



Escuela de Ciencias e
Ingeniería

SOLUCIÓN DE LOS PATRONES DE TURING EMPLEANDO EL MÉTODO DE LOS ELEMENTOS FINITOS

Presentado para obtener el título de

Profesional en Matemáticas Aplicadas y Ciencias de la Computación

Juan Esteban Beltrán Ríos

Dirección:

José Alejandro Guerrero Vargas, PhD

Universidad del Rosario
Escuela de Ciencias e Ingeniería

2025

DEDICATORIA

ii

Dedico este trabajo principalmente a Dios por la oportunidad que me ha permitido tener la sabiduría, el entendimiento y la fortaleza para culminar mis estudios junto a esta investigación.

A mi padre por estar siempre animándome a seguir, ofrecer siempre su apoyo y estar siempre dispuesto a escuchar cuando lo necesitaba. A mi madre por los valores que me inculco en el tiempo que estuvo a mi lado y siempre mostrando que creía en mí, siendo una inspiración para mejorar cada día.

A mis hermanas, amigos y toda mi familia que siempre creyeron en mí y me motivaban a no rendirme en cada momento. Por todos los momentos que pude compartir con cada uno de ellos, en los que surgieron recuerdos que forjaron la persona que soy ahora.

A aquellos maestros que no solo me enseñaban materias, sino que buscaban dejar una marca en cada uno de sus alumnos para aportar en su crecimiento como persona, mostrándose dispuestos a apoyarme en los momentos más complicados. Ellos me mostraron el gran valor del conocimiento, sin dejar de lado el valor de cada persona.

AGRADECIMIENTOS

iii

Agradezco a Dios por la vida el crecimiento que me ha dado y por haberme puesto en este campo, permitiéndome ser sabio para entender y comprender todo lo visto durante mi carrera.

A mi padre por la paciencia y el esfuerzo que hizo para darme todo lo que necesite para poder llegar a realizar este trabajo y culminar mis estudios exitosamente.

A mi profesor José Alejandro Guerrero por su paciencia, esmero y profesionalismo para compartir sus conocimientos y poner a disposición todos sus medios para la realización de este trabajo.

Resumen

En este trabajo se muestra la solución numérica de los patrones de Turing, tratando específicamente el caso del modelo de Schnakenberg. Estos patrones surgen a partir de las inestabilidades generadas por la interacción entre químicos difusores y reactivos en sistemas inicialmente homogéneos, siendo fundamentales para comprender procesos como la morfogénesis y autoorganización biológica. Por la complejidad no lineal de las ecuaciones que los describen, se requiere de métodos numéricos que aproximen su solución.

La investigación realizada propone una implementación del modelo de Schnakenberg en el software libre FEniCSx, partiendo de la formulación variacional de las ecuaciones y su discretización mediante elementos finitos. Para esto se definen condiciones de frontera de tipo Neumann homogéneas y un esquema temporal implícito resuelto por el método de Newton-Raphson. La simulación se realiza sobre un dominio cuadrado unitario con diferentes combinaciones de parámetros que controlan la difusión, la intensidad de reacción y el tiempo de integración, con lo que se reproducen distintas configuraciones espacio-temporales de los patrones. Los resultados obtenidos se visualizan con archivos XDMF y el software ParaView, mostrando la evolución dinámica de las concentraciones.

Los patrones generados son coherentes con los realizados en otros estudios, confirmando la validez del enfoque numérico. Se observa como el contraste difusivo entre las especies hace de generador de inestabilidad y el parámetro de reacción γ controla la frecuencia y el número de estructuras formadas. Es así que, el modelo reproduce desde manchas aisladas hasta configuraciones reticulares estables, dependiendo de los parámetros empleados.

El correcto funcionamiento de FEniCSx demuestra que es una alternativa poderosa y accesible comparado a herramientas de código cerrado, facilitando el avance de la modelación matemática en biología, ingeniería y materiales. De esta forma, se dan las bases para futuras investigaciones orientadas a dominios más complejos, análisis de bifurcaciones y optimización de modelos reacción-difusión.

This paper presents the numerical solution of Turing, specifically addressing the Schnakenberg model. These patterns arise from instabilities generated by the interaction between diffusing and reacting chemicals in initially homogeneous systems, and are fundamental to understanding processes such as morphogenesis and biological self-organization. Due to the nonlinear complexity of the equations that describe them, numerical methods are required to approximate their solution.

This research proposes an implementation of the Schnakenberg model in the free software FEniCSx, starting from the variational formulation of the equations and their discretization using finite elements. For this purpose, homogeneous Neumann boundary conditions and an implicit time scheme solved by the Newton-Raphson method are defined. The simulation is performed on a unitary square domain with different combinations of parameters that control diffusion, reaction intensity, and integration time, thereby reproducing different spatiotemporal configurations of the patterns. The results obtained are visualized using XDMF files and ParaView software, showing the dynamic evolution of concentrations.

The generated patterns are consistent with those of other studies, confirming the validity of the numerical approach. It is observed how the diffusion contrast between species acts as a generator of instability, and the reaction parameter γ controls the frequency and number of structures formed. Thus, the model reproduces everything from isolated spots to stable lattice configurations, depending on the parameters used.

The correct operation of FEniCSx demonstrates that it is a powerful and accessible alternative to closed-source tools, facilitating the advancement of mathematical modeling in biology, engineering, and materials. This lays the groundwork for future research focused on more complex domains, bifurcation analysis, and optimization of reaction-diffusion models.

Tabla de contenido

Capítulo 1 INTRODUCCIÓN	1
Capítulo 2 OBJETIVOS	4
1. Objetivo general	4
2. Objetivos específicos	4
3. Modificaciones	4
Capítulo 3 PROBLEMA Y JUSTIFICACIÓN.....	5
Capítulo 4 MARCO TEÓRICO Y ESTADO DEL ARTE	7
1. Conceptos fundamentales	7
2. Estudios descriptivos y experimentales de patrones de Turing	7
3. Sistemas reacción-difusión	9
4. Selección de parámetros	10
5. Método de los elementos finitos.....	12
6. Investigaciones previas	13
Capítulo 5 METODOLOGÍA	16
1. Implementación numérica.....	16
2. Implementación Numérica en FeniCS	18
Capítulo 6 RESULTADOS Y DISCUSIÓN	24
1. Caso 1	24
2. Caso 2	25
3. Caso 3	26
4. Caso 4	27
Capítulo 7 CONCLUSIONES	30
1. Conclusión general	30
2. Limitaciones del estudio	30
3. Líneas futuras de investigación	31
Referencias	32

LISTA DE TABLAS

Tabla 1 – Parámetros para simulaciones 11

Figura 1 Función de marcado de bordes	18
Figura 2 Funciones de base tipo Langragiano	19
Figura 3 Creación del espacio mixto.....	19
Figura 4 Función para hallar elementos en la frontera.....	19
Figura 5 Configuración y uso de las entidades	20
Figura 6 Definición de las funciones de prueba y solución	20
Figura 7 Iniciación de la perturbación	21
Figura 8 Funciones para inicializar la solución	21
Figura 9 Formulación débil en código para FeniCSx	21
Figura 10 Definición del tipo de frontera.....	22
Figura 11 Definición y solución del problema	22
Figura 12 Creación de los archivos xdmf y bucle temporal.....	23
Figura 13 Concentraciones de u y v en el caso 1. Tiempo adimensional [4] para a=0, b=5, c=10-50.....	24
Figura 14 Comparación con la literatura Caso 1. A. Resultado de Vanegas, Landinez y Garzón [4] b. Resultado de este estudio	25
Figura 15 Concentraciones de u y v en el caso 2. Tiempo adimensional [4] para a=0, b=10, c=20	25
Figura 16 Comparación con la literatura Caso 2. A. Resultado de Vanegas, Landinez y Garzón [4] b. Resultado de este estudio	26
Figura 17 Concentraciones de u y v en el caso 3. Tiempo adimensional [15] para a=0, b=5, c=10	26
Figura 18 Comparación con la literatura Caso 3. A. Resultado de Vanegas, Gonzales y Garzón [15] b. Resultado de este estudio.....	27

Figura 19 Concentraciones de u y v en el caso 4. Tiempo adimensional [29] para a=0, b=8, c=15	ix
27	
Figura 20 Comparación con la literatura Caso 4. A. Resultado de Garzón, Galeano y Mantilla [29] b. Resultado de este estudio.....	28

Capítulo 1

INTRODUCCIÓN

Los patrones espacio-temporales son características fundamentales de los sistemas biológicos ya que pueden ser determinantes tanto en el desarrollo y la función del organismo como en su supervivencia y adaptación al medio ambiente [1]. Por ejemplo, las rayas de las cebras que les permite camuflarse con la hierba de la jungla, o la coloración y patrones de las alas de las mariposas para comunicar que son venosas y evitar que sean depredadas [2]. Desde el punto de vista biológico, los patrones formados por estos modelos pueden ser clasificados en dos categorías: patrones químicos y patrones de movimiento celular [3]. En el caso de los patrones químicos están los sistemas reacción-difusión que describen las interacciones moleculares que generan patrones complejos en el espacio y/o el tiempo, debido a la presencia de términos de transporte, síntesis y degradación que dependen de todas las sustancias presentes en el dominio de análisis [4].

Todo esto fue descrito de forma numérica por el matemático británico Alan Turing, quien en 1952 propuso un modelo revolucionario para los científicos experimentales en biología teórica en su artículo *The chemical basis of morphogenesis*, donde planteó la aparición espontánea de patrones en sistemas inicialmente homogéneos como resultado de procesos de reacción y difusión [5] [6]. Este establece las condiciones para las cuales un proceso reactivo en equilibrio puede ser inestable ante la presencia de términos difusivos, generando unos patrones espaciales heterogéneos, denominados inestabilidades impulsadas por difusión o inestabilidades de Turing [7]. Estos generan los llamados patrones de Turing, y son una pieza clave en el entendimiento de la autoorganización en sistemas biológicos y otras disciplinas.

No obstante, para poder comprender la morfología final, la estabilidad y la dinámica en geometrías complejas, la resolución de las ecuaciones que modelan estos patrones es necesaria. Esto plantea un desafío significativo debido a la naturaleza no lineal, acoplada y dependiente del espacio y tiempo de estos modelos, que requiere de la aplicación de herramientas numéricas que sean precisas y estables. La no linealidad,

surge por las tasas de reacción que no son proporcionales a las concentraciones, haciendo que pequeñas variaciones produzcan respuestas no proporcionales o cambios cualitativos en el comportamiento del sistema [8]. Además, los modelos son acoplados porque cada especie química influye en la evolución de las demás mediante términos reactivos y difusivos, formando un sistema interdependiente en el que ninguna ecuación puede resolverse de manera aislada. Finalmente, estos procesos son espacio-temporales ya que las concentraciones varían simultáneamente en el tiempo y en cada punto del dominio, lo que implica la necesidad de trabajar con sistemas que evolucionan a lo largo del tiempo.

Para solucionar esto, se tiene el método de los elementos finitos (MEF), ya que permite abordar problemas en geometrías complejas gracias a la discretización del espacio en pequeños puntos llamados nodos [9]. Al hacer esto, se consigue la representación de un fenómeno físico a través de un problema numérico de dimensión finita con mallas simples y elementos adecuados [10]. En la actualidad existen simulaciones que, con este método, dan solución a estos patrones; sin embargo, muchas se encuentran en softwares de código cerrado o de licencia [1] [4] [5] [11], que debido a su alto costo de uso dificulta el avance de sus estudios y posibles aplicaciones.

Metodológicamente, este trabajo se sustenta en la formulación variacional de las ecuaciones que describen los patrones de Turing y su implementación numérica mediante herramientas de código abierto como el software FEniCS. La estrategia incluye la discretización espacial, junto con técnicas de linealización que permiten resolver el sistema no lineal acoplado asociado al modelo de Schnakenberg. Las simulaciones se validarán a partir de configuraciones de prueba documentadas en la literatura, comparando la morfología de los patrones obtenidos, así como la reproducción de características esperadas bajo diferentes valores de los parámetros. La elección de esta metodología permite un enfoque riguroso y flexible que combina el análisis matemático con la simulación computacional.

La importancia de este trabajo radica en su potencial para aportar herramientas que satisfagan los requisitos para el correcto análisis de estos patrones, permitiendo que puedan ser empleadas en diferentes áreas del conocimiento, como la ingeniería de tejidos, la ecología y el diseño de materiales funcionales [12]. Asimismo, este estudio abre la

puerta a futuras investigaciones orientadas a la optimización computacional de modelos en dominios tridimensionales, integración de otros modelos numéricos o incluso el análisis de sensibilidad y bifurcación frente a parámetros clave de un sistema.

Capítulo 2

OBJETIVOS

1. Objetivo general

Evaluar la respuesta espacio-temporal de los patrones de Turing aplicando el método de los elementos finitos en software libre.

2. Objetivos específicos

- Realizar una aproximación teórica a los sistemas reacción-difusión y al método de los elementos finitos.
- Encontrar la solución aproximada al modelo de Schnakenberg empleando el método de los elementos finitos.
- Implementar la solución de los modelos estudiados en software libre incorporando las condiciones de frontera y la discretización temporal necesarias para resolver el sistema.

3. Modificaciones

Durante el desarrollo del trabajo, se modificó el segundo objetivo de encontrar la solución de los modelos de Schnakenberg y quimiotaxis a solamente el primero, debido a una menor disponibilidad de los parámetros y estudios para validar el modelo. Sumado a qué la metodología implementada en una herramienta de código abierto, que es el propósito de este proyecto, es la misma que con Schnakenberg.

Capítulo 3

PROBLEMA Y JUSTIFICACIÓN

El estudio de los patrones espacio-temporales en sistemas biológicos ha sido una herramienta esencial para comprender la organización en la naturaleza. Fenómenos como las pigmentaciones animales y estructuras complejas de tejidos, que son causados por la autoorganización celular, se han podido explicar a partir de modelos matemáticos. Para trasladar estas ideas a simulaciones computacionales es necesario comprender los modelos que describen dichos patrones y las limitaciones que existen para implementarlos numéricamente.

El trabajo de Turing demostró teóricamente que, debido a las inestabilidades de Turing, surge la posibilidad de modelar el comportamiento de un sistema reacción-difusión ante ligeras perturbaciones que generan espontáneamente patrones heterogéneos en el espacio desde un estado uniforme [11]. Este fenómeno es fundamental en biología, medicina y diversas ciencias aplicadas, ya que ayuda a explicar procesos como la formación de tejidos, patrones de pigmentación o la organización espacial de las células [1]. Pero para llegar a simular geometrías reales se necesitan de herramientas que sean capaces de modelar correctamente los dominios complejos que encontramos en la naturaleza, y esta complejidad conlleva a altos costos computacionales o de licenciamiento que no todas las industrias están dispuestas a invertir [13] [14].

Dentro de los modelos más estudiados para la formación de patrones se encuentra el modelo de Schnakenberg, propuesto en 1979 como un sistema de reacción química capaz de exhibir oscilaciones y comportamientos no triviales [15]. Su simplicidad matemática, junto con su capacidad para generar inestabilidades de Turing de manera clara, lo ha convertido en un modelo de referencia para estudiar la autoorganización en sistemas de reacción-difusión [4]. Gracias a que permite identificar con precisión las condiciones bajo las cuales surgen patrones estables, el modelo de Schnakenberg es ampliamente usado para validar implementaciones numéricas, analizar bifurcaciones y estudiar el efecto de la geometría del dominio sobre la aparición de patrones.

En la biología, el entender la formación de estos patrones espacio-temporales permite comprender mejor el desarrollo y la formación de los organismos tanto en su

supervivencia como en su adaptación al medio ambiente [11]. Resolver las ecuaciones que acoplan los procesos de transporte de masa y de reacciones químicas que son necesarios aporta una explicación cuantitativa para comprender mejor las observaciones experimentales de procesos morfogenéticos y explorar hipótesis sobre la formación de los patrones espaciales en la piel animal y la modificación de estructuras originales dada por la evolución [1].

Otro ejemplo concreto de la relevancia de este problema es la aplicación médica que se le puede dar en el procedimiento ortodóntico de la expansión maxilar sobre la sutura medial palatina para tratar la deficiencia transversal del maxilar [16]. Para evitar dañar al paciente durante este tratamiento es necesario comprender el comportamiento de los tejidos y la sutura ante la presión de cargas externas generadas durante la expansión [16]. El uso de una herramienta de simulación numérica validado permite modelar los procesos mecanobiológicos en la sutura palatina durante la expansión maxilar. Esto ayudaría a profundizar en la comprensión de la recuperación y la optimización de los procesos de aplicación de cargas para mejorar la efectividad del tratamiento y reducir su duración.

A pesar de la importancia de los patrones de Turing en casos como los mencionados, actualmente la mayoría de las simulaciones numéricas se realizan con software de código cerrado o de alto costo, que restringen el acceso a investigadores y profesionales [17]. Además, muchos de estos entornos no ofrecen la flexibilidad necesaria para modelar dominios geométricos complejos o implementar condiciones de frontera específicas, lo que dificulta la aplicación de estos modelos en problemas reales [18]. Esta falta de herramientas accesibles y adaptables constituye un obstáculo para avanzar en la investigación y la transferencia del conocimiento hacia aplicaciones prácticas [17] [13].

Con todo lo anterior, este trabajo busca aportar al desarrollo de herramientas computacionales abiertas, eficientes y robustas que faciliten la simulación y análisis de los patrones de Turing. Con esto, se mejorará la integración de los patrones de Turing en diversas áreas del conocimiento de forma práctica y a bajo costo.

Capítulo 4

MARCO TEÓRICO Y ESTADO DEL ARTE

1. Conceptos fundamentales

- **Perturbación:** Es un pequeño cambio sobre el estado homogéneo del sistema. Ese estado representa una solución inicial donde las concentraciones de los morfógenos son constantes en todo el dominio [19]. Para estudiar si el sistema presenta inestabilidad de Turing, se introduce una variación artificial en ese equilibrio inicial. Dependiendo de la magnitud de esta variación y de si gobierna o no la inestabilidad del modelo se considera ligera o pesada [4].
- **Bifurcación:** La teoría de la bifurcación proporciona la metodología matemática para analizar los puntos críticos en el espacio de parámetros de un sistema dinámico donde un cambio mínimo provoca un gran cambio cualitativo en el comportamiento del sistema [4]. En los sistemas de reacción-difusión, el análisis de estabilidad lineal alrededor del estado estacionario homogéneo es la herramienta clave para identificar estas transiciones [4].
- **Residuo:** En el contexto del método de los elementos finitos, el residuo se define como la diferencia cuando se sustituye la solución aproximada en la ecuación original, lo que nos da a entender de forma cuantificable cuanto se desvía la solución numérica de satisfacer las ecuaciones diferenciales parciales en cualquier punto del dominio [20].
- **Heterogeneidad:** Los patrones de Turing son inherentemente heterogéneos, ya que rompen la simetría espacial de un estado uniforme, y para modelar este cambio en un sistema celular real es fundamental considerar la dinámica del dominio, dándole más robustez a los patrones formados [15].

2. Estudios descriptivos y experimentales de patrones de Turing

Turing en su investigación notó que, si bien la reacción química y la difusión molecular son homogeneizadores por separado, juntos pueden dar lugar a patrones estacionarios cuando se les combina de una forma especial. A partir de esto estableció:

primero, como se pueden formar patrones químicos espaciales, y segundo, como relacionar estos patrones con aquellos observados en la naturaleza [6].

La morfogénesis, proceso en el que se desarrollan los órganos de un embrión, es uno de los componentes clave en la biología del desarrollo. Turing propuso que la generación y forma de este proceso está directamente relacionado con un pre-patrón químico, controlado por las células, que dará una diferenciación de órganos, tejidos, patrón de la piel, etc. [1].

Estos fenómenos regidos por los procesos de reacción y difusión se caracterizan por la presencia de distribuciones espacio-temporales de las especies involucradas, siendo lo comúnmente llamado patrón. Turing definió las condiciones bajo las cuales un proceso reactivo en equilibrio puede ser inestabilizado por la presencia de un término difusivo, generándose así unos patrones espaciales heterogéneos, denominados inestabilidades por difusión o inestabilidades de Turing [5]. Tras un estudio a las soluciones de modelos biológicos definidos por ecuaciones de reacción-difusión encontró que pueden existir tres tipos de inestabilidades [4]:

- Oscilatorias en el tiempo y uniformes en el espacio, asociadas a las inestabilidades de Hopf independientes del espacio: Recibe este nombre por las soluciones de las ecuaciones sometidas a esta inestabilidad, que están justificadas con la bifurcación de Hopf [21], y se puede observar en reacciones químicas oscilantes como la de Belousov-Zhabotinsky, donde las concentraciones de reactivos varían de forma periódica en el tiempo sin formación de estructuras espaciales [22].
- Estacionarias en el tiempo y periódicas en el espacio: Este es el caso típico de los patrones de Turing, siendo responsable de la formación de estructuras como las manchas y rayas en la piel de animales [11].
- Oscilatorias tanto en el espacio como en el tiempo: Se les dice ondas viajeras, ya que se manifiestan como ondas que se propagan en el espacio y surgen por la interacción entre las anteriores inestabilidades, se puede usar para modelos depredador-presa [23].

3. Sistemas reacción-difusión

Normalmente, la difusión tiende a dispersar la concentración para crear una distribución homogénea, pero la interacción bajo ciertas condiciones específicas de reacción y difusión entre dos químicos puede generar una inestabilidad de Turing, lo que causa la formación de patrones heterogéneos en el espacio [7]. Esto inició la investigación de modelos matemáticos que expliquen una amplia variedad de fenómenos biológicos que involucran la interacción de individuos, células o especies químicas dentro de una región determinada [11]. Entre estos modelos nos encontramos con el de glucólisis, proceso en el que una célula descompone las moléculas de glucosa en piruvato [16], el de Gierer-Meinhardt para la regeneración y la formación de estructuras corporales en organismos biológicos [24] y el de FitzHugh-Nagumo para simular la propagación de ondas excitables en medios continuos, como la propagación de señales en fibras nerviosas y las ondas espirales en reacciones químicas oscilantes [25]. Como se mencionó en los objetivos, se abordará de forma particular el modelo de Schnakenberg.

Schnakenberg desarrolló su modelo en búsqueda de un sistema de reacción química más sencillo y químicamente plausible, que exhibiera el llamado comportamiento de soluciones periódicas temporales [26]. La inspiración para tal idea provino de la investigación en diferentes áreas biológicas, como los ciclos ecológicos de densidades de población y la regulación metabólica, que muestran un comportamiento dependiente del tiempo [27].

El sistema está definido por dos ecuaciones diferenciales parciales acopladas por las concentraciones de dos químicos o especies, un activador (u) que interactúa con un inhibidor (v), ambos con su respectivo coeficiente de difusión D_u o D_v , en el tiempo (t) y el espacio. Las ecuaciones son [16] [28]:

$$\frac{\partial u}{\partial t} = \nabla^2 u + \gamma(a - u + u^2 v) \quad (1a)$$

$$\frac{\partial v}{\partial t} = d\nabla^2 v + \gamma(b - u^2 v) \quad (1b)$$

Donde:

- u y v son las concentraciones de las dos especies reaccionantes.
- ∇^2 representa el operador laplaciano, que explica la difusión de la sustancia en el espacio. Está definido como la divergencia del gradiente.
- γ es la escala de velocidad de las reacciones químicas respecto a la difusión.
- a y b son términos relacionados con la producción de u y v respectivamente
- $d = D_u/D_v$ es el coeficiente de difusión.

Ahora, cada termino tiene su respectivo significado en la ecuación:

- Los términos $\nabla^2 u$ y $\nabla^2 v$ representan la difusión de las sustancias químicas u y v en el espacio.
- El coeficiente de difusión d modula la tasa de difusión de la especie v .
- Los términos $\gamma(a - u + u^2v)$ y $\gamma(b - u^2v)$ representan las reacciones locales entre las especies.

En el caso de $a - u + u^2v$ tenemos que a es un término fuente constante que impulsa la producción de u , $-u$ representa la tasa natural de decaimiento de u y u^2v representa el término de interacción no lineal entre sustancias, que conduce a una producción autocatalítica de u basada en su concentración y la de v .

En el caso de $b - u^2v$ tenemos que b es un término fuente constante para la producción de v y $-u^2v$ representa un término de degradación para v , impulsado por su interacción con u .

4. Selección de parámetros

Como se mencionó antes, existen tres tipos de inestabilidades, pero para conseguir la de Turing es fundamental hallar los valores para los que se obtiene la solución del sistema. Para el caso del modelo de Schnakenberg, los valores de estado estacionario de las concentraciones de especies (u^*, v^*) se calculan como se muestra a continuación [16]:

$$f(u, v) = \gamma(a - u + u^2v) = 0$$

$$g(u, v) = \gamma(b - u^2v) = 0$$

Donde $f(u, v)$ y $g(u, v)$ son las funciones reactivas de la ecuación. Al solucionarlas tenemos:

$$u^* = a + b$$

$$v^* = \frac{b}{(a + b)^2}$$

En la literatura se ha demostrado que asignar los valores $a = 0.1$ y $b = 0.9$ producen buenas respuestas al sistema de reacción-difusión. Por tanto, las condiciones iniciales se definieron como concentraciones perturbadas alrededor del estado estacionario de la siguiente manera:

$$u(t = 0) = u^* + p_u$$

$$v(t = 0) = v^* + p_v$$

Donde p_u y p_v son las perturbaciones de u^* y v^* . De esta manera, es posible encontrar el área geométrica de los posibles valores que pueden tomar los parámetros d y γ para satisfacer las condiciones necesarias que generan las inestabilidades de Turing.

$$f_u g_v - f_v g_u > 0$$

$$f_u + g_v < 0$$

$$d f_u + g_v < 0$$

$$(d f_u + g_v)^2 - 4d(f_u g_v - f_v g_u) > 0$$

Donde el subíndice de la función indica la variable por la cual se debe derivar parcialmente la función respectiva.

Los parámetros d y γ se calcularon mediante experimentación numérica. Los valores utilizados para la prueba de los parámetros se muestran en la Tabla 1 y corresponden a los valores reportados en la literatura para los modos de vibración de un dominio cuadrado [4] [15] [29].

Caso	D_v	dt	T	γ
1	9.1676	0.05	50	176.12
2	8.6676	0.02	20	230.82
3	8.6076	0.01	10	909.66
4	8.61	0.1	15	535.09

Tabla 1 – Parámetros para simulaciones

5. Método de los elementos finitos

El método de los elementos finitos surge como una técnica numérica que aprovecha el poder computacional para obtener soluciones aproximadas de ecuaciones diferenciales, tanto ordinarias como parciales, como los son los sistemas reacción-difusión. El fundamento de este método está en su formulación variacional (o débil) que permite reformular una ecuación diferencial parcial continua como un problema discreto de algebra lineal [30].

Para esto, se empieza por definir correctamente el problema tomando en cuenta el fenómeno a simular, el dominio del problema, y las condiciones de frontera. Seguido a esto, se hace la discretización del dominio o mallado, que divide el dominio en pequeños subdominios, denominados elementos finitos, conectados entre sí a través de puntos nodales, los cuales tienen en cuenta unas ecuaciones de equilibrio y la compatibilidad nodal, aproximando de manera precisa el problema original con infinitos grados de libertad a uno con una cantidad finita de grados de libertad [31].

Tras el mallado, se debe usar una técnica de aproximación para resolver la versión debilitada del problema. Aquí entra el método de residuos ponderados, que utiliza funciones de prueba que satisfacen las condiciones de frontera prescritas y una formulación integral para minimizar el error, error promedio, sobre todo el dominio del problema. Este método busca una solución aproximada de la forma [30]:

$$y^*(x) = \sum_{i=1}^n c_i N_i(x)$$

Donde y^* es la solución aproximada expresada como el producto entre los parámetros desconocidos c_i y las funciones de prueba N_i . Estas últimas, deben ser continuas sobre el dominio de interés y satisfacer las condiciones de frontera especificadas y deben ser seleccionadas para satisfacer la física del problema. En casos como el del modelo de Schnakenberg tenemos que aplicar el teorema de Green para reducir el orden de las derivadas, haciendo que solo requiera de primeras derivadas, simplificando el cálculo numérico. Al ser una aproximación la ecuación diferencial generará un residuo que puede ser calculado [32]:

$$R(x) = D[y^*, x] \neq 0$$

Por lo tanto, $R(x)$ también depende de los valores desconocidos. Este método requiere que los parámetros c_i sean evaluados tal que el error residual sobre el dominio sea cero:

$$\int_a^b w_i(x)R(x) dx = 0 \quad \text{con } i = 1, n$$

Aquí, w_i corresponde a n funciones de peso arbitrarias que generan n ecuaciones algebraicas. Ahora, con el método de Galerkin se establece:

$$w_i(x) = N_i(x) \quad \text{con } i = 1, n$$

La resolución numérica de los sistemas de ecuaciones matriciales resultantes del ensamblaje es posible gracias al poder de cálculo computacional, que integra todos los comportamientos individuales de cada elemento para predecir el comportamiento global del sistema [33]. Esta fase de procesamiento entonces, resuelve las ecuaciones lineales o no lineales simultáneamente para obtener resultados globales, como los valores de cada nodo [31].

Finalmente, se usan pos-procesadores modernos que eliminan la necesidad de evaluar los resultados de casa paso al facilitarle al usuario una interacción rápida y eficiente con los datos resultantes [16]. De aquí se obtienen las variables de interés para el estudio que se esté llevando a cabo.

Muchos autores han aplicado este método en el contexto de ecuaciones de reacción-difusión, mostrando su potencial para abordar geometrías complejas y condiciones de frontera variadas según el problema que se esté tratando [5] [10] [11] [34].

6. Investigaciones previas

En 1972, Gierer y Meinhardt generalizaron las ideas de Turing en el principio de patrón de activación de corto alcance, inhibición de largo alcance o activación local, inhibición lateral – LALI y aplicaron el modelo a la formación y regulación de patrones en la Hydra [35]. Desde entonces, Meinhardt ha aplicado extensiones de este modelo a una vasta gama de procesos biológicos de formación de patrones, incluyendo la filotaxis,

la segmentación, las venas y, lo más relevante para este estudio, los patrones de pigmentación en las conchas marinas [36]. Luego se llega a investigadores como Garzón, quien es de los más influyentes en el área de la simulación de sistemas reacción-difusión, que en 2007 propondría usar el método de los elementos finitos a la morfogénesis del tejido óseo, e incluso para el diseño de la geometría interna de matrices de generación ósea para mejorar los procesos de fabricación de implantes [3]. Otro gran aporte es el realizado por Vanegas y Garzón que hicieron énfasis en la implementación computacional de los modelos biológicos de glucólisis y quimiotaxis utilizando específicamente el método de los elementos finitos [28].

Toda esta formulación requiere de herramientas computacionales capaces de solucionar estas ecuaciones. Es aquí que aparecen trabajos como el de Rodríguez y Leiva, quienes en 2010 desarrollaron una herramienta de fácil uso y adquisición capaz de aplicar el método de los elementos finitos para dar solución a este tipo de problemas, incluyendo un visualizador gráfico que permite la creación de geometrías, asignación de propiedades de materiales y asignación de restricciones y cargas [37]. Aunque es una investigación valiosa, hoy se tienen medios para crear herramientas totalmente gratuitas y que son de bajo costo computacional.

Respecto a investigaciones que estén orientadas a la solución de patrones de Turing usando este tipo de herramientas, nos encontramos con trabajos como los de Duque-Daza, Garzón-Alvarado y Ramirez, quienes en 2015 usaron el software ABAQUS para dar solución a las ecuaciones reacción-difusión que describen la formación de patrones de Turing en superficies que simulan epidermis de animales, consiguiendo resultados que pueden compararse con los fenómenos naturales que se buscaban simular [10]. Incluso se ha llegado a dar solución a estos patrones en espacios tridimensionales, en los que se ha visto el gran impacto que sufren las sustancias activadoras ante las condiciones de frontera en comparación a las inhibidoras y la dependencia de la trayectoria del patrón evolutivo dentro de un dominio en crecimiento, pero también se concluyó que todavía no hay una comprensión total de los factores que llevan a la formación de patrones específicos en casos tridimensionales [34].

También se encuentran trabajos avanzados que integran diferentes modelos para lograr simulaciones más complejas que dependen de múltiples condiciones. Entre estos esta la investigación de Guerrero en la que realizo una simulación computacional de un modelo biomecánico y mecanobiológico aplicado a la formación de la sutura media palatina, con un fuerte enfoque biológico que pudo ser validado con casos de pacientes reales [16]. Otro ejemplo sería el trabajo de Garzón, Galeano y Mantilla en el que simularon sistemas de reacción-convección-difusión en dominios fijos, introduciendo los campos de velocidad toroidal que distorsionan los patrones originales al tener desviaciones proporcionales a la velocidad del giro a las que eran sometidas, sin llegar a causar la eliminación de la inestabilidad [5].

Todos estos trabajos han permitido avances significativos, pero aún existen limitaciones como la dependencia de software cerrado, las restricciones de estos softwares a dominios simples, y la configuración de las condiciones de frontera siguen siendo un desafío. Esto inspira el uso de herramientas abiertas y flexibles como la que se propone en esta investigación.

Capítulo 5

METODOLOGÍA

1. Implementación numérica

Para aplicar el MEF a las ecuaciones 1^a y 1b se requiere llegar a la formulación débil usando residuos ponderados. Para hacer esto, primero se igualan ambas ecuaciones a 0:

$$\frac{\partial u}{\partial t} - \nabla^2 u + \gamma(a - u + u^2 v) = 0 \quad (2^a)$$

$$\frac{\partial v}{\partial t} - d\nabla^2 v + \gamma(b - u^2 v) = 0 \quad (2b)$$

Luego se usa la formulación de residuos ponderados para debilitar las ecuaciones:

$$\int_{\Omega} f_{pu} \left[\frac{\partial u}{\partial t} - \nabla^2 u - \gamma(a - u + u^2 v) \right] d\Omega = 0 \quad (3^a)$$

$$\int_{\Omega} f_{pv} \left[\frac{\partial v}{\partial t} - d\nabla^2 v - \gamma(b - u^2 v) \right] d\Omega = 0 \quad (3b)$$

Siendo Ω el dominio de la ecuación, y f_{pu} , f_{pv} son las funciones de prueba para u y v respectivamente. Reorganizando las ecuaciones se tiene:

$$ru = \int_{\Omega} f_{pu} \frac{\partial u}{\partial t} d\Omega - \int_{\Omega} f_{pu} \gamma a d\Omega + \int_{\Omega} f_{pu} \gamma u d\Omega - \int_{\Omega} f_{pu} \gamma u^2 v d\Omega - \int_{\Omega} f_{pu} \nabla^2 u d\Omega \quad (4^a)$$

$$rv = \int_{\Omega} f_{pv} \frac{\partial v}{\partial t} d\Omega - \int_{\Omega} f_{pv} \gamma b d\Omega + \int_{\Omega} f_{pv} \gamma u^2 v d\Omega - \int_{\Omega} d f_{pv} \nabla^2 v d\Omega \quad (4b)$$

Ahora se aplica el teorema de Green para debilitar las integrales:

$$ru = \int_{\Omega} f_{pu} \frac{\partial u}{\partial t} d\Omega - \int_{\Omega} f_{pu} \gamma a d\Omega + \int_{\Omega} f_{pu} \gamma u d\Omega - \int_{\Omega} f_{pu} \gamma u^2 v d\Omega \quad (5^a)$$

$$+ \int_{\Omega} \nabla f_{pu} \nabla u d\Omega - \int_{\Gamma} f_{pu} (\nabla u \cdot \vec{n}) d\Gamma$$

$$rv = \int_{\Omega} f_{pv} \frac{\partial v}{\partial t} d\Omega - \int_{\Omega} f_{pv} \gamma b d\Omega + \int_{\Omega} f_{pv} \gamma u^2 v d\Omega + \int_{\Omega} d \nabla f_{pv} \nabla v d\Omega \quad (5b)$$

$$- \int_{\Gamma} f_{pv} (\nabla v \cdot \vec{n}) d\Gamma$$

Donde Γ representa el contorno del sistema. Con esta formulación los valores de u y v se aproximan como una combinación lineal de las funciones de forma. Aquí u_j y v_j son los valores nodales, y N_{u_j} y N_{v_j} son las funciones de base:

$$u^a \approx \sum_j u_j N_{u_j} \quad (6^a)$$

$$v^a \approx \sum_j v_j N_{v_j} \quad (6b)$$

Esta aproximación discretiza las ecuaciones diferenciales parciales sobre la malla de elementos finitos, dividiendo el dominio continuo en elementos más pequeños, donde la solución se aproxima localmente por las funciones de forma. Finalmente, se aplica una formulación de Galerkin y la ecuación cambia:

$$ru^a = \int_{\Omega} N_u \frac{\partial u}{\partial t} d\Omega - \int_{\Omega} N_u \gamma a d\Omega + \int_{\Omega} N_u \gamma u d\Omega - \int_{\Omega} N_u \gamma u^2 v d\Omega + \int_{\Omega} N_u \nabla u \nabla d\Omega \quad (6^a)$$

$$rv^a = \int_{\Omega} N_v \frac{\partial v}{\partial t} d\Omega - \int_{\Omega} N_v \gamma b d\Omega + \int_{\Omega} N_v \gamma u^2 v d\Omega + \int_{\Omega} N_v \nabla^2 v d\Omega \quad (6b)$$

La forma débil conduce a un sistema de ecuaciones en forma matricial. Se definen una matriz de rigidez K , un vector nodal U , y un vector de fuerza F , los cuales resultan de la discretización de ambos términos de difusión y de reacción. La forma general del sistema discretizado para cada variable se verá así:

$$[K]U = F$$

Donde K representa las contribuciones de difusión y reacción (de términos como $\nabla^2 u$ y $u^2 v$), U contiene los valores nodales desconocidos de la variable (como por ejemplo concentraciones de u_i) y F representa las contribuciones de fuentes externas o reacciones (como por ejemplo los términos $a - u + u^2 v$). El último paso es ensamblar las matrices y vectores de cada elemento para formar el sistema global. Este sistema global se resolverá numéricamente en cada paso de tiempo usando la librería FeniCS Project en Python.

2. Implementación Numérica en FeniCS

Para la implementación en FeniCS se tomaron como referencias los trabajos de Vanegas y Garzón-Alvarado [4] [15] [29], en los que se definieron los parámetros para conseguir los patrones deseados.

Se estableció una tolerancia de 10^{-14} para la comparación de coordenadas, con el fin de identificar correctamente los bordes de la malla, evitando errores causados por la representación numérica de los puntos en coma flotante. Luego, se definió una función de marcado de bordes que asigna identificadores enteros a las fronteras del dominio, de forma que se diferencien los contornos y se apliquen las condiciones de frontera específicas en cada uno dentro del método de los elementos finitos.

```
# Marcadores para cada borde
def marcador_borde(x):
    marcadores = np.full(x.shape[1], -1, dtype=np.int32)
    marcadores[np.isclose(x[0], 0.0, atol=tol)] = 1 # Borde izquierdo (x = 0)
    marcadores[np.isclose(x[0], 1.0, atol=tol)] = 2 # Borde derecho (x = 1)
    marcadores[np.isclose(x[1], 0.0, atol=tol)] = 3 # Borde inferior (y = 0)
    marcadores[np.isclose(x[1], 1.0, atol=tol)] = 4 # Borde superior (y = 1)
    return marcadores
```

Figura 1 Función de marcado de bordes

Seguido a esto se definen los valores de los parámetros, siendo $D_u = 1.0$, $a = 0.1$ y $b = 0.9$ constantes, y los demás dependen del fenómeno que se quiera replicar. Para validar la implementación se eligieron 4 casos de la literatura que se encuentra en la Tabla 1.

El dominio de la simulación se hace sobre un cuadrado unitario $[0,1] \times [0,1]$ que se discretiza en una malla de elementos cuadriláteros mixtos de 25×25 para los casos 1 y 2, o 50×50 para los casos 3 y 4. Para la aproximación numérica de u y v se seleccionaron funciones de base de tipo Lagrangiano de primer orden, lo que les permite ser continuas en los nodos compartidos entre elementos.

```
# Espacio base para funciones
elemento_u = ufl.FiniteElement("Lagrange", malla.ufl_cell(), 1)
elemento_v = ufl.FiniteElement("Lagrange", malla.ufl_cell(), 1)
```

Figura 2 Funciones de base tipo Langragiano

Dado que se trabaja con dos variables u y v se proyectan los espacios mixtos en subespacios independientes para cada una, lo que facilita la interpolación y asignación de condiciones iniciales específicas.

```
# Elemento mixto
elemento_mixto = ufl.MixedElement([elemento_u, elemento_v])
V = fem.FunctionSpace(malla, elemento_mixto)
```

Figura 3 Creación del espacio mixto

Gracias a las funciones de FeniCS se hallan los nodos en la frontera a los que se les aplicaran la función de marcado.

```
# Localizar fronteras
facets = mesh.locate_entities_boundary(
    malla,
    dim=1,
    marker=lambda x: np.logical_or.reduce((
        np.isclose(x[0], 0.0, atol=tol),
        np.isclose(x[0], 1.0, atol=tol),
        np.isclose(x[1], 0.0, atol=tol),
        np.isclose(x[1], 1.0, atol=tol)
    )))
)
```

Figura 4 Función para hallar elementos en la frontera

Después, se procede a identificar las coordenadas de los nodos asociados a la frontera, aplicarles el marcaje, crear los elementos que se rigen por estos, obtener los subespacios para u y v , asignar valores constantes en la frontera y aplicar las condiciones de frontera sobre los nodos que se sabe que están el borde izquierdo ya que están marcados con 1.

```

# Encuentra los grados de libertad
x = fem.locate_dofs_topological(V, entity_dim=1, entities=facets)
facet_coords = mesh.compute_midpoints(malla, 1, facets).T

# Aplicar la función de marcaje
valores_marcaje = marcador_borde(facet_coords)

# Crea objetos MeshTags que asocian los datos con un subset de entidades de la malla
facet_tags = mesh.meshtags(malla, 1, facets, valores_marcaje)

# Obtener subespacios para u y v
V_u, V_v = V.sub(0), V.sub(1)

# Valores constantes en la frontera
u_D_expr = fem.Constant(malla, ScalarType(1.0))
v_D_expr = fem.Constant(malla, ScalarType(0.0))

# Localizar los DoFs sobre el borde izquierdo (marcado con 1)
dofs_u = fem.locate_dofs_topological(V_u, entity_dim=1, entities=facet_tags.indices[facet_tags.values == 1])
dofs_v = fem.locate_dofs_topological(V_v, entity_dim=1, entities=facet_tags.indices[facet_tags.values == 1])

```

Figura 5 Configuración y uso de las entidades

Posterior a esto, se definieron las funciones de prueba y las funciones de solución (en el paso actual y anterior), para aplicar un esquema de avance temporal implícito.

```

# Funciones de prueba
phi = ufl.TestFunction(V)
phi_u, phi_v = ufl.split(phi)

# Función de la solución en el paso actual
w = fem.Function(V)
u, v = ufl.split(w)

# Función de la solución en el paso anterior (para el paso en el tiempo)
w_n = fem.Function(V)
u_n, v_n = ufl.split(w_n)

```

Figura 6 Definición de las funciones de prueba y solución

Para inducir a la inestabilidad de Turing las condiciones iniciales se establecieron como perturbaciones aleatorias alrededor de un estado homogéneo. El sistema no lineal resultante se resolvió mediante un método de Newton-Raphson implementado en la librería FeniCSx.

```
# Inicialización aleatoria suave con funciones Python
def u0_function(x):
    return np.full(x.shape[1], a + b) + 0.05 * np.random.rand(x.shape[1])

def v0_function(x):
    return np.full(x.shape[1], b / (a + b)**2) + 0.05 * np.random.rand(x.shape[1])
```

Figura 7 Iniciación de la perturbación

Se deben crear las funciones para inicializar la solución y asignar un w_n

```
# Crear funciones para inicializar
u0 = fem.Function(V_u_c)
v0 = fem.Function(V_v_c)

u0.interpolate(u0_function)
v0.interpolate(v0_function)

# Asignar a w_n
with w_n.vector.localForm() as loc:
    loc.set(0.0)

w0 = fem.Function(V)
w0.sub(0).interpolate(u0_function)
w0.sub(1).interpolate(v0_function)

w_n.x.array[:] = w0.x.array
```

Figura 8 Funciones para inicializar la solución

En este punto, se usa la formulación débil descrita anteriormente, que permitió resolver el sistema en su forma variacional y se acopla.

```
# Definir ecuaciones débiles
F_u = ((u - u_n)/dt)*phi_u*uf1.dx + D_u*uf1.dot(uf1.grad(u), uf1.grad(phi_u))*uf1.dx - gamma * (a - u + u*u*v)*phi_u*uf1.dx
F_v = ((v - v_n)/dt)*phi_v*uf1.dx + D_v*uf1.dot(uf1.grad(v), uf1.grad(phi_v))*uf1.dx - gamma * (b - u*u*v)*phi_v*uf1.dx
F = F_u + F_v
```

Figura 9 Formulación débil en código para FeniCSx

Para poder seguir con la solución, se debe definir el tipo de condición de frontera que se va a usar. En este caso usamos las condiciones de frontera de Neumann iguales a cero, porque no hay intercambio de sustancias en el borde de nuestro cuadrado unitario,

lo que numéricamente se puede interpretar como una razón de cambio o derivada nula. En FeniCS esto se ve representado al estar vacío el argumento.

```
# Crear condiciones de Neumann homogéneas  
bcs = []
```

Figura 10 Definición del tipo de frontera

Con esto se define el problema no lineal y se soluciona usando el NewtonSolver de FeniCS

```
# Resolver el problema no lineal  
problem = NonlinearProblem(F, w, bcs)  
solver = NewtonSolver(malla.comm, problem)
```

Figura 11 Definición y solución del problema

Los resultados de este sistema se guardan en un archivo xdmf individual para u y v . Estos archivos se usan en el visualizador de ParaView, generando una animación de la evolución espacio-temporal del sistema.

```

# Crear archivo para guardar resultados
u_f_write = fem.Function(V.sub(0).collapse()[0])
u_file = XDMFFile(malla.comm, "schnakenberg_u.xdmf", "w")
u_file.write_mesh(malla)

v_f_write = fem.Function(V.sub(1).collapse()[0])
v_file = XDMFFile(malla.comm, "schnakenberg_v.xdmf", "w")
v_file.write_mesh(malla)

# Bucle temporal
t = 0.0
while t < T + 1e-8:
    # Resolver sistema no lineal
    n, converged = solver.solve(w)
    if not converged:
        print(f"El solver no convergió en t = {t:.3f}")
    else:
        print(f"t = {t:.3f}, iteraciones = {n}")

    # Interpolar u para escribir
    u_f_write.interpolate(w.sub(0).collapse())
    u_file.write_function(u_f_write, t)

    v_f_write.interpolate(w.sub(1).collapse())
    v_file.write_function(v_f_write, t)

    # Avanzar en el tiempo
    w_n.x.array[:] = w.x.array
    t += dt

u_file.close()
v_file.close()

```

Figura 12 Creación de los archivos xdmf y bucle temporal

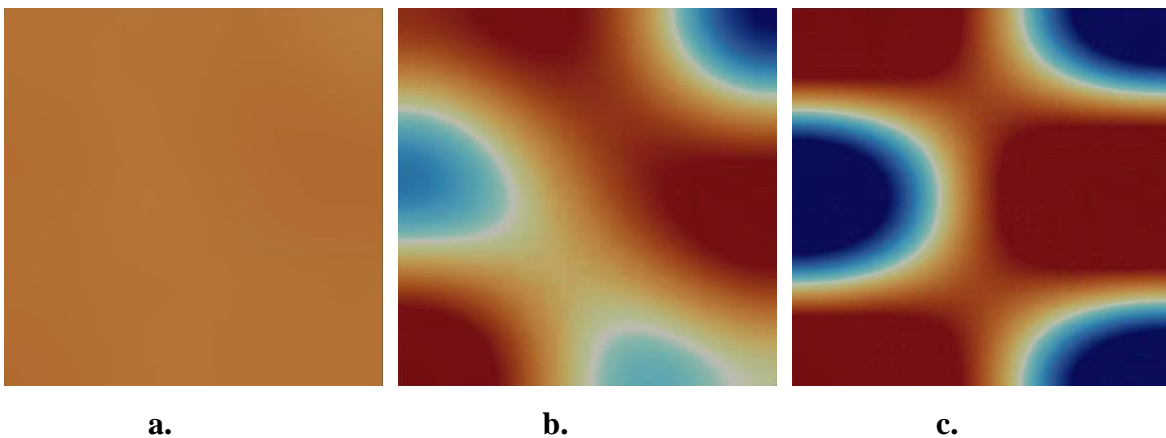
De esta forma, se parte de la formulación débil del problema, pasa por la discretización en FeniCSx y termina con la visualización de resultados en ParaView, lo que permite analizar de manera detallada la dinámica espacio-temporal de los patrones de Turing en diferentes condiciones de simulación. El código mostrado se encuentra disponible para uso en un repositorio de la plataforma GitHub, al que se puede acceder a través del siguiente link: <https://github.com/JuanesBeltran/Modelo-de-Schnakenberg>

Capítulo 6

RESULTADOS Y DISCUSIÓN

Teniendo los resultados de la aproximación del modelo de Schnakenberg bajo diferentes parámetros, comprobando que concuerden con investigaciones realizadas por otros estudios reportados en la literatura, se muestran los casos mencionados anteriormente en la Tabla 1 – Parámetros para simulacione y su evolución espacio-temporal. Todos parten desde un estado inicial homogéneo que sufre una perturbación inicial y se muestra la evolución en el tiempo.

1. Caso 1



*Figura 13 Concentraciones de u y v en el caso 1. Tiempo adimensional [4] para $a=0$,
 $b=5$, $c=10-50$*

Se muestra la aparición de estructuras en forma de manchas de gran tamaño que desde un tiempo intermedio se empieza a consolidar hasta llegar a su forma consolidada y estable al terminar la simulación. El alto coeficiente de difusión favorece la rápida difusión de las sustancias y la formación de manchas. El valor bajo de γ impide que las perturbaciones aumenten, dando como resultado pocas manchas de gran tamaño pero que son estables.

Estos resultados son coherentes con los de Vanegas, Landinez y Garzón [4], quienes hicieron sus simulaciones en el software ABAQUS.

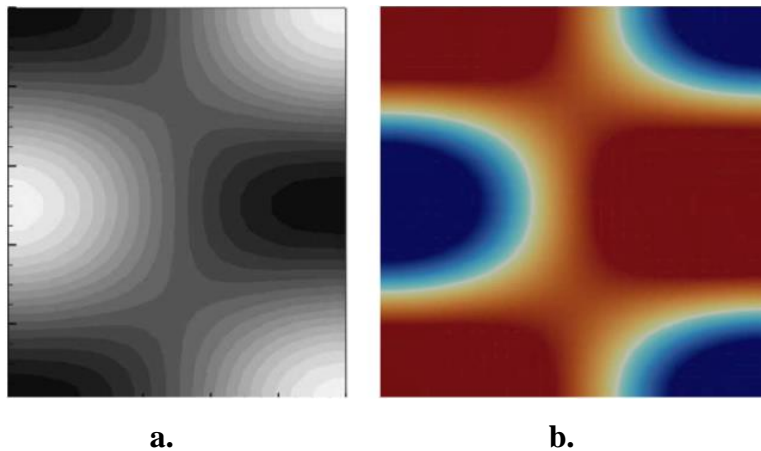


Figura 14 Comparación con la literatura Caso 1. A. Resultado de Vanegas, Landinez y Garzón [4] b. Resultado de este estudio

2. Caso 2

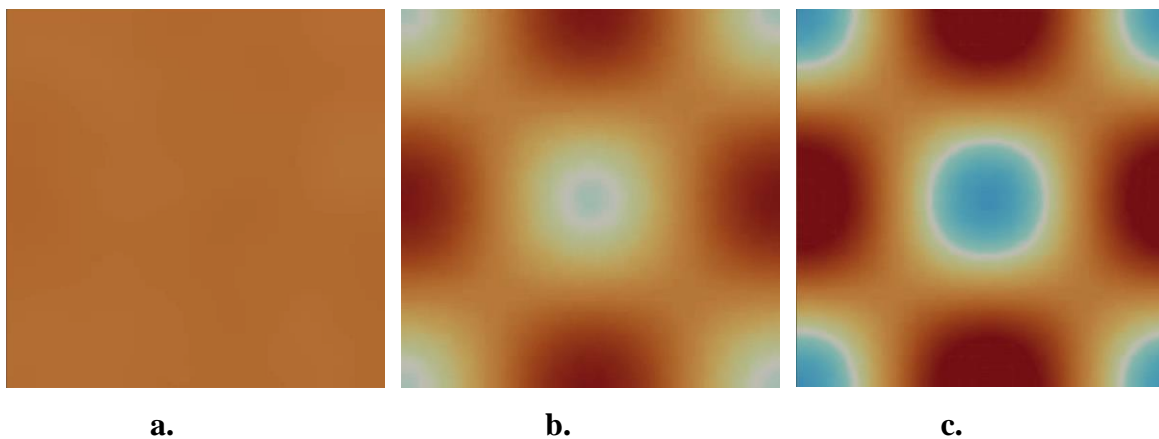


Figura 15 Concentraciones de u y v en el caso 2. Tiempo adimensional [4] para $a=0$, $b=10$, $c=20$

Se ve un aumento en las zonas de concentración de las sustancias que tienen formas de manchas o retículas casi circulares algo más pequeñas que en el caso anterior. La reducción del coeficiente de difusión reduce el contraste difusivo, lo que explica la forma espacial más reducida. Subir el γ incrementa la intensidad de las reacciones, acelerando la amplificación de perturbaciones y la cantidad de focos de activación.

Al igual que en el caso anterior, los resultados del trabajo de Vanegas y sus colegas coinciden [4].

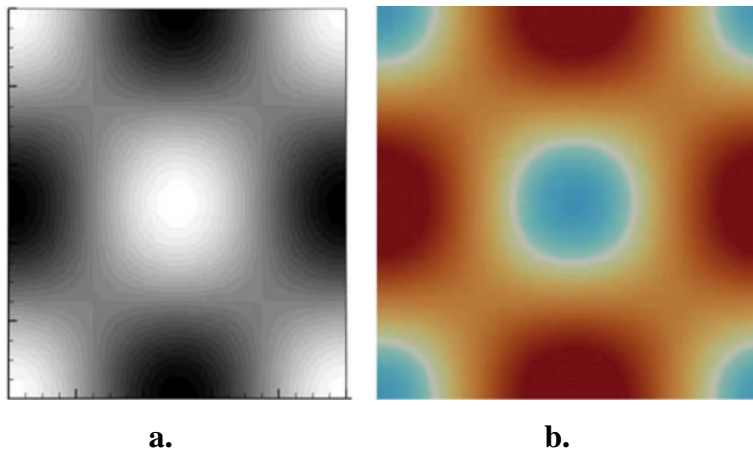


Figura 16 Comparación con la literatura Caso 2. A. Resultado de Vanegas, Landinez y Garzón [4] b. Resultado de este estudio

3. Caso 3

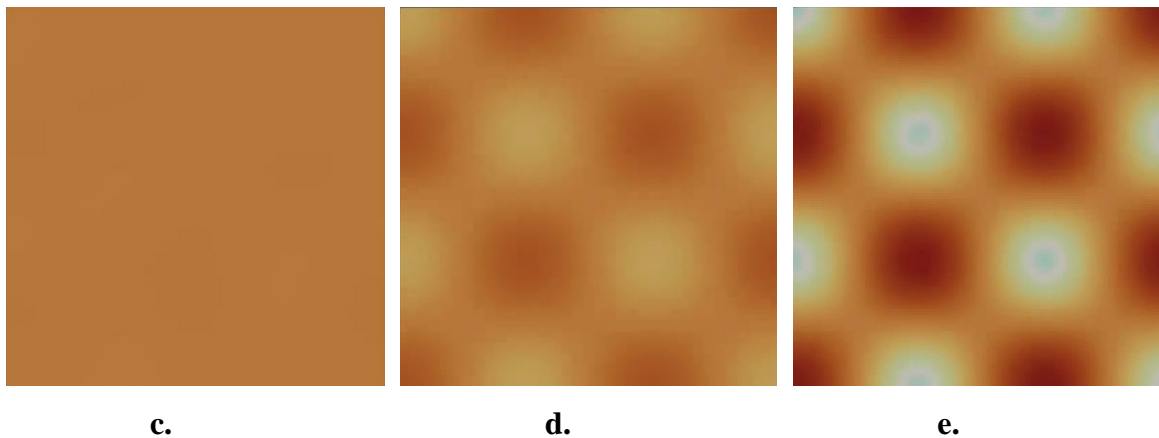


Figura 17 Concentraciones de u y v en el caso 3. Tiempo adimensional [15] para $a=0$, $b=5$, $c=10$

Se forma un patrón con mayor frecuencia espacial que los anteriores casos. Se puede ver que las estructuras tienden a ser más numerosas y de menor tamaño, aunque mantienen la forma de mancha casi circular. El valor de γ tan alto incrementa la aparición

de múltiples focos de concentración en el espacio. Para poder apreciar este incremento es que se modifica el proceso de mallado del espacio para aumentar la cantidad de veces que se divide y los elementos que se generan. La disminución del coeficiente de difusión también ayuda a reducir la longitud de área dominada por la sustancia.

Estos resultados son coherentes con los de Vanegas, Gonzales y Garzón [15], quienes volvieron a usar el software ABAQUS para las simulaciones.

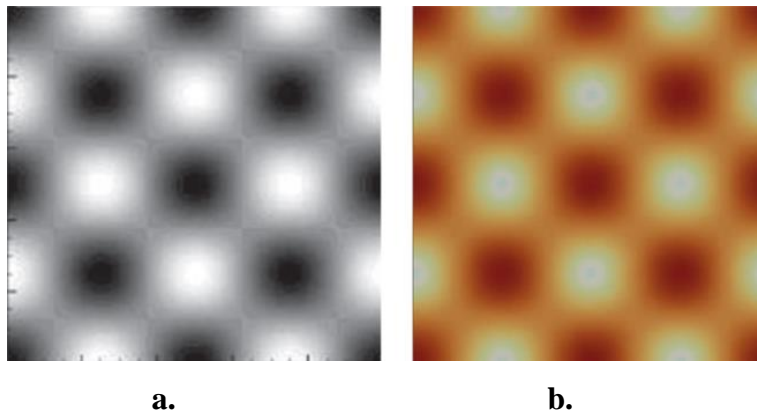


Figura 18 Comparación con la literatura Caso 3. A. Resultado de Vanegas, Gonzales y Garzón [15] b. Resultado de este estudio

4. Caso 4

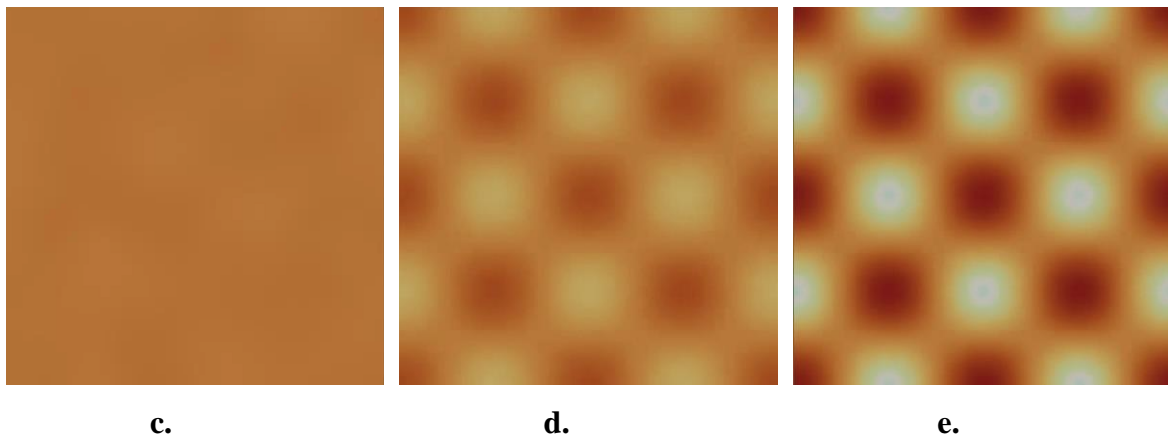


Figura 19 Concentraciones de u y v en el caso 4. Tiempo adimensional [29] para $a=0$, $b=8$, $c=15$

Se forma el patrón con mayor frecuencia espacial de todos, contando con numerosas estructuras de menor tamaño que se mantienen estables en el tiempo. La combinación de un coeficiente de difusión como el de los anteriores casos, junto a un γ intermedio genera una selección de longitudes de área que favorece que tomen la forma de rejillas regulares. A pesar de no tener sus limitaciones tan marcadas, se mantiene estable en el tiempo.

Este último resultado coincide con lo mostrado por Garzón, Galeano y Mantilla [29], quienes también usaron ABAQUS para sus simulaciones.

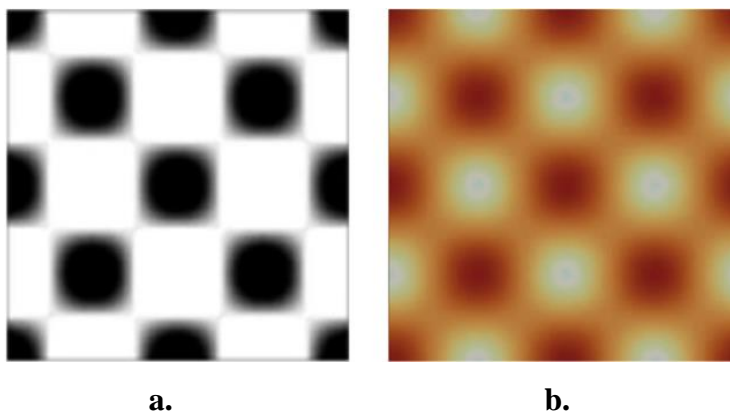


Figura 20 Comparación con la literatura Caso 4. A. Resultado de Garzón, Galeano y Mantilla [29] b. Resultado de este estudio

En todos los casos se evidencio como el contraste difusivo es el motor principal de la inestabilidad de Turing por las diferencias entre los valores de D_u y D_v . Los valores ligeramente mayores tienden a producir manchas más separadas (longitud de onda mayor), mientras que valores más bajos favorecen patrones de mayor frecuencia espacial (más estructuras).

El efecto de γ es controlar la velocidad de crecimiento de las perturbaciones y la intensidad de las interacciones entre sustancias. Un γ alto (como el caso 3) produce muchos focos pequeños y dinámica rápida; uno moderado (como el caso 1) produce patrones más estacionarios y claramente diferenciados.

Los tipos de patrones observados y la sensibilidad a los parámetros son consistentes con trabajos clásicos y con simulaciones reportadas en la literatura [4] [15]

[29], en los que usaron herramientas de pago como ABAQUS, todo esto con herramientas que no tienen restricciones por licencias.

Capítulo 7

CONCLUSIONES

1. Conclusión general

Los patrones de Turing son una de las principales manifestaciones de la autoorganización en sistemas naturales, siendo claves para explicar fenómenos biológicos y fisicoquímicos complejos a partir de las interacciones de reacción y difusión. Sin embargo, llevar a cabo la implementación computacional de estos tiene limitaciones importantes debido a las condiciones de uso del software de código cerrado.

Con este trabajo se demuestra la utilidad y potencia del software libre, específicamente de la librería FEniCSx, como una alternativa robusta y accesible para la simulación de modelos de reacción-difusión. Tras la implementación numérica, se logró reproducir patrones de Turing con diferentes configuraciones paramétricas y se validaron los resultados con estudios previos citados en la literatura. La formulación débil del modelo, en combinación con el método de los elementos finitos, permitió obtener soluciones numéricas estables y precisas que capturan adecuadamente la dinámica espacio-temporal de los sistemas estudiados.

Con todo esto, el trabajo cumple su objetivo principal de implementar la solución por MEF en una herramienta abierta, reproducible y funcional que facilita el estudio y análisis de los patrones de Turing. Así, se aporta al avance del conocimiento en el uso de la modelación matemática para fenómenos naturales y se abren las posibilidades a aplicaciones en áreas como la biología del desarrollo, la medicina, la física de materiales o la ingeniería biomédica. Además, al promover el uso de software libre, se alienta a la colaboración y la accesibilidad del conocimiento, abriendo las puertas para el desarrollo de modelos más complejos y la exploración de otros fenómenos.

2. Limitaciones del estudio

Este trabajo presenta varias limitaciones que se deben reconocer. En primer lugar, la simulación se realizó en un dominio bidimensional y una geometría sencilla. Esto permitió validar la metodología al reproducir los patrones reportados en la literatura, pero

también limita la aplicación de los resultados en sistemas biológicos más realistas, que involucran geometrías irregulares y complejas.

En segundo lugar, a pesar de que existen otros modelos de reacción-difusión que explican la formación de patrones de Turing, el estudio se centró únicamente en el modelo de Schnakenberg, ya que sirve como base para la formulación de otros modelos y que se puede encontrar bastante documentación de referencia para su implementación.

Por último, aunque FEniCSx permitió una implementación efectiva, el proyecto se enfoca únicamente en dominios fijos y no incluye fenómenos adicionales como crecimiento del dominio ni el acoplamiento mecánico, que también desempeñan un papel relevante en la formación de patrones biológicos.

3. Líneas futuras de investigación

Este trabajo cumplió con los objetivos propuestos y sirve como fundamento para continuar y profundizar la investigación en el área del modelamiento matemático. Una primera dirección es la extensión del análisis a dominios irregulares y superficies tridimensionales, aprovechando las capacidades geométricas que ofrece FEniCSx. Esto permitiría estudiar patrones más cercanos a estructuras biológicas reales, como tejidos curvos o morfologías embrionarias.

También se podrían explorar otros modelos de reacción-difusión y sistemas acoplados complejos, realizando los ajustes necesarios en la implementación, para evaluar las capacidades del MEF y del software libre para capturar fenómenos más complejos.

Finalmente, este trabajo sirve como base inicial de un repositorio de código abierto y una interfaz simplificada para que otros investigadores puedan replicar y adaptar las simulaciones sin depender de software propietario. Esto ampliaría el impacto del trabajo y contribuiría al avance de la modelación computacional.

Referencias

- [1] A. L. Durán, Patrones de Turing en sistemas Biológicos, Iztapalapa, 2012.
- [2] M. F. Erickson, «The Ecology of Chemically Defended,» Macquarie University, Sydney, 2025.
- [3] D. A. Garzón, Simulación de procesos de reacción-difusión: aplicación a la morfogénesis del tejido óseo, Zaragoza: Centro Politécnico Superior de la Universidad de Zaragoza, 2007.
- [4] J. Venegas, N. Landinez y D. Garzón, Análisis de la inestabilidad de Turing en modelos biológicos, Medellín: Dyna rev.fac.nac.minas, 2009.
- [5] D. A. Garzón, C. H. Galeano y J. M. Mantilla, «Formación de patrones de turing para sistemas de reacción-convección-difusión en dominios fijos sometidos a campos de velocidad toroidal,» *Revista Facultad de Ingeniería Universidad de Antioquia*, pp. 75-87, 2010.
- [6] A. M. Turing, The Chemical Basis of Morphogenesis, Philosophical Transactions of the Royal Society of London, Londres: Series B, 1952.
- [7] R. Burgüer, Una introducción a la formación de patrones de Turing, Concepción, 2023.
- [8] G. G. Malinietski, Fundamentos matemáticos de la sinérgica. Caos, estructuras y simulación por ordenador, Moscú: Серия: Sinérgica: del pasado al futuro, 2006.
- [9] L. L. Otero Pereiro, Aprendiendo sobre el Método de los Elementos Finitos, Ciudad de La Habana: Instituto Superior Politécnico José Antonio Echeverría, 2006.
- [10] C. Duque-Daza, A. Ramirez y D. Garzón-Alvarado, «Turing patterns formation on surfaces under deformation: A total lagrangian method approach,» *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, 1 12 2012.
- [11] L. A. Gonzales , J. C. Vanegas y D. A. Garzón, «Formación de patrones en sistemas de reacción-difusión en dominios crecientes,» *Revista Internacional de Métodos Numéricos*, vol. 25, n° 2, pp. 145-161, 2009.
- [12] N. Jar, «Servicio de Información y Noticias Científicas,» 20 6 2018. [En línea]. Available: <https://www.agenciasinc.es/Noticias/Turing-tenia-razon-sus-patrones-biologicos-son-una-promesa-en-ingenieria-de-tejidos>.
- [13] R. Luna, «Librería de Tijera,» 07 09 2024. [En línea]. Available: <https://libreriatijera.com.ar/por-que-un-software-debe-ser-libre/>.
- [14] SOAINT, «Altos costos de licenciamiento impiden la innovación en empresas,» 7 Agosto 2024. [En línea]. Available: <https://soaint.com/blog/altos-costos-de-licenciamiento-impiden-la-innovacion-en-empresas/>.
- [15] J. C. Vanegas, L. A. González y D. A. Garzón, «Solución numérica de modelos biológicos de reacción difusión en dominios fijos mediante el método de los elementos finitos,» *Revista Facultad de Ingeniería Universidad de Antioquia*, pp. 65-75, Jun 2009.

- [16] J. A. Guerrero Vargas, «Biomechanical and mechanobiological model of the midpalatal suture formation process, by computational simulation.,» 10 Dec 2019. [En línea]. Available: <https://repositorio.unal.edu.co/handle/unal/77286>.
- [17] C. Pear, «Open-Source vs. Proprietary Tools: Key Differences,» Anvil Labs, 21 Agosto 2025. [En línea]. Available: <https://anvil.so/post/open-source-vs-proprietary-tools-key-differences>.
- [18] S. Gratkowski, «Implementation of Asymptotic Boundary Conditions in Commercial FEM Software,» de *14th IGTE Symposium*, Graz, 2010.
- [19] P. Grindrod, «Patterns and Waves: The Theory and Applications of Reaction–Diffusion Equations,» *Oxford University Press*, 1996.
- [20] S. Romero, F. J. Moreno y I. M. Rodríguez, *Introducción a las Ecuaciones en Derivadas Parciales*, Huelva: Servicio de Publicaciones. Universidad de Huelva, 2001.
- [21] J. B. Peña Muñoz, *Inestabilidad de Turing en sistemas biológicos*, Iztapalapa, 2022.
- [22] V. Petrov, V. Gáspár, J. Masere y K. Showalter, «Controlling chaos in the Belousov—Zhabotinsky reaction,» *Nature*, vol. 361, n° 6409, pp. 240-243, 1993.
- [23] G. A. Salcedo Varela, «Bifurcación de Turing-Hopf en sistemas de reacción-difusión,» Universidad de Sonora, Sonora, 2017.
- [24] A. Gierer y H. Meinhardt, «A Theory of Biological Pattern Formation,» *Kybernetik*, vol. 12, n° 1, p. 30–39, 1972.
- [25] J. Nagumo, S. Arimoto y S. Yoshizawa, «An Active Pulse Transmission Line Simulating Nerve Axon,» *Proc. IRE*, vol. 50, n° 10, p. 2061–2070, 1962.
- [26] J. Schnakenberg, «Simple Chemical Reaction System with Limit Cycle Behaviour,» *J. Theor. Biol.*, vol. 81, n° 12, pp. 389-400, 1979.
- [27] C. L. Beentjes, *Pattern formation analysis in the Schnakenberg model*, Oxford: Mathematical Institute, University of Oxford, 2014.
- [28] J. C. Vanegas y D. A. Garzón, «Biological modelling and computational implementation using the finite elements method,» *Computational and Applied Mathematics*, vol. 33, n° 3, pp. 621-640, 2013.
- [29] D. A. Garzón, C. Galeano y J. Mantilla, «Numerical tests on pattern formation in 2D heterogeneous mediums: an approach,» *Dyna*, pp. 56-66, Abril 2012.
- [30] D. L. Logan, *A first course in the finite element method*, Platteville: Thomson, 2011.
- [31] A. Ortiz Prado, O. Ruiz Cervantes y J. A. Ortiz Valera, *Modelado de procesos de manufactura*, Ciudad de Mexico: Facultad de Ingeniería de la UNAM, 2013.
- [32] D. V. Hutton, *Fundamentals of Finite Element Analysis*, New York: McGraw-Hill, 2004.
- [33] «Software de análisis de elementos finitos,» Autodesk, [En línea]. Available: <https://www.autodesk.com/mx/solutions/finite-element-analysis>.
- [34] S. B. Tahar, J. J. Muñoz, S. J. Shefelbine y E. Comellas, «Turing pattern prediction in three-dimensional domains: the role of initial conditions and growth,» *bioRxiv*, 19

Apr 2023.

- [35] A. Gierer y H. Meinhardt, A theory of biological pattern formation, Tübingen: Kybernetik, 1972.
- [36] H. Meinhardt, P. Prusinkiewicz y D. R. Fowler, The Algorithmic Beauty of Sea Shells, Berlin: Springer-Verlag, 2003.
- [37] H. A. Rodríguez y H. R. Leiva, Desarrollo de una herramienta computacional basada en el método de elementos finitos como instrumento didáctico-académico, Naguanagua: Facultad de ingeniería universidad de Carabobo, 2010.