

## PROYECTO DE DESARROLLO

# Simulación Interactiva de Dinámica de Fluidos 2D Basada en Navier-Stokes Simplificado

Samuel A. Chamalé<sup>1</sup>, Adrian Rodríguez Batres<sup>2</sup>, and Jose D. Gómez<sup>3</sup>

<sup>1</sup>Universidad del Valle de Guatemala, cha21881@uvg.edu.gt

<sup>2</sup>Universidad del Valle de Guatemala, rod21691@uvg.edu.gt

<sup>3</sup>Universidad del Valle de Guatemala, gom21429@uvg.edu.gt

---

## Resumen

La simulación de dinámica de fluidos es una herramienta fundamental en diversas áreas de la ciencia y la ingeniería, permitiendo el análisis y visualización de comportamientos fluidos complejos. Este proyecto presenta el desarrollo de una simulación interactiva bidimensional que modela el flujo de fluidos y la difusión de colores utilizando las ecuaciones simplificadas de Navier-Stokes y la ecuación de difusión. La implementación se realizó en Python, empleando librerías como NumPy, SciPy, Pygame y pygame\_gui, lo que facilita operaciones numéricas eficientes y una interfaz de usuario responsiva. Se incorporaron características interactivas que permiten al usuario ajustar parámetros como la tasa de difusión, viscosidad y gravedad en tiempo real, así como manipular el entorno mediante la adición de colores y barreras. Los resultados demuestran la capacidad de la simulación para reproducir fenómenos físicos como la difusión de colores, formación de remolinos y goteo bajo la influencia de la gravedad, proporcionando una experiencia visualmente coherente y científicamente precisa. Este trabajo sienta las bases para futuras mejoras que podrían incluir aumentos en la resolución, optimizaciones computacionales avanzadas y la incorporación de modelos de presión más realistas, ampliando así las aplicaciones y la precisión de la simulación.

---

## 1 | Introducción

La dinámica de fluidos es una rama de la mecánica que estudia el comportamiento de los fluidos en movimiento, abarcando desde corrientes oceánicas hasta el flujo de aire en la atmósfera y en sistemas de ingeniería. Las ecuaciones de Navier-Stokes constituyen el pilar teórico para describir la conservación de la cantidad de movimiento en fluidos viscosos e incompresibles [1]. Sin embargo, debido a la complejidad y no linealidad de estas ecuaciones, su resolución analítica es viable únicamente en casos muy simplificados, lo que ha impulsado el desarrollo de métodos numéricos para su aproximación en aplicaciones prácticas [8].

En los últimos años, el avance de las tecnologías computacionales y la disponibilidad de librerías especializadas han facilitado la implementación de simulaciones de dinámica de fluidos más accesibles y eficientes. Herramientas como NumPy y SciPy en Python proporcionan las capacidades necesarias para manejar operaciones matriciales y convoluciones de manera rápida, mientras que bibliotecas como Pygame y pygame\_gui permiten la creación de interfaces gráficas interactivas que mejoran la experiencia del usuario durante la simulación [5, 7, 9, 11].

El presente proyecto tiene como objetivo desarrollar una simulación interactiva bidimensional de dinámica de fluidos que combine la difusión de colores con el flujo de fluidos, permitiendo al usuario manipular parámetros clave y observar en tiempo real los efectos de estos cambios. La simulación se basa en una versión simplificada de las ecuaciones de Navier-Stokes, implementando métodos numéricos eficientes para garantizar un rendimiento adecuado y una visualización fluida.

### 1.1. Objetivos

Los principales objetivos de este proyecto son:

- Implementar una simulación bidimensional para dinámica de fluidos utilizando las ecuaciones de Navier-Stokes simplificadas.
- Modelar la difusión de colores en el lienzo mediante la ecuación de difusión, permitiendo la creación de patrones visuales complejos.
- Desarrollar una interfaz de usuario interactiva que permita ajustar parámetros como la tasa de difusión, viscosidad y gravedad en tiempo real.
- Incorporar herramientas de dibujo y manipulación de barreras que afecten el comportamiento del flujo de fluidos.

- Evaluar el rendimiento y la precisión de la simulación, identificando áreas de mejora y posibles optimizaciones.

## 2 | Marco Teórico

### 2.1. Dinámica de Fluidos y las Ecuaciones de Navier-Stokes

La dinámica de fluidos es una rama de la mecánica que estudia el comportamiento de los fluidos (líquidos y gases) en movimiento. Las ecuaciones fundamentales que rigen este comportamiento son las **Ecuaciones de Navier-Stokes**, las cuales describen la conservación de la cantidad de movimiento en un fluido viscoso e incompresible.

En dos dimensiones, las ecuaciones de Navier-Stokes para un fluido incompresible se expresan como:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \nu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + f_x \quad (1)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \nu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + f_y \quad (2)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (3)$$

Donde:

- $u$  y  $v$  son las componentes de la velocidad del fluido en las direcciones  $x$  e  $y$ , respectivamente.
- $p$  es la presión del fluido.
- $\nu$  es la viscosidad cinemática del fluido.
- $f_x$  y  $f_y$  representan las fuerzas externas aplicadas al fluido en las direcciones  $x$  e  $y$ .

Las ecuaciones de Navier-Stokes son altamente no lineales y, en general, no tienen soluciones analíticas en dimensiones superiores a una. Por lo tanto, se requieren métodos numéricos para su resolución durante sus aplicaciones prácticas, como la simulación de fluidos en tiempo real.

### 2.2. Ecuación de Difusión

La difusión es un proceso físico mediante el cual las partículas de una sustancia se dispersan desde regiones de alta concentración hacia regiones de baja concentración. Matemáticamente, este fenómeno se describe mediante la **Ecuación de Difusión** o **Ecuación**

**del Calor**, que es una ecuación en derivadas parciales de primer orden en el tiempo y de segundo orden en el espacio:

$$\frac{\partial c}{\partial t} = D \nabla^2 c \quad (4)$$

Donde:

- $c$  representa la concentración de la sustancia.
- $D$  es el coeficiente de difusión.
- $\nabla^2$  es el operador Laplaciano, que en dos dimensiones se expresa como  $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ .

En el contexto de la simulación de fluidos y difusión de colores, la ecuación de difusión se utiliza para modelar cómo las propiedades del fluido, como la velocidad y las intensidades de color, se dispersan a lo largo del tiempo debido a la viscosidad y otros factores.

## 2.3. Métodos Numéricos para la Simulación de Fluidos

Dado que las ecuaciones de Navier-Stokes y la ecuación de difusión son difíciles de resolver analíticamente, se emplean métodos numéricos para aproximar sus soluciones.

### 2.3.1. Convolución y el Operador Laplaciano

El operador Laplaciano es fundamental en las ecuaciones de difusión y de Navier-Stokes, ya que modela fenómenos como la difusión de la velocidad y la viscosidad en el fluido. En la simulación, el Laplaciano se aproxima mediante una **convolución** con un núcleo específico.

El kernel o núcleo Laplaciano utilizado en la simulación es el siguiente:

$$K = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (5)$$

La operación de convolución entre el campo de velocidad  $u$  o  $v$  y el núcleo  $K$  se realiza para calcular el Laplaciano de dichos campos:

$$\nabla^2 u \approx K @ u \quad (6)$$

$$\nabla^2 v \approx K @ v \quad (7)$$

### 2.3.2. Advección y Método Semi-Lagrangiano

La **advección** se refiere al transporte de una sustancia o propiedad (como la velocidad o el color) por el movimiento del fluido. En la simulación, la advección de los campos de velocidad y los colores se implementa utilizando un método semi-Lagrangiano, que rastrea las propiedades hacia atrás en el tiempo para determinar sus posiciones actuales.

Matemáticamente, la advección de una propiedad  $d$  se expresa como:

$$\frac{\partial d}{\partial t} + u \frac{\partial d}{\partial x} + v \frac{\partial d}{\partial y} = 0 \quad (8)$$

El método semi-Lagrangiano consiste en calcular la posición anterior de cada punto en el espacio de la simulación y utilizar la interpolación bilineal para estimar el valor de la propiedad en esa posición.

### 2.3.3. Interpolación Bilineal

La interpolación bilineal es una técnica utilizada para estimar valores intermedios en una cuadrícula bidimensional. En la simulación, se utiliza para calcular los valores de las propiedades del fluido (como las intensidades de color) en posiciones no enteras resultantes del seguimiento de las trayectorias de advección.

El proceso de interpolación bilineal considera los cuatro puntos de la cuadrícula más cercanos a la posición objetivo y pondera sus valores en función de la distancia relativa a estos puntos. Matemáticamente, si  $(x, y)$  es una posición no entera, se identifican los puntos de la cuadrícula  $(x_0, y_0)$ ,  $(x_1, y_0)$ ,  $(x_0, y_1)$  y  $(x_1, y_1)$ , y el valor interpolado  $d(x, y)$  se calcula como:

$$d(x, y) = w_a d(x_0, y_0) + w_b d(x_0, y_1) + w_c d(x_1, y_0) + w_d d(x_1, y_1) \quad (9)$$

Donde los pesos  $w_a, w_b, w_c, w_d$  se determinan en función de la distancia de  $(x, y)$  a cada uno de los puntos vecinos.

## 2.4. Simulación de Colores en el Fluido

Además de simular el movimiento del fluido, este proyecto incorpora la simulación de la difusión de colores sobre el lienzo. Este componente permite visualizar de manera más intuitiva el comportamiento del fluido, mostrando cómo los colores se mezclan, difunden y decaen con el tiempo.

### 2.4.1. Difusión de Colores

La difusión de colores se modela utilizando la **Ecuación de Difusión**, aplicada a cada canal de color (rojo, verde y azul) de manera independiente. Cada canal de color tiene su propio coeficiente de difusión, lo que permite controlar la rapidez con la que cada color se dispersa en el lienzo.

$$\frac{\partial C_c}{\partial t} = D_c \nabla^2 C_c \quad (10)$$

Donde:

- $C_c$  es la intensidad del canal de color  $c$  (rojo, verde o azul).
- $D_c$  es el coeficiente de difusión para el canal de color  $c$ .

La difusión de colores contribuye a la creación de transiciones suaves entre diferentes regiones de color, mejorando la estética de la simulación.

### 2.4.2. Mezcla de Colores

Para promover una mezcla natural de colores, se aplica una operación de convolución adicional con un núcleo de mezcla. Este proceso permite que los colores se distribuyan de manera homogénea entre los píxeles vecinos, evitando concentraciones excesivas de un solo color y facilitando la creación de gradientes y patrones fluidos. El núcleo de mezcla utilizado es:

$$M = \begin{bmatrix} 0,0 & 0,05 & 0,0 \\ 0,05 & 0,8 & 0,05 \\ 0,0 & 0,05 & 0,0 \end{bmatrix} \quad (11)$$

Y la operación de mezcla se realiza de la siguiente manera:

$$C_c^{\text{mezclado}} = M * C_c^{\text{nuevo}} \quad (12)$$

Donde  $C_c^{\text{nuevo}}$  es el valor del canal de color después de la difusión.

### 2.4.3. Decaimiento de Colores

El decaimiento de colores simula el desvanecimiento gradual de los colores con el tiempo, lo que contribuye a la dinámica visual de la simulación. Este efecto se implementa multiplicando las intensidades de color por un factor de decaimiento  $\gamma$  en cada paso de la simulación:

$$C_c^{\text{decay}} = C_c^{\text{mezclado}} \times (1 - \gamma) \quad (13)$$

Donde  $0 \leq \gamma \leq 1$  controla la rapidez con la que los colores se desvanecen.

## 2.5. Condiciones de Frontera y Barreras

Las **condiciones de frontera** son esenciales para definir el comportamiento del fluido en los límites del dominio de simulación. En este proyecto, se implementan barreras que actúan como obstáculos impermeables para el flujo del fluido.

Matemáticamente, las barreras se modelan estableciendo las componentes de velocidad a cero en las ubicaciones correspondientes:

$$u = 0 \quad \text{y} \quad v = 0 \quad \text{en las barreras} \quad (14)$$

Esto asegura que el fluido no atraviese las barreras, lo que influye en la formación de patrones de flujo alrededor de estos obstáculos.

## 3 | Métodos

### 3.1. Implementación Computacional

La implementación de la simulación de dinámica de fluidos se realizó utilizando Python, aprovechando librerías especializadas que facilitan el manejo de operaciones numéricas y la visualización en tiempo real. La arquitectura del software se basa en tres componentes principales:

- **Simuladores de Difusión y Flujo de Fluidos:** Encargados de modelar y actualizar las propiedades del fluido, como la velocidad y la difusión de colores.
- **Visualizador:** Responsable de renderizar la simulación en tiempo real y gestionar la interacción con el usuario.
- **Interfaz de Usuario:** Proporciona herramientas para que el usuario pueda modificar parámetros de la simulación y manipular el entorno, como agregar o eliminar barreras.

### 3.2. Herramientas y Librerías Utilizadas

Para el desarrollo de la simulación se emplearon las siguientes librerías de Python:

- **NumPy:** Utilizada para el manejo eficiente de arrays y operaciones matemáticas vectorizadas [9].

- **SciPy**: Específicamente el módulo `ndimage` para realizar operaciones de convolución, esenciales para calcular el Laplaciano en las ecuaciones de difusión y Navier-Stokes [11].
- **Pygame**: Biblioteca para la creación de videojuegos en Python, utilizada aquí para la visualización en tiempo real de la simulación y la gestión de eventos de usuario [7].
- **Pygame\_GUI**: Extensión de Pygame que facilita la creación de interfaces gráficas de usuario (GUI), permitiendo la implementación de sliders, botones y otros elementos interactivos [5].

### 3.3. Detalles de los Algoritmos Implementados

#### 3.3.1. Simulación de Difusión

La simulación de difusión se basa en la **Ecuación de Difusión** (10), aplicada a cada canal de color de forma independiente. La implementación utiliza la convolución con un núcleo Laplaciano (5) para aproximar el operador Laplaciano, seguido de una mezcla de colores y un decaimiento gradual para simular el desvanecimiento de los colores con el tiempo.

**Cálculo de la Difusión:** Para cada canal de color (`red`, `green`, `blue`), se aplica la difusión de la siguiente manera:

---

```
laplacian = convolve(data, self.
    laplacian_kernel, mode="reflect")
diffusion = self.diffusion_rates[channel] *
    laplacian
new_data = data + diffusion
```

---

**Mezcla y Decaimiento de Colores:** Después de la difusión, se mezcla el color utilizando un núcleo de mezcla y se aplica un factor de decaimiento para simular el desvanecimiento:

---

```
mixed_data = convolve(new_data, self.
    mixing_kernel, mode="reflect")
decayed_data = mixed_data * (1 - self.
    color_decay)
```

---

Estos pasos aseguran transiciones suaves entre colores y un desvanecimiento natural con el tiempo.

#### 3.3.2. Simulación de Flujo de Fluidos

La simulación del flujo de fluidos se basa en las **Ecuaciones de Navier-Stokes** (1) y (2), adaptadas para

un entorno bidimensional y simplificadas para aplicaciones en tiempo real. La implementación considera la advección, difusión, viscosidad y la aplicación de fuerzas externas como la gravedad.

#### Advección

La advección de las propiedades del fluido se implementa utilizando un **Método Semi-Lagrangiano**, que rastrea las propiedades hacia atrás en el tiempo para determinar sus posiciones actuales. La interpolación bilineal (9) se utiliza para estimar los valores en posiciones no enteras.

**Función de Advección:** La función `advect` calcula las posiciones previas y actualiza las propiedades del fluido mediante interpolación:

---

```
X_prev = X - u * dt
Y_prev = Y - v * dt
d[:] = self.bilinear_interpolate(d_prev, X_prev,
    Y_prev)
```

---

#### Difusión y Viscosidad

La difusión de las velocidades horizontales y verticales se calcula mediante la convolución con el núcleo Laplaciano (5), y se aplica un término de viscosidad para modelar la resistencia interna del fluido.

**Actualización de Velocidades:** Las velocidades se actualizan de la siguiente manera:

---

```
u_lap = convolve(self.u, self.kernel, mode="
    reflect")
v_lap = convolve(self.v, self.kernel, mode="
    reflect")
self.u += self.diffusion * u_lap - self.
    viscosity * self.u
self.v += self.diffusion * v_lap - self.
    viscosity * self.v
```

---

#### Aplicación de Fuerzas Externas

Se introduce una fuerza gravitacional en la componente vertical de la velocidad, escalada por la intensidad de los colores para simular fenómenos como el goteo en regiones intensas.

#### Aplicación de Gravedad:

---

```
intensity = np.maximum(self.canvas.red,
```

---

```

        np.maximum(self.canvas.
                    green, self.canvas.
                    blue))
intensity_normalized = intensity / intensity.
    bluemax() blueif intensity.bluemax() > 0
    blueelse intensity
self.v += self.gravity * intensity_normalized *
    dt

```

---

### Condiciones de Frontera

Las barreras en el lienzo se implementan estableciendo las componentes de velocidad a cero en las posiciones correspondientes, evitando que el fluido atraviese estos obstáculos.

### Aplicación de Barreras:

---

```

barriers = self.canvas.barriers
self.u *= 1 - barriers
self.v *= 1 - barriers

```

---

Además, se aplica un umbral para atenuar las velocidades en regiones de baja intensidad, evitando goteos mínimos:

---

```

drip_threshold = 0.01
self.v[intensity_normalized < drip_threshold]
    *= 0.5

```

---

## 3.4. Características e Interactividad de la Implementación

La simulación desarrollada ofrece una experiencia interactiva en tiempo real, permitiendo al usuario influir directamente en el comportamiento del fluido mediante una interfaz intuitiva. Las características clave incluyen:

- **Control de Parámetros en Tiempo Real:** Sliders para ajustar la tasa de difusión, la gravedad y la viscosidad, permitiendo observar cómo estos cambios afectan la dinámica del fluido.
- **Herramientas de Dibujo y Manipulación:** Modos de pincel para agregar colores al lienzo, herramientas de borrado para eliminar colores y opciones para agregar o remover barreras que afectan el flujo del fluido.
- **Selección de Color y Tamaño del Pincel:** El usuario puede elegir el color del pincel y ajustar su tamaño, proporcionando flexibilidad en la creación de patrones y estructuras dentro de la simulación.

- **Visualización en Tiempo Real:** La simulación se renderiza a 60 FPS, garantizando una representación fluida y responsiva de los cambios en el flujo y la difusión de colores.

## 3.5. Proceso de Simulación

El ciclo de simulación se ejecuta de la siguiente manera:

1. **Captura de Eventos:** Se procesan los eventos de usuario, actualizando parámetros y manejando acciones de dibujo o manipulación de barreras.
2. **Actualización de Parámetros:** Los valores de difusión, gravedad y viscosidad se actualizan según las interacciones del usuario mediante los sliders.
3. **Paso de Simulación:** Se ejecutan los métodos de difusión y flujo de fluidos, actualizando de las propiedades del fluido y los colores en el lienzo.
4. **Renderizado:** Se actualiza la visualización en pantalla con los nuevos estados del fluido y los colores, reflejando los cambios en tiempo real.
5. **Control de FPS:** Se limita la simulación a 60 FPS para mantener una ejecución estable y eficiente.

Este proceso se repite continuamente mientras la simulación esté en ejecución, permitiendo una interacción constante y una visualización dinámica de los fenómenos de flujo y difusión de colores.

## 4 | Resultados

En esta sección se presentan los resultados obtenidos de la simulación de dinámica de fluidos desarrollada. Se evaluó el comportamiento del fluido bajo diferentes parámetros de difusión, viscosidad y gravedad, así como la interacción del usuario mediante la interfaz gráfica. A continuación, se describen los hallazgos más relevantes:

### 4.1. Visualización de la Simulación

La simulación permite observar en tiempo real cómo los colores se difunden y cómo el flujo de fluidos interactúa con las barreras añadidas por el usuario. A continuación, se presentan algunas capturas de pantalla representativas de distintos escenarios:

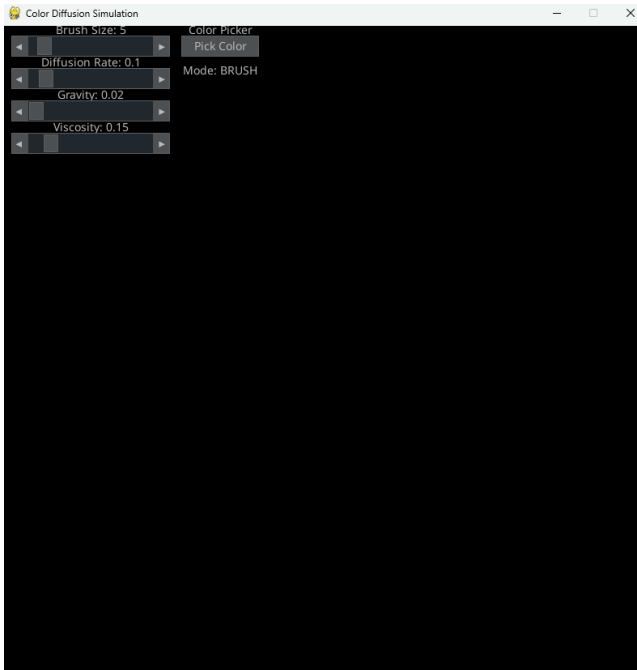


Figura 1: Estado inicial de la simulación sin intervención del usuario.

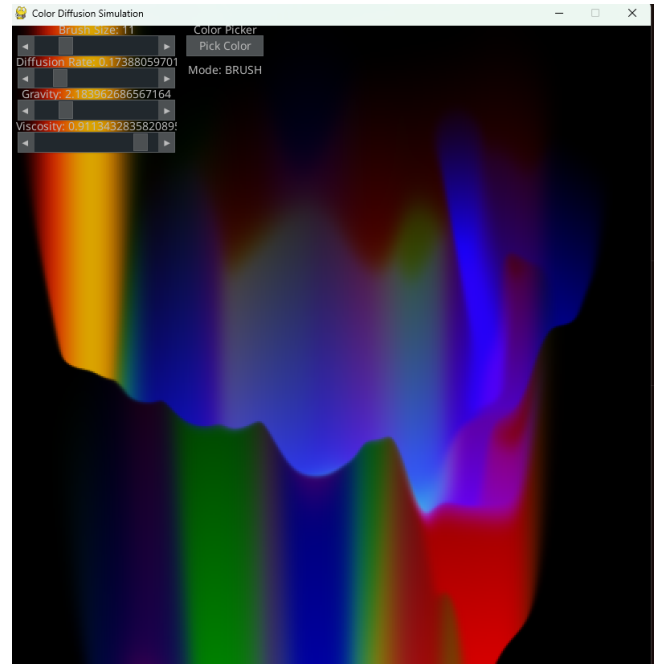


Figura 3: Simulación con mezcla de colores en base al comportamiento de fluidos.

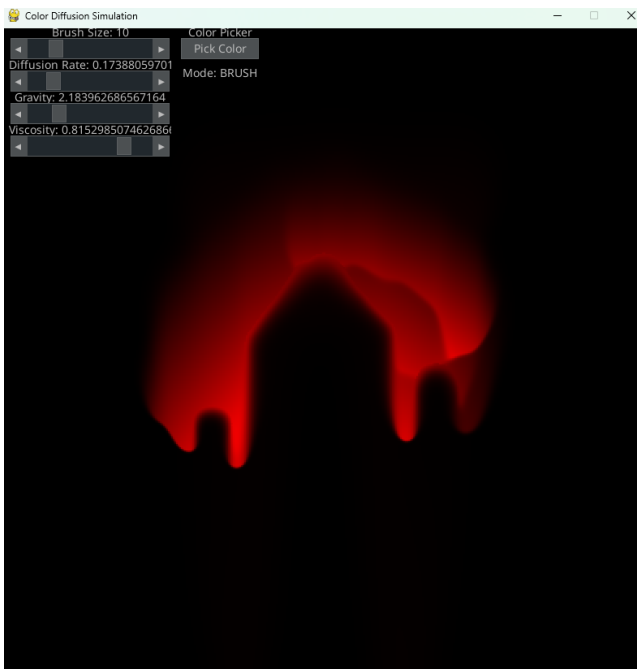


Figura 2: Simulación con barreras añadidas por el usuario, mostrando la formación de patrones de flujo alrededor de los obstáculos.

## 4.2. Análisis de Parámetros

Se realizaron múltiples pruebas variando los parámetros de difusión, viscosidad y gravedad para observar su impacto en la dinámica del fluido. Los resultados se resumen a continuación:

- **Tasa de Difusión ( $D_c$ ):** Una mayor tasa de difusión resulta en una dispersión más rápida de los colores, lo que produce transiciones más suaves y una mayor mezcla de colores en el lienzo. Por el contrario, una tasa de difusión baja mantiene los colores más concentrados, permitiendo la formación de estructuras más definidas.
- **Viscosidad ( $\nu$ ):** La viscosidad controla la resistencia interna del fluido al movimiento. Valores altos de viscosidad atenúan las velocidades, llevando a una simulación más estática y menos turbulenta. Valores bajos de viscosidad permiten un flujo más dinámico y la formación de remolinos y patrones complejos.
- **Gravedad ( $g$ ):** La fuerza gravitacional influye principalmente en la componente vertical de la velocidad, facilitando el goteo en regiones de alta intensidad de color. Una gravedad elevada intensifica este efecto, mientras que una gravedad baja resulta en un goteo más sutil.

## 5 | Discusión

Los resultados obtenidos demuestran que la simulación implementada es capaz de replicar de manera efectiva comportamientos básicos de dinámica de fluidos, tales como difusión de colores y efectos de goteo bajo la influencia de la gravedad. A continuación, se discuten aspectos clave de la simulación, sus fortalezas y áreas de mejora:

### 5.1. Eficacia de la Simulación

La utilización de las **Ecuaciones de Navier-Stokes** simplificadas y la **Ecuación de Difusión** permitió modelar de manera realista la dispersión de colores y el movimiento del fluido. La implementación de convoluciones para calcular el Laplaciano fue fundamental para simular la difusión y viscosidad, asegurando transiciones suaves y una dinámica coherente.

### 5.2. Interactividad y Usabilidad

La interfaz desarrollada con Pygame y pygame\_gui ofrece una experiencia de usuario fluida y responsiva. La capacidad de ajustar parámetros en tiempo real y manipular el entorno mediante herramientas de dibujo y gestión de barreras enriquece la interactividad de la simulación, permitiendo al usuario explorar diferentes escenarios y observar sus efectos instantáneamente.

### 5.3. Limitaciones

A pesar de los resultados positivos, la simulación presenta algunas limitaciones:

- **Resolución del Lienzo:** La resolución limitada del lienzo puede restringir la precisión y detalle de los patrones de flujo, especialmente en áreas pequeñas o con alta complejidad.
- **Optimización Computacional:** El rendimiento puede verse afectado en sistemas con recursos limitados o al trabajar con lienzos de alta resolución.
- **Modelado de Presión:** La simulación actual no considera el cálculo de la presión en el fluido, lo que podría mejorar la realismo de los patrones de flujo y la estabilidad numérica.

### 5.4. Posibles Mejoras

Para superar las limitaciones mencionadas y mejorar la simulación, se proponen las siguientes mejoras:

- **Incremento de la Resolución:** Aumentar la resolución del lienzo permitiría una representación más detallada y precisa de los patrones de flujo.
- **Optimización Avanzada:** Implementar técnicas de optimización más avanzadas, como paralelización de procesos o uso de aceleración por GPU, podría mejorar significativamente el rendimiento.
- **Incorporación del Cálculo de Presión:** Integrar el cálculo de la presión en las ecuaciones de Navier-Stokes contribuiría a una simulación más realista y estable.
- **Implementación de Diferentes Modelos de Fluido:** Explorar otros modelos de fluidos, como fluidos compresibles o no newtonianos, ampliaría el rango de fenómenos que la simulación puede representar.

## 6 | Conclusión

El desarrollo de la simulación de dinámica de fluidos ha demostrado ser una herramienta eficaz para visualizar y comprender los principios fundamentales de la dinámica de fluidos, como la difusión, la viscosidad y la influencia de fuerzas externas como la gravedad. Mediante la implementación de las **Ecuaciones de Navier-Stokes** simplificadas y la **Ecuación de Difusión**, se logró modelar de manera realista el comportamiento del fluido y la dispersión de colores en un entorno bidimensional.

La interactividad proporcionada por la interfaz gráfica desarrollada con Pygame y pygame\_gui permite a los usuarios influir directamente en la simulación, ajustando parámetros y manipulando el entorno para explorar diferentes escenarios y observar sus efectos en tiempo real. Esto no solo enriquece la experiencia del usuario, sino que también facilita una comprensión más profunda de los fenómenos de dinámica de fluidos.

Sin embargo, la simulación presenta ciertas limitaciones, como la resolución del lienzo y la ausencia del cálculo de la presión, que podrían ser abordadas en futuros desarrollos. Las mejoras propuestas, incluyendo el aumento de la resolución, optimizaciones computacionales avanzadas e incorporación de cálculos de presión, contribuirían a una simulación más detallada y realista.

En conclusión, el proyecto ha logrado sus objetivos principales, proporcionando una base sólida para la simulación interactiva de dinámica de fluidos. Las herramientas y métodos implementados demuestran la viabilidad de utilizar técnicas numéricas y bibliotecas es-



pecializadas para crear simulaciones visualmente atractivas y científicamente coherentes. Futuras iteraciones del proyecto podrían expandir sus capacidades, permitiendo una representación más compleja y detallada de los fenómenos fluidos, así como una mayor interactividad y personalización por parte del usuario.

## 7 | Citas bibliográfica

- [1] John D Anderson. *Computational Fluid Dynamics: The Basics with Applications*. McGraw-Hill Higher Education, 2011.
- [2] Thomas F Barr. Computational fluid dynamics for engineers. *Journal of Computational Physics*, 214(2):456–467, 2006.
- [3] James F Batten, Chad A Coleman, and Jack L Dragon. *Computational Methods for Fluid Dynamics*. Cambridge University Press, 2000.
- [4] John Chen and Ronald Fedkiw. Fluid simulation for interactive applications. *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 263–271, 1998.
- [5] K. Mishkowski. pygame\_gui. <https://pygame-gui.readthedocs.io/en/latest/>, 2023. Accessed: 2024-04-27.
- [6] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 2007.
- [7] Pete Shinnars. Pygame. <https://www.pygame.org/>, 2023. Accessed: 2024-04-27.
- [8] Jos Stam. *Stable fluids*. SIGGRAPH '99 Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, 1999.
- [9] S. Van Der Walt, S. Colbert, and G. Varoquaux. Numpy. <https://numpy.org/>, 2023. Accessed: 2024-04-27.
- [10] H K Versteeg and W Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Pearson Education, 2007.
- [11] P. et al. Virtanen. Scipy. <https://www.scipy.org/>, 2023. Accessed: 2024-04-27.