naturalmente fracturados, por lo que es importante su modelado numérico. Asimismo, la simulación numérica de yacimientos requiere de grandes cantidades de recursos computacionales por lo que es necesario su optimización, por otro lado la simulación numérica de yacimientos no contempla todos los fenómenos físico-químicos involucrados. Por este motivo, se empleó inteligencia artificial entrenada a partir de principios de la teoría del transporte para el modelado de un bloque de matriz rodeado de fracturas.

En el capítulo 1 se presentan conceptos fundamentales sobre inteligencia artificial, en el cual se hace énfasis en las redes neuronales debido a la importancia de éstas en este trabajo. La teoría sobre el modelo de doble porosidad-doble permeabilidad desarrollado por Warren y Root (1963) es presentado en el capítulo 2. Por otro lado, en el capítulo 3 se habla sobre el modelo conceptual así como el objetivo, hipótesis y alcances de este trabajo. En el capítulo 4 es explicado ampliamente el modelado asintótico de un bloque de matriz poroso rodeado de fracturas, en donde son obtenidas las ecuaciones esenciales para el entrenamiento de la red neuronal, consecuentemente en el capítulo 5 se define la red neuronal utilizada así como el entrenamiento de dicho modelo de inteligencia artificial. Se realizaron comparaciones entre la red neuronal, una discretización de las ecuaciones en diferencia finitas utilizando un método de descomposición LU. Por otro lado, el modelo de IA también fue comparado con un modelo

resuelto por Gauss-Seidel empleando sobre-relajación sucesiva, los resultados se encuentran en el capítulo 6. Finalmente, las conclusiones y recomendaciones obtenidas del presente trabajo son expuestas en el capítulo 8. En las **Figuras 0-1** y **0-2** se muestra un diagrama general sobre la metodología seguida en el presente trabajo.

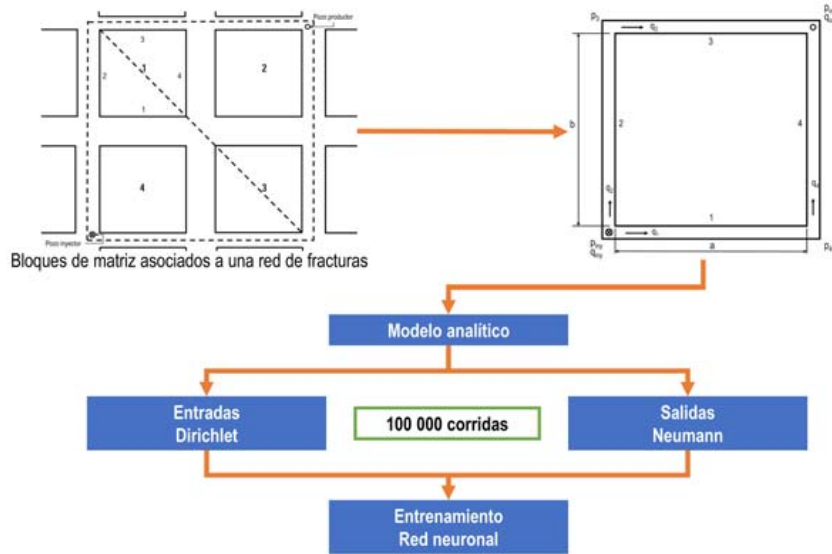


Figura 0-1: Diagrama general sobre el desarrollo del trabajo.

#### Comparación entre diferentes modelos

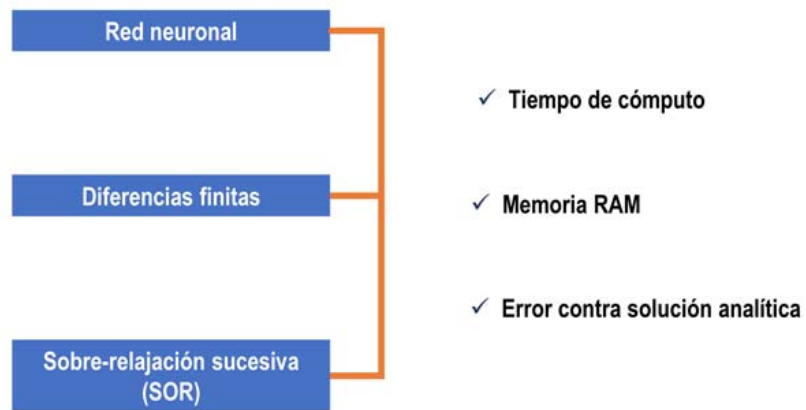


Figura 0-2: Diagrama general sobre los modelos numéricos comparados.



# Conceptos fundamentales de inteligencia artificial

## 1.1. Redes Neuronales

El estudio de las redes neuronales artificiales, comunmente llamadas redes neuronales, fue inicialmente motivado por la capacidad del cerebro humano para procesar información de una manera completamente distinta a la de una computadora. El cerebro es un sistema de procesamiento de información altamente complejo y no lineal, a través de un procesamiento paralelo, tiene la capacidad de conectar sus principales componentes (las neuronas), para realizar actividades de reconocimiento de patrones, percepción y control motriz, algunas veces más rápido que las tecnologías de inteligencia artificial actuales. El cerebro humano tiene la habilidad de poderse adaptar al ambiente que lo rodea, esta habilidad es llamada plasticidad.

En la manera más general, una red neuronal es una máquina diseñada para emular el comportamiento del cerebro para desarrollar una tarea en específico. Usualmente la red es implementada mediante componentes electrónicos o es programada en una computadora. Para tener un buen rendimiento, las redes neuronales emplean una interconexión masiva, la cual se asemeja al cerebro en dos aspectos:

1. El conocimiento de la red es adquirido por un proceso de entrenamiento.
2. Las interconexiones, que son conocidas como pesos sinápticos, son las responsables de

almacenar la información aprendida.

El procedimiento utilizado para realizar el aprendizaje se denomina algoritmo de entrenamiento, cuya función es modificar los pesos sinápticos de la red de forma ordenada para alcanzar un objetivo de diseño deseado. La modificación de los pesos sinápticos proporciona el método tradicional para el diseño de redes neuronales, cuyo enfoque es el más cercano a la teoría del filtro adaptativo lineal, que ya está bien establecido y se aplica con éxito en diversos campos (Haykin, 2009). Sin embargo, también es posible que una red neuronal modifique su propia topología, lo cual está motivado por el hecho de que las neuronas en el cerebro humano mueren y crean nuevas conexiones sinápticas.

Una red neuronal deriva su poder de cálculo a través de su estructura distribuida de forma paralela y de su capacidad para aprender y para generalizar. La generalización se refiere a la generación de salidas razonables de la red neuronal para entradas de información que no se encuentran durante el entrenamiento. Estas dos capacidades de procesamiento de información hacen posible que las redes neuronales encuentren soluciones aproximadas dentro de rangos aceptables a problemas complejos.

No obstante, las redes neuronales no pueden proporcionar una solución trabajando de una manera individual, más bien, un problema complejo de interés se descompone en un número de tareas relativamente simples, y a cada una de las redes se les asigna un subconjunto de las tareas que corresponden a sus capacidades inherentes. Es importante reconocer que se tiene un largo camino por recorrer antes de que se simule una arquitectura informática que imite al cerebro humano.

### **1.1.1. Arquitectura de una red neuronal**

Una neurona es una unidad de procesamiento de información que es fundamental para el funcionamiento de una red neuronal (**Figura 1-1**). A continuación se describen los tres elementos básicos del modelo neuronal:

- a) Enlaces de conexión, cada uno de los cuales se caracteriza por su propio peso o fuerza, específicamente, una señal  $x_j$  a la entrada de la sinapsis  $j$  conectada a la neurona  $k$  se multiplica por el peso  $w_{kj}$ . El subíndice  $k$  se refiere a la neurona en cuestión, y el subíndice

$j$  se refiere al extremo de entrada de la sinapsis a la que se refiere el peso. A diferencia del peso de una sinapsis en el cerebro, el peso sináptico de una neurona artificial puede estar en un rango que incluye tanto valores negativos como positivos.

- b) Un sumador para agregar las señales de entrada ponderadas por los respectivos pesos de la neurona, el cual constituye un combinador lineal.
- c) Una función de activación para limitar la amplitud de la salida de una neurona a un valor finito.

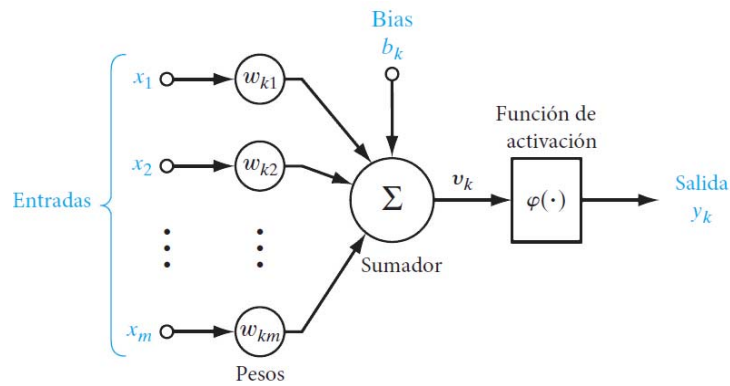


Figura 1-1: Diagrama de una neurona (Modificado de Haykin, 2019)

El modelo de la **Figura 1-1** también incluye un sesgo aplicado externamente. En el lenguaje de Ciencia de Datos es mejor conocido por su nombre en inglés (*bias*) que es denotado por  $b_k$ , el cual tiene el efecto de aumentar o disminuir la entrada neta de la función de activación, dependiendo de si ésta es positiva o negativa, respectivamente.

En términos matemáticos la neurona  $k$ , representada en la **Figura 1-1**, se describe a través de las siguientes ecuaciones:

$$n_k = \sum_{j=1}^m w_{kj}x_j \quad (1.1)$$

y

$$y_k = \varphi(n_k - b_k) \quad (1.2)$$

donde,

- $n_k \triangleq$  Salida del combinador lineal debido a las señales de entrada
- $x_j \triangleq$  Entradas de la neurona
- $w_{kj} \triangleq$  Pesos
- $b_k \triangleq$  Bias
- $\varphi \triangleq$  Función de activación
- $y_k \triangleq$  Salidas de la neurona

Un modelo equivalente al mostrado en la **Figura 1-1** es el ilustrado en la **Figura 1-2**, que es el resultado de adicionar:

- a) Una nueva señal de entrada fijada en  $+1$ .
- b) Un nuevo peso igual al bias  $b_k$ .

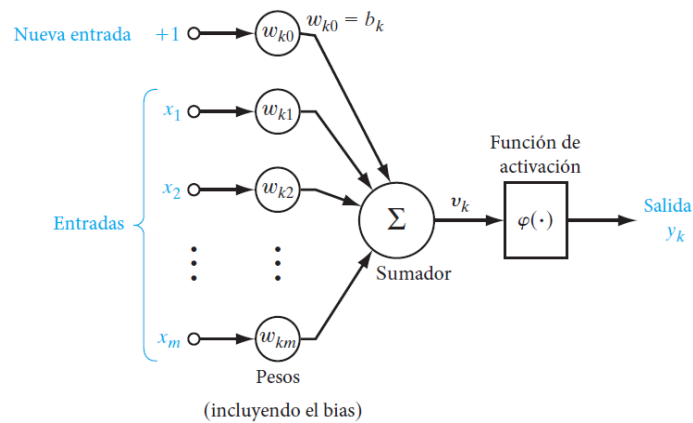


Figura 1-2: Diagrama de una neurona considerando bias como entrada (Modificado de Haykin, 2019)

La manera en que se estructuran las neuronas de una red depende principalmente del algoritmo de aprendizaje utilizado para entrenarla. A continuación se describen dos categorías de arquitectura de redes neuronales que son generalmente usadas:

## 1. Redes de una sola capa

En la forma más simple de una red neuronal, se tiene una capa de entrada de nodos fuente que se proyecta directamente sobre una capa de salida de neuronas como es mostrado en la **Figura 1-3** . Se hace referencia a una sola capa a las neuronas de salida debido a que en la capa de entrada no se realizan cálculos debido a que meramente se ingresan datos.

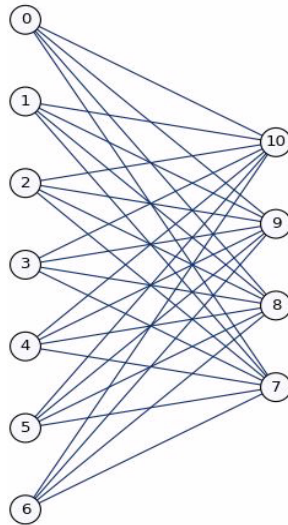


Figura 1-3: Red neuronal de una capa

## 2. Redes multicapa

Las redes neuronales multicapa se distinguen por tener diversas capas ocultas (**Figura 1-4**) con la función de intervenir entre las capas de entrada y de salida. Al contar con capas ocultas en la arquitectura de la red neuronal es posible obtener mejores parámetros de los datos de entrada, por lo que la red ahora contiene una mejor perspectiva gracias al mayor número de conexiones establecidas. La interacción entre las neuronas de las capas ocultas entonces se entendería como un aprendizaje profundo. En el presente trabajo fueron entrenadas redes neuronales multicapa.

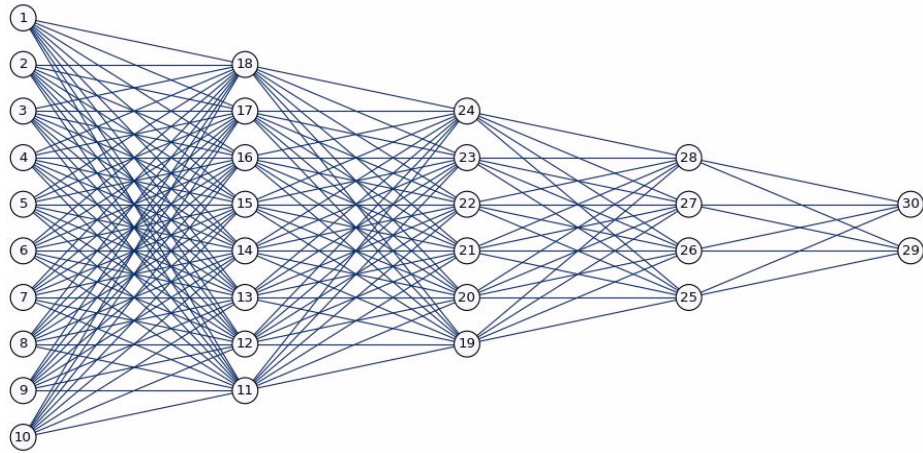


Figura 1-4: Red neuronal de tres capas ocultas

### 1.1.2. Funciones de activación

La función de activación más sencilla es la a) **función de umbral binario** que sólo puede asumir dos valores, si la entrada está por encima de un determinado límite, la función cambia de un valor a otro y lo demás permanece constante (**Figura 1-5**), por lo tanto implica que la función no es derivable en el límite y para el resto la derivada es 0. De otra manera la b) **función tangencial** restringe los datos de  $[-1,1]$  (**Figura 1-6**) y finalmente, otra función que es muy utilizada es la c) **función logística** que opera de  $[0,1]$  (**Figura 1-7**).

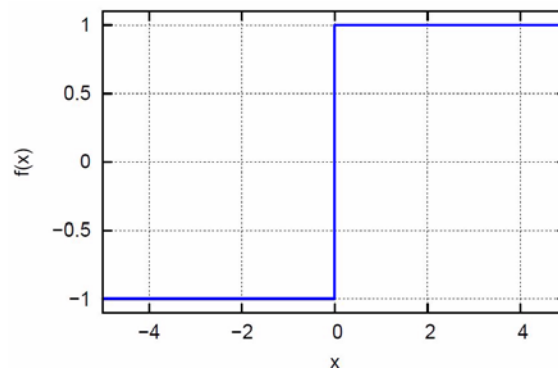


Figura 1-5: a) Función de umbral binario

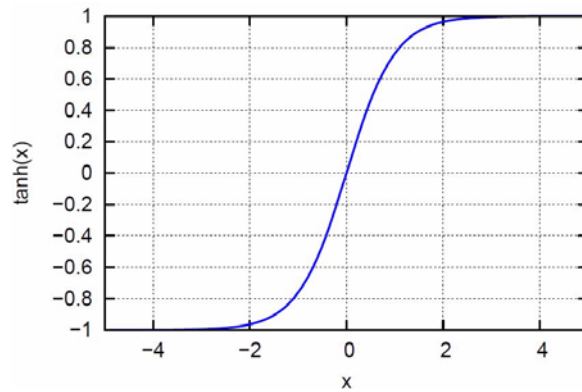


Figura 1-6: b) Función tangente hiperbólica

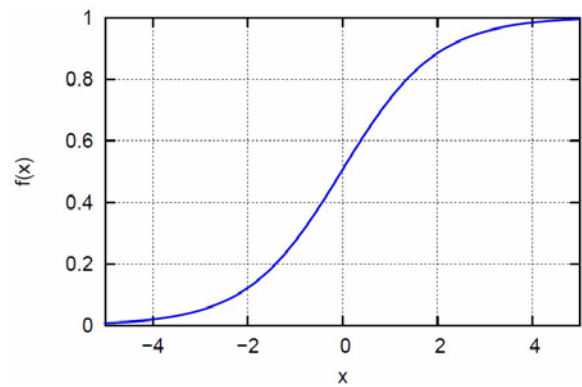


Figura 1-7: c) Función logística

### 1.1.3. Proceso de entrenamiento

Existen distintos métodos para entrenar una red neuronal; en este trabajo el entrenamiento fue de tipo supervisado donde el conjunto de entrenamiento consiste en datos de entrada, así como sus resultados correctos para lograr una activación precisa de todas las neuronas de salida. Posteriormente se compara directamente el rendimiento de cada uno de los datos que se introducen en la red con la solución correcta y se pueden modificar los pesos de la red neuronal en función de la diferencia obtenida; el objetivo es modificar las ponderaciones para que la red no sólo pueda asociar los patrones de entrada y salida de forma independiente después de la formación, sino que pueda proporcionar resultados acertados a patrones de entrada similares y

desconocidos.

Este procedimiento de aprendizaje no siempre es biológicamente posible, pero es muy efectivo y por lo tanto muy utilizado. Los procesos básicos de entrenamiento de una red neuronal se enlistan a continuación:

1. Ingresar los datos de entrada y salida.
2. Propagación directa de la entrada por la red neuronal, generación de la salida.
3. Comparación de la salida generada (paso 2) con los datos de salida ingresados (paso 1), se calcula el error.
4. Se realizan correcciones considerando el error y los datos de salida generados.

## **1.2. Machine learning**

El aprendizaje automático o machine learning es una forma de IA que permite a un sistema aprender de los datos en lugar de a través de una programación explícita. Ya que el aprendizaje automático le permite al sistema aprender continuamente de los datos y adquiere su capacidad para predecir el futuro de acuerdo a estos. Este potente conjunto de algoritmos y modelos se está utilizando en todos los sectores para mejorar los procesos y obtener información sobre patrones y anomalías en los datos. Machine learning es un proceso en equipo que requiere la colaboración de científicos de datos, ingenieros de datos, analistas de negocios y líderes de negocios.

El machine learning utiliza una variedad de algoritmos que aprenden iterativamente de los datos para mejorar, describir los datos y predecir los resultados. A medida que los algoritmos ingieren datos de entrenamiento, es posible producir modelos más precisos basados en estos datos, por lo que un modelo de aprendizaje automático es el resultado del entrenamiento del algoritmo de machine learning con datos. Por lo tanto después del entrenamiento, cuando se proporciona un modelo con una entrada, se generará un resultado (Nilsson, 1998). Por ejemplo, un algoritmo creará un modelo predictivo, luego, cuando se proporcionen datos a dicho modelo, se recibirá una predicción basada en los datos que entrenaron al modelo. El machine learning es ahora una herramienta para emplearse en la administración integral de yacimientos mediante el desarrollo de plataformas basadas en metodologías de ciencia de datos .



A lo largo de las décadas, las técnicas de IA se han utilizado ampliamente como un método para mejorar el rendimiento, pero, en los últimos años, con las mejoras en modelos de computación y almacenamiento más baratos, ha habido un aumento del interés en la IA y en el machine learning que ha llevado a que se inviertan grandes cantidades de dinero en empresas generadoras de software. A continuación, se enlistan algunos avances tecnológicos que han generado valor para que la IA tenga un mayor impacto:

- Los procesadores modernos se han vuelto cada vez más potentes y densos. La relación densidad/rendimiento ha mejorado drásticamente.
- El costo de almacenar y administrar grandes cantidades de datos se ha reducido drásticamente. Además, las nuevas innovaciones de almacenamiento han llevado a un rendimiento mayor, así como la capacidad de analizar conjuntos de datos mucho más grandes.
- La capacidad de distribuir el procesamiento de cómputo a través de clústeres de equipos ha mejorado considerablemente la capacidad de analizar datos complejos en un tiempo récord.
- Hay más conjuntos de datos comerciales disponibles para apoyar el análisis, muchos de ellos están disponibles como servicios en la nube y como interfaces de programación de aplicaciones (API) por sus siglas en inglés, bien definidas.
- Los algoritmos de machine learning se han puesto a disposición a través de comunidades open source con grandes bases de usuarios, por lo tanto, hay más recursos, marcos y bibliotecas que han facilitado el desarrollo.
- La visualización se ha vuelto más sencilla, no es necesario ser un científico de datos para interpretar los resultados, por lo que el machine learning es ampliamente utilizado en muchas industrias.

### **1.3. Deep learning**

El aprendizaje profundo o deep learning es un método específico de machine learning que incorpora redes neuronales en capas sucesivas para aprender de los datos de forma iterativa. El

deep learning es especialmente útil cuando se trata de aprender patrones a partir de datos no estructurados. Además, permite construir conceptos complejos a partir de conceptos más sencillos; la idea de comprender la representación correcta de los datos proporciona una perspectiva sobre el deep learning. Cada capa de la representación puede considerarse como el estado de la memoria de la computadora después de ejecutar otro conjunto de instrucciones en paralelo. Las redes con mayor profundidad pueden ejecutar más instrucciones en secuencia. Las instrucciones secuenciales ofrecen una gran potencia porque las instrucciones posteriores pueden referirse a los resultados de instrucciones anteriores. Según esta visión de aprendizaje profundo, no toda la información en las activaciones de una capa codifica necesariamente factores de variación que explican la entrada. La representación también almacena información de estado que ayuda a ejecutar un programa que puede dar sentido a la entrada, esta información de estado podría ser análoga a la de un contador o puntero de un programa tradicional.

Hay dos formas principales de medir la profundidad de un modelo, se basa en el número de instrucciones secuenciales que deben ejecutarse para evaluar la arquitectura, se asimila como la longitud del camino más largo a través de un diagrama de flujo que describen cómo calcular cada uno de los resultados del modelo dadas sus datos de entrada. Así como dos códigos equivalentes tendrán longitudes diferentes dependiendo del lenguaje en el que se escriba el programa, la misma función puede ser esquematizada como un diagrama de flujo con diferentes alternativas.

En resumen, deep learning, específicamente, es un tipo de machine learning, esta técnica permite a los sistemas computacionales mejorar con experiencia y datos. El machine learning es el único enfoque viable para construir sistemas de IA que pueden funcionar en entornos complicados del mundo real. El aprendizaje profundo es un tipo particular de aprendizaje de máquina que logra gran poder y flexibilidad, con cada concepto definido en relación con conceptos más simples, y representaciones más abstractas calculadas en términos de otros menos abstractos (Sutskever *et al.*, 2013).

# Modelado de Yacimientos Naturalmente Fracturados

Los yacimientos naturalmente fracturados (YNF) se diferencian de los yacimientos homogéneos desde distintos puntos de vista: geológicos, petrofísicos, de producción y económicos. La principal característica es que están compuestos por dos medios principales: el de la matriz y las fracturas. Los YNF están compuestos de distribuciones aleatorias de fracturas, bloques de matriz y frecuentemente con vóculos o cavidades de disolución. Dada esta complejidad geométrica no siempre es fácil reproducir el comportamiento de estos yacimientos, específicamente para predecir su producción. Una fractura es un plano discontinuo resultante de los esfuerzos que exceden el punto de ruptura de la roca o bien una discontinuidad macroscópica de origen natural debida a la deformación o diagénesis; en conjunto la red de fracturas conlleva efectos positivos o negativos en la recuperación de aceite. Estadísticamente el 80 % de la producción total de hidrocarburos del país proviene de los yacimientos naturalmente fracturados. (Centro Nacional de Información de Hidrocarburos, 2019), debido a procesos diagénéticos y los esfuerzos regionales estos yacimientos son altamente heterogéneos y anisotrópicos. Dado que el patrón de fracturas naturales es un reflejo del estado local de esfuerzos cuando se inició el fracturamiento, las fracturas naturales muestran orientaciones preferentes en una región específica, lo que añade anisotropía a la permeabilidad del yacimiento. Además, un gran número de fracturas naturales altamente conectadas pueden mejorar la permeabilidad efectiva del yacimiento (Oda, 1985).

Algunos métodos geológicos, como la inversión sísmica (Far *et al.*, 2013), son desarrollados para caracterizar con precisión las orientaciones y densidad de las fracturas naturales.

En un yacimiento fracturado, los fenómenos físicos pueden ser completamente diferentes si se comparan con los de un yacimiento homogéneo; las fracturas proporcionan canales complejos para el movimiento de los fluidos e introducen una permeabilidad máxima y mínima, lo que puede tener un impacto positivo o negativo en la recuperación total (Nelson, 2001). Gracias al sistema fracturado se presentan varios problemas como lo es la canalización de agua temprana, dejando un gran volumen de aceite sin extraer. Por lo tanto, es difícil predecir con precisión la recuperación de estos yacimientos. Además, en este tipo de yacimientos también aumenta la dificultad de realizar esquemas de producción adecuados. La eficiencia de un esquema de producción puede depender en gran medida de la conectividad de la red de fractura, que no es fácil de medir directamente (Lee *et al.*, 1993).

Algunos métodos de recuperación mejorada de aceite (EOR, por sus siglas en inglés) también pueden enfrentar problemas en yacimientos fracturados, ya que es muy probable que los fluidos inyectados se desplacen a través de las fracturas y no tengan efecto alguno en la matriz.

Dado que la presencia de fracturas juega un papel tan importante en la producción, es importante modelar con precisión la influencia de las fracturas en la simulación de yacimientos. El modelo de doble porosidad para el estudio de los yacimientos naturalmente fracturados es el más aceptado ya que representa los YNF con una buena aproximación. Por otro lado, existe el modelo de triple porosidad, donde desafortunadamente no hay suficiente información de los tres sistemas porosos, además representa una mayor complejidad matemática, lo que hace más viable el modelo de doble porosidad en la práctica.

## **2.1. Modelo de doble porosidad**

En un modelo de doble porosidad se parte de la idea de que un yacimiento heterogéneo puede ser representado como dos medios homogéneos conectados. Warren y Root (1963) tomando como base la conceptualización de flujo de fluidos en medios fracturados desarrollado por Barenblantt y Zheltov (1960), propusieron un modelo idealizado de una red de fracturas que a su vez están altamente interconectadas.

La teoría de Warren y Root está fundamentada en que un yacimiento fracturado es descrito por dos tipos de porosidades:

- a) la porosidad primaria de la matriz, y
- b) la secundaria que corresponde a la fractura.

Los dos sistemas son continuos y están unidos por funciones de transferencia que representan el intercambio de flujo entre los dos medios. Las funciones de transferencia depende de las dimensiones y de la forma del bloque de matriz, permeabilidad relativa, transmisibilidad del bloque, presión capilar y diferencia de densidades entre las fases (Mattax *et al.*, 1990).

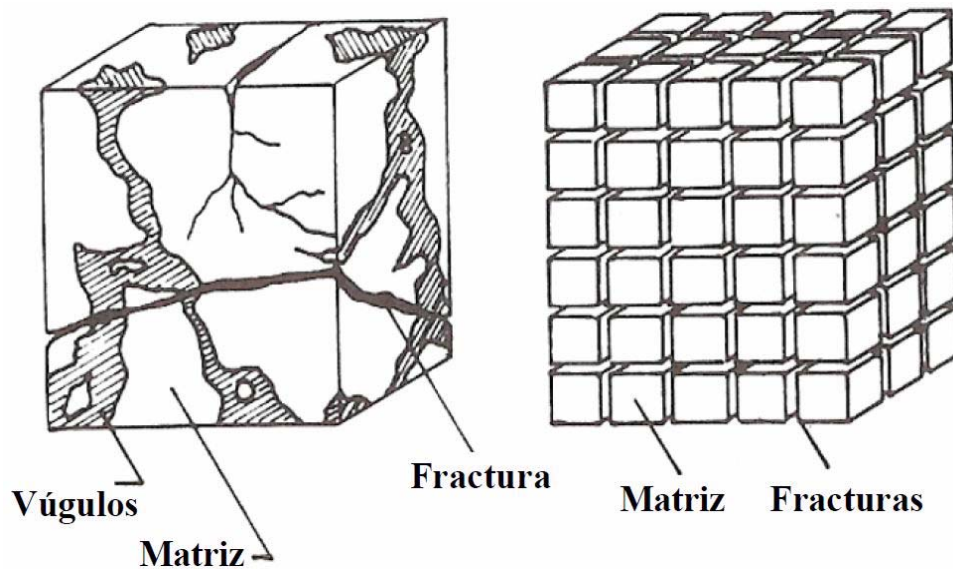


Figura 2-1: Modelo conceptual de Warren y Root (Modificado de Warren *et al.*, 2019)

Warren y Root (1963) indican que si la escala de heterogeneidad es pequeña comparada con la escala del yacimiento, éste es prácticamente considerado como homogéneo y se deriva la solución analítica para un yacimiento infinito con las siguientes consideraciones:

1. Porosidad de la matriz es homogénea, isotrópica y además contenida en un arreglo de paralelepípedos rectangulares idénticos (**Figura 2-1**).
2. El sistema de fracturas consiste en un arreglo ortogonal uniforme de fracturas, donde se encuentra alineado con los principales tensores de permeabilidad.

3. El flujo ocurre entre la matriz y las fracturas pero el flujo entre dos bloques de matriz no puede ocurrir.
4. En los bloques de matriz siempre se presenta el estado estacionario.
5. Sólo es considerado flujo monofásico y ligeramente compresible.

Warren y Root definieron los siguientes parámetros adimensionales:

$$p_D = \left( \frac{2\pi k_f h}{q\mu} \right) (p_0 - p(r, t)) \quad (2.1)$$

$$t_D = \frac{k_f t}{(\phi_f c_f + \phi_m c_m) \mu r_w^2} \quad (2.2)$$

$$r_D = \frac{r}{r_w} \quad (2.3)$$

donde,

$k \triangleq$  Permeabilidad

$h \triangleq$  Espesor de yacimiento

$q \triangleq$  Gasto del pozo

$\mu \triangleq$  Viscosidad

$p_0 \triangleq$  Presión inicial

$t \triangleq$  Tiempo

$\phi_f \triangleq$  Porosidad de fractura

$\phi_m \triangleq$  Porosidad de matriz

$c_m \triangleq$  Compresibilidad de matriz

$c_f \triangleq$  Compresibilidad de fractura

$r_w \triangleq$  Radio del pozo

Warren y Root obtuvieron las siguientes ecuaciones para un yacimiento infinito a partir de las consideraciones antes mencionadas y los parámetros adimensionales obtenidos:

$$P_D(1, t_D) = \frac{1}{2} \left[ 0.80908 + \ln t_D + \text{Ei} \left( -\frac{\lambda t_D}{\omega(1-\omega)} \right) - \text{Ei} \left( -\frac{\lambda t_D}{(1-\omega)} \right) \right] \quad (2.4)$$

donde Ei es la integral exponencial y está definida como:

$$\text{Ei}(-x) = - \int_x^{\infty} \frac{e^{-u}}{u} du \quad (2.5)$$

La ecuación 2.4 derivada por Warren *et al.*, representa que la caída de presión es completamente descrita por dos parámetros  $\omega$  y  $\lambda$ .  $\omega$  es el almacenamiento del fluido dada la porosidad secundaria y está definido por:

$$\omega = \frac{\phi_f c_f}{\phi_f c_f + \phi_m c_m} \quad (2.6)$$

El parámetro  $\lambda$ , está referido a la capacidad de flujo de los bloques de matriz y la relación entre la permeabilidad de la matriz y de la fractura:

$$\lambda = \alpha \frac{k_m r_w^2}{k_f} \quad (2.7)$$

donde,  $\alpha$  es el factor de forma:

$$\alpha = \frac{4m(m+2)}{l^2} \quad (2.8)$$

$m$  es el número de fracturas ortogonales y la variable  $l$  definida como:

$$\begin{aligned} l &= x, m = 1 \\ l &= \frac{2xy}{(x+y)}, m = 2 \\ l &= \frac{3xy}{(xy + yz + zx)}, m = 3 \end{aligned} \quad (2.9)$$

donde,  $x, y, z$  son las dimensiones de los bloques de matriz.

El modelo de doble porosidad predice pruebas de caída de presión que muestran dos rectas

paralelas en un gráfico adimensional de presión-tiempo (**Figura 2-2**), las líneas paralelas están altamente relacionadas la capacidad relativa de almacenamiento ( $\omega$ ). La caída de presión adimensional en la curva de transición está relacionado con la capacidad de flujo y el contraste entre las propiedades de la fractura y de la matriz( $\lambda$ ). Cuando  $\omega \rightarrow 1$  y  $\lambda \rightarrow \infty$  la solución se reduce a un yacimiento homogéneo; cuando se tiene que  $\omega \rightarrow 1$  indica que el almacenamiento solamente ocurre en el sistema de fracturas y cuando  $\lambda \rightarrow \infty$  indica que no se presenta impedancia para flujo interpososo. Para un yacimiento finito radial fracturado, Kazemi (1969) observó una respuesta similar a la **Figura 2-2**.

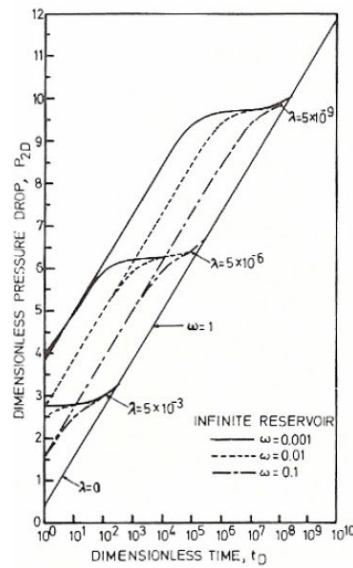


Figura 2-2: Comportamiento de la caída de presión para un yacimiento infinito de doble porosidad (Warren *et al.*, 1963)

Una alternativa natural del enfoque de doble porosidad es permitir el flujo entre bloques de matriz, llamado el modelo de doble permeabilidad. Para resolver el flujo de fluidos a través del bloque de matriz así como el sistema de fractura hay que resolver las ecuaciones de Barenblatt-Zhel'tov (1960). Chen (1990) presenta un resumen de algunas de las soluciones analíticas existentes. La ventaja de este enfoque de doble permeabilidad es que permite el flujo entre bloques de matriz, sin embargo, el planteamiento de doble permeabilidad es computacionalmente más costoso que el modelo de doble porosidad.



## Definición del problema

### 3.1. Modelo conceptual

La mayor producción de hidrocarburos en México proviene de los yacimientos naturalmente fracturados, por lo que es importante su modelado numérico. La simulación de estos yacimientos no contempla todos los fenómenos fisicoquímicos involucrados y consume enormes recursos computacionales, lo que reduce su certidumbre y dificulta su implementación.

Tradicionalmente, la producción de yacimientos naturalmente fracturados se simula con modelos de doble porosidad como es el de Warren y Root, modelo que fue ampliamente definido en el capítulo anterior.

En este trabajo el problema planteado consiste en resolver un bloque de matriz asociado a un sistema fracturado, en el cual se tienen las siguientes consideraciones:

1. La matriz es un medio homogéneo e isotrópico
2. Fluido newtoniano incompresible
3. Proceso isotérmico
4. Efectos gravitacionales despreciables

En la **Figura 3-1** se muestra una representación gráfica general de un modelo en 2D de un medio poroso fracturado, en donde se han colocado dos pozos, un inyector y un productor.

Se enumeran los bloques de matriz porosa así como sus respectivos lados, además las líneas interrumpidas son las simetrías del modelo. Para obtener la solución estacionaria al problema se deben resolver las ecuaciones: de conservación de masa, de Navier-Stokes en las fracturas y de Darcy en la matriz porosa. Sin perder generalidad, se utilizó un modelo de un solo bloque poroso rectangular rodeado por fracturas debido a que el presente trabajo de tesis se enfocó a una prueba de principio, sin embargo existen esfuerzos de modelado de redes de fractura (Villareal Sánchez, 2017 y Luna Barrera, 2018) más generales al propuesto en la **Figura 3-1** y en donde es relativamente simple agregar esta metodología.

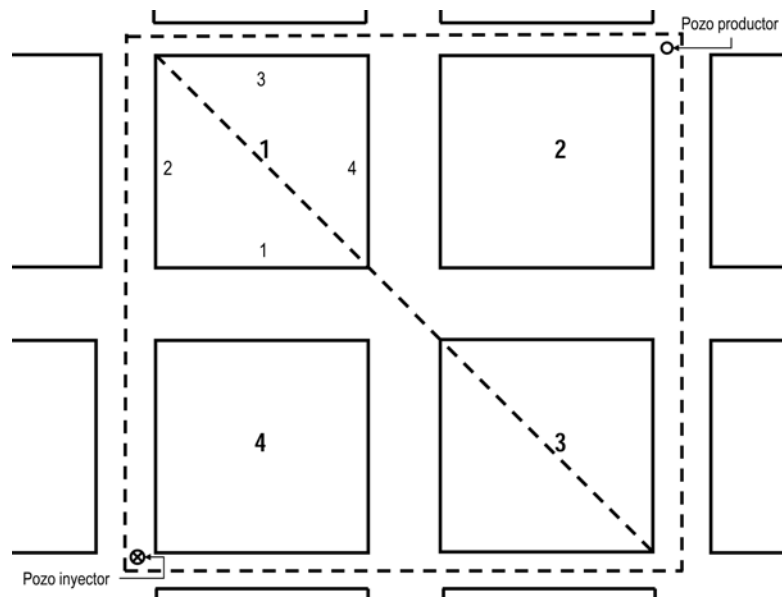


Figura 3-1: Modelo de un sistema poroso fracturado.

### 3.2. Objetivo

Simular numéricamente un sistema poroso fracturado con una inteligencia artificial entrenada con física para representar el comportamiento de un bloque de matriz.

### **3.3. Hipótesis**

Es posible emplear inteligencia artificial entrenada a partir de principios físicos, para optimizar el tiempo de cálculo y recursos computacionales al simular un medio poroso fracturado.

### **3.4. Alcances**

- Integración de tecnologías novedosas para resolver problemas de modelado de yacimientos.
- Construcción de un modelo numérico para un medio poroso fracturado acoplando redes neuronales.
- Reducción de tiempo de cálculo empleado en la simulación de yacimientos naturalmente fracturados.

# Capítulo 4

## Modelado analítico de un bloque de matriz poroso rodeado de fracturas

Considere un bloque de matriz en dos dimensiones de largo  $a$  y alto  $b$ , de porosidad  $\phi$  y permeabilidad  $k$ , saturado de un fluido newtoniano de viscosidad  $\mu$  y densidad constante  $\rho$ .

(Figura 4-1)

Dentro del bloque se satisface la Ley de Darcy:

$$\vec{u} = -\frac{k}{\mu}\nabla p \quad (4.1)$$

donde  $\vec{u}$  es la velocidad de Darcy y  $p$  es la presión. Esta ecuación debe resolverse de manera simultánea con la de conservación de masa para un fluido incompresible; para el caso estacionario se tiene:

$$\nabla \cdot \vec{u} = 0$$

Para obtener la solución analítica del sistema a resolver se parte de la ecuación de Laplace en dos dimensiones:

$$\nabla^2 p = 0 \quad (4.2)$$

ambas ecuaciones existen en el dominio

$$x \in [0, a] \quad (4.3)$$

$$y \in [0, b] \quad (4.4)$$

El bloque de matriz está rodeado por cuatro fracturas abiertas, en donde la ecuación que rige el flujo laminar en canales con paredes porosas se toma como base de las expresiones analíticas propuestas por Karode\* (2001):

$$-\frac{2}{3} \frac{h^3 W}{\mu} \frac{d^2 p}{dx^2} = W \left( \frac{k}{\mu} \frac{\partial p}{\partial y} \right) = \frac{dQ}{dx} \quad (4.5)$$

donde,

$h \triangleq$  abertura de fractura

$W \triangleq$  dimensión de bloque de matriz en el eje  $z$

$\mu \triangleq$  viscosidad dinámica del fluido

$Q \triangleq$  gasto a través de fractura

$p \triangleq$  presión

$k \triangleq$  permeabilidad de matriz

Sean las ecuaciones de las 4 fracturas que lo rodean:

Fractura en  $y = 0$

$$-\frac{2}{3} \frac{h_1^3 W}{\mu} \frac{d^2 p}{dx^2} = W \left( \frac{k}{\mu} \frac{\partial p}{\partial y} \Big|_{y=0} \right) = \frac{dQ}{dx} \quad (4.6)$$

Fractura en  $y = b$

$$-\frac{2}{3} \frac{h_3^3 W}{\mu} \frac{d^2 p}{dx^2} = W \left( -\frac{k}{\mu} \frac{\partial p}{\partial y} \Big|_{y=b} \right) = \frac{dQ}{dx} \quad (4.7)$$

---

\* Consulte el desarrollo matemático de la ecuación en el Apéndice A.

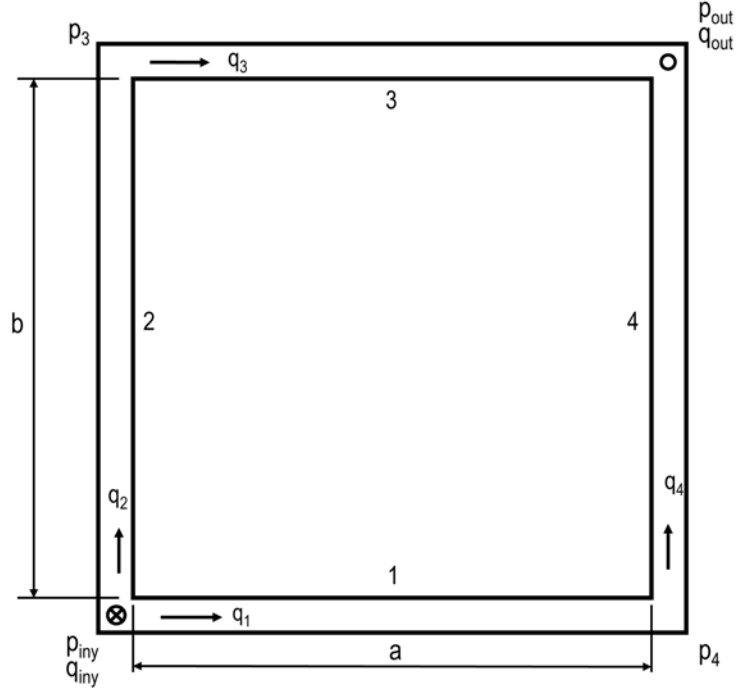


Figura 4-1: Bloque de matriz poroso rodeado por cuatro fracturas.

Fractura en  $x = 0$

$$-\frac{2}{3} \frac{h_2^3 W}{\mu} \frac{d^2 p}{dy^2} = W \left( \frac{k}{\mu} \frac{\partial p}{\partial x} \Big|_{x=0} \right) = \frac{dQ}{dy} \quad (4.8)$$

Fractura en  $x = a$

$$-\frac{2}{3} \frac{h_4^3 W}{\mu} \frac{d^2 p}{dy^2} = W \left( -\frac{k}{\mu} \frac{\partial p}{\partial x} \Big|_{x=a} \right) = \frac{dQ}{dy} \quad (4.9)$$

El sistema es resuelto bajo las siguientes condiciones de frontera, las cuales son:

$$\begin{aligned} p(x_+ \rightarrow 0, y = 0) &= p_1 \\ p(x = 0, y = b) &= p_3 \\ p(x = 0, y_+ \rightarrow 0) &= p_2 = p_1 \\ p(x = a, y = 0) &= p_4 \end{aligned} \quad (4.10)$$

se considera que la presión debe ser una función continua en los nodos donde se cruzan las

fracturas,

$$\begin{aligned}
-\frac{dP}{dx}\Big|_{x_+ \rightarrow 0, y=0} &= \frac{3}{2} \frac{\mu}{h_1^3 W} Q_1 \\
-\frac{dP}{dx}\Big|_{x=0, y=b} &= \frac{3}{2} \frac{\mu}{h_3^3 W} Q_3 \\
-\frac{dP}{dy}\Big|_{x=0, y_+ \rightarrow 0} &= \frac{3}{2} \frac{\mu}{h_3^3 W} Q_2 \\
-\frac{dP}{dy}\Big|_{x=a, y=0} &= \frac{3}{2} \frac{\mu}{h_4^3 W} Q_4
\end{aligned} \tag{4.11}$$

Con el fin de reducir el espacio paramétrico y facilitar el manejo algebraico de las ecuaciones se propone el siguiente grupo de transformación adimensional:

$$\begin{aligned}
\xi &= \frac{x}{a} \\
\eta &= \frac{y}{b} \\
u &= \frac{p - p_{out}}{p_{in} - p_{out}} \\
p_{in} &= \text{máx } p \\
p_{out} &= \text{mín } p \\
\epsilon &= \frac{a}{b}
\end{aligned} \tag{4.12}$$

De esta manera resultan las ecuaciones para la matriz 4.13 y las ecuaciones para las fracturas 4.14-4.17:

$$\frac{\partial^2 u}{\partial \xi^2} + \epsilon^2 \frac{\partial^2 u}{\partial \eta^2} = 0 \tag{4.13}$$

$$-\frac{d^2 u}{d\xi^2} = \frac{3a^2}{2b} \frac{k}{h_1^3} \left( \frac{\partial u}{\partial \eta} \Big|_{\eta=0} \right) \tag{4.14}$$

$$-\frac{d^2 u}{d\xi^2} = \frac{3a^2}{2b} \frac{k}{h_3^3} \left( -\frac{\partial u}{\partial \eta} \Big|_{\eta=1} \right) \tag{4.15}$$

$$-\frac{d^2u}{d\eta^2} = \frac{3b^2}{2a} \frac{k}{h_2^3} \left( \frac{\partial u}{\partial \xi} \Big|_{\xi=0} \right) \quad (4.16)$$

$$-\frac{d^2u}{d\eta^2} = \frac{3b^2}{2a} \frac{k}{h_4^3} \left( -\frac{\partial u}{\partial \xi} \Big|_{\xi=1} \right) \quad (4.17)$$

Asimismo, sea  $h$  un valor promedio de abertura de fractura y  $\gamma_i = \frac{h}{h_i}$ , por lo que finalmente se obtienen las siguientes ecuaciones adimensionales:

$$\frac{\partial^2 u}{\partial \xi^2} + \epsilon^2 \frac{\partial^2 u}{\partial \eta^2} = 0 \quad (4.18)$$

$$-\frac{d^2u}{d\xi^2} = \lambda \epsilon \gamma_1^3 \left( \frac{\partial u}{\partial \eta} \Big|_{\eta=0} \right) \quad (4.19)$$

$$-\frac{d^2u}{d\xi^2} = \lambda \epsilon \gamma_3^3 \left( -\frac{\partial u}{\partial \eta} \Big|_{\eta=1} \right) \quad (4.20)$$

$$-\frac{d^2u}{d\eta^2} = \frac{\lambda \gamma_2^3}{\epsilon^2} \left( \frac{\partial u}{\partial \xi} \Big|_{\xi=0} \right) \quad (4.21)$$

$$-\frac{d^2u}{d\eta^2} = \frac{\lambda \gamma_4^3}{\epsilon^2} \left( -\frac{\partial u}{\partial \xi} \Big|_{\xi=1} \right) \quad (4.22)$$

donde,  $\lambda = \frac{3ak}{2h^3}$ .

Ahora bien, para las condiciones de frontera se realiza el mismo proceso algebraico y se obtienen las siguientes ecuaciones:

$$u(\xi_+ \rightarrow 0, \eta = 0) = u_1 \quad (4.23)$$

$$u(\xi = 0, \eta = 1) = u_3$$

$$u(\xi = 0, \eta = 0) = u_2$$

$$u(\xi = 1, \eta = 0) = u_4$$



$$\begin{aligned}
-\left. \frac{du}{d\xi} \right|_{\xi_+ \rightarrow 0, \eta=0} &= \gamma_1 q_1 \\
-\left. \frac{du}{d\xi} \right|_{\xi=0, \eta=1} &= \gamma_3 q_3 \\
-\left. \frac{du}{d\eta} \right|_{\xi=0, \eta_+ \rightarrow 0} &= \frac{\gamma_2}{\epsilon} q_2 \\
-\left. \frac{du}{d\eta} \right|_{\xi=1, \eta_+ \rightarrow 0} &= \frac{\gamma_4}{\epsilon} q_4
\end{aligned} \tag{4.24}$$

donde

$$q_i = \frac{3}{2} \frac{a\mu Q_i}{(p_{in} - p_{out}) h^3 W} \tag{4.25}$$

## 4.1. Solución asintótica

La teoría de perturbaciones obtiene soluciones aproximadas a problemas que implican un pequeño parámetro. El fundamento de la teoría de perturbación consiste en descomponer un problema difícil en un número infinito de problemas relativamente fáciles (Bender y Orzag, 1978). En este caso la metodología es utilizada para resolver la ecuación diferencial de Laplace acoplada con el flujo en las fracturas.

Las fracturas de un yacimiento son de razón de aspecto mucho menor a la unidad, se considera un largo  $a$  de 1 [m], una permeabilidad  $k$  de  $0.01 \times 10^{-12}$  [m<sup>2</sup>] y una apertura promedio de las fracturas  $h$  de  $1 \times 10^{-3}$  [m], entonces:

$$\lambda = \frac{3ak}{2h^3} = \frac{3(1[m])(0.01 \times 10^{-12} [m^2])}{2(1 \times 10^{-3} [m])^3} = 0.015 \ll 1$$

Debido a lo anterior se propone una solución del tipo asintótico, utilizando como parámetro de expansión a  $\lambda \ll 1$ , es decir:

$$u = \sum_{n=0}^{\infty} u_n \lambda^n = u_0 + u_1 \lambda + u_2 \lambda^2 + \mathcal{O}(\lambda^3) \tag{4.26}$$

Sustituyendo esta propuesta en la ecuación de Laplace, se obtiene la siguiente expansión

para la matriz:

$$\frac{\partial^2 u_0}{\partial \xi^2} + \lambda \frac{\partial^2 u_1}{\partial \xi^2} + \lambda^2 \frac{\partial^2 u_2}{\partial \xi^2} + \epsilon^2 \frac{\partial^2 u_0}{\partial \eta^2} + \lambda \epsilon^2 \frac{\partial^2 u_1}{\partial \eta^2} + \lambda^2 \epsilon^2 \frac{\partial^2 u_2}{\partial \eta^2} + \mathcal{O}(\lambda^3) = 0 \quad (4.27)$$

O bien, para cada una de las fracturas del bloque de matriz:

$$-\frac{d^2 u_0}{d\xi^2} - \lambda \frac{d^2 u_1}{d\xi^2} - \lambda^2 \frac{d^2 u_2}{d\xi^2} + \mathcal{O}(\lambda^3) = \lambda \epsilon \gamma_1^3 \left( \frac{\partial u_0}{\partial \eta} \Big|_{\eta=0} + \lambda \frac{\partial u_1}{\partial \eta} \Big|_{\eta=0} \right) \quad (4.28)$$

$$-\frac{d^2 u_0}{d\xi^2} - \lambda \frac{d^2 u_1}{d\xi^2} - \lambda^2 \frac{d^2 u_2}{d\xi^2} + \mathcal{O}(\lambda^3) = \lambda \epsilon \gamma_3^3 \left( -\frac{\partial u_0}{\partial \eta} \Big|_{\eta=1} - \lambda \frac{\partial u_1}{\partial \eta} \Big|_{\eta=1} \right) \quad (4.29)$$

$$-\frac{d^2 u_0}{d\eta^2} - \lambda \frac{d^2 u_1}{d\eta^2} - \lambda^2 \frac{d^2 u_2}{d\eta^2} + \mathcal{O}(\lambda^3) = \frac{\lambda \gamma_2^3}{\epsilon^2} \left( \frac{\partial u_0}{\partial \xi} \Big|_{\xi=0} + \lambda \frac{\partial u_1}{\partial \xi} \Big|_{\xi=0} \right) \quad (4.30)$$

$$-\frac{d^2 u_0}{d\eta^2} - \lambda \frac{d^2 u_1}{d\eta^2} - \lambda^2 \frac{d^2 u_2}{d\eta^2} + \mathcal{O}(\lambda^3) = \frac{\lambda \gamma_4^3}{\epsilon^2} \left( -\frac{\partial u_0}{\partial \xi} \Big|_{\xi=1} - \lambda \frac{\partial u_1}{\partial \xi} \Big|_{\xi=1} \right) \quad (4.31)$$

Adicionalmente, dadas las condiciones de frontera desde orden cero hasta orden dos:

$$u_0(\xi_+ \rightarrow 0, \eta = 0) + \lambda u_1(\xi_+ \rightarrow 0, \eta = 0) + \lambda^2 u_2(\xi_+ \rightarrow 0, \eta = 0) + \mathcal{O}(\lambda^3) = u_1 \quad (4.32)$$

$$u_0(\xi = 0, \eta = 1) + \lambda u_1(\xi = 0, \eta = 1) + \lambda^2 u_2(\xi = 0, \eta = 1) + \mathcal{O}(\lambda^3) = u_3 \quad (4.33)$$

$$u_0(\xi = 0, \eta = 0) + \lambda u_1(\xi = 0, \eta = 0) + \lambda^2 u_2(\xi = 0, \eta = 0) + \mathcal{O}(\lambda^3) = u_2 \quad (4.34)$$

$$u_0(\xi = 1, \eta = 0) + \lambda u_1(\xi = 1, \eta = 0) + \lambda^2 u_2(\xi = 1, \eta = 0) + \mathcal{O}(\lambda^3) = u_4 \quad (4.35)$$

Finalmente,

$$-\frac{du_0}{d\xi}\Big|_{\xi_+\rightarrow 0, \eta=0} - \lambda \frac{du_1}{d\xi}\Big|_{\xi_+\rightarrow 0, \eta=0} - \lambda^2 \frac{du_2}{d\xi}\Big|_{\xi_+\rightarrow 0, \eta=0} + \mathcal{O}(\lambda^3) = \gamma_1 q_{1,0} + \lambda \gamma_1 q_{1,1} \quad (4.36)$$

$$-\frac{du_0}{d\xi}\Big|_{\xi=0, \eta=1} - \lambda \frac{du_1}{d\xi}\Big|_{\xi=0, \eta=1} - \lambda^2 \frac{du_2}{d\xi}\Big|_{\xi=0, \eta=1} + \mathcal{O}(\lambda^3) = \gamma_3 q_{3,0} + \lambda \gamma_3 q_{3,1} \quad (4.37)$$

$$-\frac{du_0}{d\eta}\Big|_{\xi=0, \eta_+\rightarrow 0} - \lambda \frac{du_1}{d\eta}\Big|_{\xi=0, \eta_+\rightarrow 0} - \lambda^2 \frac{du_2}{d\eta}\Big|_{\xi=0, \eta_+\rightarrow 0} + \mathcal{O}(\lambda^3) = \frac{\gamma_2}{\epsilon} q_{2,0} + \lambda \frac{\gamma_2}{\epsilon} q_{2,1} \quad (4.38)$$

$$-\frac{du_0}{d\eta}\Big|_{\xi=1, \eta_+\rightarrow 0} - \lambda \frac{du_1}{d\eta}\Big|_{\xi=1, \eta_+\rightarrow 0} - \lambda^2 \frac{du_2}{d\eta}\Big|_{\xi=1, \eta_+\rightarrow 0} + \mathcal{O}(\lambda^3) = \frac{\gamma_4}{\epsilon} q_{4,0} + \lambda \frac{\gamma_4}{\epsilon} q_{4,1} \quad (4.39)$$

#### 4.1.1. Solución a orden cero

Dado que de la teoría de métodos asintóticos  $\lambda$  y sus potencias son linealmente independientes, la ecuación en la matriz a orden cero es:

$$\frac{\partial^2 u_0}{\partial \xi^2} + \epsilon^2 \frac{\partial^2 u_0}{\partial \eta^2} = 0 \quad (4.40)$$

Asimismo, las ecuaciones de las fracturas son:

$$-\frac{d^2 u_0}{d\xi^2} = 0 \quad (4.41)$$

$$-\frac{d^2 u_0}{d\xi^2} = 0 \quad (4.42)$$

$$-\frac{d^2 u_0}{d\eta^2} = 0 \quad (4.43)$$

$$-\frac{d^2 u_0}{d\eta^2} = 0 \quad (4.44)$$

$$\begin{aligned}
-\left. \frac{du_0}{d\xi} \right|_{\xi_+ \rightarrow 0, \eta=0} &= \gamma_1 q_{1,0} \\
-\left. \frac{du_0}{d\xi} \right|_{\xi=0, \eta=1} &= \gamma_3 q_{3,0} \\
-\left. \frac{du_0}{d\eta} \right|_{\xi=0, \eta_+ \rightarrow 0} &= \frac{\gamma_2}{\epsilon} q_{2,0} \\
-\left. \frac{du_0}{d\eta} \right|_{\xi=1, \eta_+ \rightarrow 0} &= \frac{\gamma_4}{\epsilon} q_{4,0}
\end{aligned} \tag{4.45}$$

La solución en las fracturas a orden cero es:

$$\begin{aligned}
u_0(\xi, \eta = 0) &= \alpha_1 + \beta_1 \xi \\
u_0(\xi, \eta = 1) &= \alpha_3 + \beta_3 \xi \\
u_0(\xi = 0, \eta) &= \alpha_2 + \beta_2 \eta \\
u_0(\xi = 1, \eta) &= \alpha_4 + \beta_4 \eta
\end{aligned} \tag{4.46}$$

sujeta a las condiciones de frontera:

$$\begin{aligned}
u_0(\xi_+ \rightarrow 0, \eta = 0) &= u_1 \\
u_0(\xi = 0, \eta = 1) &= u_3 \\
u_0(\xi = 0, \eta = 0) &= u_2 \\
u_0(\xi = 1, \eta = 0) &= u_4
\end{aligned}$$

o bien, en las fracturas:

$$\begin{aligned}
u_{0,1}(\xi, \eta = 0) &= u_1 + \beta_1 \xi \\
u_{0,3}(\xi, \eta = 1) &= u_3 + \beta_3 \xi \\
u_{0,2}(\xi = 0, \eta) &= u_2 + \beta_2 \eta \\
u_{0,4}(\xi = 1, \eta) &= u_4 + \beta_4 \eta
\end{aligned} \tag{4.47}$$

usando estas otras condiciones de frontera

$$\begin{aligned}
-\frac{du_0}{d\xi}\Big|_{\xi_+ \rightarrow 0, \eta=0} &= \gamma_1 q_1 \\
-\frac{du_0}{d\xi}\Big|_{\xi=0, \eta=1} &= \gamma_3 q_3 \\
-\frac{du_0}{d\eta}\Big|_{\xi=0, \eta_+ \rightarrow 0} &= \frac{\gamma_2}{\epsilon} q_2 \\
-\frac{du_0}{d\eta}\Big|_{\xi=1, \eta_+ \rightarrow 0} &= \frac{\gamma_4}{\epsilon} q_4
\end{aligned} \tag{4.48}$$

resultando

$$\begin{aligned}
u_{0,1}(\xi, \eta = 0) &= u_1 - \xi \gamma_1 q_1 \\
u_{0,3}(\xi, \eta = 1) &= u_3 - \xi \gamma_3 q_3 \\
u_{0,2}(\xi = 0, \eta) &= u_2 - \eta \frac{\gamma_2}{\epsilon} q_2 \\
u_{0,4}(\xi = 1, \eta) &= u_4 - \eta \frac{\gamma_4}{\epsilon} q_4
\end{aligned} \tag{4.49}$$

donde, el subíndice es el orden y la cara del bloque de matriz, respectivamente. En los nodos de las fracturas la presión debe ser una función continua, entonces, las ecuaciones se encuentran sujetas bajo las siguientes restricciones:

$$\begin{aligned}
u_0(0, \eta = 0) &= \alpha_1 + \beta_1(0) = u_0(\xi = 0, 0) = \alpha_2 + \beta_2(0) \\
\alpha_1 &= \alpha_2
\end{aligned} \tag{4.50}$$

$$\begin{aligned}
u_0(\xi = 0, 1) &= \alpha_2 + \beta_2(1) = u_0(0, \eta = 1) = \alpha_3 + \beta_3(0) \\
\alpha_2 + \beta_2 &= \alpha_3
\end{aligned} \tag{4.51}$$

$$\begin{aligned}
u_0(1, \eta = 1) &= \alpha_3 + \beta_3(1) = u_0(\xi = 1, 1) = \alpha_4 + \beta_4(1) & (4.52) \\
\alpha_3 + \beta_3 &= \alpha_4 + \beta_4
\end{aligned}$$

$$\begin{aligned}
u_0(\xi = 1, 0) &= \alpha_4 + \beta_4(0) = u_0(1, \eta = 0) = \alpha_1 + \beta_1(1) & (4.53) \\
\alpha_4 &= \alpha_1 + \beta_1
\end{aligned}$$

de aquí se llega a:

$$\begin{aligned}
\alpha_2 &= \widehat{\alpha_1} & (4.54) \\
\beta_2 &= \widehat{\alpha_3} - \alpha_2 \\
\beta_1 &= \widehat{\alpha_4} - \alpha_1 \\
\beta_3 &= \alpha_4 + \widehat{\beta_4} - \alpha_3
\end{aligned}$$

Donde  $\alpha_1, \alpha_3, \alpha_4$  y  $\beta_4$  son los parámetros que se eligieron como linealmente independientes, los cuales son señalados con el símbolo  $\widehat{\phantom{x}}$ , cuyo valor define a las condiciones de frontera de Dirichlet entre el sistema de fracturas y la matriz. Como se verá más adelante en el capítulo 5 estos parámetros serán la clave para el entrenamiento de los modelos de redes neuronales artificiales empleadas en este trabajo.

Ahora bien, debido a que la presión adimensional está en el intervalo  $[0, 1]$  los parámetros están sujetos a las siguientes restricciones:

$$\begin{aligned}
-\alpha_4 &\leq \beta_4 \leq 1 - \alpha_4 & (4.55) \\
0 &\leq \alpha_1 \leq 1 \\
0 &\leq \alpha_3 \leq 1 \\
0 &\leq \alpha_4 \leq 1
\end{aligned}$$

Finalmente, el problema numérico a resolver a orden cero es:

$$\frac{\partial^2 u_0}{\partial \xi^2} + \epsilon^2 \frac{\partial^2 u_0}{\partial \eta^2} = 0 \quad (4.56)$$

$$u_{0,1}(\xi, \eta = 0) = u_1 - \xi \gamma_1 q_1 \quad (4.57)$$

$$u_{0,3}(\xi, \eta = 1) = u_3 - \xi \gamma_3 q_3$$

$$u_{0,2}(\xi = 0, \eta) = u_2 - \eta \frac{\gamma_2}{\epsilon} q_2$$

$$u_{0,4}(\xi = 1, \eta) = u_4 - \eta \frac{\gamma_4}{\epsilon} q_4$$

### Cálculo de gastos a orden cero

El sistema del bloque de matriz poroso rodeado de cuatro fracturas es un sistema conservativo, por continuidad en flujo en las esquinas los gastos a orden cero en cada una de las fracturas se tiene:

$$u_{iny} - \gamma_1 q_1 = u_4 \quad (4.58)$$

$$u_{iny} - \frac{\gamma_2}{\epsilon} q_2 = u_3 \quad (4.59)$$

$$u_{out} = u_3 - \gamma_3 q_3 \quad (4.60)$$

$$u_{out} = u_4 - \frac{\gamma_4}{\epsilon} q_4 \quad (4.61)$$

$$q_1 = q_4 \quad (4.62)$$

$$q_2 = q_3$$

Igualando ecuaciones y simplificando, los modelos matemáticos para calcular los gastos a orden cero en las fracturas:

$$q_1 = \frac{u_{iny} - u_{out}}{\left(\gamma_1 + \frac{\gamma_4}{\epsilon}\right)} \quad (4.63)$$

$$q_2 = \frac{u_{iny} - u_{out}}{\gamma_3 - \frac{\gamma_2}{\epsilon}} \quad (4.64)$$

#### 4.1.2. Solución a orden uno

Sea la ecuación en la matriz:

$$\frac{\partial^2 u_1}{\partial \xi^2} + \epsilon^2 \frac{\partial^2 u_1}{\partial \eta^2} = 0 \quad (4.65)$$

en las fracturas:

$$-\frac{d^2 u_{1,1}}{d\xi^2} = \epsilon \gamma_1^3 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) \quad (4.66)$$

$$-\frac{d^2 u_{1,3}}{d\xi^2} = \epsilon \gamma_3^3 \left( -\frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) \quad (4.67)$$

$$-\frac{d^2 u_{1,2}}{d\eta^2} = \frac{\gamma_2^3}{\epsilon^2} \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) \quad (4.68)$$

$$-\frac{d^2 u_{1,4}}{d\eta^2} = \frac{\gamma_4^3}{\epsilon^2} \left( -\frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) \quad (4.69)$$

bajo las condiciones de frontera:

$$u_{1,1} (\xi_+ \rightarrow 0, \eta = 0) = 0 \quad (4.70)$$

$$u_{1,2} (\xi = 0, \eta = 0) = 0$$

$$u_{1,3} (\xi \rightarrow 1, \eta = 1) = 0$$

$$u_{1,4} (\xi = 1, \eta \rightarrow 1) = 0$$



$$\begin{aligned}
\left. \frac{du_{1,1}}{d\xi} \right|_{\xi_+ \rightarrow 0, \eta=0} &= \gamma_1 q_{1,1} \\
\left. \frac{du_{1,3}}{d\xi} \right|_{\xi=0, \eta=1} &= \gamma_3 q_{3,1} \\
\left. \frac{du_{1,2}}{d\eta} \right|_{\xi=0, \eta_+ \rightarrow 0} &= \frac{\gamma_2}{\epsilon} q_{2,1} \\
\left. \frac{du_{1,4}}{d\eta} \right|_{\xi=1, \eta_+ \rightarrow 0} &= \frac{\gamma_4}{\epsilon} q_{4,1}
\end{aligned} \tag{4.71}$$

La solución general para las ecuaciones en la fractura es:

Para  $\eta = 0$

$$-\frac{d^2 u_{1,1}}{d\xi^2} = \epsilon \gamma_1^3 \left( \left. \frac{\partial u_{0,1}}{\partial \eta} \right|_{\eta=0} \right) \tag{4.72}$$

$$u_{1,1} = \int_0^\xi \xi \epsilon \gamma_1^3 \left( \left. \frac{\partial u_{0,1}}{\partial \eta} \right|_{\eta=0} \right) d\xi - \xi \int_0^\xi \epsilon \gamma_1^3 \left( \left. \frac{\partial u_{0,1}}{\partial \eta} \right|_{\eta=0} \right) d\xi + a_1 \xi + b_1 \tag{4.73}$$

Para  $\eta = 1$

$$-\frac{d^2 u_{1,3}}{d\xi^2} = \epsilon \gamma_3^3 \left( - \left. \frac{\partial u_{0,3}}{\partial \eta} \right|_{\eta=1} \right) \tag{4.74}$$

$$u_{1,3} = \int_0^\xi \xi \epsilon \gamma_3^3 \left( - \left. \frac{\partial u_{0,3}}{\partial \eta} \right|_{\eta=1} \right) d\xi - \xi \int_0^\xi \epsilon \gamma_3^3 \left( - \left. \frac{\partial u_{0,3}}{\partial \eta} \right|_{\eta=1} \right) d\xi + a_3 \xi + b_3 \tag{4.75}$$

Para  $\xi = 0$

$$-\frac{d^2 u_{1,2}}{d\eta^2} = \frac{\gamma_2^3}{\epsilon^2} \left( \left. \frac{\partial u_{0,2}}{\partial \xi} \right|_{\xi=0} \right) \tag{4.76}$$

$$u_{1,2} = \int_0^\eta \eta \frac{\gamma_2^3}{\epsilon^2} \left( \left. \frac{\partial u_{0,2}}{\partial \xi} \right|_{\xi=0} \right) d\eta - \eta \int_0^\eta \frac{\gamma_2^3}{\epsilon^2} \left( \left. \frac{\partial u_{0,2}}{\partial \xi} \right|_{\xi=0} \right) d\eta + a_2 \eta + b_2 \tag{4.77}$$

Para  $\xi = 1$

$$-\frac{d^2 u_{1,4}}{d\eta^2} = \frac{\gamma_{4,4}^3}{\epsilon^2} \left( -\frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) \quad (4.78)$$

$$u_{1,4} = \int_0^\eta \eta \frac{\gamma_4^3}{\epsilon^2} \left( -\frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta - \eta \int_0^\eta \frac{\gamma_4^3}{\epsilon^2} \left( -\frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta + a_4 \eta + b_4 \quad (4.79)$$

ahora bien, con las condiciones de frontera finalmente se obtienen las ecuaciones a resolver a orden uno:

$$u_{1,1}(\xi_+ \rightarrow 0, \eta = 0) = 0$$

$$u_{1,2}(\xi = 0, \eta = 0) = 0$$

$$u_{1,3}(\xi \rightarrow 1, \eta = 1) = 0$$

$$u_{1,4}(\xi = 1, \eta \rightarrow 1) = 0$$

$$u_{1,1}(\xi_+ \rightarrow 0, \eta = 0) = 0 \rightarrow b_1 = 0 \quad (4.80)$$

$$u_{1,1} = \int_0^\xi \xi \epsilon \gamma_1^3 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi - \xi \int_0^\xi \epsilon \gamma_1^3 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi + a_1 \xi$$

$$u_{1,2}(\xi = 0, \eta = 0) = 0 \rightarrow b_2 = 0 \quad (4.81)$$

$$u_{1,2} = \int_0^\eta \eta \frac{\gamma_2^3}{\epsilon^2} \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta - \eta \int_0^\eta \frac{\gamma_2^3}{\epsilon^2} \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta + a_2 \eta$$

$$u_{1,3}(\xi \rightarrow 1, \eta = 1) = 0 \quad (4.82)$$

$$u_{1,3} = \epsilon \gamma_3^3 \int_0^1 \xi \left( -\frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi - \epsilon \gamma_3^3 \int_0^1 \left( -\frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi + a_3 + b_3 = 0$$

$$u_{1,4}(\xi = 1, \eta \rightarrow 1) = 0 \quad (4.83)$$

$$u_{1,4} = \frac{\gamma_4^3}{\epsilon^2} \int_0^1 \eta \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta - \frac{\gamma_4^3}{\epsilon^2} \int_0^1 \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta + a_4 + b_4 = 0$$

Algebraicamente se determinan los valores de  $a_{1,\dots,4}$  y  $b_{1,\dots,4}$  que satisfacen las ecuaciones:

$$\begin{aligned} a_1 = & \quad (4.84) \\ & + \frac{1}{\left(1 + \frac{\gamma_4}{\epsilon\gamma_1}\right)} \left[ (\gamma_4\gamma_1^2 + \epsilon\gamma_1^3) \int_0^1 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi - \epsilon\gamma_1^3 \int_0^1 \xi \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi \right] \\ & + \frac{\gamma_4^3}{\left(1 + \frac{\gamma_4}{\epsilon\gamma_1}\right) \epsilon^2} \left[ \int_0^1 \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta - \int_0^1 \eta \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta \right] \end{aligned}$$

$$\begin{aligned} a_2 = & \quad (4.85) \\ & \frac{\epsilon\gamma_3^3}{\left(1 + \frac{\gamma_4}{\epsilon\gamma_1}\right)} \left[ \int_0^1 \left( - \frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi - \int_0^1 \xi \left( - \frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi \right] \\ & + \frac{1}{\left(1 + \frac{\gamma_4}{\epsilon\gamma_1}\right)} \left[ \left( \frac{\gamma_3\gamma_2^2}{\epsilon} + \frac{\gamma_2^3}{\epsilon^2} \right) \int_0^1 \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta - \frac{\gamma_2^3}{\epsilon^2} \int_0^1 \eta \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta \right] \end{aligned}$$

$$\begin{aligned} a_3 = & \quad (4.86) \\ & - \frac{\gamma_3\gamma_2^2}{\epsilon} \int_0^1 \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) \\ & + \frac{\gamma_3\epsilon}{\gamma_2} \left( \begin{aligned} & \frac{\epsilon\gamma_3^3}{\left(1 + \frac{\gamma_4}{\epsilon\gamma_1}\right)} \left[ \int_0^1 \left( - \frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi - \int_0^1 \xi \left( - \frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi \right] \\ & + \frac{1}{\left(1 + \frac{\gamma_4}{\epsilon\gamma_1}\right)} \left[ \left( \frac{\gamma_3\gamma_2^2}{\epsilon} + \frac{\gamma_2^3}{\epsilon^2} \right) \int_0^1 \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta - \frac{\gamma_2^3}{\epsilon^2} \int_0^1 \eta \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta \right] \end{aligned} \right) \end{aligned}$$

$$a_4 = \tag{4.87}$$

$$\begin{aligned}
& -\gamma_4 \gamma_1^2 \int_0^1 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi \\
& + \frac{\gamma_4}{\epsilon \gamma_1} \left( + \frac{1}{\left(1 + \frac{\gamma_4}{\epsilon \gamma_1}\right)} \left[ (\gamma_4 \gamma_1^2 + \epsilon \gamma_1^3) \int_0^1 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi - \epsilon \gamma_1^3 \int_0^1 \xi \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi \right] \right. \\
& \quad \left. + \frac{\gamma_4^3}{\left(1 + \frac{\gamma_4}{\epsilon \gamma_1}\right) \epsilon^2} \left[ \int_0^1 \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta - \int_0^1 \eta \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta \right] \right)
\end{aligned}$$

$$b_1 = 0 \tag{4.88}$$

$$b_2 = 0 \tag{4.89}$$

$$b_3 = \tag{4.90}$$

$$\begin{aligned}
& \frac{\gamma_2^3}{\epsilon^2} \int_0^1 \eta \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta - \frac{\gamma_2^3}{\epsilon^2} \int_0^1 \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta \\
& + \frac{\epsilon \gamma_3^3}{\left(1 + \frac{\gamma_4}{\epsilon \gamma_1}\right)} \left[ \int_0^1 \left( - \frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi - \int_0^1 \xi \left( - \frac{\partial u_{0,3}}{\partial \eta} \Big|_{\eta=1} \right) d\xi \right] \\
& + \frac{1}{\left(1 + \frac{\gamma_4}{\epsilon \gamma_1}\right)} \left[ \left( \frac{\gamma_3 \gamma_2^2}{\epsilon} + \frac{\gamma_2^3}{\epsilon^2} \right) \int_0^1 \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta - \frac{\gamma_2^3}{\epsilon^2} \int_0^1 \eta \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} \right) d\eta \right]
\end{aligned}$$

$$b_4 = \tag{4.91}$$

$$\begin{aligned}
& \epsilon \gamma_1^3 \int_0^1 \xi \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi - \epsilon \gamma_1^3 \int_0^1 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi \\
& + \frac{1}{\left(1 + \frac{\gamma_4}{\epsilon \gamma_1}\right)} \left[ (\gamma_4 \gamma_1^2 + \epsilon \gamma_1^3) \int_0^1 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi - \epsilon \gamma_1^3 \int_0^1 \xi \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} \right) d\xi \right] \\
& + \frac{\gamma_4^3}{\left(1 + \frac{\gamma_4}{\epsilon \gamma_1}\right) \epsilon^2} \left[ \int_0^1 \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta - \int_0^1 \eta \left( - \frac{\partial u_{0,4}}{\partial \xi} \Big|_{\xi=1} \right) d\eta \right]
\end{aligned}$$

### Cálculo de gastos a orden uno

Para el cálculo de los gastos, por continuidad en flujo en las esquinas a orden uno en cada una de las fracturas se obtiene como:

$$\begin{aligned}
 q_{1,2}(\xi = 0, \eta = 1) &= q_{1,3}(\xi = 0, \eta = 1) \\
 \frac{\epsilon}{\gamma_2} \frac{du_{1,2}}{d\eta} \Big|_{(\xi=0, \eta=1)} &= \frac{1}{\gamma_3} \frac{du_{1,3}}{d\xi} \Big|_{(\xi=0, \eta=1)}
 \end{aligned} \tag{4.92}$$

$$\begin{aligned}
 q_{1,1}(\xi = 1, \eta = 0) &= q_{1,4}(\xi = 1, \eta = 0) \\
 \frac{1}{\gamma_1} \frac{du_{1,1}}{d\xi} \Big|_{(\xi=1, \eta=0)} &= \frac{\epsilon}{\gamma_4} \frac{du_{1,4}}{d\eta} \Big|_{(\xi=1, \eta=0)}
 \end{aligned} \tag{4.93}$$

Derivando y simplificando las ecuaciones 4.80, 4.81, 4.82, 4.83, los modelos matemáticos resultantes para calcular los gastos en las fracturas que rodean el bloque de matriz poroso son:

$$\begin{aligned}
 q_{1,2}(\xi = 0, \eta = 1) &= q_{1,3}(\xi = 0, \eta = 1) \\
 \frac{\epsilon}{\gamma_2} \frac{du_{1,2}}{d\eta} \Big|_{(\xi=0, \eta=1)} &= \frac{1}{\gamma_3} \frac{du_{1,3}}{d\xi} \Big|_{(\xi=0, \eta=1)} \\
 -\frac{\gamma_2^2}{\epsilon} \int_0^1 \left( \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=0} - \frac{\partial u_{0,2}}{\partial \xi} \Big|_{\xi=-\frac{h_2}{a}} \right) + \frac{\epsilon}{\gamma_2} a_2 &= \frac{1}{\gamma_3} a_3
 \end{aligned} \tag{4.94}$$

$$\begin{aligned}
 q_{1,1}(\xi = 1, \eta = 0) &= q_{1,4}(\xi = 1, \eta = 0) \\
 \frac{1}{\gamma_1} \frac{du_{1,1}}{d\xi} \Big|_{(\xi=1, \eta=0)} &= \frac{\epsilon}{\gamma_4} \frac{du_{1,4}}{d\eta} \Big|_{(\xi=1, \eta=0)} \\
 -\epsilon \gamma_1^2 \int_0^1 \left( \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=0} - \frac{\partial u_{0,1}}{\partial \eta} \Big|_{\eta=-\frac{h_1}{b}} \right) d\xi + \frac{a_1}{\gamma_1} &= \frac{\epsilon}{\gamma_4} a_4
 \end{aligned} \tag{4.95}$$

Entonces, por continuidad global a orden uno:

$$q_{1,1}(\xi = 0, \eta = 0) + q_{1,2}(\xi = 0, \eta = 0) = q_{1,3}(\xi = 1, \eta = 1) + q_{1,4}(\xi = 1, \eta = 1) \quad (4.96)$$

En este trabajo sólo se consideraron los órdenes cero y uno. Sin embargo, es posible demostrar que los problemas a orden superior son idénticos a de orden 1. Lo anterior implica resolver la ecuación de Laplace con condiciones de frontera lineales para cada orden, lo cual es posible resolverse mediante una red neuronal entrenada.

## 4.2. Solución general de la ecuación de Laplace en dos dimensiones

Sea la ecuación de Laplace en dos dimensiones

$$\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = 0 \quad (4.97)$$

en el dominio rectangular

$$0 < x < x_0$$

$$0 < y < y_0$$

en donde una solución polinomial es:

$$\varphi(x, y) = axy + bx + cy + d \quad (4.98)$$

bajo las condiciones de frontera de Dirichlet lineales:

$$\begin{aligned}
\varphi(x, 0) &= \varphi_1 = A_1 + B_1x & (4.99) \\
\varphi(0, y) &= \varphi_2 = A_2 + B_2y \\
\varphi(x, y_0) &= \varphi_3 = A_3 + B_3x \\
\varphi(x_0, y) &= \varphi_4 = A_4 + B_4y
\end{aligned}$$

Si las condiciones de frontera son continuas en cada esquina:  
para el origen  $(0, 0)$

$$\varphi_1 = A_1 + B_1 0 = \varphi_2 = A_2 + B_2 0$$

$$A_1 = A_2 \quad (4.100)$$

en la coordenada  $(0, y_0)$

$$\varphi_2 = A_2 + B_2 y_0 = \varphi_3 = A_3 + B_3 0$$

$$B_2 = \frac{A_3 - A_2}{y_0} \quad (4.101)$$

en el punto  $(x_0, 0)$

$$\varphi_1 = A_1 + B_1 x_0 = \varphi_4 = A_4 + B_4 0$$

$$B_1 = \frac{A_4 - A_1}{x_0} \quad (4.102)$$

en el punto  $(x_0, y_0)$

$$B_3 = \frac{A_4 + B_4 y_0 - A_3}{x_0} \quad (4.103)$$

entonces dejando como parámetros independientes a  $A_1, A_3, A_4$  y  $B_4$

$$\begin{aligned}
\varphi(x, 0) &= \varphi_1 = A_1 + \left(\frac{A_4 - A_1}{x_0}\right)x \\
\varphi(0, y) &= \varphi_2 = A_1 + \left(\frac{A_3 - A_1}{y_0}\right)y \\
\varphi(x, y_0) &= \varphi_3 = A_3 + \left(\frac{A_4 + B_4 y_0 - A_3}{x_0}\right)x \\
\varphi(x_0, y) &= \varphi_4 = A_4 + B_4 y
\end{aligned}$$

Por lo tanto, la solución particular:

$$\varphi(x, y) = \frac{B_4 y_0 - A_3 + A_1}{y_0 x_0} xy + \frac{A_4 - A_1}{x_0} x + \frac{A_3 - A_1}{y_0} y + A_1 \quad (4.104)$$

Ahora bien, adimensionalizando con el siguiente grupo de transformación:

$$\begin{aligned}
\xi &= \frac{x}{x_0} \\
\eta &= \frac{y}{y_0} \\
u &= \frac{\varphi}{\varphi_c}
\end{aligned}$$

se tiene:

$$\frac{\partial^2 \varphi}{\partial \xi^2} + \varepsilon^2 \frac{\partial^2 \varphi}{\partial \eta^2} = 0$$

donde

$$\varepsilon = \frac{x_0}{y_0}$$

en el dominio rectangular

$$0 < \xi < 1$$

$$0 < \eta < 1$$



con condiciones

$$\begin{aligned}\varphi(x, 0) &= \varphi_1 = A_1 + \left(\frac{A_4 - A_1}{x_0}\right)x \\ u(\xi, 0) &= \alpha_1 + (\alpha_4 - \alpha_1)\xi\end{aligned}\tag{4.105}$$

$$\begin{aligned}\varphi(0, y) &= \varphi_2 = A_1 + \left(\frac{A_3 - A_1}{y_0}\right)y \\ u(0, \eta) &= \alpha_1 + (\alpha_3 - \alpha_1)\eta\end{aligned}\tag{4.106}$$

$$\begin{aligned}\varphi(x, y_0) &= \varphi_3 = A_3 + \left(\frac{A_4 + B_4 y_0 - A_3}{x_0}\right)x \\ u(\xi, 1) &= \alpha_3 + (\alpha_4 + \beta_4 - \alpha_3)\xi\end{aligned}\tag{4.107}$$

$$\begin{aligned}\varphi(x_0, y) &= \varphi_4 = A_4 + B_4 y \\ u(1, \eta) &= \alpha_4 + \beta_4 \eta\end{aligned}$$

Finalmente, la solución a la ecuación de Laplace en dos dimensiones es:

$$u(\xi, \eta) = (\beta_4 - \alpha_3 + \alpha_1)\xi\eta + (\alpha_4 - \alpha_1)\xi + (\alpha_3 - \alpha_1)\eta + \alpha_1\tag{4.108}$$

En la **Figura 4-2** se muestra la gráfica de la solución completa.

Una vez obtenida la solución analítica de la ecuación de Laplace (ecuación 4.108), es necesario obtener ecuaciones complementarias que son requeridas para obtener los valores numéricos de los modelos matemáticos a solucionar. Estas se obtienen mediante cálculo diferencial e integral de las ecuaciones siguientes:

Sea la derivada de la ecuación 4.108 con respecto a  $\eta$  evaluada en  $\xi = 0$  y  $\xi = 1$ :

$$\left.\frac{\partial u}{\partial \eta}\right|_{\xi=0} = \frac{\varphi_c}{y_0}(\alpha_3 - \alpha_1)\tag{4.109}$$

$$\left. \frac{\partial u}{\partial \eta} \right|_{\xi=1} = \frac{\varphi_c}{y_0} \beta_4 \quad (4.110)$$

Por otro lado, la derivada con respecto a  $\xi$  evaluada en  $\eta = 0$  y  $\eta = 1$  :

$$\left. \frac{\partial u}{\partial \xi} \right|_{\eta=0} = \frac{\varphi_c}{x_0} (\alpha_4 - \alpha_1) \quad (4.111)$$

$$\left. \frac{\partial u}{\partial \xi} \right|_{\eta=1} = \frac{\varphi_c}{x_0} (\beta_4 - \alpha_3 + \alpha_4) \quad (4.112)$$

Asimismo, al integrar la ecuación 4.108, se obtienen los flujos totales:

$$\int_0^{x_0} \left. \frac{\partial \varphi}{\partial x} \right|_{y=0} \mathbf{d}x = \varphi_c (\alpha_4 - \alpha_1) \quad (4.113)$$

$$\int_0^{x_0} \left. \frac{\partial \varphi}{\partial x} \right|_{y=y_0} \mathbf{d}x = \varphi_c (\beta_4 - \alpha_3 + \alpha_4) \quad (4.114)$$

$$\int_0^{y_0} \left. \frac{\partial \varphi}{\partial y} \right|_{x=0} \mathbf{d}y = \varphi_c (\alpha_3 - \alpha_1) \quad (4.115)$$

$$\int_0^{y_0} \left. \frac{\partial \varphi}{\partial y} \right|_{x=x_0} \mathbf{d}y = \varphi_c \beta_4 \quad (4.116)$$

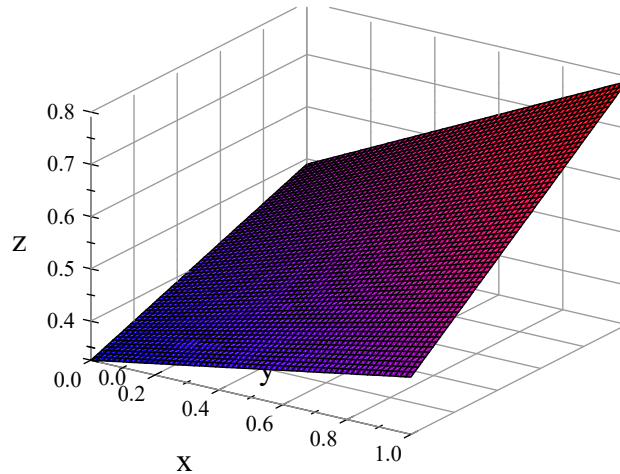


Figura 4-2: Gráfica de la solución particular a la ecuación de Laplace en dos dimensiones.

Finalmente, de acuerdo con las ecuaciones obtenidas en el análisis asintótico a orden uno (4.80-4.83) es requerido obtener una función de la forma  $\int x f'(y) dx$ , en donde  $f'(x)$  es la derivada de la ecuación 4.108, entonces, por cálculo integral las ecuaciones resultantes son:

$$\int_0^1 \eta \left. \frac{\partial u}{\partial \xi} \right|_{\xi=1,0} d\eta = \frac{1}{2} \alpha_4 - \frac{1}{3} \alpha_3 - \frac{1}{6} \alpha_1 + \frac{1}{3} \beta_4 \quad (4.117)$$

$$\int_0^1 \xi \left. \frac{\partial u}{\partial \eta} \right|_{\eta=1,0} d\xi = \frac{1}{6} \alpha_3 - \frac{1}{6} \alpha_1 + \frac{1}{3} \beta_4 \quad (4.118)$$

## Definición y entrenamiento de la red neuronal

En el capítulo 4, fueron obtenidas las ecuaciones para modelar el comportamiento de un bloque de matriz con razón de aspecto  $\varepsilon$ . La solución asintótica que se encontró implica que en cada orden se resuelva la ecuación de Laplace bajo condiciones de frontera lineales de Dirichlet, continuas a lo largo de la fractura. Este hecho se puede utilizar para entrenar una red neuronal en este tipo de problemas, para ello se utiliza un generador de números aleatorios y se define un conjunto de condiciones de frontera que junto a la solución analítica del capítulo 3 generen un conjunto de entradas (condiciones de frontera de Dirichlet) y salidas, como se muestra en la **Figura 5-1**.

En el presente capítulo se profundizará sobre la implementación de redes neuronales artificiales que serán utilizadas para el modelado de un bloque de matriz rodeado de cuatro fracturas.

Cabe mencionar que para el entrenamiento de la red neuronal fue empleado un conjunto de 100 000 datos generados aleatoriamente, de los cuales el 90 % fueron usados para entrenamiento y el 10 % para validación.

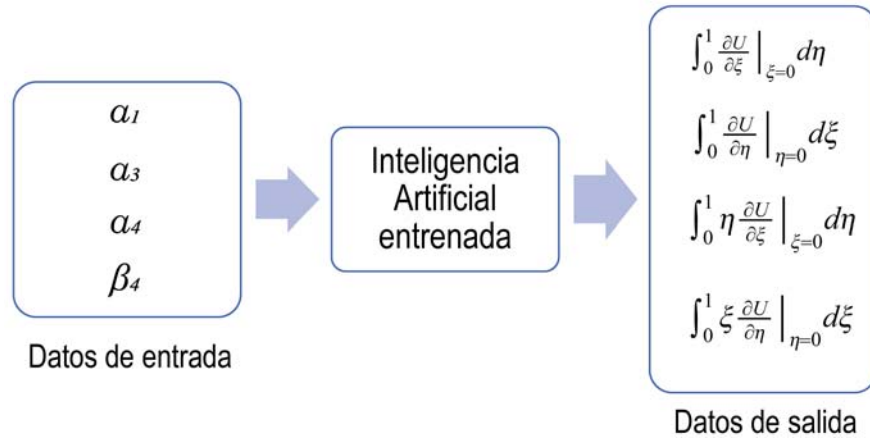


Figura 5-1: Entradas y salidas de la red neuronal.

En primer lugar, se explicará el algoritmo de generación de datos los cuales son la base del entrenamiento de las redes neuronales; el código está estructurado de acuerdo al diagrama de flujo (**Figura 5-2**), el cual inicia ingresando los parámetros inherentes al sistema (permeabilidad, factor de forma, abertura de fracturas, presión de inyección y presión de salida), posteriormente, de manera aleatoria se generan las condiciones de frontera correspondientes a los valores  $\alpha_{1,\dots,4}$  y  $\beta_{1,\dots,4}$  (en la entrada también se establece el número de set de datos que se desea generar), por ejemplo para este trabajo se generó un conjunto de cien mil datos de  $\alpha_{1,\dots,4}$  y  $\beta_{1,\dots,4}$ . Posteriormente, se realizan los cálculos analíticos empleando las ecuaciones obtenidas en el capítulo anterior, una vez finalizado este proceso de generación aleatoria, se generan tres bases de datos: la primera con los valores aleatoriamente generados, la segunda con los cálculos de los flujos y por último, los cálculos que resultan de las ecuaciones con la forma  $\int x f'(y) dx$ . El código numérico empleado para el desarrollo de este trabajo de tesis fue realizado en Fortran GNU y se usaron los algoritmos de Python para el entrenamiento.

El sistema a resolver es un sistema conservativo, por lo tanto, siguiendo el principio de conservación de masa así como continuidad en la presión (lo que genera condiciones similares a la Ley de Kirchhoff para el flujo de corriente en circuitos), y considerando que en equilibrio la

suma de los flujos en las fracturas es igual a cero, se tiene:

$$F_{tot} = \int_0^1 \frac{\partial U}{\partial \xi} \Big|_{\xi=0} d\eta + \int_0^1 \frac{\partial U}{\partial \xi} \Big|_{\xi=1} d\eta + \int_0^1 \frac{\partial U}{\partial \eta} \Big|_{\eta=0} d\xi + \int_0^1 \frac{\partial U}{\partial \eta} \Big|_{\eta=1} d\xi \quad (5.1)$$

$$F_{tot} = \varphi_c (\beta_4 - (\beta_4 - \alpha_3 + \alpha_4) - (\alpha_3 - \alpha_1) + (\alpha_4 - \alpha_1)) \quad (5.2)$$

$$F_{tot} = 0 \quad (5.3)$$

La ecuación 5.1 fue empleada como parámetro de validación.

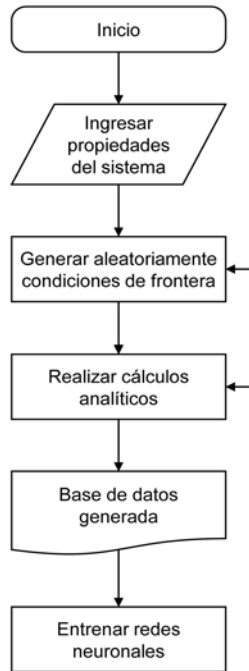


Figura 5-2: Diagrama de flujo general de algoritmo

## 5.1. Entrenamiento de redes neuronales

El entrenamiento de las redes neuronales fue realizado en el software “ffnet” desarrollado por Marek Wojciechowski (2016); con este software se logra un excelente rendimiento computacional implementando funciones básicas en Fortran 77, así como en Python. Además, se tiene la posibilidad de que las redes entrenadas sean exportadas al lenguaje Fortran, para compilar y

llamar desde distintos lenguajes de programación. Para optimizar el entrenamiento de las redes neuronales el número de procesadores a utilizar es elegible.

Ahora bien, debido a la simetría de los datos de salida se optó por utilizar una red neuronal en donde se obtengan dos salidas para calcular el comportamiento de los flujos y por otro lado otras dos salidas que corresponden a las ecuaciones de la forma  $\int x f'(y) dx$ , la red elegida tiene una arquitectura 4-12-8-4 (**Figura 5-3**) en donde el número 4 indica los datos de entrada los cuales corresponden a  $\alpha_1, \alpha_3, \alpha_4$  y  $\beta_4$ , asimismo cuenta con dos capas ocultas de 12 y 8 neuronas, finalmente cuatro nodos que son definidos para los datos de salida anteriormente explicados.

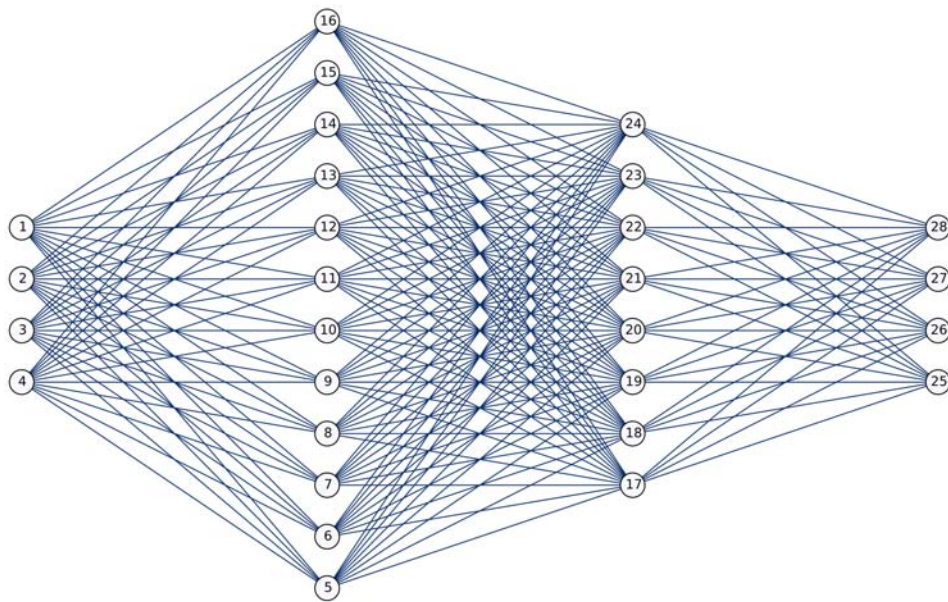


Figura 5-3: Arquitectura de red neuronal 4-12-8-4.

Cabe destacar que para definir la arquitectura de una red neuronal no existe un método o regla, por lo que ésta es obtenida de manera experimental y se elige debido al mejor ajuste a los datos y el menor error de entrenamiento y validación de la red.

El entrenamiento de la red neuronal resultó exitoso debido a que los errores de entrenamiento y validación fueron pequeños. Se realizaron más de 4700 iteraciones. En la **Figura 5-4** de acuerdo a la gráfica se observa que el error de entrenamiento es alrededor de  $4 \times 10^{-3}$ , por otro lado el error de validación es del orden de  $4 \times 10^{-4}$ .

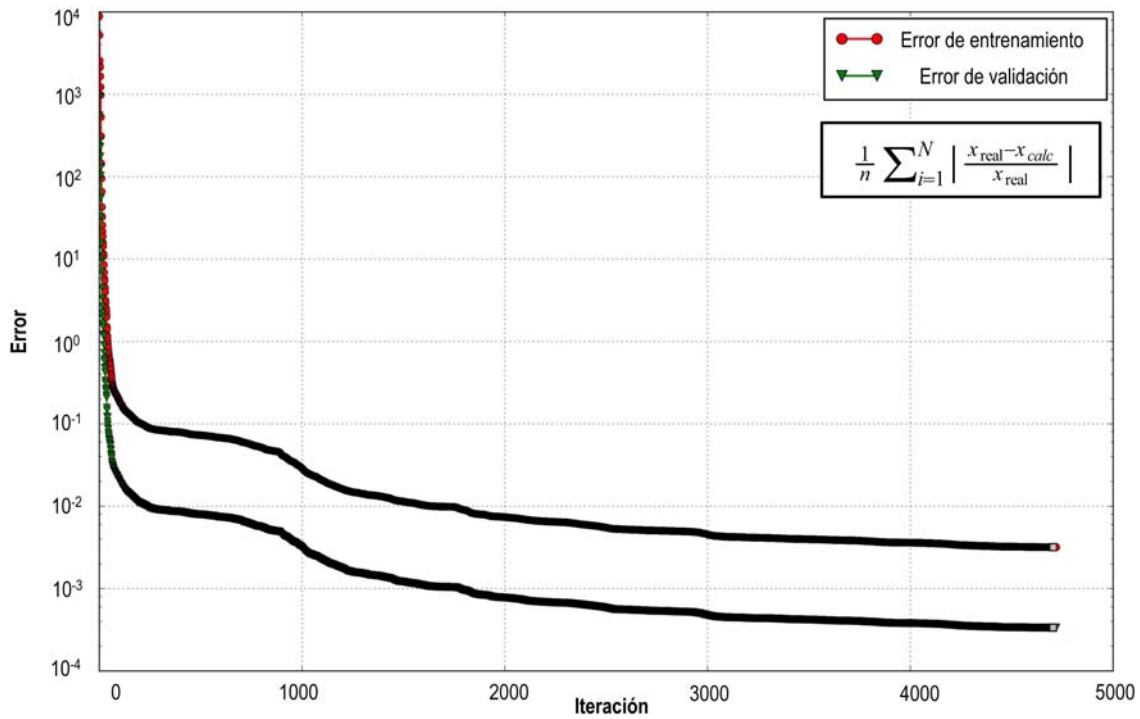


Figura 5-4: Gráfica de error de entrenamiento y validación de la red neuronal.

Con el fin de verificar el ajuste de los datos, se generaron gráficas de regresión lineal, las cuales son generadas para cada una de las salidas (**Figuras 5-5,5-8**). El valor de  $R^2$  para cada una de las salidas (**Tabla 5.1**) es muy cercano a 1, por lo tanto, los datos que se obtuvieron como resultado y, que serán explicados ampliamente en el capítulo siguiente, tienen un error pequeño con respecto a los cálculos analíticos.

	Salida	$R^2$
1	$\int_0^1 \frac{\partial U}{\partial \xi} \Big _{\xi=0} d\eta$	0.99999986
2	$\int_0^1 \frac{\partial U}{\partial \eta} \Big _{\eta=0} d\xi$	0.99999984
3	$\int_0^1 \eta \frac{\partial U}{\partial \xi} \Big _{\xi=0} d\eta$	0.99999983
4	$\int_0^1 \xi \frac{\partial U}{\partial \eta} \Big _{\eta=0} d\xi$	0.99999985

Tabla 5.1: Valores de regresión lineal para cada una de las salidas



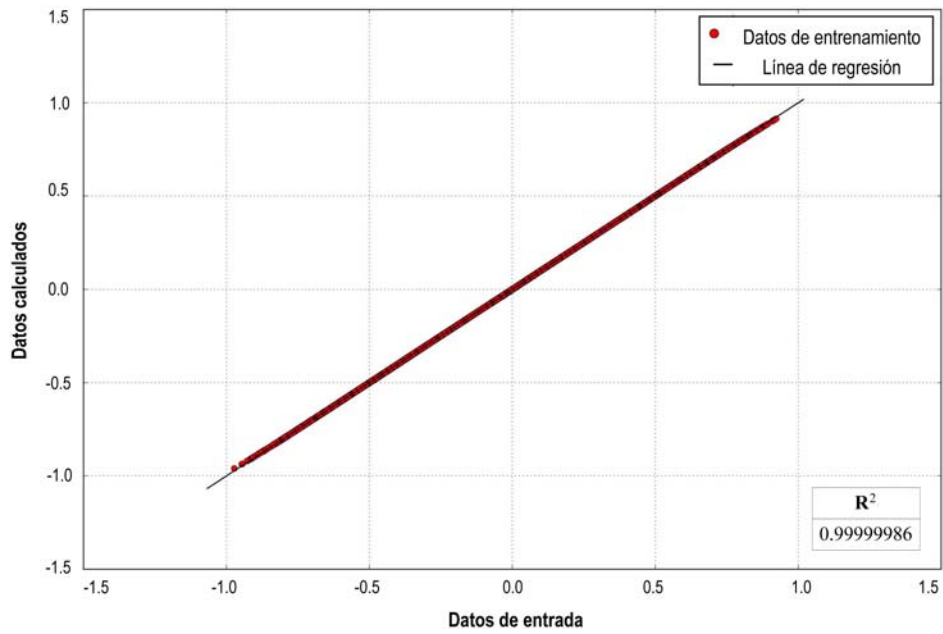


Figura 5-5: Ajuste de datos para la salida 1 de la red neuronal entrenada.

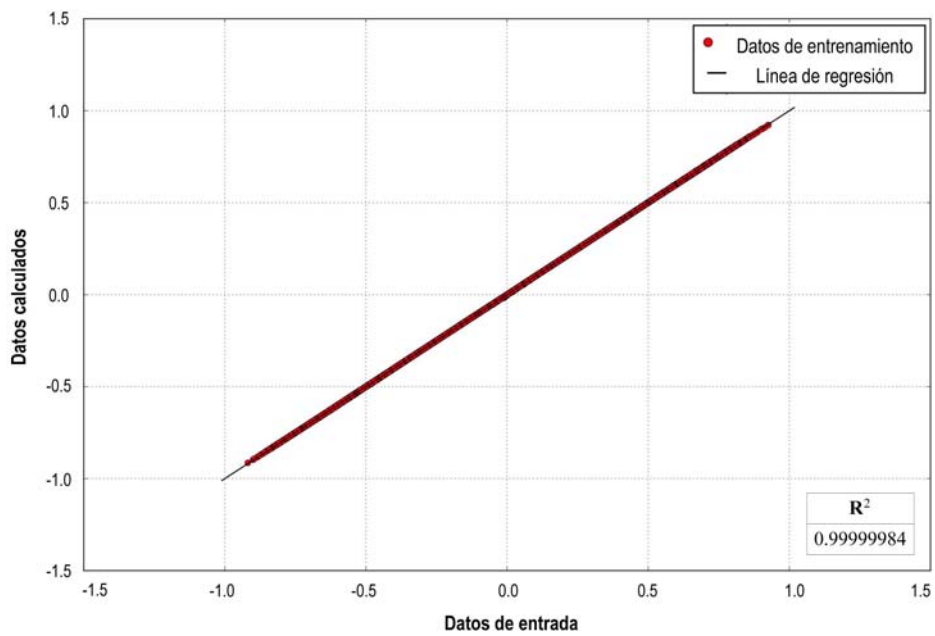


Figura 5-6: Ajuste de datos para la salida 2 de la red neuronal entrenada.

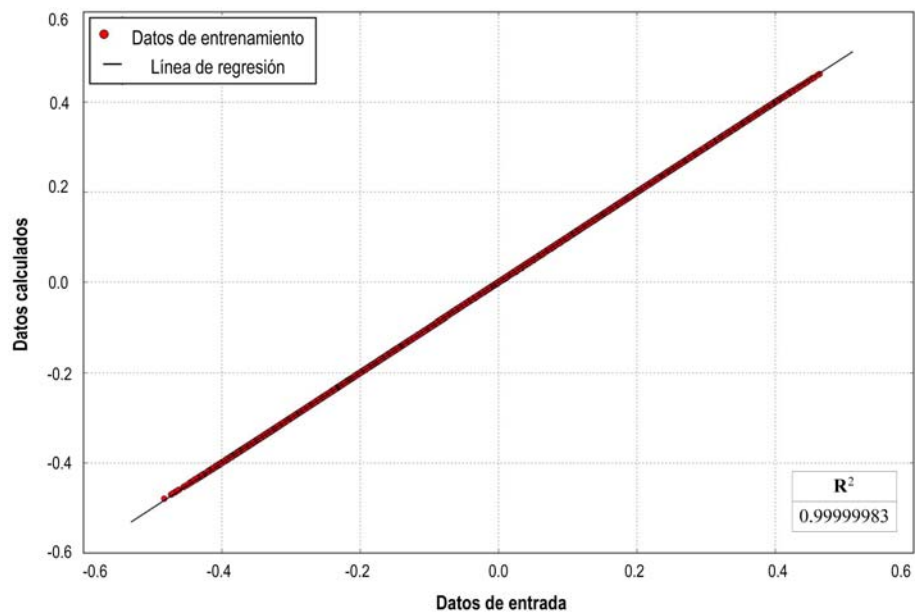


Figura 5-7: Ajuste de datos para la salida 3 de la red neuronal entrenada.

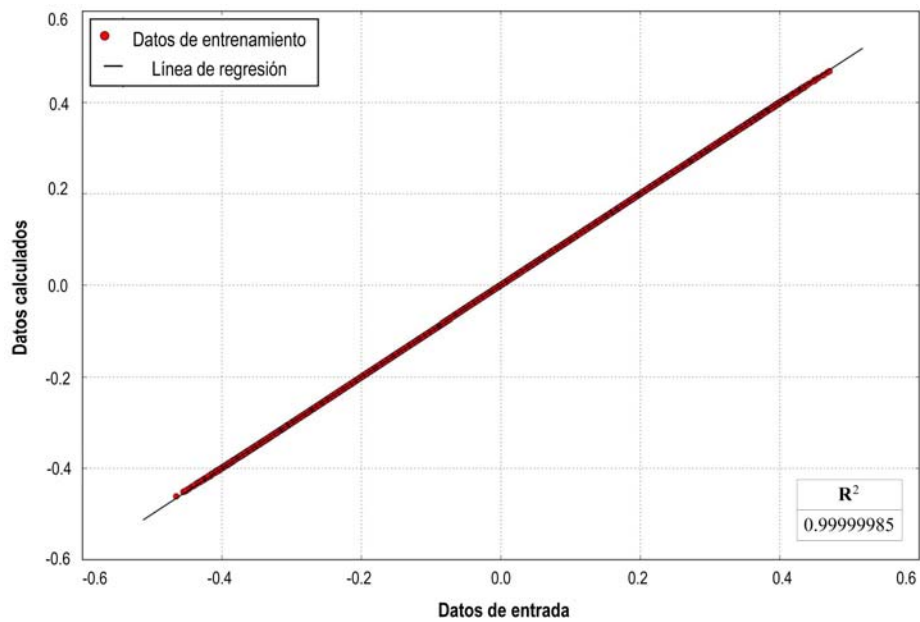


Figura 5-8: Ajuste de datos para la salida 4 de la red neuronal entrenada.

# Comparación entre modelos tradicionales y la red neuronal

En el capítulo 4 se obtuvieron los modelos matemáticos mediante un método asintótico posteriormente, en el capítulo 5 se desarrolló una aproximación a los modelos anteriormente mencionados y se propuso una aproximación a la solución mediante redes neuronales; en el presente capítulo se presentará un análisis comparativo de ambos métodos para modelar el comportamiento de un bloque poroso de matriz rodeado por cuatro fracturas.

## 6.1. Validación de la red neuronal

En primer lugar se validó el modelo de la red neuronal para la solución del bloque de matriz comparándolo contra la solución de un modelo de matriz porosa en dos dimensiones de geometría rectangular resuelto mediante una discretización de las ecuaciones en diferencia finitas utilizando un método de descomposición LU. Por otro lado, el modelo de IA también fue comparado con un modelo resuelto por Gauss-Seidel empleando sobre-relajación sucesiva, SOR por sus siglas en inglés para llegar a la solución. Para ello se programó el sistema en Fortran 95.

Se compararon las soluciones con dos grupos de condiciones de frontera, para el primer grupo considere los siguientes valores obtenidos del generador aleatorio:  $\alpha_1 = 0.32512924471$ ,  $\alpha_3 = 0.48976966821$ ,  $\alpha_4 = 0.43473339276$  y  $\beta_4 = 0.35508255364$ , los errores relativos entre los mode-

los y la red neuronal se muestra en las **Tablas 6.1** y **6.2** , cabe mencionar que para el modelo numérico de diferencias finitas y de Gauss-Seidel se definió una malla de  $200 \times 200$ .

<b>Modelo</b>	$\int_0^1 \frac{\partial U}{\partial \xi} \Big _{\xi=0} d\eta$	$\int_0^1 \frac{\partial U}{\partial \eta} \Big _{\eta=0} d\xi$
Solución Analítica	0.2048252131	0,2598614886
Diferencias finitas	0.2048252131	0.2598614885
Gauss-Seidel	0.2047880125	0.2598242879
Red neuronal	0.2046060234	0.2600066388
<b>Comparación de modelos</b>	<b>Error relativo</b>	
Solución analítica <i>vs</i> Diferencias finitas	0.0000000051 %	0.0000000005 %
Solución analítica <i>vs</i> Gauss-Seidel	0.0616133773 %	0.0485642682 %
Solución analítica <i>vs</i> Red neuronal	0.1070130385 %	0.0558567598 %

Tabla 6.1: Resultados obtenidos para el cálculo de flujos, comparando red neuronal con modelos tradicionales.

<b>Modelo</b>	$\int_0^1 \eta \frac{\partial U}{\partial \xi} \Big _{\xi=0} d\eta$	$\int_0^1 \xi \frac{\partial U}{\partial \eta} \Big _{\eta=0} d\xi$
Solución Analítica	0.1182827841	0.1458009218
Diferencias finitas	0.1182835856	0.1458017232
Gauss-Seidel	0.1182437344	0.1457618721
Red neuronal	0.1183334482	0.1458864505
<b>Comparación de modelos</b>	<b>Error relativo</b>	
Solución analítica <i>vs</i> Diferencias finitas	0.0006776186 %	0.0005496850 %
Solución analítica <i>vs</i> Gauss-Seidel	0.0428330261 %	0.0586613202 %
Solución analítica <i>vs</i> Red neuronal	0.0330138554 %	0.0267829132 %

Tabla 6.2: Resultados obtenidos de las integrales requeridas por la solución asintótica, comparando red neuronal con modelos tradicionales.

Ahora bien, para el segundo grupo de condiciones de frontera en las **Tablas 6.3** y **6.4** se muestran los resultados generados por el programa. Considere los siguientes valores:  $\alpha_1 = 0.8, \alpha_3 = 0.45, \alpha_4 = 0.45$  y  $\beta_4 = -0.35$ .

Por otra parte, de acuerdo a los resultados obtenidos y posterior al haber realizado un análisis comparativo, se valida el modelo de Inteligencia Artificial dada la buena aproximación resultante con los modelos anteriormente mencionados, con un error relativo promedio de 0.0556474059 % en comparación con el modelo de la solución analítica empleado para el entrenamiento de la Inteligencia Artificial.

<b>Modelo</b>	$\int_0^1 \frac{\partial U}{\partial \xi} \Big _{\xi=0} d\eta$	$\int_0^1 \frac{\partial U}{\partial \eta} \Big _{\eta=0} d\xi$
Solución Analítica	0.35	0.35
Diferencias finitas	0.35000000000129	0.35000000000190
Gauss-Seidel	0.35003720065003	0.35003720065002
Red neuronal	0,35020553119522	0,35013922854848
<b>Comparación de modelos</b>	<b>Error relativo</b>	
Solución analítica <i>vs</i> Diferencias finitas	0.0000000001 %	0.0000000002 %
Solución analítica <i>vs</i> Gauss-Seidel	0.0106287572 %	0.0106287571 %
Solución analítica <i>vs</i> Red neuronal	0.0587231986 %	0.0397795853 %

Tabla 6.3: Resultados obtenidos con otras condiciones de frontera para el cálculo de flujos, comparando red neuronal con modelos tradicionales.

<b>Modelo</b>	$\int_0^1 \eta \frac{\partial U}{\partial \xi} \Big _{\xi=0} d\eta$	$\int_0^1 \xi \frac{\partial U}{\partial \eta} \Big _{\eta=0} d\xi$
Solución Analítica	-0.175	-0.175
Diferencias finitas	-0.17500000000034	-0.17500000000089
Gauss-Seidel	-0.175039851210525	-0.175039851210518
Red neuronal	-0.175085226400468	-0.174941179842245
<b>Comparación de modelos</b>	<b>Error relativo</b>	
Solución analítica <i>vs</i> Diferencias finitas	0.0000000002 %	0.0000000005 %
Solución analítica <i>vs</i> Gauss-Seidel	0.0227721203 %	0.0227721203 %
Solución analítica <i>vs</i> Red neuronal	0.0487008003 %	0.0336115187 %

Tabla 6.4: Resultados obtenidos con otras condiciones de frontera de las integrales requeridas por la solución asintótica, comparando red neuronal con modelos tradicionales.

En la **Figura 6-1** se presenta una gráfica en dos dimensiones donde se observa la distribución de presión en el bloque de matriz. Se utilizó el método de Gauss-Seidel con SOR para la obtención de dicho gráfico, se omiten los resultados por otros métodos dada la similitud de la solución.

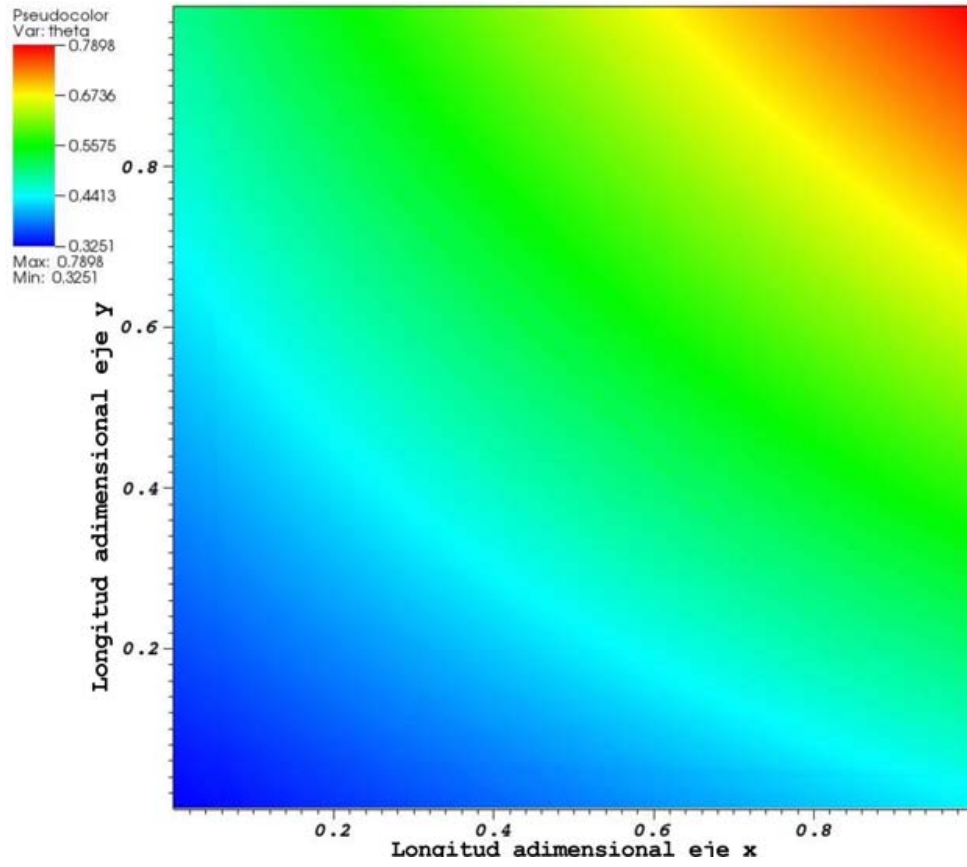


Figura 6-1: Gráfico en 2D de la solución del problema propuesto.

# Resultados de modelos tradicionales y red neuronal

Como se mencionó en el capítulo 5 se generaron dos bases de datos con las cuales fue realizado el entrenamiento de la Inteligencia Artificial, donde se obtuvieron errores de entrenamiento y validación en promedio del orden de  $10^{-3}$  y  $10^{-4}$ , respectivamente; en la **Figura 7-1** se esquematiza el proceso para el modelado de un bloque de matriz poroso rodeado de cuatro fracturas por el método analítico y el método de las redes neuronales.

De esta manera en la **Tabla 7.1** se enlistan los datos de entrada del programa:

<b>Datos entrada</b>
Longitud eje $x$ ( $a$ )
Longitud eje $y$ ( $b$ )
Permeabilidad de matriz ( $k$ )
Apertura de fractura ( $h_1$ )
Apertura de fractura ( $h_2$ )
Apertura de fractura ( $h_3$ )
Apertura de fractura ( $h_4$ )
Presión de inyección ( $p_{iny}$ )
Presión de salida ( $p_{out}$ )

Tabla 7.1: Datos de entrada de programa

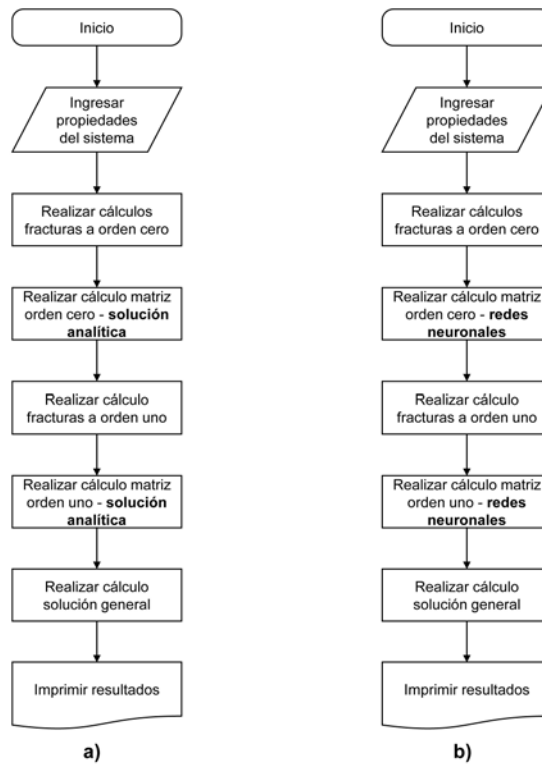


Figura 7-1: Diagramas de flujo de procesos a comparar, a)Modelo analítico, b)Modelo de redes neuronales

Una vez introducidos los datos se realiza el cálculo analítico de los flujos de fluido en las fracturas a orden cero, con este resultado se calcula la solución en matriz a orden cero, ya sea analíticamente o mediante la red neuronal entrenada. Posteriormente, se calcula el flujo en las fracturas a orden uno (analíticamente) y se obtiene la solución a orden uno en la matriz utilizando la solución analítica o la red neuronal.

Una vez que se cuentan con los datos de entrada necesarios para la red neuronal que son las condiciones de frontera y el factor de forma, se llama a la subrutina de la red neuronal la cual realiza los cálculos de flujos y referentes a las ecuaciones de la forma  $\int f'(x) dx$ . Por último, es generado un archivo con los datos obtenidos tanto analíticamente como por redes neuronales, y en este caso, se realizó el análisis de exactitud de los resultados.

Suponga los datos adimensionales de entrada de la **Tabla 7.2** para un caso donde existe simetría.



Datos entrada	Valor
Longitud eje $x$ ( $a$ )	1
Longitud eje $y$ ( $b$ )	1
Permeabilidad de matriz ( $k$ )	$1 \times 10^{-12}$
Apertura de fractura ( $h_1$ )	0.1
Apertura de fractura ( $h_2$ )	0.1
Apertura de fractura ( $h_3$ )	0.1
Apertura de fractura ( $h_4$ )	0.1
Presión de inyección ( $p_{iny}$ )	0.8
Presión de salida ( $p_{out}$ )	0.1

Tabla 7.2: Datos adimensionales de entrada de programa para un caso simétrico

De acuerdo con los datos obtenidos, el error relativo promedio entre los cálculos analíticos y los modelos desarrollados en cuestión de cálculo de flujos es de 0.0199600498 %, ahora bien, para el caso de los cálculos de las ecuaciones de la forma  $\int x f'(y) dx$  el error relativo promedio es 0.0213094267 %, lo cual demuestra una buena aproximación.

Asimismo, las presiones resultantes en las fronteras son los mostrados en la **Tabla 7.3**.

$p_3$	0.45
$p_4$	0.45

Tabla 7.3: Presiones que resultan de los cálculos en un caso de simetría

Las presiones resultante son el promedio de la presión de entrada y salida dado al equilibrio que se encuentra el sistema. Los gastos totales (orden cero y uno) en cada una de las fracturas se muestran en la **Tabla 7.4**, debido a que el sistema es simétrico se aprecia en los resultados generados que el gasto se distribuye de manera equitativa por las fracturas.

$q_1$	0.35
$q_2$	0.350000000262500
$q_3$	0.350000000262500
$q_4$	0.35

Tabla 7.4: Gastos totales que resultan de los cálculos en un caso de simetría

Por otro lado, se propone un caso general, en donde se tienen distintas aberturas de fractura así como un factor de forma de  $\varepsilon = 0.2105$ , considere los datos adimensionales de la **Tabla 7.5**. Debido a que los cálculos correspondientes al cálculo de flujos y ecuaciones de la forma  $\int x f'(y) dx$  no son dependientes del factor de forma  $\varepsilon$ , de la apertura de fracturas y

permeabilidad de la matriz, por lo que resultarán en los mismos cálculos de las **Tablas 6.3 y 6.4**.

En las **Tablas 7.6 y 7.7** se muestran las presiones que resultaron en las fronteras y los gastos totales en cada una de las fracturas, respectivamente. De acuerdo a los gastos que resultaron se observa que existe conservación de la materia independientemente de que el sistema no es geoméricamente simétrico respecto al factor de forma y apertura de fracturas.

Datos entrada	Valor
Longitud eje $x$ ( $a$ )	0.4
Longitud eje $y$ ( $b$ )	1.9
Permeabilidad de matriz ( $k$ )	$1 \times 10^{-12}$
Apertura de fractura ( $h_1$ )	0.1
Apertura de fractura ( $h_2$ )	0.2
Apertura de fractura ( $h_3$ )	0.3
Apertura de fractura ( $h_4$ )	0.4
Presión de inyección ( $p_{iny}$ )	0.8
Presión de salida ( $p_{out}$ )	0.1

Tabla 7.5: Datos adimensionales de entrada de programa para un caso general

$p_3$	0.45
$p_4$	0.45

Tabla 7.6: Presiones que resultan de los cálculos de un caso general

$q_1$	0.12799999997273
$q_2$	0.103384615359728
$q_3$	0.103384615359728
$q_4$	0.127999999982979

Tabla 7.7: Gastos totales que resultan de los cálculos de un caso general

Continuando con el análisis de resultados, se realizó una medición del tiempo de cómputo que fue empleado por los distintos modelos que fueron explicados en el capítulo anterior, resultando en los tiempos de la **Tabla 7.8**, al realizar una relación entre la red neuronal, diferencias finitas y el modelo empleando Gauss-Seidel como método de solución, el modelo de inteligencia artificial es 4 638 020.41 y 458 034.99 veces más rápido que los respectivos métodos de solución, asimismo en la **Figura 7-2** se muestra una representación gráfica sobre esta relación de tiempo de cómputo.

Modelo	Tiempo de cómputo [s]
Diferencias finitas	5.7261
Gauss-Seidel	0.56549
Red neuronal	$1.23 \times 10^{-6}$

Tabla 7.8: Tiempo de cómputo para los distintos métodos de solución

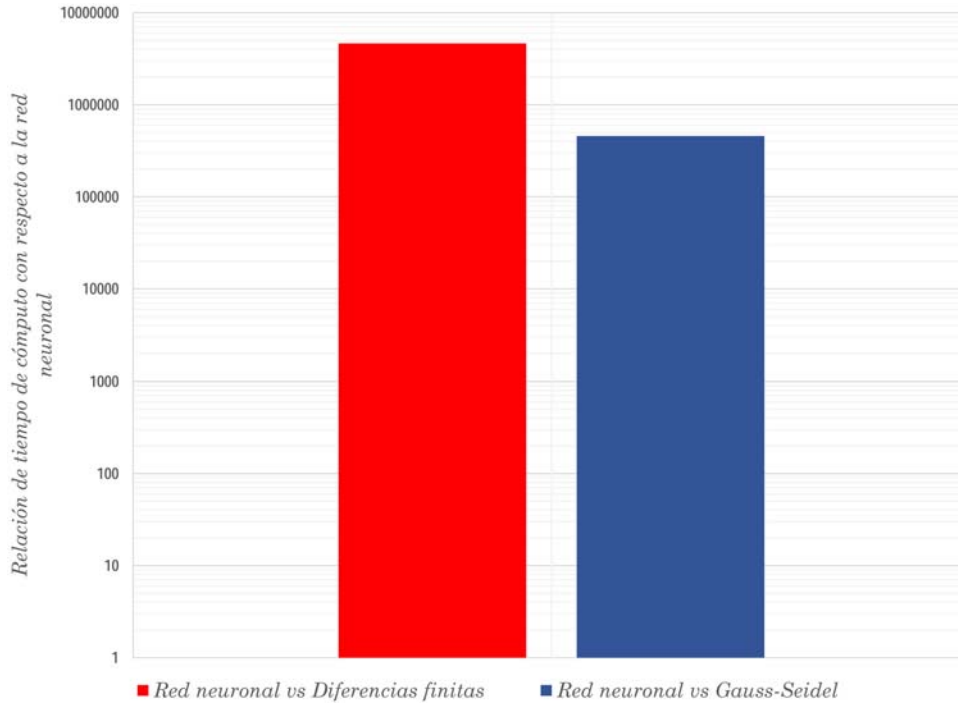


Figura 7-2: Comparación de tiempo de cómputo con respecto a la red neuronal.

Como último parámetro de comparación entre la red neuronal y los modelos anteriormente expuestos, se determinó la cantidad de memoria RAM que fue utilizada en cada uno de los cálculos, se refleja que el modelo de diferencias finitas es el que más memoria consume, mientras que la red neuronal emplea una pequeña cantidad de memoria RAM (**Tabla 7.9**).

Cabe mencionar que las mediciones de los recursos computacionales que fueron realizadas para el tiempo de cómputo y la memoria RAM fue llevada a cabo en una estación de trabajo que cuenta con 24 núcleos para procesamiento de información y 48 GB de memoria RAM.

<b>Modelo</b>	<b>Memoria RAM</b>
Diferencias finitas	12.1 <i>GB</i>
Gauss-Seidel	288 <i>KB</i>
Red neuronal	104 <i>KB</i>

Tabla 7.9: Memoria RAM usada por los métodos numéricos y la red neuronal.

## Conclusiones

La Inteligencia Artificial en el sector energético, especialmente en la ingeniería de yacimientos tiene un gran potencial en investigación donde existen muchas áreas de oportunidad. En el presente trabajo se abordó en la simulación numérica de yacimientos, primordialmente como una prueba de principio para determinar la viabilidad del uso de IA.

- Se demostró que es posible emplear inteligencia artificial entrenada a partir de principios de la teoría del transporte que modelen los bloques de matriz, para optimizar el tiempo de cálculo y recursos computacionales, lo cual se formuló como hipótesis de este trabajo.
- Los resultados son muy prometedores, ya que el análisis sobre el tiempo de cómputo usando Inteligencia Artificial es del orden de millones y cientos de miles de veces más rápido que los métodos numéricos tradicionales los cuales son empleados para la solución de problemas en ecuaciones diferenciales principalmente.
- En cuanto al consumo de memoria RAM, la red neuronal entrenada para la solución del problema ocupó una menor memoria con respecto a los otros métodos con los que se validó la red neuronal. Los análisis anteriores demuestran la viabilidad de la implementación para la disminución considerable del uso de los recursos computacionales haciendo más eficientes y rentables los procesos en la ingeniería de yacimientos.
- Por otro lado, con respecto al error y exactitud del modelo se presenta una muy buena aproximación al educar una red neuronal para un conjunto de datos de los cálculos nece-

sarios que son: los flujos (ecuaciones de la forma  $\int f'(x) dx$ ) y las ecuaciones de la forma  $\int x f'(y) dx$ , sin embargo, se espera que se obtengan mejores resultados en cuanto a error de entrenamiento y validación así como en exactitud, si se entrenara una red neuronal independiente para cada una de las salidas de los resultados anteriormente mencionados. Es posible emplear distintos softwares especializados sobre inteligencia artificial, por ejemplo, Anaconda que trabaja con las librerías de Phyton o también en el lenguaje de programación R, ambos son ampliamente utilizados por científicos de datos.

- En este trabajo solamente se realizaron los modelos para el comportamiento de un bloque de matriz poroso rodeado de cuatro fracturas en estado estacionario, no obstante, es factible extender el modelo a múltiples bloques de matriz asociados a una red de fracturas. Como un esfuerzo adicional se propone realizar esfuerzos para una solución dependiente del tiempo utilizando redes neuronales. Finalmente es posible la implementación de fenómenos no considerados como lo son los fenómenos termodinámicos y químicos.
- Se debe tomar con precaución el uso de IA, Ben-David *et al.* encontraron un problema que no tiene respuesta matemática porque está vinculada a paradojas lógicas descubiertas por el matemático austriaco Kurt Gödel en la década de 1930, las cuales no pueden resolverse utilizando matemáticas ordinarias. Los matemáticos que realizaron trabajos en un problema de machine learning y mostraron que la “capacidad de aprendizaje”, está vinculada a una paradoja conocida como la hipótesis del continuo, el vínculo con el problema es que una pequeña muestra puede ser suficiente para hacer el entrenamiento de la Inteligencia Artificial, sin embargo no existe un tamaño finito que sea idóneo para que el entrenamiento sea considerado perfecto.
- Finalmente, es necesario incursionar en mayor medida en el campo de Inteligencia Artificial aplicada a la ingeniería de yacimientos, dado que para la elaboración de este trabajo no se detectaron publicaciones relacionadas con la simulación numérica de yacimientos; sin embargo en otras áreas de estudio de ingeniería petrolera ya son empleadas distintas formas de IA y manejo de grandes datos (Big Data).

# Bibliografía

- [1] Barenblatt, G.I., and Zheltov, Y.P., 1960, On Fundamental Equations of Flow of Homogeneous Liquids in Naturally Fractured Rocks, Dokl. Akad. Nauk. URSS, Vol. 132, No. 3, p. 545-548.
- [2] Ben-David, Shai, *et al.*, 2019, Learnability can be undecidable. Nature Machine Intelligence, vol. 1, no 1, p. 44.
- [3] Bender, Carl, *et al.*, 1978, Advanced Mathematical Methods for Scientists and Engineers, McGraw-Hill Book Company
- [4] Chen, Z.-X., 1990, Analytical Solutions for Double-Porosity, Double-Permeability and Layered Systems, Journal of Petroleum Science and Engineering, Vol. 5, p. 1-24.
- [5] Centro Nacional de Información de Hidrocarburos, Comisión Nacional de Hidrocarburos, <https://hidrocarburos.gob.mx/>, consultado: abril 2019.
- [6] Far, M.E., Sayers, C.M., Thomsen, L. et al. 2013. Seismic Characterization of Naturally Fractured Reservoirs Using Amplitude versus Offset and Azimuth Analysis. Geophysical Prospecting 61 (2): 427–47. <http://dx.doi.org/10.1111/1365-2478.12011>.
- [7] Frank Rosenblatt, 1957, The Perceptron A Perceiving and Recognizing Automaton. Reporte 85-460-1, Cornell Aeronautical Laboratory, Buffalo, New York.
- [8] Haykin, Simon. 2009. Neural networks and learning machines, tercera edición. Upper Saddle River, New Jersey: Pearson Prentice Hall

- [9] Ilya Sutskever, James Martens, George Dahl, Geoffrey Hinton. 2013. On the importance of initialization and momentum in deep learning, Proceedings of the 30th International Conference on Machine Learning, PMLR 28(3):1139-1147.
- [10] Karode, S.K. 2001, Laminar flow in channels with porous walls, revisited. Journal of Membrane Science, Ottawa, Canada. 237-241. Junio.
- [11] Kazemi, H., 1969, Pressure Transient Analysis of Naturally Fractured Reservoirs with Uniform Fracture Distribution, Society of Petroleum Engineers Journal, SPE 2156A, p. 451-462.
- [12] Lee, C.H., Yu, J.-L., and Hwung, H.-H. 1993. Fluid Flow and Connectivity in Fractured Rock. Water Resources Management 7 (2): 169–184. <http://dx.doi.org/10.1007/BF00872480>.
- [13] Luna Barrera, D. 2018. Modelo de pérdidas de circulación de un pozo en un yacimiento fracturado considerando interacción matriz-fractura. Tesis de Maestría en Ingeniería, Universidad Nacional Autónoma de México, Ciudad de México, México (Octubre 2018).
- [14] Marek Wojciechowski, Solving differential equations by means of feed-forward artificial neural networks, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7267(1): 187-195, 2012
- [15] Marek Wojciechowski, Application of artificial neural network in soil parameter identification for deep excavation numerical model, Computer Assisted Mechanics and Engineering Sciences, 18(4): 303-311, 2011
- [16] Marek Wojciechowski, Feed-forward neural network for python, Technical University of Łódź, <http://ffnet.sourceforge.net/>, 2005-2016, Łódź, Polonia
- [17] Mattax, C.C., and Dalton, R.L., 1990, Reservoir Simulation Monograph Volume 13, Society of Petroleum Engineers, p. 184
- [18] J. McCarthy, Marvin L. Minsky, Nathaniel Rochester, and Claude E. Shannon, 1955, A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence. <https://doi.org/10.1609/aimag.v27i4.1904>



- [19] Nelson, R. 2001. *Geologic Analysis of Naturally Fractured Reservoirs*. Gulf Professional Publishing.
- [20] Nilsson, N. J.. 1998, *The Mathematical Foundations of Learning Machines*, San Francisco: Morgan Kaufmann.
- [21] Oda, M. 1985. Permeability Tensor for Discontinuous Rock Masses. *Geotechnique*, 35 (4), 483-495. <http://dx.doi.org/10.1680/geot.1985.35.4.483>.
- [22] Villareal Sánchez, G. 2017. Modelo para cuantificar las pérdidas de circulación para fluidos no newtonianos en yacimientos naturalmente fracturados. Tesis de Maestría en Ingeniería, Universidad Nacional Autónoma de México, Ciudad de México, México (Marzo 2017).
- [23] Warren, J.E., and Root, P.J., 1963, The Behavior of Naturally Fractured Reservoirs, *Society of Petroleum Engineers Journal*, SPE 426, p. 245-255.

## Ecuación de Karode

Considere una fractura entre dos bloques de matriz poroso, con una geometría rectangular, paralelos y de dimensiones  $L \times B \times W$ , separadas por una distancia de abertura de fractura  $d_i = 2h$ , donde el eje  $z$  es paralelo a la fractura.

Cuando las paredes o caras de la fractura son impermeables la relación entre el gasto y el gradiente de presión es:

$$q = -\frac{2}{3} \frac{h^3 W}{\mu} \frac{\partial p}{\partial z} \quad (\text{A.1})$$

donde,

$$\begin{aligned} q &\triangleq \text{gasto a través de fractura} \\ p &\triangleq \text{presión} \\ \mu &\triangleq \text{viscosidad dinámica del fluido} \end{aligned}$$

Si las caras de la fractura son impermeables,  $q$  es independiente de  $z$  y el gradiente de presión es constante. El sistema está gobernado por las ecuaciones de conservación de masa y de Navier-Stokes, suponiendo que el gradiente de presión se presenta en la dirección  $z$  y el único componente no cero de la velocidad también es en la dirección  $z$ , la solución se conoce como la ecuación de Hagen-Poiseuille (Karode, 2001).

Ahora bien, para paredes permeables el gasto cambia a lo largo de la longitud de la fractura, debido a las pérdidas a través de las paredes; realizando un balance del gasto en un diferencial de  $z$  con base en la **Figura A-1**.

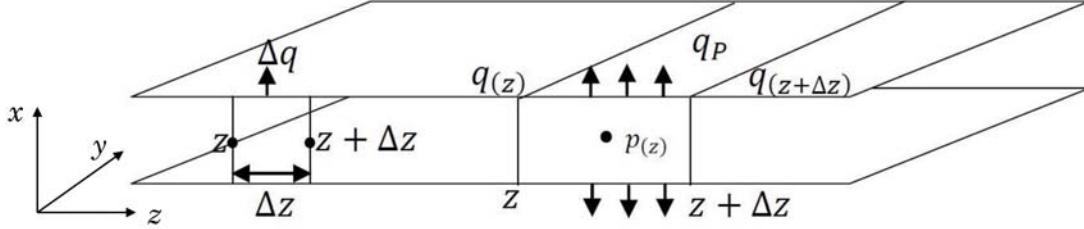


Figura A-1: Balance del gasto en un diferencial de  $z$ . (Modificado de Luna Barrera, 2019)

En primer lugar, el gasto que entra en  $z$

$$q(z) = -\frac{2 h^3 W}{3 \mu} \left. \frac{\partial p}{\partial z} \right|_z \quad (\text{A.2})$$

por otro lado, el gasto que sale en  $z + \Delta z$

$$q(z + \Delta z) = -\frac{2 h^3 W}{3 \mu} \left. \frac{\partial p}{\partial z} \right|_{z + \Delta z} \quad (\text{A.3})$$

o bien,

$$\frac{q(z + \Delta z) - q(z)}{\Delta z} = -\frac{2 h^3 W}{3 \mu} \frac{\left( \left. \frac{\partial p}{\partial z} \right|_{z + \Delta z} - \left. \frac{\partial p}{\partial z} \right|_z \right)}{\Delta z} \quad (\text{A.4})$$

calculando el límite cuando  $\Delta z \rightarrow 0$

$$\lim_{\Delta z \rightarrow 0} \left( \frac{q(z + \Delta z) - q(z)}{\Delta z} \right) = -\frac{2 h^3 W}{3 \mu} \lim_{\Delta z \rightarrow 0} \left( \frac{\left( \left. \frac{\partial p}{\partial z} \right|_{z + \Delta z} - \left. \frac{\partial p}{\partial z} \right|_z \right)}{\Delta z} \right) \quad (\text{A.5})$$

Se considera que la diferencia de gastos con respecto al diferencial de  $z$  es proporcional al gasto del fluido que entra en los dos bloques de matriz:

$$\frac{q(z+\Delta z) - q(z)}{\Delta z} = -2 \frac{q_{exit}}{\Delta z} \quad (\text{A.6})$$

donde el gasto volumétrico ( $q_{exit}$ ) que entra por la cara  $W$  es:

$$q_{exit} = u_{exit} \Delta z W \quad (\text{A.7})$$

donde  $u_{exit}$  es la velocidad del fluido que interacciona en la matriz porosa.

Sustituyendo la ec. A.7 en la ec. A.6

$$\frac{q(z+\Delta z) - q(z)}{\Delta z} = -2u_{exit}W$$

Calculando el límite cuando  $\Delta z \rightarrow 0$

$$\lim_{\Delta z \rightarrow 0} \left( \frac{q(z+\Delta z) - q(z)}{\Delta z} \right) = -2u_{exit}W$$

Finalmente se tiene:

$$\frac{dq}{dz} = -2u_{exit}W \quad (\text{A.8})$$

además

$$\frac{dq}{dz} = -\frac{2}{3} \frac{h^3 W}{\mu} \frac{d^2 p}{dz^2} \quad (\text{A.9})$$

o bien

$$\frac{dq}{dz} = -\frac{2}{3} \frac{h^3 W}{\mu} \frac{d^2 p}{dz^2} = -2u_{exit}W \quad (\text{A.10})$$