



Diseño, implementación y caracterización numérica de un modelo WCSPH para el vaciado de tanques en dos dimensiones

Harold David Vergel Sánchez

Trabajo de grado para optar al título de Astrónomo

Asesores

Hernán David Salinas Jiménez, PhD

Luis Fernando Quiroga Peláez, PhD

Universidad de Antioquia

Instituto de Física

Pregrado en Astronomía

Medellín

2026

Cita	(Vergel Sánchez, 2026)
Referencia	Vergel Sánchez, H. D. (2026). <i>Diseño, implementación y caracterización numérica de un modelo WCSPH para el vaciado de tanques en dos dimensiones</i> [Trabajo de grado de pregrado, Investigación]. Universidad de Antioquia, Medellín, Ciudad Universitaria.
Estilo APA 7 (2020)	



Biblioteca Carlos Gaviria Díaz.

Repositorio Institucional:

<http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia – www.udea.edu.co

Rector: Héctor Iván García García.

Decana: Adriana Echavarría.

Director de Instituto: Jorge Mario Osorio Guillén.

Coordinador de Pregrado: Jhon Jairo Zuluaga Quintero.

El contenido de esta obra corresponde al derecho de expresión del autor y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. El autor asume la responsabilidad por los derechos de autor y conexos.

Agradecimientos

Me gustaría empezar agradeciendo a mis asesores Hernán David Salinas y Luis Fernando Quiroga, por su apoyo, amistad y disposición que tuvieron durante todo este proyecto, que comenzó desde hace ya unos años en las clases del profesor Luis Quiroga y que nunca permitió que la idea de trabajar en simulación computacional muriera, a pesar de las dificultades; también agradezco al profesor Hernán por apoyar esta idea desde el primer momento. Les agradezco profundamente todo su tiempo y dedicación en las largas jornadas que fueron necesarias para la realización de la tesis, así como por los consejos que me dieron en cada una de las decisiones que tomamos. Sin su ayuda y ambición de investigar, este proyecto no se habría llevado a cabo.

A todos y cada uno de mis compañeros de la carrera de Astronomía, especialmente a mi grupo de amigos Astrococos, por hacerme sentir parte de un grupo incluso siendo foráneo. Al resto de compañeros de instituto, facultad y universidad, por su compañía y también comprensión en momentos donde la vida era compleja de procesar; hicieron de la vida universitaria mucho más amena. A Jhon Alejandro de la oficina de programación del Instituto de Física de la universidad, y a quienes fueron mis compañeros y maestros en ese trabajo. A quienes me apoyaron en el emprendimiento de *Bolegancho* que me sostuvo estos últimos años de carrera, permitiéndome llevar a cabo este trabajo. También a aquellas personas que, por un motivo u otro, ya no hacen parte de mi vida, pero que guardo con cariño en la memoria, pues en mayor o menor medida construyeron lo que soy hoy.

A mis queridos abuelos, estén o no en este mundo, por enseñarme y acompañarme durante toda la vida; a mis padres que me apoyaron en este proyecto tan largo como fue mi carrera. Al resto de mis tíos, primos y demás familiares que siempre me demostraron su confianza y cariño. A mi hermano que siempre estuvo a mi lado cuando regresaba al pueblo.

Finalmente, a todas las personas que me apoyaron, instruyeron, aconsejaron o se interesaron por este proyecto, gracias por darme el impulso necesario para terminarlo.

Índice general

1. Agradecimientos	I
2. Resumen	VIII
3. Abstract	IX
4. Introducción	1
5. Objetivos	3
5.1. Objetivos Generales:	3
5.2. Objetivos Específicos:	3
6. Mecánica de fluidos	4
6.1. Fundamentos de la mecánica de fluidos	4
6.1.1. Descripción Lagrangiana y Euleriana de los fluidos	4
6.1.2. Conservación de la masa	4
6.1.3. Conservación del momento	5
6.1.4. Viscosidad	5
6.1.5. Hidrostática y equilibrio de fluidos	6
6.1.6. Flujo ideal bajo gravedad	7
6.1.7. Flujos de superficie libre: descripción física y complejidad matemática.	9
7. Métodos numéricos de simulación hidrodinámica.	11
7.1. Métodos numéricos para flujos de superficie libre	11
7.1.1. Métodos basados en malla	11
7.1.2. Métodos libres de malla y ventaja del enfoque lagrangiano	12
7.2. Método <i>Smoothed Particle Hydrodynamics</i> (SPH)	13
7.2.1. Representación del dominio y discretización por partículas	13
7.2.2. Funciones de suavizado (kernels) y criterios de selección	14

7.3.	Formulaciones del método SPH	16
7.3.1.	SPH incompresible (ISPH)	16
7.3.2.	SPH débilmente compresible (WCSPH)	17
7.4.	Formulación matemática del método WCSPH implementado	17
7.4.1.	Aproximación SPH y ecuaciones de conservación	17
7.4.2.	Modelo de presión y control de compresibilidad	18
7.4.3.	Modelado de la viscosidad y fuerzas externas	20
7.4.4.	Corrección en la velocidad usando el modelo de XSPH	21
7.4.5.	Integración temporal en SPH	22
7.5.	Tratamiento de condiciones de frontera	23
7.5.1.	Fronteras sólidas y tratamiento numérico	23
7.5.2.	Fronteras de partículas fantasma	24
7.5.3.	Modelos de interacción fluido-frontera adicionales	24
7.6.	Resolución espacial, paso del tiempo y condición CFL	25
7.7.	Métodos de búsqueda de partículas vecinas.	26
7.7.1.	Método de fuerza bruta	27
7.7.2.	Método de árbol cuaternario	27
8.	Diseño y arquitectura del sistema	28
8.1.	Arquitectura general del sistema	28
8.1.1.	Visión general del sistema	28
8.1.2.	Diagrama de Contexto	28
8.1.3.	Diagrama de casos de uso.	29
8.1.4.	Requisitos no funcionales.	29
8.1.5.	Diagrama de descomposición funcional	30
8.1.6.	Diagrama de componentes	30
9.	Metodología	32
9.1.	Algoritmo y flujo general de ejecución de una simulación	32
9.1.1.	Implementación del módulo de condiciones iniciales	33

9.1.2.	Migración e implementación del módulo algorítmico	34
9.1.3.	Módulo de configuración	34
9.2.	Selección del modelo hidrodinámico (tipo de SPH)	34
9.2.1.	Modelo hidrodinámico adoptado	35
9.2.2.	Construcción de geometrías de prueba	36
9.3.	Pruebas experimentales y parámetros implementados	37
9.3.1.	Pruebas iniciales	38
9.3.2.	Pruebas con la ecuación de estado tipo Tait	38
9.4.	Proceso de relajación inicial	39
9.4.1.	Determinación de la estabilidad con base a la densidad	39
9.4.2.	Barrido de parámetros de presión B y velocidad del sonido c	40
9.5.	Validación numérica de leyes físicas	41
9.5.1.	Validación ley de Torricelli	41
9.5.2.	Vaciado por vasos comunicantes	42
9.6.	Tiempo de simulación dependiendo de método de búsqueda de vecinos	42
10.	Resultados	43
10.1.	Pruebas experimentales y parámetros implementados	43
10.1.1.	Pruebas iniciales	43
10.1.2.	Pruebas con la ecuación de estado tipo Tait	44
10.2.	Resultados del barrido del parámetro de presión de la ecuación de Tait B y de la velocidad del sonido artificial c	47
10.3.	Simulaciones estables de validación numérica en base a leyes físicas	51
10.3.1.	Vaciado por caída libre (ley de Torricelli)	51
10.3.2.	Vaciado por vasos comunicantes	55
11.	Discusión de los resultados	58
12.	Perspectivas	59
13.	Conclusiones	60

A. Apéndice A: Perfil de usuarios	64
B. Apéndice B: Historias de usuario.	65
C. Apéndice C: Requisitos funcionales.	67
D. Apéndice D: Requisitos no funcionales.	74

Índice de figuras

6.1. Geometría vasos comunicantes, corto longitudinal (ejes x,y).	7
6.2. Geometría aplicable a la ley de Torricelli, corte longitudinal (ejes x,y).	8
8.1. Digrama de contexto para implementación final	29
8.2. Digrama de casos de uso	30
8.3. Diagrama de descomposición funcional	30
8.4. Diagrama de componentes	31
9.1. Campo de velocidad en el <i>Shear driven cavity problem</i> con la implementación de código anterior para 4000 partículas de fluido.	32
9.2. Diagrama de flujo del sistema implementado para simulaciones individuales.	33
9.3. Geometría de los experimentos numéricos con leyes físicas	37
9.4. Distribución inicial de las partículas bajo la geometría aplicada para probar la ley de Torricelli	38
10.1. Resultados obtenidos al emplear la ecuación de estado lineal	43
10.2. Geometría relajada de las partículas de fluido con la ecuación de Tait y los parámetros de la literatura	44
10.3. Variación de la densidad con cada paso de tiempo, simulación con ecuación de Tait y parámetros en base a la literatura.	45
10.4. Pruebas de comportamiento para la ecuación de Tait con parámetros de la literatura.	45
10.5. Análisis del último paso estable en la relajación de la geometría de Torricelli, utilizando parámetros de la literatura y presión extrapolada a fronteras.	46
10.6. Proceso de relajación del fluido usando los parámetros iniciales encontrados de forma empírica por López (2025)	47
10.7. Estabilización y vaciado de tipo Torricelli con los parámetros empíricos: $B = 1 Pa$ y $c = 1 \times 10^{-2} m/s$	48
10.8. Gráfica de velocidad del sonido vs densidad promedio [kg/m^3] ejecutando 500 pasos de tiempo.	48
10.9. Gráfica de velocidad del sonido vs densidad promedio [kg/m^3] ejecutando 3000 pasos de tiempo.	49
10.10 Gráfica del parámetro de presión B vs densidad promedio [kg/m^3] ejecutando 500 pasos de tiempo.	49

10.11	Gráfica del número de paso de tiempo vs. densidad promedio [kg/m^3] del fluido, simulación con parámetros empíricos.	50
10.12	Escape de partículas durante la simulación usando el parámetro empírico de la ecuación de estado tipo Tait $B = 1.27 Pa$	51
10.13	Relajación del fluido para los parámetros encontrados empíricamente $B = 0.48 Pa$ y $c = 1 \times 10^{-3} m/s$	52
10.14	Vaciado de tanque usando la geometría de Torricelli y parámetros empíricos, paso de tiempo 1000.	52
10.15	Partículas consideradas para los cálculos de validación de la ley de Torricelli.	53
10.16	Trayectorias de las partículas seleccionadas para el cálculo de Torricelli, entre la salida del recipiente superior y antes del rebote en el inferior.	54
10.17	Evolución temporal de las superficies libres del fluido en los vasos durante la simulación.	56
10.18	Mapas de color de la presión del fluido en distintos instantes de la simulación.	57

Resumen

El estudio de flujos de superficie libre mediante métodos numéricos presenta desafíos asociados al tratamiento de la presión, las fronteras y la estabilidad numérica. En este trabajo se diseña, implementa y caracteriza numéricamente un prototipo de simulación hidrodinámica basado en el método *Smoothed Particle Hydrodynamics* (SPH), utilizando la formulación débilmente compresible (WCSPH), aplicado al problema del vaciado de tanques.

A partir del análisis del estado del arte, se desarrolló una arquitectura de software modular y escalable que permite la construcción de geometrías representativas del vaciado de tanques, basado en un corte longitudinal y asociado a una representación en rectángulos, y el estudio sistemático de los parámetros del modelo, centrándose en aquellos asociados a la ecuación de estado, en particular el parámetro de presión base B y la velocidad artificial del sonido c . Mediante experimentos computacionales y barridos se analizó la sensibilidad del método a los parámetros, los modelos de presión, la velocidad del sonido artificial y la interacción con fronteras. Se observó que valores comúnmente reportados en la literatura no garantizan regiones amplias de estabilidad, especialmente al emplear fronteras discretizadas con una sola línea de partículas.

La validación del modelo se realizó mediante la simulación de la ley de Torricelli y del principio de vasos comunicantes, mostrando un comportamiento físicamente coherente, aunque con desviaciones respecto a los modelos ideales debido a efectos viscosos, discretización del fluido e interacción fluido-frontera. Los resultados confirmaron la capacidad del modelo WCSPH para reproducir comportamientos físicos realistas, poniendo en evidencia sus limitaciones y la necesidad de mejoras en el tratamiento de fronteras y en la calibración de parámetros para simulaciones más estables y de mayor complejidad.

Palabras clave: *Smoothed Particle Hydrodynamics* (SPH), WCSPH, flujos de superficie libre, vaciado de tanques, simulación hidrodinámica, métodos numéricos.

Abstract

The study of free-surface flows using numerical methods presents challenges related to the treatment of pressure, boundaries, and numerical stability. In this work, a hydrodynamic simulation prototype based on the Smoothed Particle Hydrodynamics (SPH) method is designed, implemented, and numerically characterized, using the weakly compressible formulation (WCSPH), and applied to the problem of tank draining.

Based on the analysis of the state of the art, a modular and scalable software architecture was developed that allows the construction of geometries representative of tank draining, based on a longitudinal section and associated with a rectangular representation, as well as the systematic study of the model parameters, focusing on those related to the equation of state, in particular the base pressure parameter B and the artificial speed of sound c . Through computational experiments and parameter sweeps, the sensitivity of the method to the parameters, pressure models, artificial speed of sound, and boundary interaction was analyzed. It was observed that values commonly reported in the literature do not guarantee wide regions of stability, especially when using boundaries discretized with a single line of particles.

The validation of the model was carried out through the simulation of Torricelli's law and the principle of communicating vessels, showing physically consistent behavior, although with deviations from ideal models due to viscous effects, fluid discretization, and fluid–boundary interaction. The results confirmed the ability of the WCSPH model to reproduce realistic physical behaviors, highlighting its limitations and the need for improvements in boundary treatment and parameter calibration for more stable and more complex simulations.

Keywords: Smoothed Particle Hydrodynamics (SPH), WCSPH, free-surface flows, tank draining, hydrodynamic simulation, numerical methods

Introducción

El estudio de procesos hidrodinámicos, de gran utilidad en el contexto de ciencia e ingeniería, representan grandes desafíos para implementarse en problemas reales debido a que requiere cálculos numéricos complejos, dado que las ecuaciones de Navier-Stokes no poseen solución analítica general, excepto para casos muy particulares. Su utilidad engloba desde entornos astrofísicos, hasta aplicaciones de física e ingeniería en entornos cotidianos. Para estudiar fenómenos hidrodinámicos se requiere de un análisis teórico sólido y de simulación computacional, con sus respectivas aproximaciones o enfoques. En el caso de los flujos de superficie libre, en los cuales el fluido no está completamente atrapado en un recipiente, se presentan complejidades inherentes a los fenómenos como grandes deformaciones, interfaces móviles y comportamientos no lineales, debido a estas complejidades, se han desarrollado modelos de simulación hidrodinámicos libres de malla que permitían hacer estudio directo sobre el fluido haciendo cálculos respecto a las partículas de material, en contraste con los enfoques Eulerianos tradicionales, que describen el flujo sobre dominios espaciales fijos.

El método de simulación computacional SPH (*Smoothed Particle Hydrodynamics*) es una técnica numérica de tipo Lagrangiana, libre de mallas, diseñada para modelar medios continuos mediante partículas discretas. Por lo que inicialmente fue pensado para aplicarse en astrofísica como se menciona en Liu and Liu (2003) y que aún es usado a día de hoy, especialmente en simulaciones cosmológicas (Springel, 2010). Sin embargo, el modelo ha ido evolucionando para implementarse en otra gran variedad de aplicaciones tanto en física como en ingeniería (Ye et al., 2019; Vacondio et al., 2021).

Dentro de las distintas formulaciones del método SPH, tenemos el enfoque débilmente compresible (Weakly Compressible SPH, WCSPH) que se caracteriza por calcular el valor de la presión para cada partícula de forma explícita, mediante una ecuación de estado y una velocidad del sonido artificial Monaghan (1994).

Sin embargo, una implementación completa y validación exhaustiva del método WCSPH en aplicaciones de mediana o gran escala (como pueden ser inundaciones urbanas, rupturas de presas o mezcla de varios materiales), requiere amplios recursos computacionales, datos experimentales de contraste y procesos de calibración de los parámetros propios de las ecuaciones implementadas, que dependen también de la escala física del sistema a simular y que normalmente no se encuentran disponibles por completo en la literatura de forma libre. De esta forma resulta importante realizar una revisión del método WCSPH a través de problemas estándar, en este caso, el vaciado de tanques como una primera aproximación a modelos más complejos asociados a flujos de superficie libre de inundaciones, capacidades de flujo y desagüe, entre otros. Para realizar dicho estudio es necesario planear un *software* base que permita la aplicación y construcción de geometrías generalizables, de modo que el problema pueda abordarse de forma controlada. Este enfoque permite entender el comportamiento del sistema, implementar un prototipo base y proponer los cambios o calibraciones necesarias para escalar el escenario, sin enfrentarse desde el inicio a la complejidad adicional propia de situaciones más realistas.

Bajo este contexto, el presente trabajo se centra en el diseño e implementación de un *software* de simulación hidrodinámica basado en el método numérico WCSPH (Monaghan, 1994), enfocado en el estudio del proceso de vaciado de tanques. Se aborda la formulación teórica, la implementación computacional y la

caracterización numérica del mismo, estableciendo una base metodológica dirigida a desarrollos posteriores e inserción de modelos de mayor complejidad. El trabajo presenta también un estudio adicional de los lineamientos del desarrollo de *software* asociados al diseño, basado en la guía metodológica (López Ramírez and Quiroga Peláez, 2025), que se suma a la definición de sistemas y problemas representativos en la validación numérica, la determinación de los parámetros que rigen la estabilidad del método, y la retroalimentación de estos.

Finalmente, el documento se organiza de la siguiente manera: en el Capítulo 2 se presentan los objetivos generales y específicos que delimitan el alcance del trabajo y establecen los criterios de evaluación del modelo implementado. El Capítulo 3 desarrolla el marco teórico necesario, comenzando con los fundamentos de la mecánica de fluidos y los flujos de superficie libre, para luego introducir los métodos numéricos existentes y profundizar en la formulación del método SPH, con especial énfasis en su variante débilmente compresible (WCSPH) y en los modelos asociados de presión, viscosidad, integración temporal y tratamiento de fronteras. Asimismo, se discuten aspectos clave para la estabilidad numérica, como la resolución espacial, el paso de tiempo y los métodos de búsqueda de vecinos.

En el Capítulo 4 se describe la metodología adoptada, tanto desde el punto de vista científico como computacional. Se detallan las decisiones de modelado, la construcción de geometrías de prueba, la arquitectura general del *software* desarrollado y los módulos implementados, junto con el proceso de calibración y relajación inicial del sistema. También se presentan las pruebas experimentales realizadas, el barrido de parámetros relevantes del modelo WCSPH y los procedimientos de validación numérica mediante problemas físicos clásicos, como la ley de Torricelli y el sistema de vasos comunicantes.

El Capítulo 5 expone los resultados obtenidos a partir de las simulaciones realizadas, analizando la estabilidad del método, la influencia de los parámetros de presión y velocidad del sonido, el comportamiento del fluido en presencia de fronteras sólidas y la eficiencia computacional de los métodos de búsqueda de vecinos. Estos resultados permiten evaluar tanto la fidelidad física del modelo como las limitaciones inherentes a la formulación adoptada y a los modelos de frontera implementados.

Tenemos entonces que este trabajo presenta una implementación funcional inicial del método WCSPH aplicada al vaciado de tanques, proporcionando un análisis de capacidades y complejidades, y estableciendo una base sólida para futuras extensiones del *software* hacia simulaciones de mayor complejidad, que pueden presentarse como: escalas más realistas, modelos físicos adicionales, mejoras en el tratamiento de fronteras, efectos de tensión superficial o flujos multifásicos.

Objetivos

5.1. Objetivos Generales:

Implementar un prototipo y caracterizar numéricamente un modelo de simulación hidrodinámica basado en el método WCSPH para el estudio del proceso de vaciado de tanques.

5.2. Objetivos Específicos:

1. Analizar el estado del arte del método de simulación hidrodinámica SPH, enfocándonos en su formulación débilmente compresible (WCSPH).
2. Construir el diseño de *software* necesario para la simulación de vaciado de tanques, de forma generalizable y escalable, permitiendo su aplicación posterior en geometrías y modelos de mayor complejidad.
3. Construir una primera versión del código mediante un prototipo enfocado a geometrías representativas del problema y validables numéricamente.
4. Analizar el comportamiento físico, la estabilidad y la influencia de los parámetros del modelo WCSPH.
5. Evaluar los resultados obtenidos mediante el prototipo, usando validaciones numéricas y comparándolas con la teoría física clásica del vaciado de tanques.

Mecánica de fluidos

6.1. Fundamentos de la mecánica de fluidos

La dinámica de fluidos estudia el comportamiento de los fluidos en movimiento, y la interacción de éstos con sólidos o con otros fluidos en las fronteras. (Çengel and Cimbala, 2018, p. 2). Desde un punto de vista continuo, los fluidos se describen mediante campos físicos como la velocidad, la presión y la densidad, los cuales dependen del espacio y del tiempo (White, 2017).

6.1.1. Descripción Lagrangiana y Euleriana de los fluidos

Existen dos enfoques fundamentales para describir el movimiento de un fluido: la formulación Euleriana y la formulación Lagrangiana. La descripción lagrangiana se basa en calcular el vector posición y velocidad asociado a cada una de las partículas de fluido o agrupaciones de este; mientras que la descripción euleriana define un volumen de control, a través del cual es fluido entra y sale, y al cual se le definen variables de campo (Çengel and Cimbala, 2018, p 136-170).

El método *Smoothed Particle Hydrodynamics* (SPH) adopta una formulación puramente Lagrangiana, libre de mallas en la cual el fluido se discretiza mediante partículas que transportan masa, momento y otras propiedades físicas, cuyas derivadas son calculadas mediante una función de interpolación. Esta característica permite una representación de grandes deformaciones, interfaces móviles y fragmentación del fluido, aspectos que resultan muchas veces complejos en métodos Eulerianos tradicionales (Liu and Liu, 2003; Monaghan, 1992).

6.1.2. Conservación de la masa

Basándonos en la formulación presentada en (Lautrup, 2011, pp. 194-195), y sabiendo que para un sistema físico cerrado la masa siempre permanece constante. Tenemos que, en el formalismo Euleriano es necesario tener en cuenta que la masa que sale de un volumen debe ingresar en otro, lo cual puede expresarse como:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0, \quad (6.1)$$

donde ρ es la densidad, t es el tiempo y \vec{v} es el vector velocidad.

Esta es la ecuación local de la conservación de la masa o también llama *ecuación de continuidad*, mientras que en el formalismo lagrangiano debido a que el observador sigue al fluido estará dada en base a la derivada comóvil de la densidad, llamada tasa de cambio comóvil de la densidad y por lo tanto la ecuación de continuidad en este caso será:

$$\frac{D\rho}{Dt} = -\rho \nabla \cdot \vec{v}. \quad (6.2)$$

Donde D/Dt la derivada material, que está dada por:

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \vec{v} \cdot \nabla. \quad (6.3)$$

6.1.3. Conservación del momento

Como la mecánica de medio continuo y en nuestro caso la mecánica de fluidos sigue las mismas leyes de la mecánica clásica, tenemos entonces que la segunda ley de Newton es aplicable en la forma vectorial de las tres ecuaciones asociadas a la aceleración, que en su forma Lagrangina están dadas por:

$$\rho \frac{D\vec{v}}{Dt} = -\nabla P + \mu \nabla^2 \vec{v} + \left(\zeta + \frac{1}{3} \mu \right) \nabla(\nabla \cdot \vec{v}) + \vec{f}_{ext}, \quad (6.4)$$

donde P es la presión absoluta y μ es la viscosidad dinámica, ζ la viscosidad volumétrica y \vec{f}_{ext} reúne todas las fuerzas externas que se aplican por unidad de masa, es decir, en su efecto de aceleración.

La presión de esta ecuación se resuelve mediante una ecuación de estado para darle solución a la ecuación de momentum (Fraga Filho, 2019, p. 12).

Cuya forma Euleriana aparecen en (Lautrup, 2011, p. 256) y corresponde a:

$$\rho \left(\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla) \vec{v} \right) = -\nabla P + \eta \nabla^2 \vec{v} + \left(\zeta + \frac{1}{3} \eta \right) \nabla(\nabla \cdot \vec{v}) + \vec{f}_{ext}. \quad (6.5)$$

6.1.4. Viscosidad

Anteriormente teníamos que los valores de viscosidad aparecían como escalares, pero físicamente la viscosidad se comporta como un tensor y proviene del tensor de deformación (Lautrup, 2011, p. 256):

$$\sigma_{ij} = -P \delta_{ij} + \mu \left(\nabla_i v_j + \nabla_j v_i - \frac{2}{3} \nabla \cdot \vec{v} \delta_{ij} \right) + \zeta \nabla \cdot \vec{v} \delta_{ij}, \quad (6.6)$$

donde σ_{ij} , es el tensor de deformación, δ_{ij} es el delta de Kronecker.

La parte derecha de la ecuación anterior, se puede dividir en el primer término que corresponde a la parte isotrópica (asociada a la presión) y los demás términos que asociados a la viscosidad, si extraemos estos últimos, es decir, los que poseen los valores de μ y ζ , tenemos:

$$\tau_{ij} = \mu (\nabla_i v_j + \nabla_j v_i) + \left(\zeta - \frac{2}{3} \mu \right) \nabla \cdot \vec{v} \delta_{ij}, \quad (6.7)$$

donde τ_{ij} representa el tensor de viscosidad.

6.1.5. Hidrostática y equilibrio de fluidos

La hidrostática estudia el comportamiento de los fluidos en reposo y constituye un caso particular de la mecánica de fluidos en el cual las velocidades macroscópicas son nulas. En este régimen, las fuerzas dominantes que actúan sobre el fluido son la gravedad y las fuerzas de presión, mientras que los efectos viscosos no juegan un papel relevante en el equilibrio mecánico del sistema Batchelor (1967); White (2017).

Bajo la hipótesis de un fluido continuo, homogéneo e incompresible, el equilibrio hidrostático se describe mediante la ecuación fundamental de la hidrostática, que relaciona el gradiente de presión con el campo gravitatorio:

$$\nabla P = \rho \vec{g}, \quad (6.8)$$

donde P es la presión, ρ la densidad del fluido y \vec{g} la aceleración de la gravedad (Landau and Lifshitz, 1987, p. 5).

En el caso unidimensional, con el eje vertical z orientado positivamente hacia arriba, esta expresión se reduce a:

$$\frac{dP}{dz} = -\rho g. \quad (6.9)$$

La integración de la ecuación anterior conduce a la conocida relación lineal entre presión y profundidad:

$$P(z) = P_0 + \rho g (h - z), \quad (6.10)$$

donde P_0 es la presión de la superficie libre del fluido, ubicada a la altura h y g la magnitud de la aceleración gravitacional (Serway and Jewett, 2008, p. 392). Esta expresión muestra que, en un fluido en reposo, la presión depende únicamente de la profundidad y no de la forma del recipiente, resultado que constituye la base física de diversos fenómenos hidrostáticos clásicos.

Principio de vasos comunicantes

El principio de los vasos comunicantes establece que, en un conjunto de recipientes conectados entre sí por su parte inferior y que contienen el fluido en reposo, el nivel de la superficie libre se alcanza a la misma altura en todos los recipientes, independientemente de su forma o sección transversal (White, 2017, pp. 61-64). Una representación simplificada de este problema que va a ser usada más adelante se presenta en la *figura 6.1*.

Considerando dos recipientes comunicados por una tubería arbitraria, sea z_c la altura del punto en la que ambas se conectan o el punto medio entre la tubería, este punto pertenece al fluido continuo y por su continuidad debe ser el mismo medido desde cualquiera de los dos recipientes, esto hace que:

Vaciado entre tanques. Prueba de vasos comunicantes

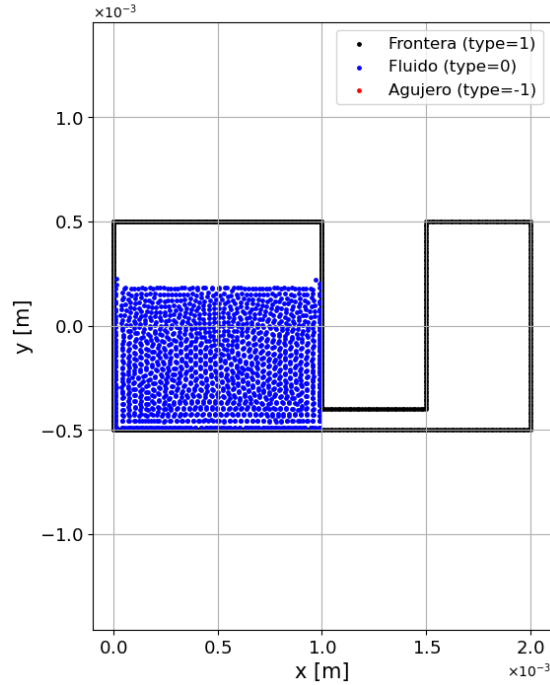


Figura 6.1: Geometría vasos comunicantes, corto longitudinal (ejes x,y).

$$P^{(1)}(z_c) = P^{(2)}(z_c), \quad (6.11)$$

donde $P^{(1)}(z_c)$ es la presión del punto z_c que conecta ambos vasos medida desde el vaso 1; mientras que $P^{(2)}(z_c)$, lo es para el vaso 2.

Sustituyendo las presiones mediante la ecuación 6.10 tenemos que:

$$P_{atm} + \rho g(h_1 - z_c) = P_{atm} + \rho g(h_2 - z_c), \quad (6.12)$$

donde h_1 y h_2 corresponden a la altura del punto z_c en el vaso 1 y 2, respectivamente; mientras que P_{atm} es la presión atmosférica.

Eliminando los términos comunes a ambos lados llegamos a que $h_1 = h_2$, lo cual nos dice que sin importar la forma de los vasos o la altura a la que se encuentren uno del otro si el punto de conexión de ambos hace parte del fluido, su superficie libre estará a la misma altura.

6.1.6. Flujo ideal bajo gravedad

Cuando un fluido no está en el régimen hidrostático y comienza a moverse bajo la acción de la gravedad, su comportamiento puede describirse, en primera aproximación, mediante modelos de flujo ideal. En estos

modelos se asume que el fluido es incompresible y que los efectos viscosos son despreciables, lo que permite obtener expresiones analíticas simples que capturan la física dominante del fenómeno Batchelor (1967); White (2017).

Bajo estas hipótesis, el movimiento del fluido puede analizarse a partir del principio de conservación de la energía mecánica, expresado mediante la ecuación de Bernoulli para un fluido ideal:

$$\frac{P}{\rho} + \frac{v^2}{2} + gz = \text{constante}, \quad (6.13)$$

donde v es la velocidad del fluido y z la coordenada vertical (White, 2017, p. 165).

Ley de Torricelli

Fue formulada originalmente por Evangelista Torricelli en el siglo XVII y puede derivarse directamente de la ecuación de Bernoulli (*ecuación 6.13* (Landau and Lifshitz, 1987; White, 2017)). La ley de Torricelli describe la velocidad de salida de un fluido ideal a través de un orificio ubicado en la pared de un recipiente, cuando el flujo es impulsado únicamente por la gravedad, una representación de una geometría asociado puede verse en la *figura 6.2*.

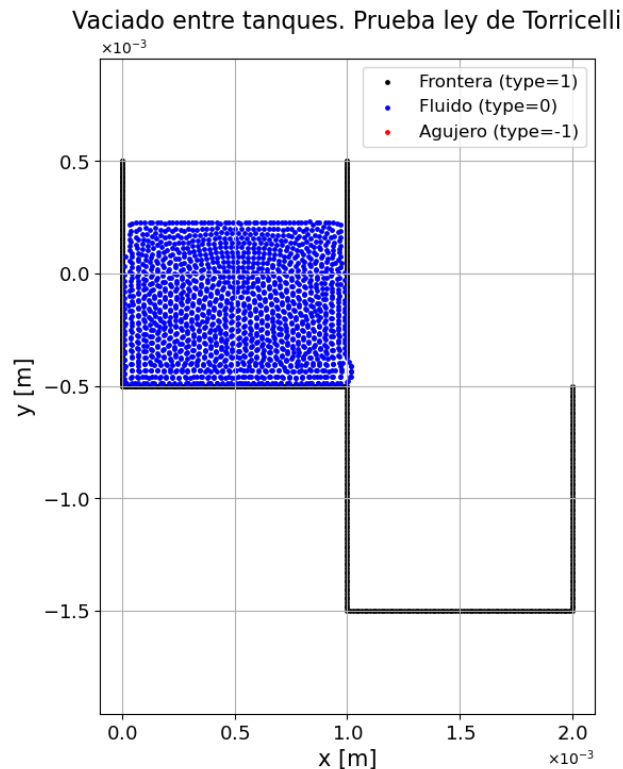


Figura 6.2: Geometría aplicable a la ley de Torricelli, corte longitudinal (ejes x,y).

Considerando un recipiente abierto a la atmósfera, con una superficie libre situada a una altura H por

encima del orificio, y suponiendo que la velocidad en la superficie es despreciable, la aplicación de la ecuación de Bernoulli entre la superficie libre y el orificio conduce a:

$$\frac{P_{atm}}{\rho} + g H = \frac{P_{atm}}{\rho} + \frac{v^2}{2}, \quad (6.14)$$

De donde se obtiene la expresión conocida como ley de Torricelli:

$$v = \sqrt{2gH}. \quad (6.15)$$

Esta ecuación indica que la velocidad de salida del fluido es equivalente a la que conseguiría una partícula en caída libre desde una altura H , mostrando la conversión de energía potencial gravitacional en energía cinética (White, 2017, p. 168-169).

El movimiento que describe el fluido corresponde a un movimiento uniformemente acelerado dado por:

$$y(t) = y_0 + v_0 t + \frac{1}{2} g t^2, \quad (6.16)$$

donde $y(t)$ es la posición de una partícula en el eje y , y_0 su altura inicial y v_0 la velocidad inicial en el eje y .

La ley de Torricelli es válida bajo las siguientes condiciones (White, 2017, p. 165):

1. Fluido ideal e incompresible (sin viscosidad ni disipación de energía mecánica).
2. Flujo estacionario (las magnitudes macroscópicas no dependen explícitamente del tiempo).
3. Orificio pequeño en comparación con el tamaño del recipiente.
4. Pérdidas viscosas y turbulentas despreciables.

6.1.7. Flujos de superficie libre: descripción física y complejidad matemática.

Las superficies libres se definen como la interfaz entre un fluido y una fase gaseosa con presión conocida y constante, que suele ser normalmente la presión atmosférica del lugar, involucrando también la presencia de un campo gravitacional. En condiciones hidrostáticas, dicha superficie adopta una geometría horizontal como consecuencia del equilibrio de fuerzas debido al campo de presión constante (White, 2017, pp. 31-33).

Desde el punto de vista del continuo, una superficie libre constituye una frontera del dominio fluido, con campo de presión constante y donde pueden aparecer discontinuidades en los campos, sin que ello invalide la descripción macroscópica de la dinámica del fluido (White, 2017, p. 35, pp. 133-135).

Desde el punto de vista numérico, los flujos de superficie libre son particularmente complejos de modelar, ya que involucran la solución tanto de las ecuaciones de movimiento como de la determinación de la posición de la interfaz fluido-gas, la cual no es conocida a priori y hace parte de la solución. Estas dificultades se

suman al carácter no lineal de las ecuaciones de momento 6.4 y ecuación de continuidad 6.2, así como por la presencia de grandes deformaciones de la interfaz y la posible aparición de gradientes abruptos en los campos. Como consecuencia, las soluciones analíticas exactas solo existen en configuraciones muy idealizadas, haciendo necesario el uso de métodos numéricos para su estudio (Tryggvason et al., 2011, pp. 1-2).

Métodos numéricos de simulación hidrodinámica.

7.1. Métodos numéricos para flujos de superficie libre

Como se discutía anteriormente, los flujos de superficie libre presentan una interfaz liquido-gas cuya posición evoluciona en el tiempo y deber ser tratada como parte del problema. La modelación numérica de los flujos de superficie libre terminan siendo un caso particular de los flujos con interfaces, donde una de las fases se simplifica significativamente. En el caso del flujo aire-agua, la presión sobre la interface depende únicamente del tiempo y no del espacio, y los esfuerzos viscosos son despreciables, convirtiendo la interface en condiciones de frontera móvil sobre el líquido al cual se le resuelven las ecuaciones de Navier-Stokes y donde la interacción con el gas está mediado solo por las condiciones mencionadas (Scardovelli and Zaleski, 1999, p. 574).

A partir de esta aproximación, se han desarrollado diversos métodos numéricos para la simulación de flujos de superficie libre, los cuales se clasifican de acuerdo con la forma en que representan el dominio computacional y la interfaz fluido-gas. En particular, se encuentran dos enfoques principales: métodos basados en malla y métodos libres de malla, teniendo cada uno ventajas y limitaciones en cuanto a precisión, estabilidad y coste computacional (Scardovelli and Zaleski, 1999; Tryggvason et al., 2011).

7.1.1. Métodos basados en malla

Los métodos basados en malla son uno de los métodos más comunes para la simulación de flujos de superficie libre. Este tipo de enfoques, presenta el dominio como una malla que lo divide, la cual puede ser fija o adaptativa. Las ecuaciones de Navier-Stokes se resuelven mediante diferencias finitas, volúmenes o elementos finitos (Scardovelli and Zaleski, 1999, p. 568). La principal dificultad reside en la precisión con que la interfaz liquido-gas se discretiza y con ella su deformación a medida que el flujo evoluciona.

Dentro de este enfoque, una primera categoría corresponde a los métodos de malla fija, donde la malla permanece inmóvil y no sigue las deformaciones de la interfaz. En este caso, la superficie libre atraviesa la malla y su posición se describe mediante variables auxiliares definidas en las celdas. Ejemplos representativos de este enfoque incluyen los métodos Volume of Fluid (VOF), Level Set y Front Capturing. No obstante, estos métodos suelen presentar dificultades asociadas a la reconstrucción geométrica precisa de la interfaz y la representación de la superficie libre (Scardovelli and Zaleski, 1999, pp. 569-571).

Una segunda categoría corresponde a los métodos de malla móvil o front-tracking, en los cuales la interfaz coincide con las fronteras del mallado, separando los subdominios de cada fase. Este enfoque permite una descripción geométrica más precisa de la superficie libre y un tratamiento más directo de las condiciones de contorno. Sin embargo, cuando la interfaz experimenta grandes deformaciones, es necesario realizar procesos de remallado lo que incrementa significativamente el costo computacional, limitando su aplicabilidad en

problemas altamente dinámicos (Scardovelli and Zaleski, 1999, pp. 571–572).

7.1.2. Métodos libres de malla y ventaja del enfoque lagrangiano

Los métodos libres de mallas están intrínsecamente asociados a la descripción lagrangiana del fluido, pues las partículas o marcadores de las mismas se mueven junto al campo de velocidades del fluido (Tryggvason et al., 2011, p. 114). Se basan en la hipótesis de que el dominio del problema puede ser dividido en un número finito de partículas, cuya interacción se define a través de esquemas de interpolación o aproximación, y cuyas propiedades físicas son evaluadas y actualizadas en cada instante de tiempo (Fraga Filho, 2019).

Dentro de este marco conceptual, se han desarrollado diversos métodos numéricos libres de malla, los cuales pueden diferenciarse tanto por los esquemas de aproximación empleados para representar los campos continuos como por los distintos enfoques adoptados para la formulación de las ecuaciones gobernantes (Belytschko et al., 1996; Nguyen et al., 2008).

Entre los enfoques más antiguos se encuentra el *Smoothed Particle Hydrodynamics* (SPH), propuesto originalmente por Gingold y Monaghan en el contexto de aplicaciones astrofísicas (Liu and Liu, 2003), donde las grandes deformaciones del dominio y la ausencia de fronteras rígidas hacían poco adecuados los métodos basados en malla.

Además de SPH, existen otros métodos libres de malla que han sido desarrollados con el objetivo de mejorar la precisión y estabilidad numérica. Entre ellos se encuentran el método Element-Free Galerkin (EFG), basado en aproximaciones de mínimos cuadrados móviles; el método de Funciones de Base Radial (RBF), que utiliza interpolaciones globales o locales centradas en puntos; y el Reproducing Kernel Particle Method (RKPM), que extiende el concepto de kernel para garantizar propiedades de reproducibilidad (Patel and Rachchh, 2020; Eirís et al., 2023).

Desde un punto de vista general, los métodos libres de malla y el enfoque lagrangiano ofrecen una serie de ventajas frente a los métodos eulerianos tradicionales, particularmente en problemas con geometrías complejas y superficies libres. En particular, de acuerdo con (Fraga Filho, 2019, p. 2), pueden destacarse las siguientes:

1. Las superficies libres y sus deformaciones, las interfaces y las fronteras móviles aparecen de forma natural en la formulación de partículas dentro del proceso de simulación.
2. Para geometrías complejas, la no necesidad de reestructurar la malla hace más simple y menos costosa computacionalmente la simulación.
3. Se cumple la conservación de la masa pues cada elemento permanece invariante y con ella se controla la difusión numérica de esta.
4. Se conserva la historia de cada partícula, entendiendo mejor el comportamiento espacio-temporal.

7.2. Método *Smoothed Particle Hydrodynamics* (SPH)

El método *Smoothed Particle Hydrodynamics* (SPH) es una técnica numérica lagrangiana y libre de malla desarrollada originalmente para el estudio de problemas astrofísicos, que ha sido ampliamente extendida a aplicaciones en dinámica de fluidos e ingeniería debido a su capacidad para modelar grandes deformaciones, superficies libres e interfaces móviles sin necesidad de reconstrucción geométrica del dominio (Gingold and Monaghan, 1977; Liu et al., 2008). En este método, el medio continuo se representa mediante un conjunto discreto de partículas que transportan las propiedades físicas del fluido y cuya interacción se define a través de funciones de interpolación suaves, permitiendo una descripción de la dinámica del sistema desde un enfoque lagrangiano (Liu and Liu, 2003).

Como se había mencionado antes, este método parte de la descripción lagrangiana de las ecuaciones de Navier-Stokes 6.2 y 6.4, en las cuales las variables se describen desde la perspectiva del movimiento de partículas de material. En este enfoque, cada partícula es un ente independiente dentro de la simulación, que transporta propiedades como masa, densidad, velocidad y energía. Esta formulación permite modelar fenómenos complejos con grandes deformaciones, como multifases o superficies libres, que se suma a la facilidad de ser invariante bajo transformaciones Galileanas (Liu et al., 2008).

7.2.1. Representación del dominio y discretización por partículas

Para representar numéricamente los campos continuos del fluido dentro del método SPH, se parte de la representación integral de una función, la cual permite introducir de manera natural la función kernel de suavizado (Liu and Liu, 2003, pp. 35–38). Dicha representación se basa en la identidad asociada a la delta de Dirac ($\delta(\vec{r} - \vec{r}')$), dada por

$$f(\vec{r}) = \int_{\Omega} f(\vec{r}') \delta(\vec{r} - \vec{r}') d\vec{r}', \quad (7.1)$$

donde Ω es dominio físico ocupado por el fluido. Si en vez de usar el delta de Dirac definimos una función más suave $W(x - x', h)$ la cual cumple que:

$$f(\vec{r}) = \int_{\Omega} f(\vec{r}') W(\vec{r} - \vec{r}', h) d\vec{r}'. \quad (7.2)$$

En la ecuación anterior, h es llamada longitud de suavizado y define el área de influencia en la cual la función W actúa. Esta función de kernel debe cumplir la misma propiedad que la delta de Dirac:

$$\int_{\Omega} W(\vec{r} - \vec{r}', h) dx' = 1. \quad (7.3)$$

La función W , también debe cumplir que en el límite cuando h tiende a 0 reencontramos la función delta de Dirac, también debe tener un dominio de soporte dado por κh , el cual cumple que:

$$W(\vec{r} - \vec{r}', h) = 0 \text{ cuando } |\vec{r} - \vec{r}'| > \kappa h. \quad (7.4)$$

En base a esta formulación se pueden definir las funciones asociadas a los campos del fluido y por lo tanto también a sus derivadas. Sin embargo, en la práctica, el dominio continuo Ω se discretiza en un conjunto finito de partículas materiales, cada una asociada a una masa m_j , una densidad ρ_j y una posición \vec{r}_j . Bajo esta discretización, la integral continua puede aproximarse mediante una sumatoria sobre las partículas vecinas dentro del dominio de soporte del kernel. De esta forma, el valor de una magnitud escalar f evaluada en la posición de una partícula i se aproxima como una interpolación basada en las contribuciones de las partículas vecinas, y está dada por la expresión (Liu and Liu, 2003; Ye et al., 2019):

$$f(\vec{r}_i) \approx \sum_j f(\vec{r}_j) \frac{m_j}{\rho_j} W(\vec{r}_i - \vec{r}_j, h). \quad (7.5)$$

7.2.2. Funciones de suavizado (kernels) y criterios de selección

Las funciones de suavizado, o kernels, constituyen un elemento central del método *Smoothed Particle Hydrodynamics* (SPH), ya que determinan la forma en que la información de los campos se transfiere a las partículas vecinas. Como antes se había visto, el kernel puede interpretarse como una aproximación a la delta de Dirac, que permite reconstruir funciones y sus derivadas espaciales a partir de un conjunto discreto de partículas, preservando la naturaleza lagrangiana del método (Liu and Liu, 2003).

Para garantizar la consistencia y estabilidad numérica del método, las funciones kernel deben satisfacer una serie de propiedades fundamentales descritas en Liu et al. (nd); Liu and Liu (2003) (Correcciones de forma y estilo por OpenAI (2025)). Entre ellas se incluyen:

1. **Normalización**, de modo que su integral sobre el dominio de soporte sea la unidad, asegurando la reproducción exacta de campos constantes, como se indica en la ecuación (7.3).
2. **Soporte compacto**, de manera que el kernel sea nulo fuera de un radio finito de influencia proporcional a la longitud de suavizado h (ecuación (7.4)).
3. **Simetría**, es decir, $W(\vec{r}_i - \vec{r}_j) = W(\vec{r}_j - \vec{r}_i)$, garantizando contribuciones equivalentes de partículas situadas a igual distancia.
4. **Positividad**, de manera que $W \geq 0$ dentro del dominio de soporte, lo cual resulta fundamental para evitar la aparición de valores no físicos.
5. **Decaimiento monótono**, de forma que la influencia de una partícula disminuya al aumentar la distancia, reflejando el carácter local de las interacciones físicas.
6. **Consistencia con la delta de Dirac**, de modo que el kernel tienda a la función delta cuando la longitud de suavizado h se aproxima a cero.
7. **Suavidad suficiente**, requiriendo continuidad adecuada del kernel y de sus derivadas para obtener aproximaciones precisas de gradientes y mejorar la estabilidad numérica.

Cuando miramos como se han ido implementando diferentes funciones de kernel de suavizado a través de los artículos de SPH en el tiempo, vemos que inicialmente, en el artículo de Lucy se usó la función *bell-shaped* (Liu et al., nd):

$$W(\vec{r} - \vec{r}', h) = W(R, h) = \alpha_d \begin{cases} (1 + 3R)(1 - R)^3 & R \leq 1, \\ 0 & R > 1 \end{cases} \quad (7.6)$$

Donde α_d es el factor dimensional y tendrá un valor de $5/4h$, $5/\pi h^2$ y $105/16\pi h^3$ para una, dos y tres dimensiones, respectivamente. Mientras que R es la distancia relativa entre dos partículas dada por:

$$R = \frac{r}{h}, \quad (7.7)$$

donde $r = |\vec{r} - \vec{r}'|$.

Mientras que Gingold and Monaghan (Gingold and Monaghan, 1977) usaron un Kernel Gaussiano para estrellas no esféricas. Dicho Kernel está dado por la ecuación:

$$W(R, h) = \alpha_d e^{-R^2}, \quad (7.8)$$

donde α_d es el factor dimensional y tendrá un valor de $1/\pi^{1/2}h$, $1/\pi h^2$ y $1/\pi^{3/2}h^3$ para una, dos y tres dimensiones, respectivamente.

Entre las funciones de suavizado más empleadas en aplicaciones hidrodinámicas se encuentra el kernel spline cúbico propuesto por Monaghan. Este kernel es continuo junto con sus primeras derivadas y posee soporte compacto, características que lo hacen especialmente apropiado para la simulación de flujos con superficies libres e interfaces móviles (Monaghan, 1994; Ye et al., 2019). El kernel spline cúbico se define como:

$$W(R, h) = \frac{\alpha_d}{h^d} \begin{cases} (2 - R)^2 - 1, & 0 \leq R < 1, \\ (2 - R)^2, & 1 \leq R < 2, \\ 0, & R \geq 2, \end{cases} \quad (7.9)$$

donde R fue definido anteriormente en la *ecuación 7.7*, N es la dimensión del dominio y los factores de normalización α_d para las tres posibles dimensiones son: $1/h$, $15/(7\pi h^2)$ y $3/(2\pi h^3)$.

La elección del kernel y del valor de la longitud de suavizado h influye directamente en el número de partículas vecinas involucradas en la interpolación, así como en la precisión y estabilidad del método. Un número reducido de vecinos puede introducir ruido numérico y pérdida de consistencia, mientras que un número excesivo incrementa el costo computacional y puede suavizar en exceso los gradientes del flujo (Liu and Liu, 2003; Dehnen and Aly, 2012).

Con el fin de mejorar la estabilidad del método y evitar problemas como el agrupamiento artificial de partículas, se han desarrollado kernels con mejores propiedades de positividad y regularidad, entre los cuales destacan los kernels de Wendland, basados en los polinomios de Wendland. Estos kernels presentan soporte compacto, son estrictamente positivos y poseen derivadas bien definidas, lo que los hace especialmente adecuados para simulaciones de larga duración y flujos altamente deformables (Dehnen and Aly, 2012).

Desde un punto de vista práctico, la selección del kernel y del parámetro de suavizado h debe considerar un equilibrio entre precisión, estabilidad y costo computacional. En aplicaciones hidrodinámicas, el uso de kernels con soporte compacto y buena regularidad suele ser preferido, siendo el spline cúbico y los kernels de Wendland opciones comunes y ampliamente validadas en la literatura especializada (Dehnen and Aly, 2012).

7.3. Formulaciones del método SPH

Cuando se analiza la literatura sobre el algoritmo *Smoothed Particle Hydrodynamics* (SPH), se identifican principalmente dos enfoques en la formulación de las ecuaciones gobernantes: el SPH Débilmente Compresible (WCSPH) y el SPH Incompresible (ISPH) (Shadloo et al., 2011). La diferencia fundamental entre ambas formulaciones radica en el tratamiento de la presión y en la forma en que se impone la condición de incompresibilidad del fluido.

En el enfoque WCSPH, la presión se calcula de manera explícita mediante una ecuación de estado (EOS) que relaciona la presión con la densidad, introduciendo una velocidad del sonido artificial suficientemente alta para limitar las variaciones de densidad. Por el contrario, en la formulación ISPH la presión se obtiene resolviendo una ecuación de Poisson para la Presión (Pressure Poisson Equation, PPE), lo que permite imponer la condición de incompresibilidad de forma más estricta y obtener campos de presión más suaves que los producidos por las primeras formulaciones de WCSPH (Guan et al., 2022).

7.3.1. SPH incompresible (ISPH)

La formulación SPH incompresible se basa en la resolución de una ecuación de Poisson para la presión con el objetivo de garantizar un campo de velocidades libre de divergencia. Este tratamiento permite reducir significativamente las fluctuaciones artificiales de densidad y mejorar la calidad de los campos de presión obtenidos en la simulación (Shadloo et al., 2011).

Comparaciones realizadas en la literatura indican que, si bien ISPH tiende a producir distribuciones de presión más precisas que el WCSPH clásico, la calidad de las imágenes cinemáticas obtenidas puede ser inferior, especialmente en flujos altamente transitorios o con superficies libres complejas (Afanasiev et al., 2008). Asimismo, aunque ISPH puede resultar computacionalmente más eficiente en determinados escenarios, la necesidad de resolver un sistema elíptico global para la presión incrementa la complejidad de implementación y dificulta su escalabilidad en arquitecturas altamente paralelas (Shadloo et al., 2011).

7.3.2. SPH débilmente compresible (WCSPH)

En la formulación SPH débilmente compresible, la presión se calcula directamente a partir de una ecuación de estado, dando lugar a un esquema completamente explícito. A diferencia de los métodos de proyección que requieren resolver la ecuación de Poisson para imponer incompresibilidad, el enfoque WCSPH permite pequeñas fluctuaciones de densidad, evitando la solución de sistemas globales y reduciendo significativamente el tiempo de cómputo (Becker and Teschner, 2007).

El método mantiene una buena conservación de volumen y maneja de manera natural fenómenos de superficie libre, como salpicaduras y rompimiento de olas. Además, incorpora modelos de tensión superficial basados en fuerzas de cohesión, lo que permite capturar correctamente detalles finos de la interfaz, mejorando la estabilidad y eficiencia numérica (Becker and Teschner, 2007). La naturaleza explícita y local de las interacciones hace que WCSPH sea especialmente adecuado para la paralelización en arquitecturas CPU y GPU. No obstante, la elección del paso de tiempo Δt es crítica para la estabilidad del método. Estudios teóricos y numéricos muestran que la estabilidad depende del tamaño del paso de tiempo en relación con la velocidad del sonido numérica, la viscosidad y la longitud de suavizado (Violeau and Leroy, 2013).

A pesar de permitir pequeñas variaciones de densidad, WCSPH ha demostrado ser una formulación robusta y versátil para la simulación de flujos con grandes deformaciones, impactos de fluidos y superficies libres altamente dinámicas. Estas ventajas han motivado su amplio uso en aplicaciones de ingeniería y dinámica de fluidos computacional basadas en SPH (Shadloo et al., 2016).

7.4. Formulación matemática del método WCSPH implementado

El método *Weakly Compressible Smoothed Particle Hydrodynamics* (WCSPH) combina la aproximación Lagrangiana de SPH con un tratamiento explícito de la presión a través de una ecuación de estado. Esta sección presenta la formulación matemática del método, incluyendo la interpolación SPH, las ecuaciones de conservación, el modelo de presión y el tratamiento de la viscosidad y fuerzas externas.

7.4.1. Aproximación SPH y ecuaciones de conservación

Como se introdujo en la sección 7.2.1, en SPH cualquier campo escalar $f(\vec{r})$ puede aproximarse mediante la interpolación basada en un kernel de suavizado W . Haciendo uso de la ecuación 7.5.

Mediante esta formulación se tiene que la densidad de una partícula i se puede escribir como (Liu and Liu, 2003):

$$\rho_i = \sum_{j=1}^N m_j W_{ij}, \quad (7.10)$$

donde m_j es la masa de la partícula j , mientras que:

$$W_{ij} = W(\vec{r}_i - \vec{r}_j, h). \quad (7.11)$$

Con intención de mejorar la precisión del cálculo de densidades cerca de los bordes, que resulta problemático en SPH dando valores ficticios, (Liu and Liu, 2003, p. 117) menciona la existencia de una función normalizada para el cálculo de la densidad:

$$\rho_i = \frac{\sum_{j=1}^N m_j W_{ij}}{\sum_{j=1}^N \left(\frac{m_j}{\rho_j}\right) W_{ij}}. \quad (7.12)$$

Para el caso de fluidos viscosos en WCSPH, esta interpolación permite derivar las ecuaciones de conservación de masa (6.2) y momentum (6.4), para cada partícula i (Fraga Filho, 2019):

$$\frac{D\rho_i}{Dt} = \sum_{j=1}^N m_j (\vec{v}_i - \vec{v}_j) \cdot \nabla_i W_{ij}. \quad (7.13)$$

La ecuación de conservación de momentum posee una estructura diferente a la forma continua debido a que la viscosidad se aplica como una corrección artificial de la aceleración de manera que es escrita como (Korzani et al., 2014; Monaghan, 1994):

$$\frac{D\vec{v}_i}{Dt} = - \sum_{j=1}^N m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \Pi_{ij} \right) \nabla_i W_{ij} + \vec{F}_i, \quad (7.14)$$

donde Π_{ij} corresponde a la función de viscosidad artificial que se expondrá más adelante, mientras que P_i es la presión y \vec{F}_i las fuerzas externas.

7.4.2. Modelo de presión y control de compresibilidad

En WCSPH, la presión se obtiene mediante una ecuación de estado, dependiendo del sistema que queremos simular esta ecuación debe adaptarse a los regímenes de presión en los que se manejan.

Ecuación de estado lineal

Uno de los métodos más simples propuestos en Liu and Liu (2003), es usando la ecuación de presión lineal, la cual solo depende de la velocidad del sonido, la cual corresponde a un valor artificial en el modelo WCSPH, dicha ecuación está dada por:

$$P_i = c^2 \rho_i. \quad (7.15)$$

Ecuación de estado de Tait

Sin embargo, para un sistema donde se presenta gravedad y el modelo debe ser más complejo, se hace uso de otra forma de la ecuación que permite adaptar el modelo de superficie libre como campo constante de presión, esto se hace a partir de la ecuación de Tait, dada por:

$$P_i = B \left[\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right], \quad (7.16)$$

donde ρ_0 es la densidad de referencia, γ un exponente adiabático (generalmente $\gamma = 7$ para agua) y B una constante que da cuenta del límite para el máximo cambio en la densidad (Monaghan, 1994, p. 137):

$$B = \frac{\rho_0 c^2}{\gamma}. \quad (7.17)$$

Número de Mach

Para determinar el valor necesario de la velocidad de sonido artificial, importante como parámetro en los dos modelos antes mencionados, es necesario definir el número de Mach como la razón entre una velocidad característica del flujo y la velocidad del sonido, de acuerdo con:

$$Ma = \frac{V_b}{c}, \quad (7.18)$$

donde V_b representa la velocidad máxima característica del fluido y c corresponde a la velocidad del sonido artificial. En el marco del enfoque WCSPH, se impone un número de Mach suficientemente pequeño con el fin de limitar las variaciones de densidad y aproximar el comportamiento incompresible del fluido. Normalmente, se adopta la condición $Ma \leq 0.1$, lo que conduce a la relación:

$$0.1 = \frac{V_b}{c}, \quad (7.19)$$

de la cual se obtiene directamente:

$$c = \frac{V_b}{0.1} = 10 V_b. \quad (7.20)$$

En problemas de flujo con superficie libre, la velocidad característica máxima del fluido puede estimarse a partir de la velocidad de caída libre asociada a una altura representativa H . Bajo esta aproximación, se tiene:

$$V_b = \sqrt{2gH}, \quad (7.21)$$

donde g es la aceleración de la gravedad. Sustituyendo esta expresión en la definición de la velocidad del sonido artificial, se obtiene:

$$c = 10\sqrt{2gH}, \quad (7.22)$$

o, de forma equivalente:

$$c = \sqrt{200gH}. \quad (7.23)$$

Esta elección de la velocidad del sonido garantiza que las variaciones de densidad permanezcan suficientemente pequeñas durante la simulación y corresponde a la misma ecuación mencionada en Monaghan (1992).

7.4.3. Modelado de la viscosidad y fuerzas externas

El término viscoso que aparece en la ecuación de momentum discretizada 7.14 tal como se discute en Monaghan and Gingold (1983) requiere de una construcción artificial para actuar de forma más directa sobre el movimiento de las partículas. Una forma de viscosidad artificial aplicada en superficies libre es mostrada en Becker and Teschner (2007). Este artículo propone un modelo lineal respecto a la velocidad que corresponde a:

$$\Pi_{ij} = \begin{cases} -\nu \frac{\vec{v}_{ij} \cdot \vec{r}_{ij}}{|\vec{r}_{ij}|^2 + \epsilon h^2}, & \text{si } \vec{v}_{ij} \cdot \vec{r}_{ij} < 0, \\ 0, & \text{en otro caso,} \end{cases} \quad (7.24)$$

donde:

$$\nu = \frac{2\alpha h c_s}{\rho_i + \rho_j}, \quad (7.25)$$

donde α es un parámetro que caracteriza el término de viscosidad de corte y volumétrica. En el caso de las superficies libres Monaghan (1994) menciona que los cambios despreciables en la densidad hacen que la viscosidad sea casi enteramente de corte, asociándosele un $\alpha = 0.01$. Sin embargo, en Monaghan (1992) se menciona que este valor puede tomar valores hasta llegar cerca de $\alpha = 1$. Mientras, Becker and Teschner (2007) hace experimentos con valores que van desde 0.08 hasta 0.5.

Por otra lado, el valor de ϵ se introduce para evitar singularidades de las partículas cercanas tomando un valor aproximado de $\epsilon \approx 0.01$.

Una formulación más compleja de la viscosidad artificial es mencionada en Monaghan (1992) y corresponde a una dependencia cuadrática con el valor de la velocidad:

$$\Pi_{ij} = \begin{cases} \frac{-\alpha c_{ij}\phi_{ij} + \beta \phi_{ij}^2}{\rho_{ij}}, & \text{si } \vec{v}_{ij} \cdot \vec{r}_{ij} < 0, \\ 0, & \text{en otro caso,} \end{cases} \quad (7.26)$$

donde:

$$\phi_{ij} = \frac{h_{ij} (\vec{v}_{ij} \cdot \vec{r}_{ij})}{|\vec{r}_{ij}|^2 + \varepsilon h_{ij}^2}. \quad (7.27)$$

Las fuerzas externas incluyen gravedad, campos de presión o fuerzas específicas de la simulación. En Becker and Teschner (2007) la fuerza de gravedad se añade como un valor adicional en la ecuación de conservación de momentum.

$$\frac{D\vec{v}_i}{Dt} = - \sum_{j=1}^N m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \Pi_{ij} \right) \nabla_i W_{ij} + \vec{g}_i, \quad (7.28)$$

donde:

$$\vec{g}_i = \vec{g} + \vec{f}_i^{\text{externa}}. \quad (7.29)$$

Esta formulación completa el modelo matemático básico de WCSPH, en el cual las fuerzas se escriben dentro de la ecuación de momentum como fuerza por unidad de masa de la partícula de material, correspondiendo por lo tanto a una discretización de la aceleración generada por dicha fuerza (Monaghan, 1992).

7.4.4. Corrección en la velocidad usando el modelo de XSPH

Con el fin de mejorar la estabilidad cinemática del método SPH y reducir el ruido numérico asociado al movimiento de las partículas, (Monaghan, 1992) propuso el esquema de corrección de velocidad XSPH. En este enfoque, la posición de cada partícula se actualiza utilizando una velocidad corregida que incorpora un promedio ponderado de las velocidades de las partículas vecinas. Esta corrección no modifica las ecuaciones de conservación, sino que actúa únicamente sobre la velocidad y su integración temporal. La formulación de esta corrección es:

$$\frac{D\vec{r}_i}{Dt} = \vec{v}_i + \varepsilon \sum_{j=1}^N m_j \left(\frac{\vec{v}_{ji}}{\rho_{ij}} \right), \quad (7.30)$$

donde ε tiene valores entre 0 y 1.

7.4.5. Integración temporal en SPH

En el método SPH, las ecuaciones discretizadas de movimiento conducen a un sistema de ecuaciones diferenciales ordinarias del tipo:

$$\frac{d\vec{r}_i}{dt} = \vec{v}_i, \quad \frac{d\vec{v}_i}{dt} = \vec{a}_i, \quad (7.31)$$

donde \vec{a}_i incluye las contribuciones debidas a presión, viscosidad artificial y fuerzas externas.

La integración temporal de este sistema se realiza mediante esquemas explícitos de segundo orden, especialmente los integradores simplécticos como Leapfrog y Verlet.

Los integradores simplécticos preservan la estructura geométrica del sistema dinámico, evitando la deriva sistemática de la energía total y produciendo únicamente oscilaciones acotadas alrededor del valor exacto (Hairer et al., 2006). Esta propiedad los hace especialmente adecuados para la simulación de sistemas de partículas y métodos SPH, donde la estabilidad a largo plazo es un requisito fundamental Monaghan (1992).

Método Leapfrog

El integrador Leapfrog es un método simpléctico de segundo orden, ampliamente utilizado en dinámica de partículas y en SPH debido a su buena conservación de la energía a largo plazo (Hairer et al., 2006).

La forma clásica más implementada en SPH es (Cordero, 2013, p. 53):

$$\vec{v}^{n+\frac{1}{2}} = \vec{v}^{n-\frac{1}{2}} + \Delta t \vec{a}^n, \quad (7.32)$$

$$\vec{r}^{n+1} = \vec{r}^n + \Delta t \vec{v}^{n+\frac{1}{2}}, \quad (7.33)$$

donde $\vec{v}^{n+\frac{1}{2}}$ es la velocidad en el paso $n + 1$, la cual está dada por $\vec{v}^{n-\frac{1}{2}}$ que es velocidad en el punto $n - \frac{1}{2}$ y la aceleración en el punto n (\vec{a}^n), multiplicada por el paso de tiempo Δt .

Mientras que la posición en el punto $n + 1$ ($\vec{r}^{n+\frac{1}{2}}$), estará dada por la posición en el punto n más la velocidad en el punto $n + 1$ multiplicada por el paso del tiempo.

Método Verlet

El método Verlet clásico se obtiene a partir de la expansión en series de Taylor de la posición alrededor del instante t^n . La ecuación de actualización de la posición se escribe como:

$$\vec{r}^{n+1} = 2\vec{r}^n - \vec{r}^{n-1} + \vec{a}^n \Delta t^2, \quad (7.34)$$

donde \vec{r}^n representa la posición en el instante t^n , \vec{a}^n es la aceleración evaluada en dicho instante y Δt es

el paso temporal.

Una variante ampliamente utilizada en dinámica de partículas es el método Velocity Verlet, el cual permite actualizar simultáneamente la posición y la velocidad, evitando el almacenamiento de posiciones pasadas y proporcionando una evaluación más precisa del campo de velocidades (Cordero, 2013, pp. 112-113).

Las ecuaciones de integración del método Velocity Verlet se expresan como:

$$\vec{r}^{n+1} = \vec{r}^n + \vec{v}^n \Delta t + \frac{1}{2} \vec{a}^n \Delta t^2, \quad (7.35)$$

$$\vec{v}^{n+1} = \vec{v}^n + \frac{1}{2} (\vec{a}^n + \vec{a}^{n+1}) \Delta t, \quad (7.36)$$

donde la aceleración \vec{a}^{n+1} se evalúa utilizando las posiciones actualizadas \vec{r}^{n+1} .

Este integrador mantiene el orden segundo, es reversible en el tiempo y presenta buenas propiedades de conservación de la energía (Cordero, 2013, pp. 112-114).

7.5. Tratamiento de condiciones de frontera

Las condiciones de frontera constituyen uno de los aspectos más críticos en el método SPH, ya que la ausencia de una malla fija dificulta la imposición de condiciones geométricas y dinámicas en las paredes sólida. En simulaciones de flujos con superficie libre, como el vaciado de tanques, el modelo de frontera debe garantizar simultáneamente la impermeabilidad, la estabilidad numérica y una correcta transmisión de fuerzas entre el fluido y las paredes (Monaghan, 1994; Violeau, 2012).

En el marco del método WCSPH, las condiciones de frontera suelen abordarse mediante partículas especiales que representan el sólido y que interactúan con las partículas de fluido a través de fuerzas adicionales o mediante la imposición de condiciones de presión y densidad consistentes con la ecuación de estado del fluido (Adami et al., 2012; Marrone et al., 2011).

7.5.1. Fronteras sólidas y tratamiento numérico

Una de las estrategias más utilizadas para modelar fronteras sólidas en SPH es el enfoque basado en fuerzas de repulsión, introducido originalmente por Monaghan (1994). En este modelo, las partículas de frontera ejercen una fuerza repulsiva de corto alcance sobre las partículas de fluido cuando estas se aproximan a la pared.

La fuerza repulsiva se inspira en el potencial de Lennard–Jones truncado y se define de forma que actúe únicamente dentro de una distancia característica asociada a la longitud de suavizado (Monaghan, 1994; Liu and Liu, 2003). La ecuación asociada a las fronteras se escribe como:

$$f(r) = D \left(\left(\frac{r_0}{|\vec{r}_{ij}|} \right)^{n_1} - \left(\frac{r_0}{|\vec{r}_{ij}|} \right)^{n_2} \right) \frac{\vec{r}}{|\vec{r}_{ij}|^2}, \quad (7.37)$$

siempre que $|\vec{r}_{ij}| < r_0$. Donde los valores de $n_1 = 12$ y $n_2 = 4$, mientras que D es un parámetro dependiente de la escala del modelo (Liu and Liu, 2003).

Las fronteras sólidas se representan mediante partículas de tipo I, es decir, partículas fijas que no se integran en el tiempo ni participan en la ecuación de movimiento. Estas partículas únicamente sirven como soporte geométrico y como fuente de interacción con el fluido.

7.5.2. Fronteras de partículas fantasma

Una alternativa al uso de partículas de frontera fijas es el empleo de *partículas fantasma* (*ghost particles*), también denominadas partículas espejo, correspondientes a las fronteras de tipo II. En este enfoque, la pared sólida se define geoméricamente mediante partículas de borde (tipo I), al mismo tiempo que se crean nuevas partículas asociadas a partículas de fluido situadas a una distancia menor a un múltiplo del radio de suavizado (κh) respecto de la frontera, se generan partículas espejo ubicadas simétricamente al otro lado de la pared (Liu and Liu, 2003, pp. 138-139). Este procedimiento permite completar la vecindad de las partículas cercanas a la frontera y reducir la pérdida de soporte del kernel en regiones próximas a las paredes (Violeau, 2012, pp. 428-429).

En la formulación propuesta por Liu and Liu (2003), las partículas fantasma se construyen asignándoles la misma densidad y presión que la partícula de fluido correspondiente, con una velocidad con el signo opuesto, lo que facilita la imposición de condiciones de no deslizamiento en la pared. Estas partículas no representan material físico y no se integran dinámicamente, sino que actúan como un artificio numérico destinado a restaurar la simetría de las ecuaciones discretas del método SPH cerca de las fronteras (Liu and Liu, 2003, pp. 138-139).

Desde el punto de vista dinámico, las partículas fantasma participan directamente en el cálculo de densidad y presión del fluido próximo a la pared, influyendo en la ecuación de continuidad y, a través de la ecuación de estado, en el campo de presión. Sin embargo, este mecanismo por sí solo no resulta suficiente para impedir completamente la penetración del fluido fuera del dominio, por lo que suele complementarse con partículas de frontera de tipo I o fuerzas de repulsión adicionales. La combinación de ambos enfoques mejora la estabilidad numérica y garantiza la impermeabilidad de la frontera en simulaciones WCSPH (Liu and Liu, 2003; Monaghan, 1994, pp. 138-139).

7.5.3. Modelos de interacción fluido-frontera adicionales

Dentro de la formulación SPH para flujos con superficie libre, la interacción entre el fluido y las fronteras sólidas puede abordarse desde distintos niveles de complejidad. En ausencia de un modelo explícito de viscosidad fluido-pared, como ocurre en el presente trabajo, la interacción fluido-sólido se restringe a los efectos normales inducidos por el campo de presión y, en algunos casos, por fuerzas de repulsión de corto alcance (Monaghan, 1994; Violeau, 2012).

En simulaciones de flujos con superficie libre, como el vaciado de tanques, la correcta evaluación de la presión en las cercanías de la frontera resulta crucial para evitar inestabilidades numéricas y gradientes de presión no físicos. En formulaciones WCSPH, la presión del fluido se calcula típicamente mediante una ecuación de estado, mientras que las partículas de frontera requieren un tratamiento especial debido a la ausencia de partículas en el exterior del dominio, lo que provoca una pérdida de consistencia en las sumas discretas del método (Violeau, 2012).

Dos enfoques para el tratamiento de la presión en las partículas de frontera se describen a continuación:

Presión prescrita en partículas de frontera:

Un primer enfoque, conceptualmente simple, consiste en imponer directamente un valor de presión prescrito en las partículas de frontera, usualmente igual a cero, debido a que al aplicar la ecuación de Tait (ecuación 7.16) a unas partículas con densidad igual a la densidad de referencia hacen nula la presión resultante. Este tratamiento equivale a suponer que las partículas de frontera se encuentran en contacto con un medio a presión constante y ha sido utilizado en simulaciones preliminares de flujos con superficie libre (Violeau, 2012).

Extrapolación de presión desde el fluido:

Otro enfoque, se basa en la extrapolación de la presión desde las partículas de fluido de la vecindad cercana hacia las partículas de frontera, siguiendo la metodología propuesta por Adami et al. (2012). En este modelo, las partículas de frontera participan activamente en las sumas SPH, pero su presión no se calcula mediante la ecuación de estado. En su lugar, se determina a partir de un promedio ponderado de las presiones de las partículas de fluido vecinas, incorporando un término hidrostático que garantiza la consistencia con el campo gravitacional.

La presión en una partícula de frontera i se define como:

$$P_i = \frac{\sum_j (p_j + \rho_j \vec{g} \cdot (\vec{r}_i - \vec{r}_j)) W_{ij}}{\sum_j W_{ij}}, \quad (7.38)$$

donde p_j y ρ_j corresponden a la presión y densidad de las partículas de fluido vecinas, \vec{g} es la aceleración gravitacional y W_{ij} es el núcleo de suavizado evaluado entre la partícula de frontera y la partícula de fluido j .

7.6. Resolución espacial, paso del tiempo y condición CFL

La resolución espacial en el método SPH está determinada por la relación entre el espaciado inicial entre partículas Δx y la longitud de suavizado h , la cual define el radio de influencia del kernel de suavizado en el cálculo de las propiedades del fluido. En base a la ecuación 7.4 tenemos que $|\vec{r}' - \vec{r}| \leq \kappa h$ (Liu and Liu, 2003, p. 37). Además, siguiendo la consideración de (Liu and Liu, 2003, p. 145) para una longitud de suavizado

constante tenemos que $\kappa = 2$. Sin embargo, para que se cumpla la desigualdad anterior para cualquier $\kappa \geq 1$ se puede escoger $h = 1.1\Delta x_i$.

Esta relación asegura que cada partícula posea un número suficiente de vecinos dentro de su radio de influencia, lo cual es esencial para mantener la consistencia y estabilidad del método. Una elección inadecuada de h puede conducir a errores de interpolación, ruido numérico o pérdida de precisión en el cálculo de gradientes (Violeau and Leroy, 2013).

Mientras el paso de tiempo Δt en SPH se encuentra restringido por criterios de estabilidad numérica, siendo el más importante la condición de Courant–Friedrichs–Lewy (CFL). Tal como se menciona en (Liu and Liu, 2003, p. 142) esta condición hace que el paso del tiempo sea proporcional a la más pequeña resolución espacial, cuya representación en términos de la longitud de suavizado es:

$$\Delta t = \min\left(\frac{h_i}{c}\right), \quad (7.39)$$

donde c es la velocidad del sonido artificial y h_i deberá ser el valor mínimo de longitud de suavizado que se haya definido para el sistema.

Otra forma de estimar este valor del paso del tiempo fue presentado en (Morris et al., 1997, p. 219) que se basa en la condición habitual usada para diferencias finitas y que corresponde a:

$$\Delta t \leq 0.125 \frac{h^2}{\nu}, \quad (7.40)$$

donde ν es la viscosidad cinemática teórica del fluido (Liu and Liu, 2003, p. 142) y está dada por $\nu = \mu/\rho$. Valor que para el agua a temperatura ambiente ronda los $\nu \approx 1 \times 10^{-6} m^2/s$ (SE, 2026).

Una forma de integrar estos dos modelos es presentada en Yoo et al. (2024), donde para WCSPH se agrupa tanto la condición CFL como la condición de viscosidad introduciendo un parámetro llamado CFL_{WCSPH} el cual se presenta con un valor de 0.3. La ecuación entonces se presenta como:

$$\Delta t \leq \left(CFL_{WCSPH} \frac{h}{c_0}, 0.125 \frac{h^2}{\nu}\right). \quad (7.41)$$

7.7. Métodos de búsqueda de partículas vecinas.

La búsqueda de partículas vecinas corresponde a la funcionalidad del código que consume mayor tiempo de ejecución dentro de la simulación, puesto que requiere determinar que partículas se encuentran dentro del rango de distancia del dominio de soporte del kernel de tamaño κh . A diferencia de los métodos basados en malla que definen la ubicación de las partículas en el momento en que se construyen las mallas, el método SPH puede hacer uso de diferentes métodos de búsqueda unos más óptimos que otros Liu and Liu (2003).

7.7.1. Método de fuerza bruta

Se basa en calcular la distancia a todas las partículas en el dominio de la simulación y tomar como vecinas todas las partículas cuya distancia sea menor o igual al dominio del kernel κh , haciendo que si la longitud de suavizado es simétrica la partícula de referencia i también haga parte de los vecinos de la partícula considerada j . El tiempo de cómputo de este método de búsqueda crece con una complejidad computacional de $O(N^2)$ (Liu and Liu, 2003).

7.7.2. Método de árbol cuaternario

Este método se basa en la construcción de una estructura jerárquica ordenada a partir de las posiciones de las partículas, que posteriormente se utiliza para realizar la búsqueda de vecinos (Liu and Liu, 2003).

En el caso bidimensional, el dominio computacional se divide recursivamente en cuatro subdominios, dando lugar a un árbol cuaternario (quadtree). La subdivisión continúa hasta que las hojas del árbol representan partículas individuales. Una vez construido el árbol, se procede a la búsqueda de vecinos. Para una partícula i , se define una región de búsqueda cuadrada centrada en dicha partícula, con un lado de longitud $2\kappa h$. En cada nivel del árbol, se comprueba si el cuadrado de búsqueda intersecta el área asociada al nodo actual del quadtree. Si no existe intersección, se descarta ese nodo y se detiene el descenso por esa rama. Si existe intersección, se continúa descendiendo en el árbol hasta alcanzar un nodo hoja. En ese punto, se verifica si la partícula contenida en el nodo se encuentra dentro del dominio de soporte de la partícula i ; en caso afirmativo, se registra como vecina (Liu and Liu, 2003).

Este algoritmo de búsqueda presenta una complejidad computacional del orden de $O(N \log N)$, lo que lo hace altamente eficiente para simulaciones con un gran número de partículas (Hernquist and Katz, 1989).

Diseño y arquitectura del sistema

En esta sección se describe el diseño y la posterior implementación computacional del simulador desarrollado para la resolución numérica de problemas hidrodinámicos mediante el método WCSPH. La implementación se concibe como un sistema modular que traduce las decisiones teóricas y los modelos físicos descritos en la sección anterior en un conjunto de componentes.

El diseño del sistema responde a la necesidad de separar la definición del experimento numérico, la ejecución del núcleo algorítmico y el análisis de los resultados. Bajo este enfoque, la arquitectura del simulador se organiza en módulos asociados a la generación de condiciones iniciales y geometría del dominio, la configuración del modelo hidrodinámico, la ejecución del algoritmo SPH y la gestión de los datos generados durante la simulación.

A continuación, se presenta la arquitectura general del sistema y posteriormente se describen en detalle los principales módulos que componen la implementación.

8.1. Arquitectura general del sistema

8.1.1. Visión general del sistema

El simulador desarrollado en este trabajo se concibe como un sistema modular orientado a la ejecución de experimentos numéricos basados en el método WCSPH. Desde una perspectiva funcional, el sistema se organiza en módulos que reflejan el flujo natural de una simulación SPH: (i) generación de condiciones iniciales y geometría del dominio, (ii) configuración del modelo físico y numérico, (iii) ejecución del núcleo algorítmico encargado de la integración temporal de las ecuaciones hidrodinámicas y (iv) almacenamiento y visualización de los resultados.

Sin embargo, el diseño del *software* se planteó desde un enfoque más global a más general en 4 fases de desarrollo que son: (i) la estructura base del simulador, (ii) la construcción de las condiciones iniciales y la geometría, (iii) la estructura algorítmica del sistema y (iv) la validación de los resultados. Esto puede verse en los Apéndices: *historias de usuario* B y los *requisitos funcionales* C.

Una visualización del diseño más simple se puede obtener mediante diferentes diagramas que representan el *software* tanto como una entidad que se relaciona con otras externas a él, como la forma en la que dichas entidades se relacionan con componentes internos del *software*, y que por lo tanto representan funcionalidades específicas del mismo. A continuación se presentan dichos diagramas junto con otras propiedades del diseño.

8.1.2. Diagrama de Contexto

El diagrama de contexto representa el sistema como una entidad única y representa las interacciones con actores externos o conexiones con otros sistemas, incluyendo las entradas y salidas clave entre estos (López Ramírez and Quiroga Peláez, 2025).

En nuestro caso pensamos el sistema final como un simulador que puede requerir grandes capacidades computacionales y por lo tanto genera una cantidad considerable de datos, por lo que el sistema está pensado para conectarse con un servidor y una unidad de almacenamiento externa, tal como se muestra en el *diagrama 8.1*

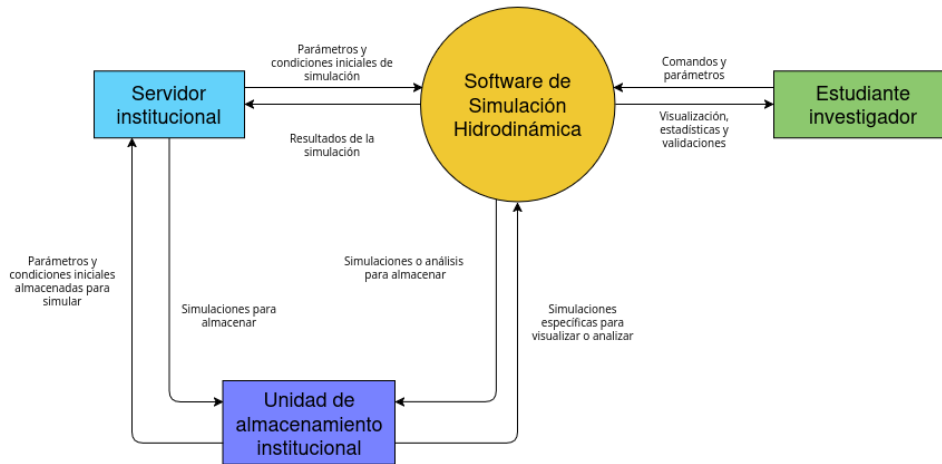


Figura 8.1: Digrama de contexto para implementación final

8.1.3. Diagrama de casos de uso.

Representa gráficamente las interacciones de los actores externos y las funcionalidades del sistema, mostrando el comportamiento desde la perspectiva del usuario o los otros sistemas externos a los que se conecte (López Ramírez and Quiroga Peláez, 2025).

De la misma manera que se presenta en el diagrama de contexto, el diagrama de caso de uso tiene tres actores externos que son el estudiante investigador, el servidor y la unidad de almacenamiento institucional, la forma en la que las funcionalidades se conectan con cada uno de ellos y entre ellas mismas se puede ver en el *diagrama 8.2*.

8.1.4. Requisitos no funcionales.

Los requisitos no funcionales corresponden a propiedades del funcionamiento del sistema como lo pueden ser características de rendimiento, escalabilidad, seguridad o compatibilidad (López Ramírez and Quiroga Peláez, 2025). En nuestro caso esas características se relacionan con la correspondencia con la física simulada, el número de partículas simuladas y los cálculos posteriores con la información resultante de la simulación, esta información se puede ver en el *Apéndice D*.

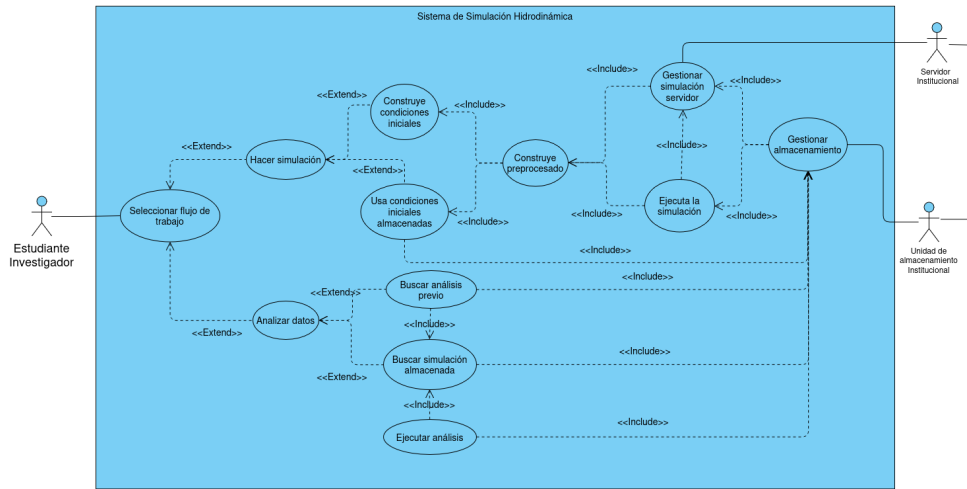


Figura 8.2: Digrama de casos de uso

8.1.5. Diagrama de descomposición funcional

Luego de haber estructurado las funcionalidades y requerimientos del *software* se puede hacer uso de un diagrama para entender como las diferentes funcionalidades se agrupan y ordenan respecto a las necesidades que desean cubrir, en nuestro caso el *Digrama 8.3*.

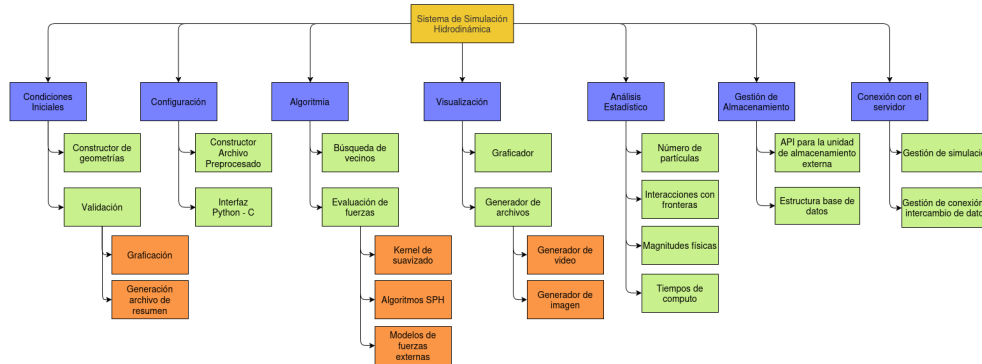


Figura 8.3: Diagrama de descomposición funcional

8.1.6. Diagrama de componentes

El diagrama de componentes es un artefacto visual modelado en UML, utilizado para representar la organización del *software*, el cual se presenta estructurado en módulo, bibliotecas o servicios y como estos interactúan mediante interfaces (López Ramírez and Quiroga Peláez, 2025).

En nuestro caso, el diagrama de componentes representa los grupos completos de funcionalidades en diferentes componentes, los cuales pueden tener asociadas una o varias interfaces, o salidas que pueden ir a

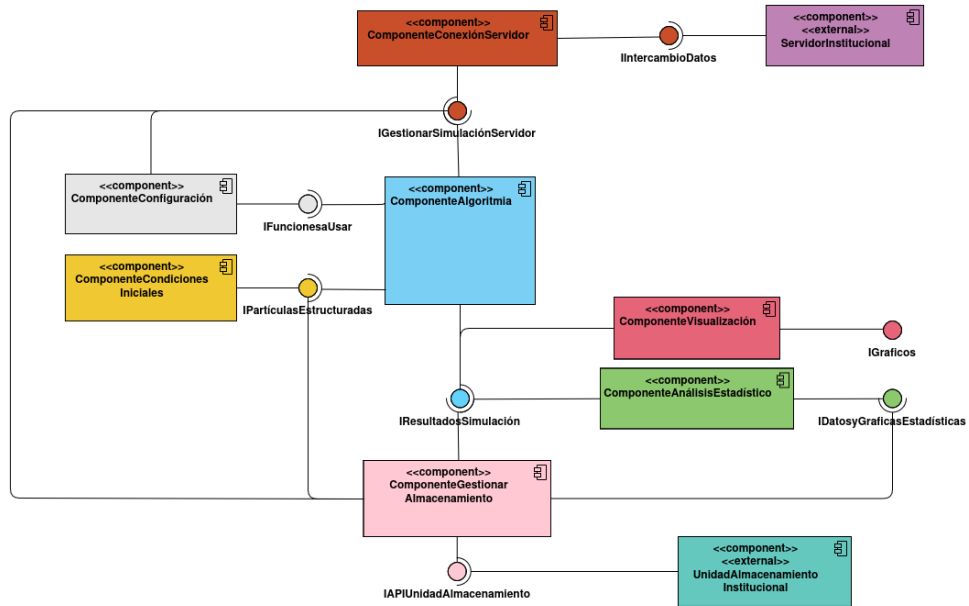


Figura 8.4: Diagrama de componentes

sistemas externos como el servidor o la unidad de almacenamiento, o simplemente generar archivos que se guarden en el local como pueden ser los gráficos. Vemos estas aplicación en el *Diagrama 8.4*.

Metodología

Esta investigación partió de una implementación realizada por el profesor Luis Quiroga ¹ del *Shear driven cavity problem* presentado en (Liu and Liu, 2003, pp. 164) durante el curso de Astrofísica Computacional del cual hice parte. El sistema simulado tenía una dimensión de $l = 10^{-3} m$ y un total de 1600 partículas de fluido asociadas a una malla de 40×40 partículas, la cual se ejecutó durante 3000 pasos obteniendo un campo de velocidades como el que se ve en 9.1. Donde vemos un recipiente de agua cortado de forma transversal, es decir, sin una componente de altura, representado por sus fronteras dadas por las partículas negras. Se graficó la velocidad del fluido para cada una de las partículas como pequeños vectores azules asociados estas, y donde todo el sistema se encuentra en presencia de un campo de velocidad externo sobre la frontera superior con dirección positiva del eje x (horizontal).

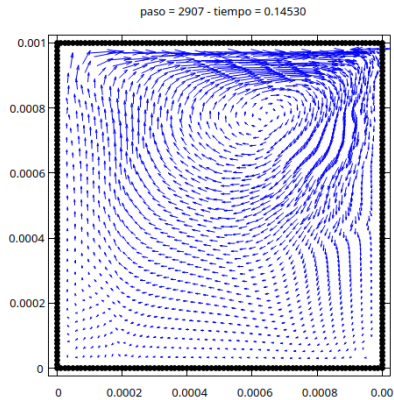


Figura 9.1: Campo de velocidad en el *Shear driven cavity problem* con la implementación de código anterior para 4000 partículas de fluido.

A partir del desarrollo de los modelos implementados en este se diseñó un código base que permitiera construir geometrías complejas y seleccionar distintas ecuaciones y parámetros para ser aplicable en inundaciones a mediana escala. Como una primera fase de implementación de este diseño se construyó un prototipo para el vaciado de tanques.

9.1. Algoritmo y flujo general de ejecución de una simulación

El flujo de ejecución de una simulación comienza con la definición del experimento mediante archivos de configuración que describen la geometría del dominio, las condiciones iniciales del fluido y los parámetros del modelo hidrodinámico. A partir de esta información, el módulo de condiciones iniciales genera la distribución inicial de partículas, que constituye el estado inicial del sistema.

Posteriormente, el núcleo algorítmico SPH utiliza estos datos como entrada para realizar la integración temporal del sistema, calculando en cada paso las magnitudes hidrodinámicas y almacenando los resultados

¹Institución Universitaria Salazar y Herrera

en archivos intermedios. Estos archivos pueden ser analizados directamente o utilizados por el módulo de visualización para examinar el comportamiento del fluido a lo largo del tiempo.

Este esquema de ejecución desacopla la definición del experimento de su resolución numérica, permitiendo repetir simulaciones con distintas configuraciones y facilitando el análisis comparativo de parámetros, la forma en la que cada paso se lleva a cabo para ejecutarse una simulación se puede observar en el *diagrama 9.2*.

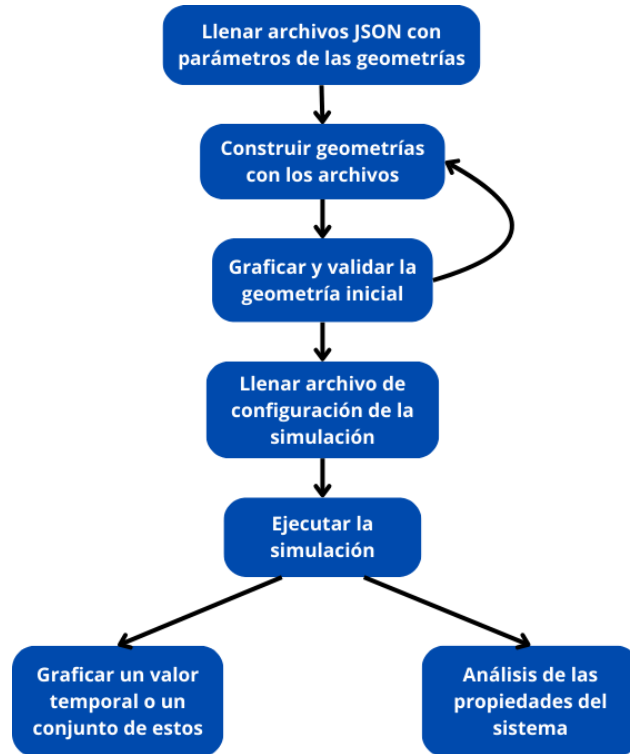


Figura 9.2: Diagrama de flujo del sistema implementado para simulaciones individuales.

9.1.1. Implementación del módulo de condiciones iniciales

El módulo de generación de condiciones iniciales se diseñó como una herramienta independiente encargada de definir el estado inicial del dominio computacional. Este módulo recibe como entrada dos archivos de configuración en formato JSON que contienen la información necesaria para determinar la ubicación espacial y el espaciado entre partículas correspondientes tanto a las fronteras sólidas como a la región de fluido, que se encuentran representadas por el mismo tipo de partículas solo que con un espaciado y mecánicas diferentes.

En el caso de las fronteras, se permitió la definición de tres entidades geométricas: un cuadrilátero principal que delimita el dominio, regiones de partículas diferenciadas que representan agujeros dentro de la geometría, permitiendo su eliminación posterior y líneas adicionales construidas sobre el dominio.

La inclusión de estas entidades geométricas responde a la adopción de una aproximación bidimensional del fenómeno de vaciado de tanques, cuya representación más general puede modelarse mediante un trapecio. Si bien el módulo fue concebido inicialmente con el objetivo de representar geometrías arbitrarias, en este

trabajo se emplea esta simplificación con el fin de capturar las características esenciales del problema físico analizado.

9.1.2. Migración e implementación del módulo algorítmico

Para la construcción del módulo de cálculos del modelo WCSPH se usaron las ecuaciones ya mencionadas en la sección 9.2.1, muchas de estas ya se encontraban implementadas en el código del sistema del Shear Cavity Problem mencionado al comienzo de la metodología por lo que para facilitar la escritura del código se usó este como base, realizando una modularización, refactorización y migración a un lenguaje más reciente, este proceso fue apoyado por OpenAI (2025).

El resultado de este proceso generó un código en C++ que producía un ejecutable el cual requiere de un archivo JSON para pasar las condiciones iniciales, los parámetros y funciones a implementar, además de las ubicaciones en las que se generan los archivos.

9.1.3. Módulo de configuración

El módulo de configuración fue llevado hasta una versión preliminar que está constituida por las plantillas que contienen los formatos permitidos de condiciones iniciales, archivos de configuración de simulación, etc. necesarios para realizar simulaciones, junto con una serie de funciones necesarias para crear nuevos archivos con el mismo formato variando los parámetros, además de lanzar simulaciones de forma individual o automatizadas en una serie de parámetros y finalmente una versión simplificada de creación de gráficas de un tiempo específico o de toda la simulación como un continuo.

9.2. Selección del modelo hidrodinámico (tipo de SPH)

El modelo del *Shear Cavity Problem* no es adecuado para entender los diferentes modelos físicos que son extrapolables al vaciado de tanques. Por esta razón, en este apartado se presenta la verificación de dichos modelos, así como su modificación y justificación, de manera que puedan ser aplicables al vaciado de tanques.

Partiendo del código WCSPH que se utilizó para simular el problema del *Shear Driven Cavity Problem* (Liu and Liu, 2003), se requirió que este se adaptara para incluir la aceleración gravitacional y la simulación de superficies libres. El modelo débilmente compresible es ampliamente utilizado para este tipo de simulaciones, tal como se menciona en la sección 7.3.2 y en Shadloo et al. (2016), donde también se señala que este método posee la ventaja de requerir un menor costo de recursos computacionales y una mayor facilidad de paralelización, aunque esta última no está planeada para este trabajo.

Todas estas consideraciones hacen que este trabajo se centre únicamente en la aplicación del método débilmente compresible de SPH, así como en su extensibilidad y en la validación de diferentes ecuaciones y modelos aplicables al mismo (correcciones de forma y estilo por OpenAI (2025)).

9.2.1. Modelo hidrodinámico adoptado

Ya que se expuso el tipo de SPH implementado, es necesario mencionar también cada una de las funciones que se implementarán inicialmente, ya que existen variaciones incluso dentro del mismo marco de WCSPH, estas diferencias proveen al método de diferentes características de funcionamiento y estabilidad que varían dependiendo del sistema física y las interacciones a simular.

Kernel de suavizado

Se usó el Kernel Spline Cúbico, correspondiente la *ecuación 7.9*, debido principalmente al soporte compacto, es decir, a su fuerte definición del intervalo de acción y que su forma funcional es ampliamente implementada y expuesta en muchos artículos del método SPH.

Densidad SPH

Se implementó también la forma funcional de la densidad normalizada que aparece en la *ecuación 7.12*, la cual presenta una mejor precisión de cálculo cerca de los bordes, tal como se menciona en Liu and Liu (2003), haciéndola muy útil para las superficies libres y sus cambios durante la simulación.

Ecuación de estado

En este caso diferentes ecuaciones de estado se implementaron, haciendo variaciones también en la forma en la que los parámetros se determinaban, partiendo del que se nos presentaba en el Shear Cavity Problem del que nos basábamos, el cual hacía uso de una ecuación de estado con densidad lineal, tal como se muestra en la *ecuación 7.15*.

También se implementó la ecuación de Tait dada en la *ecuación 7.16*, donde los parámetros B y c se calculan de manera que representen el campo de presión y gravedad natural del sistema como se describe en la sección 7.4.2. Algunas variaciones en los parámetros de la *ecuación 7.16* y la velocidad del sonido, y su estabilidad serán discutidos en capítulos posteriores.

Viscosidad

Siguiendo el modelo presentado del problema del Shear Cavity Problem, usamos una viscosidad artificial cuadrática que requiere de dos parámetros α y β , su forma funcional se representa en la *Ecuación 7.26*, esta representa la forma más robusta del modelo de viscosidad artificial, razón por la cual partimos desde esta.

Corrección del campo de velocidades

Incluimos también la corrección de velocidad del modelo XSPH, debido a que los vaciados de tanques pueden presentar cambios bruscos en el campo de velocidades por choques de las partículas.

Función de integración de velocidad y aceleración

Para integrar el paso temporal y el cálculo de los valores de posición, velocidad y aceleración, se usó el integrador simpléctico Leapfrog en su formulación básica drift-kick-drift, el cual al ser simpléctico conserva la energía del sistema a largo plazo. Las ecuaciones asociadas de posición y velocidad corresponden a 7.33 y 7.32.

Modelo de fronteras

Para este estudio se usó solamente el modelo de frontera sólida llamadas de tipo I, presentado en la sección 7.5.1 y la ecuación de repulsión tipo Lennard-Jones presentada en la *ecuación 7.37*, no se usó el modelo de fronteras fantasma o tipo II, debido a que como muestra Korzani et al. (2014) basta con las fronteras tipo I para simular un vaciado de tanques.

En cuanto a los valores macroscópicos asociados a la frontera se usó un valor de densidad constante de $\rho_0 = 1000 \text{ kg/m}^3$ con una separación entre partículas de la mitad del fluido. Este valor hace que por la definición de la ecuación de Tait 7.16, la presión sea 0, pues la densidad de referencia es la misma que la densidad de las partículas.

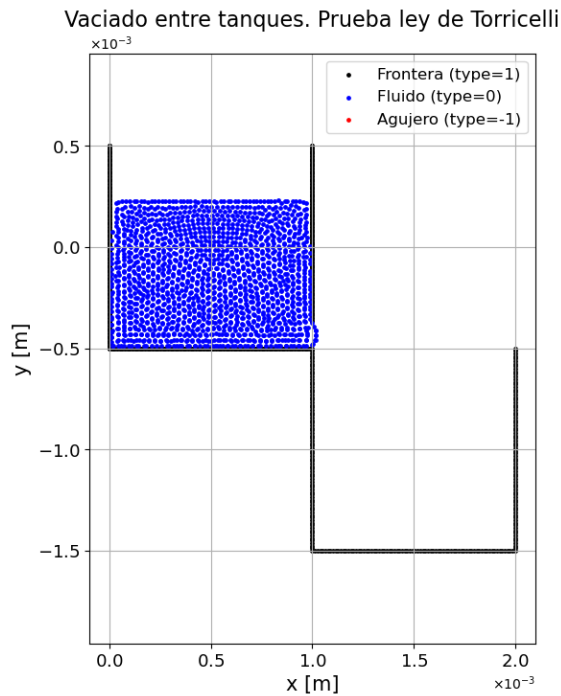
Una prueba posterior con el modelo de extrapolación de presión que se presenta en la *ecuación 7.38* fue implementado también para comparaciones entre los modelos.

9.2.2. Construcción de geometrías de prueba

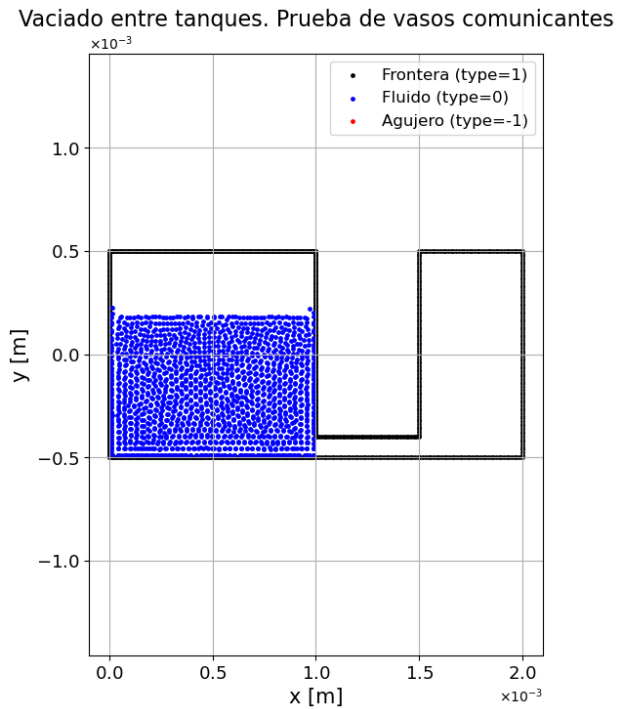
Para determinar si el funcionamiento del código y del modelo es válido físicamente, es necesario determinar unos experimentos numéricos que nos permitan determinar si el comportamiento del fluido se aproxima lo suficiente a la realidad. Para estudiar el vaciado de tanques se plantean dos geometrías asociadas a leyes físicas probadas de comportamiento de fluido, dichas geometrías deben primero visualizarse para la prueba de funcionamiento del constructor de geometrías y del visualizador. Las dos geometrías base escogidas para los experimentos son:

1. **Vaciado de un tanque a otro cercano por caída libre:** Mediante esta geometría representada en la *Figura 9.3a* se puede observar el comportamiento del fluido con la ley de Torricelli, que se explica en la *sección 6.1.6*.
2. **Vaciado de tanques por una tubería:** La geometría representada en la *Figura 9.3b* representa un caso del principio de vasos comunicantes en el cual el fondo de ambos vasos se encuentra a la misma altura, esto se hace por simplicidad, ya que como se presenta en la *sección 6.1.5*, la altura del fondo de ambos vasos no es relevante en este problema.

Analizando ambas geometrías representadas en la *Figura 9.3* vemos que ambas presentan la tapa superior del recipiente abierta, como una representación visual del efecto de la presión atmosférica, sin embargo, esto no es necesario siempre que el fluido no interactúe con la tapa debido a que la presión atmosférica se



(a) Geometría de prueba basada en la ley de Torricelli.



(b) Geometría de prueba basada en el principio de los vasos comunicantes.

Figura 9.3: Geometría de los experimentos numéricos con leyes físicas

encuentra integrada dentro de las partículas de fluido en la ecuación de estado, tal como se presenta en la sección 7.4.2. Las partículas de color azul corresponden al fluido ya estabilizado dentro del recipiente, lo cual requiere de un proceso de simulación antes de permitir el libre flujo entre recipientes y que será detallado en secciones posteriores.

9.3. Pruebas experimentales y parámetros implementados

Las pruebas iniciales del *software* fueron basadas en la ley de Torricelli (*sección 9.3a*), ya que esta permitía observar el comportamiento físico del fluido bajo cambios rápidos en velocidad y muestras visuales de la caída libre). Para estas pruebas iniciales se notó que existía una gran sensibilidad del modelo débilmente compresible a la ecuación de estado y a la velocidad del sonido artificial, tal como se menciona en Monaghan (2005).

La geometría asociada a estas pruebas puede verse en la *Figura 9.4*, teniendo esta como una representación, debido a que el tamaño de la caja puede cambiar incluso órdenes de magnitud, de la misma manera que la cantidad de partículas, dependiendo de los parámetros y consideraciones que serán expresadas en las siguientes secciones.

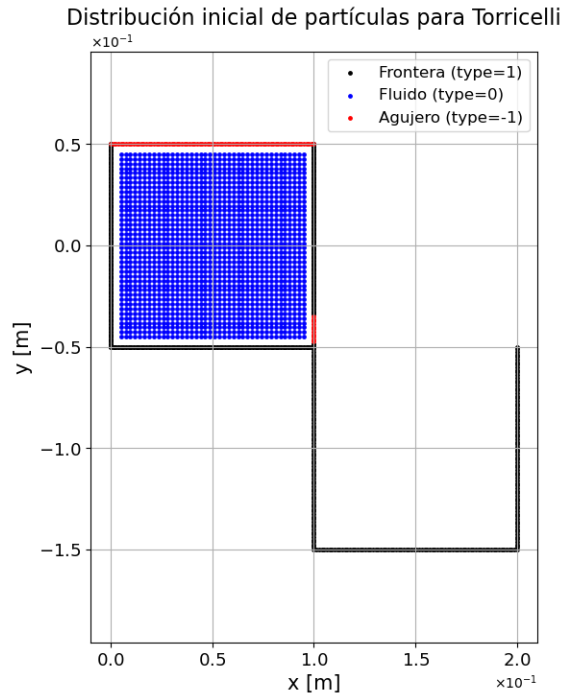


Figura 9.4: Distribución inicial de las partículas bajo la geometría aplicada para probar la ley de Torricelli

9.3.1. Pruebas iniciales

Inicialmente se realizaron pruebas de estabilidad utilizando la ecuación de estado lineal 7.15. Los parámetros empleados fueron $c = 0.01$ m/s, $g = -9.8$ m/s² y $dt = 1 \times 10^{-3}$. Se utilizó la formulación estándar de densidad SPH dada por la ecuación 7.10.

Las densidades de frontera y fluido, así como el espaciado entre partículas, se adoptaron de Korzani et al. (2014), con $\Delta x_i = 8.37 \times 10^{-4}$ para las fronteras y $\Delta x_i = 2 \times 10^{-3}$ para el fluido.

Para órdenes de magnitud del dominio de simulación de 10^{-1} m se usó un valor del parámetro $D = 1$ en la ecuación de Lennard-Jones 7.37.

9.3.2. Pruebas con la ecuación de estado tipo Tait

Parámetros de la literatura

Una búsqueda en la literatura especialmente en Monaghan (1994) mencionaba que la ecuación de estado de Tait permitía una simulación de superficies libres mucho más estables, por lo que se modifica la ecuación de estado a la *ecuación 7.16*, la geometría implementada ahora se construyó nuevamente en base a la estructura de partículas 40×40 que estaba implementada en el modelo del Shear Driven Cavity Problem, al igual que una separación de frontera del doble que la del fluido, con un orden de magnitud, ya no de 10^{-3} m sino de 10^{-1} m, lo que corresponde a simular un vaso común. Este orden de magnitud y las medidas asociadas que

este implica hacen que los valores de presión máxima y velocidad del sonido, calculadas mediante la *sección 7.4.2* sean $B = 28.000 \text{ Pa}$, $c = 14 \text{ m/s}$ y $dt \leq 5.9 \times 10^{-5} \text{ s}$, por lo que se usó un $dt = 5 \times 10^{-5} \text{ s}$. Este modelo si implementó la viscosidad artificial cuadrática (*ecuación 7.26*) y se usó una aceleración gravitacional de $g = 9.8 \text{ m/s}^2$.

Añadiendo presión a fronteras

Como una verificación adicional de estos parámetros se modificó el valor de la presión en las fronteras haciendo uso del método de extrapolación por fluido vecino mencionado en la *sección 7.5.3*, para el cual se hace uso de la *ecuación 7.38*.

Parámetros empíricos

Otros valores de los parámetros B y c fueron usados dentro de pruebas de la simulación, basado en pruebas empíricas realizadas por el estudiante Andrés López durante el curso Astrofísica Computacional, el cual encontró estabilidad del modelo de vaciado de tanques usando valores de $B = 1 \text{ Pa}$ y $c = 0.01$, con un $dt = 5 \times 10^{-5}$, tal como se puede obtener de López (2025). Sin embargo, estos parámetros no fueron orientados a los órdenes de magnitud en los que estábamos trabajando, correspondientes a 10^{-1} m sino que estaban relacionadas al orden de magnitud usado en el Shear Driven Cavity Problem, correspondiente a 10^{-3} m . Este orden de magnitud del dominio requirió un cambio del parámetro $D = 0.01$. Se usó nuevamente el modelo de presión prescrita como $p = 0 \text{ Pa}$ en las fronteras.

Un estudio sobre la estabilidad de los valores cercanos a esta región se realizó usando los valores mencionados como puntos centrales del estudio, y se realizó un estudio de la densidad en esta región para encontrar magnitudes y comportamientos realistas en el comportamiento del fluido.

9.4. Proceso de relajación inicial

En los métodos de *Smoothed Particle Hydrodynamics* (SPH), la configuración inicial de las partículas no suele satisfacer exactamente las condiciones de equilibrio consistentes con las ecuaciones discretizadas requiriendo entonces de un tiempo de relajación, como señalan Zhao et al. (2025). Las distribuciones iniciales de partículas pueden inducir fluctuaciones numéricas transitorias, particularmente en los campos de densidad y presión, que no representan el comportamiento físico real del sistema. Se requiere entonces dejar correr la simulación durante algunos pasos para encontrar el punto de estabilidad y a partir de este si poder estudiar el problema físico objetivo.

9.4.1. Determinación de la estabilidad con base a la densidad

En nuestro caso, la estabilidad del sistema se determinó a partir de la relajación del sistema hasta el reposo, partiendo de la configuración inicial (Figura 9.4), en la cual el recipiente se encuentra completamente cerrado. A medida que avanza la simulación, la densidad promedio del fluido presenta oscilaciones temporales

alrededor de un valor medio que se alcanza progresivamente durante el proceso de relajación. Dado que el valor asintótico correspondiente a un tiempo infinito no puede obtenerse de forma directa, se adoptó un criterio práctico de estabilidad basado en la magnitud relativa de dichas oscilaciones.

Se consideró que el sistema alcanza un régimen estable cuando la amplitud de una oscilación, definida como la diferencia entre un valle y una cresta consecutivos, es inferior al 1% del valor medio de densidad asociado a esa misma oscilación. Este criterio permite identificar el instante a partir del cual las variaciones de densidad pueden considerarse despreciables en comparación con el nivel medio del sistema.

Para detectar los valores de valle y cresta, fue necesario reducir previamente el ruido numérico presente en la evolución temporal de la densidad promedio, lo que facilita la identificación de máximos y mínimos locales. Para ello se aplicó un suavizado de la señal mediante un filtro de Savitzky-Golay, utilizando la función `savgol_filter` de la biblioteca SciPy (Virtanen et al., 2020). Este método suaviza la señal utilizando, en cada punto, la información de los valores vecinos, permitiendo reducir las fluctuaciones numéricas sin alterar la forma general de la curva ni la posición de los valles y crestas (Savitzky and Golay, 1964).

9.4.2. Barrido de parámetros de presión B y velocidad del sonido c

Una forma de determinar la estabilidad del modelo es analizar los valores de densidad promedio de las partículas en un determinado paso de tiempo, dado que el valor inicial es de $\rho_0 = 1000 \text{ kg/m}^3$, si este valor baja mucho implicará que las partículas han escapado del tanque debido a inestabilidad numérica, y un aumento significativo implicará problemas claros en la cinemática de las partículas; por lo tanto, luego de ejecutar algunos pasos de la relajación del fluido dentro del recipiente cerrado si el valor se mantiene cerca de ρ_0 quiere decir que los parámetros pueden ser estables a largo plazo.

Partiendo de la premisa anterior se realizó un barrido sobre los parámetros B correspondiente a la ecuación 7.16 y c asociado a la velocidad del sonido, partiendo de los parámetros y dimensiones mencionadas en la sección 9.3.2. Para tal fin se fijó primero el valor de $B = 1 \text{ Pa}$ y se ejecutaron simulaciones con 20 valores de c entre $1 \times 10^{-3} \text{ m/s}$ y 10 m/s con una separación logarítmica entre ellos, ejecutando solo 500 pasos de la simulación para ahorrar tiempo de computo. Se calculó la densidad promedio en el paso 500 de cada una de estas simulaciones y posteriormente se definió el intervalo con valores más realistas para ejecutar la simulación durante 3000 pasos, manteniendo los parámetros mencionados anteriormente.

Un proceso similar se realizó para el parámetro B usando un valor central de la región estable del paso anterior, realizando 20 simulaciones con un valor fijo de $c = 1 \times 10^{-3}$ con 500 pasos de tiempo. Dentro de la región con valores más cercanos a los de la densidad del agua ($\rho_0 = 1000 \text{ kg/m}^3$) escogiendo así como parámetros de presión: $B = (0.48, 1.27, 3.35, 4.0) \text{ Pa}$. Se realizó a estos un estudio de estabilidad a estos en base a la subsección anterior, para escoger el parámetro más óptimo que se empleó para la validación numérica en base a leyes físicas.

9.5. Validación numérica de leyes físicas

La validación numérica del modelo hidrodinámico y del simulador se hizo en base a dos situaciones físicas con solución clásica conocida y que corresponden a la ley de Torricelli y el principio de vasos comunicantes, los cuales se presentan en las secciones 6.1.5 y 6.1.6. De estos se conoce un comportamiento muy definido sobre la velocidad del fluido bajo ciertas condiciones y aproximaciones, para la ley de Torricelli; mientras que la presión y la altura que alcanzan las superficie libres de ambos vasos comunicantes debe coincidir. En base a esto se construyeron pruebas de que el fluido simulado cumple las condiciones de estas leyes físicas.

9.5.1. Validación ley de Torricelli

Bajo la aproximación de flujo ideal bajo gravedad, se utilizó como referencia la ley de Torricelli presentada en la *sección 6.1.6*, dada por la *ecuación 6.15*. La verificación del modelo se realizó comparando esta velocidad con la obtenida a partir de la simulación.

La altura efectiva del fluido se determinó a partir de las partículas que conforman la superficie libre. Definiendo $y_{\text{máx}}$ como la mayor coordenada vertical, se calculó un promedio de y considerando las partículas que cumplen $y \geq y_{\text{máx}} - \Delta y$, con $\Delta y = 0.01 y_{\text{máx}}$, siempre que dicho conjunto contenga al menos 10 partículas; en caso contrario, el criterio se aplicó de forma iterativa hasta satisfacer esta condición. El valor promedio obtenido se adoptó como la altura característica H .

Las partículas que salen por el orificio, en un instante de tiempo dado, se identificaron seleccionando aquellas cuya coordenada horizontal coincide con la posición del orificio o se encuentra a una distancia menor a $2h$, donde h es la longitud de suavizado utilizada en la simulación. Para estas partículas se realizó un seguimiento de su trayectoria desde la salida del recipiente superior hasta antes de su interacción con el fondo o el fluido del recipiente inferior. Mientras que, la fase de caída libre se delimitó identificando el instante en el cual la componente vertical de la velocidad cambia de signo, pasando de valores negativos a positivos.

Para caracterizar la cinemática de las partículas se realizaron dos ajustes independientes. En el primero, se asumió conocida la aceleración gravitacional impuesta en la simulación, ajustando la trayectoria vertical al modelo de movimiento uniformemente acelerado

$$y(t) = y_0 + v_{y0}t - \frac{1}{2}gt^2, \quad (9.1)$$

que puede reescribirse como una relación lineal

$$y(t) + \frac{1}{2}gt^2 = y_0 + v_{y0}t, \quad (9.2)$$

permitiendo estimar y_0 y v_{y0} mediante mínimos cuadrados. En el segundo ajuste no se prescribió el valor de la aceleración, ajustándose la trayectoria vertical a un polinomio de segundo grado

$$y(t) = at^2 + bt + c, \quad (9.3)$$

a partir del cual se obtuvo una aceleración efectiva $g_{\text{fit}} = -2a$, junto con $v_{y0} = b$ y $y_0 = c$.

En ambos casos, la componente horizontal del movimiento se ajustó de forma independiente mediante un modelo lineal

$$x(t) = x_0 + v_{x0}t, \quad (9.4)$$

obteniéndose la velocidad horizontal inicial v_{x0} . El tiempo se definió a partir del número de paso de simulación como $t = Num_{steps} \cdot dt$ con $dt = 5 \times 10^{-5}$ s. Finalmente, la velocidad de salida de cada partícula se determinó a partir de sus componentes iniciales y se comparó con la predicción de la ley de Torricelli.

9.5.2. Vaciado por vasos comunicantes

La validación en base al principio de vasos comunicantes está dado por la altura de las superficies libres respecto a un origen en el eje y , en este caso debido a la construcción de la geometría de la figura 9.3b pudo escogerse el fondo del recipiente, ya que la posición de las bases de los dos vasos coincidía en $y = 5 \times 10^{-4}$ m. Para determinar la altura debido a la variación de las partículas, se usó el mismo criterio mencionado en la sección anterior (9.5.1) del promedio de las partículas de la superficie libre, pero en este caso los promedios se realizaban en dos intervalos dados por el ancho de los vasos, que correspondieron a $vaso_1 \in [0, 1] \times 10^{-3}$ m y $vaso_2 \in [1.5, 2] \times 10^{-3}$ m. Se registró el valor de dichos promedios en cada paso de la variación temporal, para encontrar el valor asintótico hacia el que la altura del fluido se dirigía en cada vaso.

Otra verificación importante realizada fue observar como se comportaban los valores de presión en ambas superficies libres y las capas de fluido inferiores, haciendo promedios sobre cortes horizontales a la misma altura en ambos vasos, calculando solo en los intervalos en los que se definió el ancho del vaso, se realizaron 20 cortes en 4 momentos de la simulación: (i) el momento de apertura inicial (luego de la relajación del fluido), (ii) luego de transcurrir el 20% del tiempo total, (iii) transcurrido el 50% del tiempo total y (iv) el que se consideró el paso final.

Para determinar cual debería ser el paso final se consideró que la diferencia de las superficies libres entre cada uno de los vasos debería ser inferior al valor de la separación Δy entre dos partículas de fluido en las condiciones iniciales antes de la relajación, que correspondió a 2.5×10^{-5} m.

9.6. Tiempo de simulación dependiendo de método de búsqueda de vecinos

Durante la implementación del código y las pruebas experimentales se hizo uso inicialmente el método de búsqueda de vecinos por fuerza bruta explicado en la *sección 7.7.1*, hasta encontrar la combinación de parámetros y modelos de presión más estables. Mientras que los experimentos de validación numérica de leyes físicas se hizo mediante el método de árbol cuaternario, explicado en la *sección 7.7.2*.

Un registro de los valores de tiempo se llevó a cabo en las simulaciones de relajación de fluido en las dos geometrías haciendo uso de ambos métodos de búsqueda, en la escala de 10^{-3} m, con los parámetros $B = 0.48$ Pa y $c = 1 \times 10^{-3}$ m.

Resultados

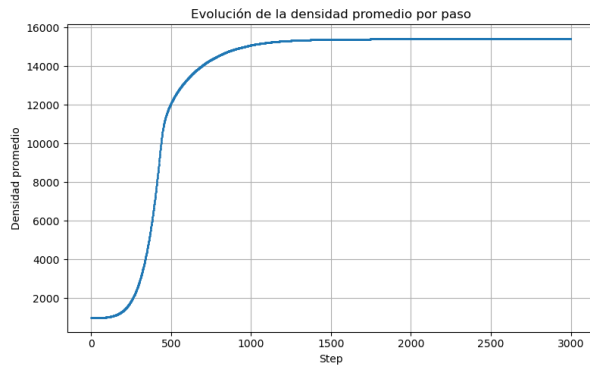
10.1. Pruebas experimentales y parámetros implementados

Diferentes comportamientos fueron obtenidos en cada uno de los experimentos realizados, el proceso de relajación permitió llegar a una configuración estable en cada una de las propuestas experimentales presentadas en 9.3. Solo una de estas configuraciones de parámetros reprodujo un comportamiento similar al de un fluido en presencia de un campo gravitacional, un análisis asociado a cada uno de los resultados en cada configuración se presentará a continuación.

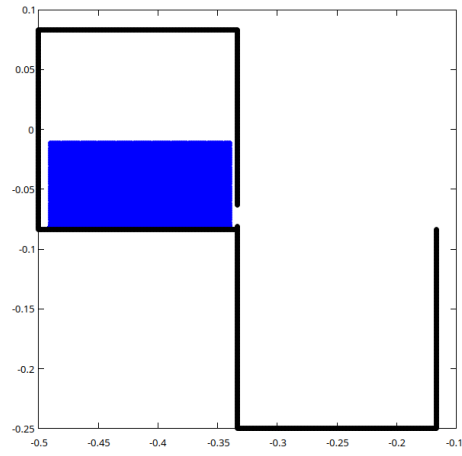
10.1.1. Pruebas iniciales

Como resultado de la implementación de la ecuación de estado lineal y los parámetros: $c = 0.01$ m/s, $g = -9.8$ m/s² y $dt = 1 \times 10^{-3}$, mencionados en la *sección 9.3.1* obtenemos una hipercompresión que llevaba la densidad promedio hasta valores cercanos a $\rho = 16000$ kg/m³, tal como se muestra en la *Figura 10.1a*.

Otra muestra clara del régimen de hipercompresión que se presentó en este modelo, fue la falta de movimiento de desbordamiento lateral, incluso al abrir el agujero lateral, pues el fluido continuaba con su compresión y no salía por el agujero, tal como se muestra en la *Imagen 10.1b*.



(a) Densidad [kg/m³] vs. número de paso de tiempo, empleando la ecuación de estado lineal



(b) Estado de hipercompresión cerca de los 1500 pasos de tiempo con ecuación de estado lineal, representada en una escala de 1×10^{-1} m en cada eje.

Figura 10.1: Resultados obtenidos al emplear la ecuación de estado lineal

El régimen de hipercompresión y los valores de densidad obtenidos en este experimento motivaron, a partir de la revisión de la literatura, el uso de una ecuación de estado de Tait (*ecuación 7.16*) en los análisis siguientes, que se basaron en la comparación del parámetro de presión B de esta, junto con la dependencia con la velocidad del sonido artificial c .

10.1.2. Pruebas con la ecuación de estado tipo Tait

Las simulaciones realizadas con ecuación de estado de Tait (*ecuación 7.16*) resultaron más estables, pues se encontraron valores diferentes que daban una estabilidad numérica para conseguir un fluido estático dentro del recipiente, sin embargo, el comportamiento para llegar a esta estabilidad y el que ocurrió luego de abrir las fronteras (partículas de color rojo en la *Figura 10.2*), presentaba comportamientos diferentes en cada caso, tanto en la estabilidad como en comportamientos asociados a viscosidad y movimiento menos fluido del esperado para un valor de densidad inicial asociado al agua. Un estudio de los valores de los parámetros empleados se muestra a continuación:

Usando los parámetros de la literatura

La simulación realizada con los parámetros B y c , calculados en base a las ecuaciones de la literatura, mostradas en la *sección 7.4.2*, y cuyos valores fueron expresados en la *sección 9.3.2*, llegó a una posición estática y estable en un menor número de pasos de tiempo respecto a configuraciones de parámetros posteriores, siendo esta estabilidad dependiente de la tapa del recipiente, tal como se muestra en la *Figura 10.2*, en la cual se redujo el tamaño de las partículas para mostrar de mejor manera la inserción de partículas de fluido en la tapa superior.

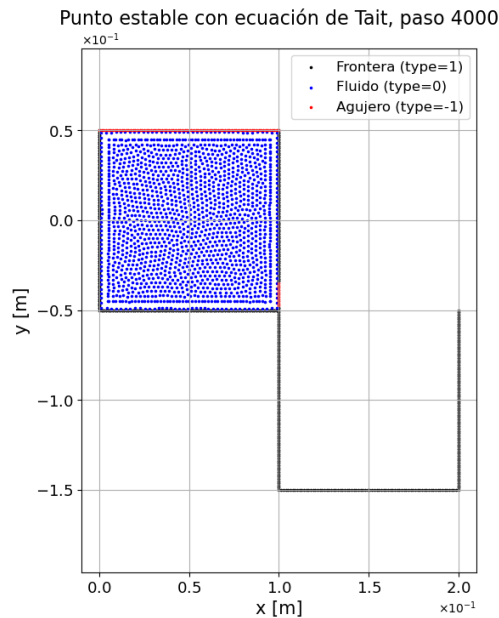


Figura 10.2: Geometría relajada de las partículas de fluido con la ecuación de Tait y los parámetros de la literatura

Cuando hacemos un análisis del comportamiento de la densidad durante la estabilización del fluido en base a las reglas expuestas en la *sección 9.4.1* vemos que este encuentra un punto sobre el cual ocurren pequeñas fluctuaciones en el paso 335 con un valor de la densidad aproximado de 1027 kg/m^3 , tal como se

puede observar en la *Figura 10.3*.

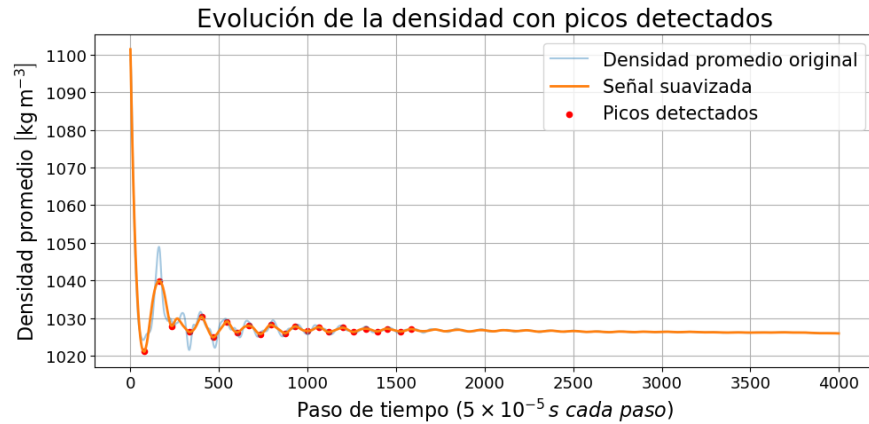
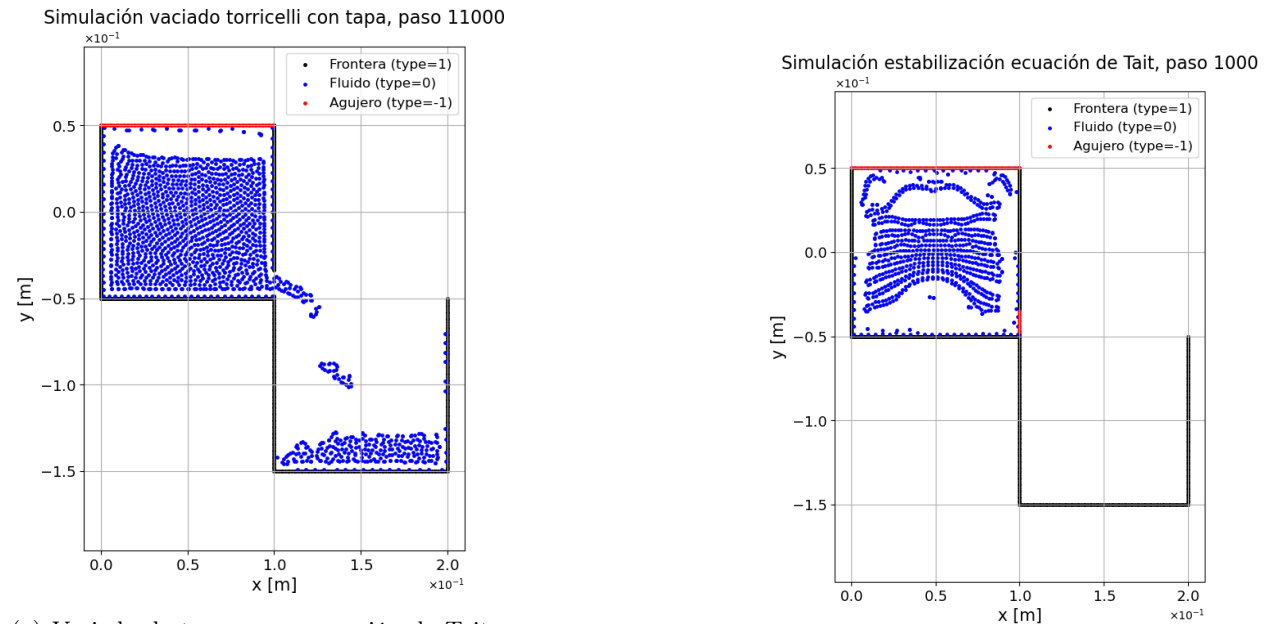


Figura 10.3: Variación de la densidad con cada paso de tiempo, simulación con ecuación de Tait y parámetros en base a la literatura.

Si realizamos el vaciado luego de la estabilización encontramos que el fluido cae de forma irregular, es decir, por segmentos. Esto puede observarse en la *Figura 10.4a*. En caso de que el fluido fuera demasiado para el recipiente, se hizo una simulación con la mitad del fluido, encontrándose un extraño comportamiento repulsivo desde la frontera inferior, tal como se aprecia en la *Figura 10.4b*.



(a) Vaciado de tanque con ecuación de Tait y parámetros de la literatura manteniendo la tapa del recipiente.

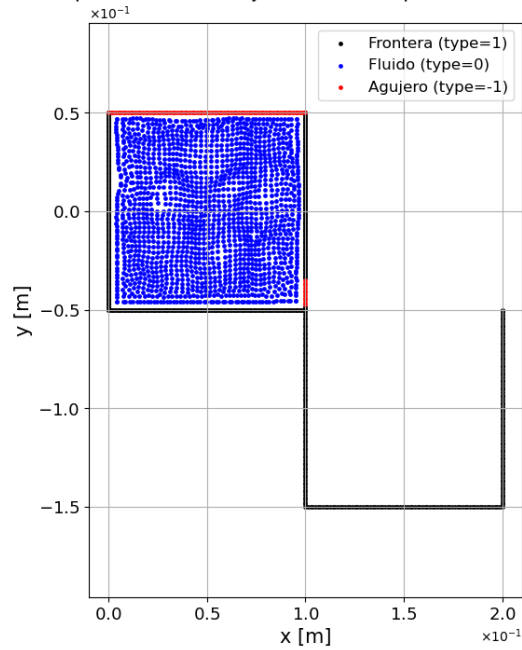
(b) Relajación de un recipiente con la mitad del fluido inicial.

Figura 10.4: Pruebas de comportamiento para la ecuación de Tait con parámetros de la literatura.

Usando la presión extrapolada con los parámetros de la literatura

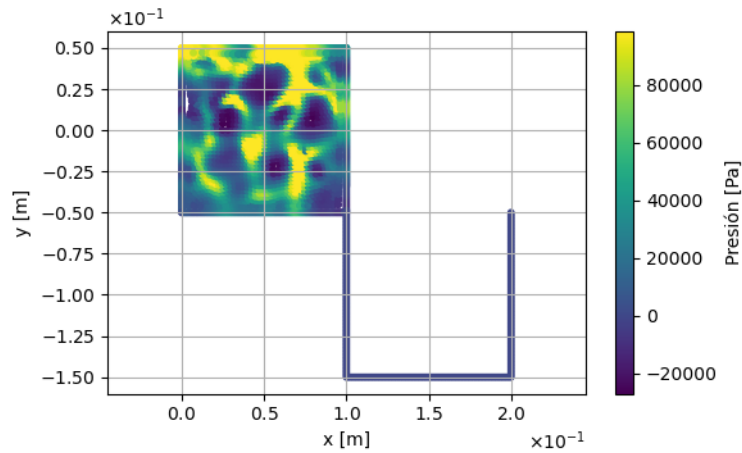
Haciendo uso de los parámetros $B = 28000 \text{ Pa}$, $c = 14 \text{ m/s}$ y $dt = 5 \times 10^{-5} \text{ s}$, mencionados en la *sección 9.3.2*, pero ahora con la presión de fronteras extrapolada desde el fluido tenemos que la simulación solo es estable hasta el paso 700, luego del cual las partículas salen de las fronteras. Graficando este último paso y el comportamiento de la presión en todas las partículas, obtenemos la *figura 10.5*.

Último paso estable relajación usando presión extrapolada



(a) Último paso estable (700) de la relajación de fluido.

Colormap relajación con presión extrapolada



(b) Colormap de la presión en el último paso estable de la relajación.

Figura 10.5: Análisis del último paso estable en la relajación de la geometría de Torricelli, utilizando parámetros de la literatura y presión extrapolada a fronteras.

Usando parámetros empíricos

Al realizar la simulación con los valores que se encontraron de forma empírica en López (2025) se llega a un estado estable que relaja el fluido, incluso siguiendo comportamientos muy similares a como ocurriría en una situación realista tal como se puede ver en la *Figura 10.6*, donde se evidencia una caída del fluido hacia puntos más bajos de donde se encontraban las condiciones iniciales (*Figura 10.6a*) y un posterior efecto rebote (*Figura 10.6b*), estos dos efectos mantienen el fluido en una condición de oscilación que va disminuyendo con cada ciclo.

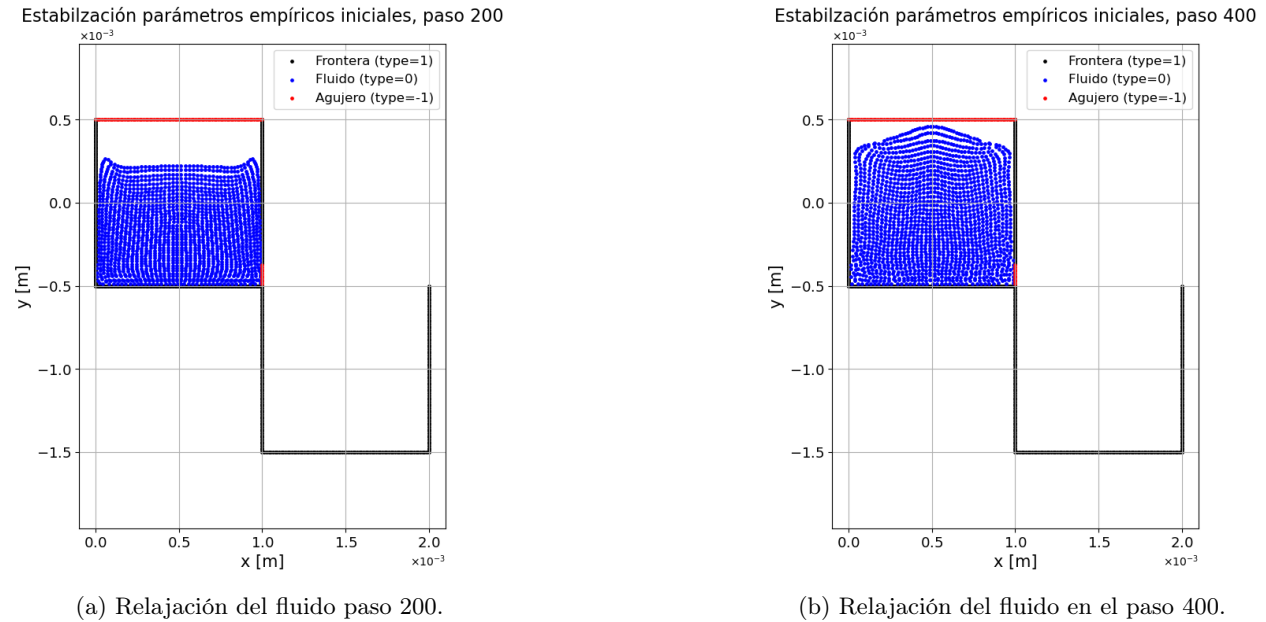


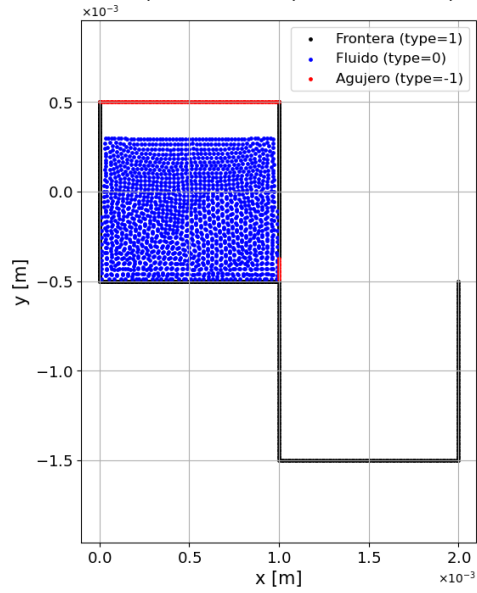
Figura 10.6: Proceso de relajación del fluido usando los parámetros iniciales encontrados de forma empírica por López (2025)

En el código propuesto en López (2025) se tiene que el punto de estabilidad suficiente para abrir la compuerta se encuentra en el paso 2300, dicho valor de tiempo se puede observar en la *Figura 10.7a*. Al abrir la compuerta para pasos posteriores se obtuvo una salida del fluido de forma segmentada en pequeñas agrupaciones de partículas, tal como se presenta en la *Figura 10.7b*.

10.2. Resultados del barrido del parámetro de presión de la ecuación de Tait B y de la velocidad del sonido artificial c .

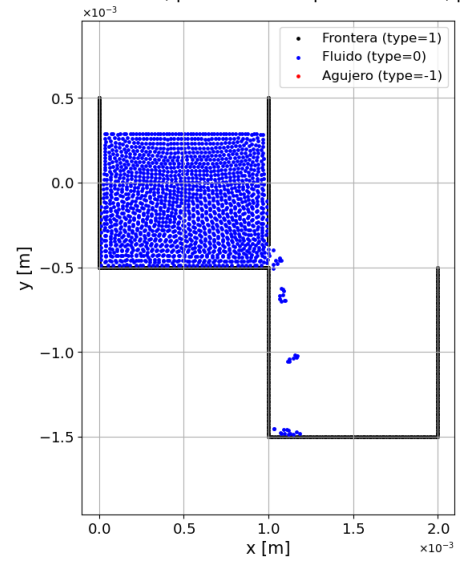
Al graficar las densidades promedio del barrido de parámetros inicial sobre la velocidad del sonido c , manteniendo un valor de $B = 1$ tal como se menciona en la *sección 9.4.2*, se obtuvo la *Figura 10.8*, en la cual es notorio que existe una zona de estabilidad que va hasta $c = 2 \times 10^{-1} \text{ m/s}$, sin embargo, esta zona empieza desde el primero valor en el eje x , por lo que analizar valores anteriores se considera necesario.

Estabilización parámetros empíricos iniciales, paso 2300



(a) Relajación del fluido, en el paso final (2300).

Vaciado de Torricelli, parámetros empíricos iniciales, paso 2800



(b) Vaciado del fluido en geometría de Torricelli, paso 2800.

Figura 10.7: Estabilización y vaciado de tipo Torricelli con los parámetros empíricos: $B = 1 \text{ Pa}$ y $c = 1 \times 10^{-2} \text{ m/s}$

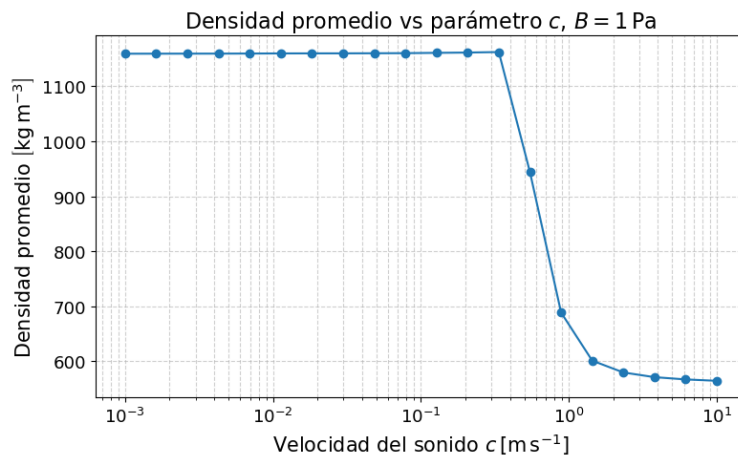


Figura 10.8: Gráfica de velocidad del sonido vs densidad promedio [kg/m^3] ejecutando 500 pasos de tiempo.

Dada esa región de estabilidad en pasos iniciales, tenemos que los valores de densidad oscilan mucho desde tiempo iniciales (500 pasos) hasta los 3000 pasos finales por lo que se define que la densidad promedio se calculará solo sobre el último 20% del tiempo, obteniendo así la *Figura 10.9*, la cual también presenta una zona de estabilidad muy marcada, decidimos entonces que para un estudio más meticuloso de la simulación se puede escoger el central de este último intervalo es decir $c = 1 \times 10^{-3} \text{ m/s}$, el cual tiene un valor aproximado, asociado a densidad promedio, en la gráfica de $\rho_0 = 1190 \text{ kg/m}^3$.

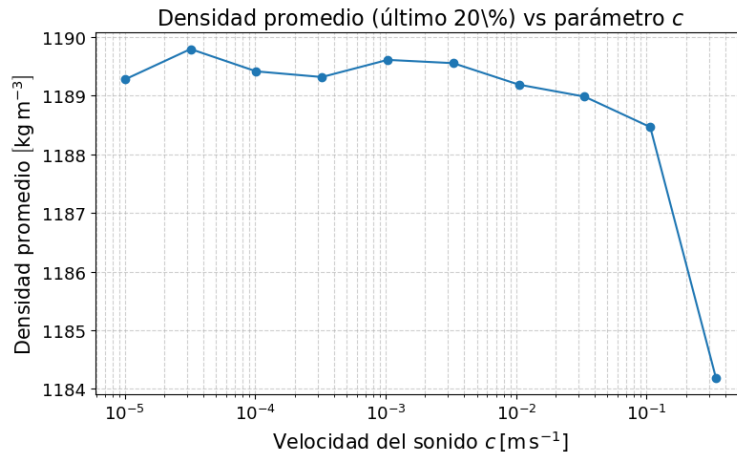


Figura 10.9: Gráfica de velocidad del sonido vs densidad promedio [kg/m^3] ejecutando 3000 pasos de tiempo.

Empleando este valor de velocidad del sonido obtenemos que al realizar el barrido sobre el parámetro B que este se comporta de manera casi lineal en un intervalo que consideramos estable, al tener densidades mayores a 1000 kg/m^3 , tal como se muestra en la *Figura 10.10*. A diferencia de la gráfica de valores de c donde la densidad promedio se mantiene constante, acá no existe una región donde el comportamiento sea siempre muy similar.

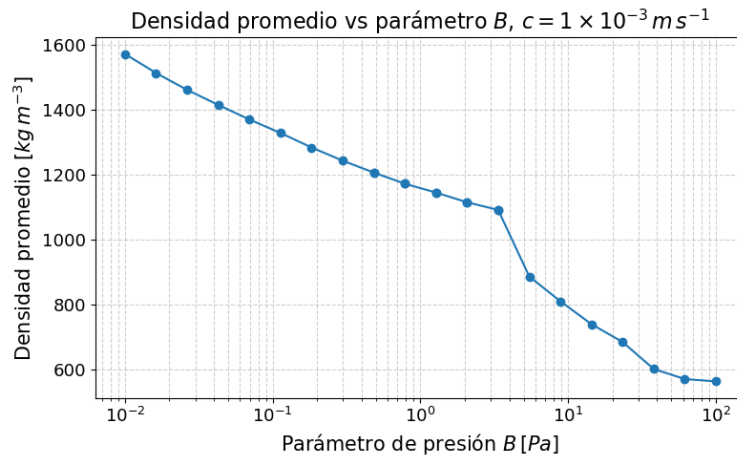


Figura 10.10: Gráfica del parámetro de presión B vs densidad promedio [kg/m^3] ejecutando 500 pasos de tiempo.

Como resultado del análisis de estabilidad (*sección 9.4.1*) a los 4 valores de B escogidos en base al criterio del párrafo anterior se obtuvieron las gráficas presentadas en la *Figura 10.11*.

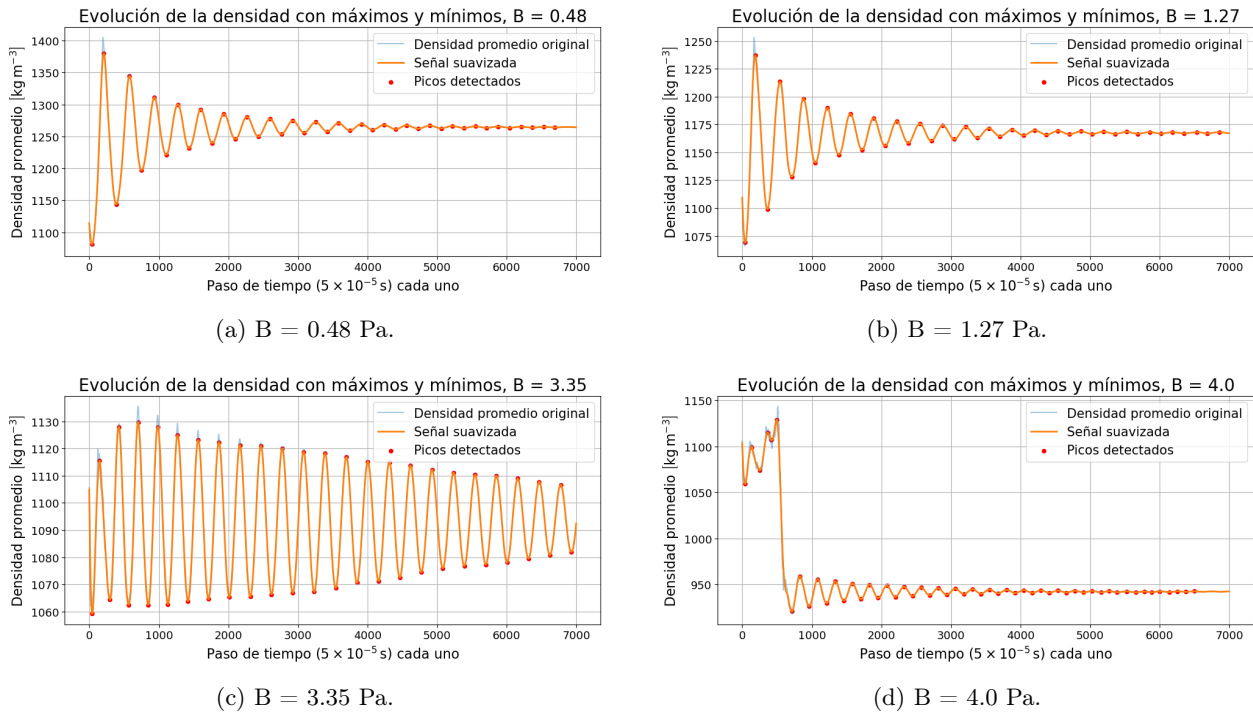


Figura 10.11: Gráfica del número de paso de tiempo vs. densidad promedio $[kg/m^3]$ del fluido, simulación con parámetros empíricos.

Solo el proceso de relajación de las figuras 10.11a y 10.11b, resultaron los suficientemente estables como para considerarse en simulaciones de validación numérica posteriores, sin embargo, la simulación con parámetro $B = 1.27$ resultó en la pérdida de unas pocas partículas al transcurrir pocos pasos como puede observarse en la *Figura 10.12*, esto sumado al hecho de que el valor de $B = 1$ Pa, con $c = 0.01$ m/s ya había sido descartado por comportamiento granular hizo que este parámetro también fuera descartado para simulaciones posteriores.

El único de los valores implementados de B para $c = 1 \times 10^{-3}$ m/s lo suficientemente estable fue $B = 0.48$ Pa, por lo se escogió como valor para análisis posteriores teniendo presente que la densidad promedio al momento de la relajación fue de 1265 kg/m^3 . Su relajación hasta un valor adecuado en base a la *sección 9.4.1*, tardó 4000 pasos de tiempo correspondientes a 0.2 s.

Los experimentos hasta este punto, incluyendo el barrido de parámetros se realizaron haciendo uso del método de búsqueda de vecinos de fuerza bruta, donde los 4000 pasos de relajación del fluido (1600 partículas) tardaban aproximadamente 766.71 segundos, lo que corresponde a un valor de 5.22 pasos por segundo; mientras que usando el método de árbol cuaternario el tiempo bajó hasta aproximadamente 212.95 s en ambas geometrías consideradas, lo que corresponde a aproximadamente 18.78 pasos por segundo. Esta mejora en la eficiencia del 72.22 % hace que a partir de acá se siga haciendo uso del método de árbol cuaternario para las

Estabilización parámetros empíricos $B = 1.27$, paso 214

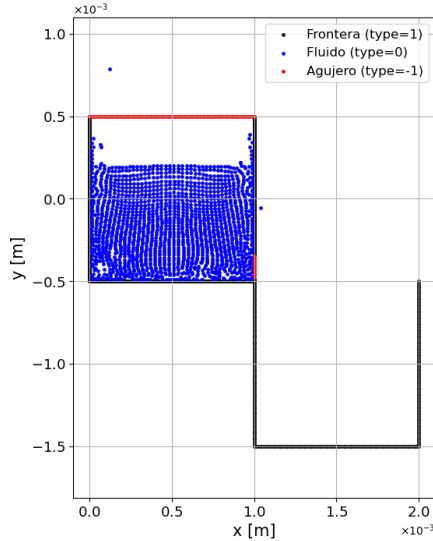


Figura 10.12: Escape de partículas durante la simulación usando el parámetro empírico de la ecuación de estado tipo Tait $B = 1.27$ Pa.

simulaciones.

Otro resultado obtenido es que al aumentar el número de partículas, la simulación solo se mantuvo estable en los 4000 pasos hasta un valor de 2401 partículas (malla de 49×49). Estos cambios en el número de partículas en un mismo espacio causan cambios en el comportamiento del fluido que no son útiles para las validaciones numéricas posteriores.

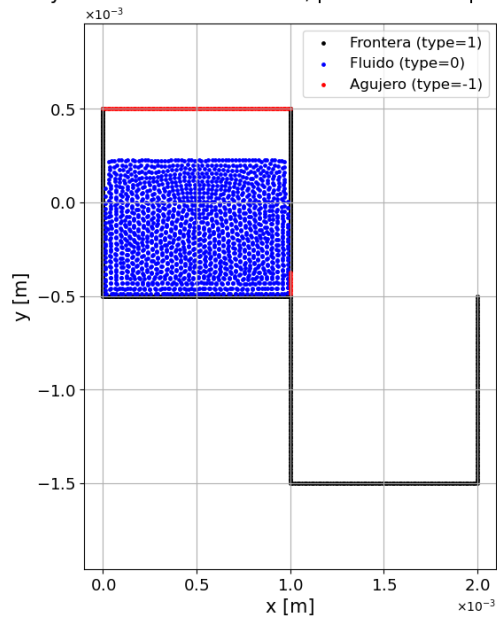
10.3. Simulaciones estables de validación numérica en base a leyes físicas

Para realizar las validaciones numéricas se usó el grupo de parámetro más estable correspondiente a $B = 0.48$ Pa, $c = 1 \times 10^{-3}$ m/s y $dt = 5 \times 10^{-5}$ s; la estabilidad conseguida tanto para la geometría de comprobación de la ley de Torricelli como para el principio de vasos comunicantes son mostrados en la *Figura 10.13*.

10.3.1. Vaciado por caída libre (ley de Torricelli)

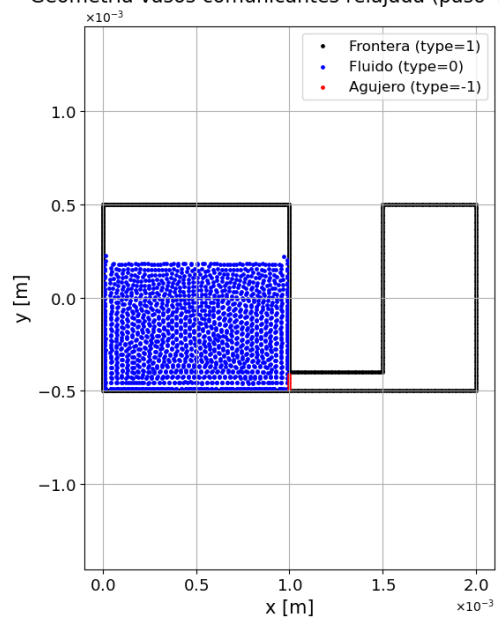
Luego de la relajación del fluido y de eliminar las compuertas (partículas en color rojo *Figura 10.13a*) las partículas salen con un comportamiento de fluidés adecuado, pero un poco agrupado, sin llegar a como ocurría en el caso de los parámetros de la literatura (*sección 10.1.2*), podemos observar también un comportamiento de empuje de las partículas cayendo, este comportamiento se ve bien reflejado en la *Figura 10.14* asociada al paso de tiempo 1000 desde que se abrieron las partículas de compuerta.

Geometría ley de torricelli estabilizada, parámetros empíricos (paso 4000)



(a) Relajación del fluido en la geometría de la ley de Torricelli.

Geometría vasos comunicantes relajada (paso 4000)



(b) Relajación de fluido en la geometría de vasos comunicantes.

Figura 10.13: Relajación del fluido para los parámetros encontrados empíricamente $B = 0.48 Pa$ y $c = 1 \times 10^{-3} m/s$.

Simulación en la geometría de la ley de Torricelli, paso 1000

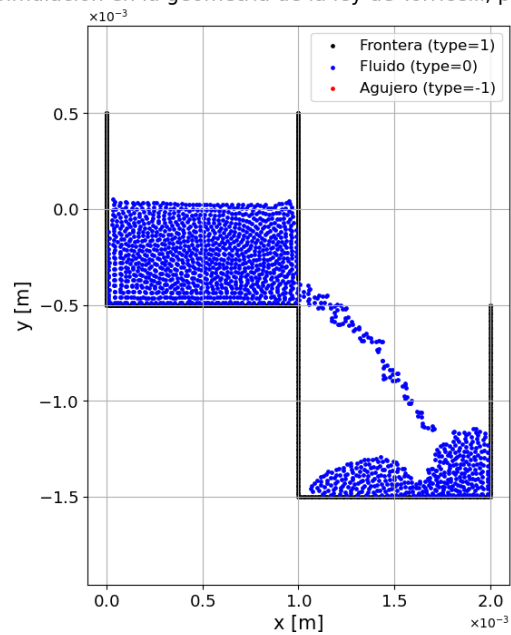
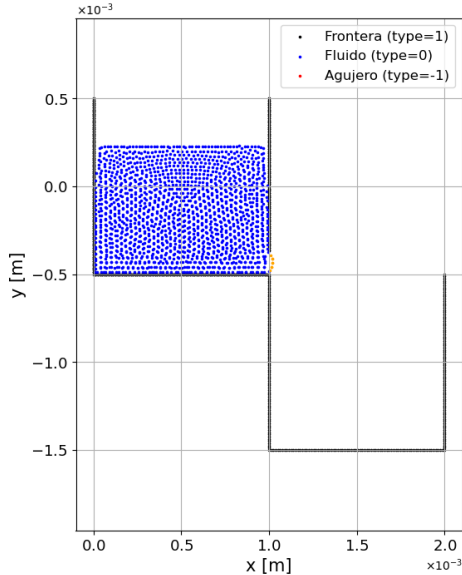


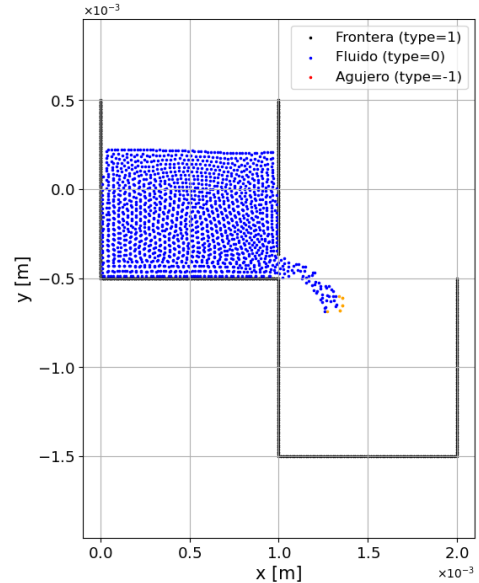
Figura 10.14: Vaciado de tanque usando la geometría de Torricelli y parámetros empíricos, paso de tiempo 1000.

Partículas para el cálculo de Torricelli usando el paso 15 (paso 15)



(a) Paso 15.

Partículas para el cálculo de Torricelli, paso 150



(b) Paso 150.

Figura 10.15: Partículas consideradas para los cálculos de validación de la ley de Torricelli.

Para verificar el cumplimiento de la ley de Torricelli y la caída libre de las partículas, se siguió el procedimiento mencionado en la *sección 9.5.1*. Se identificaron las partículas que están en la posición del agujero o un poco más a la derecha, usando el criterio de $2h$, basado en la sección antes mencionada, haciendo uso como base del 15 de la simulación, en el cual unas pocas partículas comenzaron a salir, pero tampoco habían salido las suficientes para que la caída se viera afectada por interacción entre ellas. Las 8 partículas seleccionadas con el criterio mencionado pueden observarse cayendo, en el paso 15 y 150 de tiempo, marcadas de naranja en la *Figura 10.15*, donde el tamaño de partícula se disminuyó para apreciar de mejor manera el número de estas evitando solapamientos.

La trayectoria que recorrieron las 8 partículas seleccionadas antes de chocarse con el fondo o el fluido que ya había caído anteriormente presentó una semiparábola en todos los casos, como puede observarse en la *Figura 10.16*.

Como resultado de aplicar el proceso de mínimos cuadrados a estas trayectorias encontramos tanto valores de velocidad inicial en x , como en y y un valor de la gravedad promedio cercano al valor original de 9.04 m/s , es decir, con un error de aproximadamente 7.8% asociado a interacciones entre partículas durante la caída. Los valores de cada una de las velocidades con y sin el uso del valor de g , se pueden observar en la *Tabla 10.1*.

Trayectorias de partículas para el cálculo de velocidad inicial

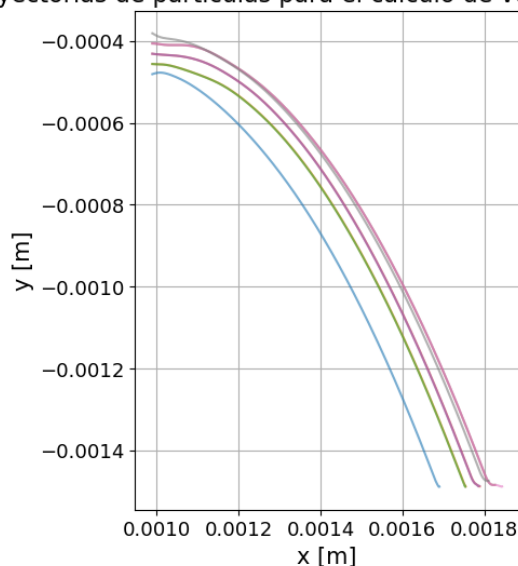


Figura 10.16: Trayectorias de las partículas seleccionadas para el cálculo de Torricelli, entre la salida del recipiente superior y antes del rebote en el inferior.

id partícula	Ajuste con $g = 9.8$		Ajuste sin valor de g		
	v_{x_0} [m]	v_{y_0} [m]	v_{x_0} [m]	v_{y_0} [m]	g obtenido [m/s]
829	0.0470	0.0117	0.047044	0.0077	9.305
909	0.0496	0.0109	0.049583	0.0044	8.998
948	0.0496	0.0108	0.049642	0.0044	8.997
949	0.0503	0.0110	0.050319	0.0040	8.949
988	0.0503	0.0110	0.050266	0.0041	8.948
989	0.0519	0.0123	0.051861	0.0076	9.246
1028	0.0518	0.0133	0.051820	0.0041	8.715
1068	0.0520	0.0120	0.052008	0.0066	9.144

Cuadro 10.1: Velocidades iniciales obtenidas por ajuste de las trayectorias de partículas. Comparación entre ajuste con gravedad conocida y ajuste sin valor de g .

Obtenemos entonces un valor promedio de la velocidad inicial en x , que coinciden pasando el valor de g como parámetro o no, de $v_{x_0} = 0.0503$ m/s. El valor de la velocidad inicial promedio en y , si varía en ambos casos teniendo $v_{y_0,g} = 0.0116$ m/s y $v_{y_0} = 0.0054$ m/s, sabemos que en teoría estos dos valores deberían ser 0, por lo que no pasar el valor de g resulta más útil ya que es un orden de magnitud menor la velocidad en y que en x .

Ahora bien, durante el paso 15 de la simulación del vaciado del tanque se encontró que la altura de la superficie libre promedio dentro del contenedor fue de aproximadamente 7.2875×10^{-4} m. Lo que implicaba en la ley de Torricelli una velocidad de aproximadamente 0.12 m/s. Esto hace que la velocidad encontrada

Cuadro 10.2: Valores de las superficies libres de los vasos en diferentes momentos de la simulación.

Porcentaje de simulación	Vaso	Número de partículas	Altura del fluido [m]	Presión [Pa]
0 %	Izquierdo	55	6.804×10^{-4}	0.1826
0 %	Derecho	Vacío	No aplica	No aplica
20 %	Izquierdo	51	5.027×10^{-4}	0.1590
20 %	Derecho	23	2.866×10^{-4}	0.1655
50 %	Izquierdo	49	4.518×10^{-4}	0.1505
50 %	Derecho	23	3.928×10^{-4}	0.1414
100 %	Izquierdo	51	4.388×10^{-4}	0.1441
100 %	Derecho	23	4.173×10^{-4}	0.1527

sea menor a la velocidad esperada, siendo esta del 41.9% de la que se espera de la ley de Torricelli.

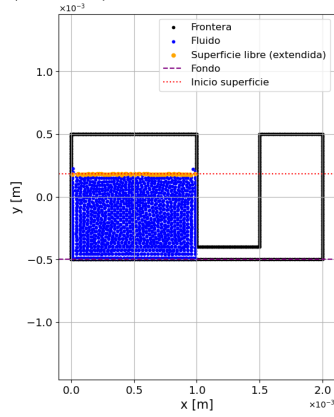
10.3.2. Vaciado por vasos comunicantes

Luego de realizar la simulación del vaciado entre los vasos comunicantes desde el vaso izquierdo (vaso 1) hacia el vaso derecho (vaso 2) de la *Figura 10.13b*, se tuvieron que simular 90000 mil pasos de tiempo, correspondientes a 4.5 s dentro de la simulación, para alcanzar una altura similar entre las dos superficies libres usando el criterio de la separación máxima en Δy de $2.5 \times 10^{-5} m$, mencionada en la *sección 9.5.2*. Los 4 momentos de la simulación, asociados a cada vaso en un momento inmediato a la relajación, al 20% de todos los pasos ejecutados hasta alcanzar la misma altura, al 50% de dichos pasos y al ejecutar todos hasta de la misma altura en ambos vasos; de los cuales se explica mejor en la *sección 9.5.2* mencionada, tuvieron valores de altura y presión que se pueden ver en la *Tabla 10.2*.

Resulta notorio que con el paso del tiempo, ambas superficies libre (vaso izquierdo y derecho) tienen a conseguir la misma altura cerca de un valor de $4 \times 10^{-4} m$, esto se puede observar en la *Figura 10.17*. Vemos también que los valores de presión en las superficies en cada tiempo son muy cercanos, mientras que el número de partículas es casi constante a través del tiempo.

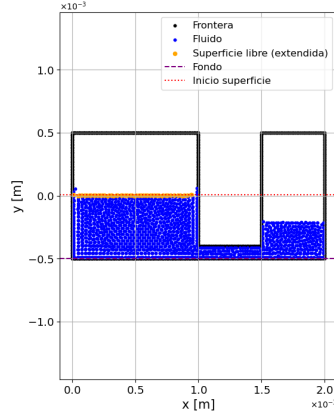
El paso de fluido entre un lugar y otro ocurre más rápido al principio de la simulación, debido al gradiente de presión que es mayor en el recipiente que contiene mayor fluido y va bajando a medida que el vaso de la derecha se llenaba, una muestra de esto se puede observar en los cambios en los valores de presión en el tubo comunicante en cada uno de los instantes de tiempo considerados anteriormente, que se muestran en la *Figura 10.18*, donde la escala de color representa la magnitud de la presión en pascales (Pa), siendo los colores más claros indicativos de mayores valores de presión.

Superficie libre paso de vaciado inicial, vasos comunicantes



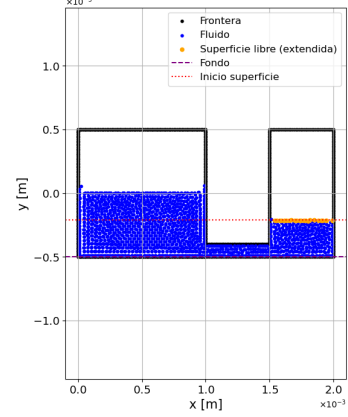
(a) 0 % de simulación.

Superficie libre vaso izquierdo en el 20% de los pasos



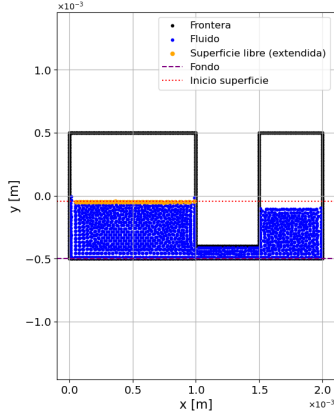
(b) 20 % – Vaso izquierdo.

Superficie libre vaso derecho en el 20% de los pasos



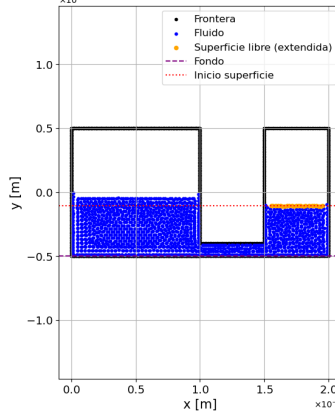
(c) 20 % – Vaso derecho.

Superficie libre vaso izquierdo en el 50% de los pasos



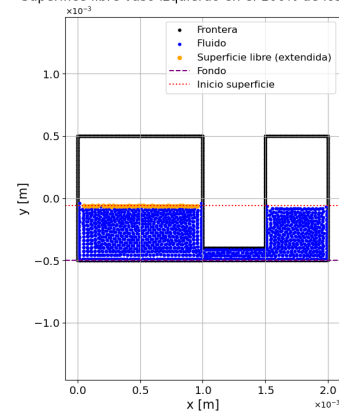
(d) 50 % – Vaso izquierdo.

Superficie libre vaso derecho en el 50% de los pasos



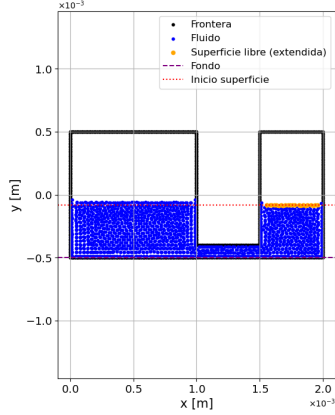
(e) 50 % – Vaso derecho.

Superficie libre vaso izquierdo en el 100% de los pasos



(f) 100 % – Vaso izquierdo.

Superficie libre vaso derecho en el 100% de los pasos



(g) 100 % – Vaso derecho.

Figura 10.17: Evolución temporal de las superficies libres del fluido en los vasos durante la simulación.

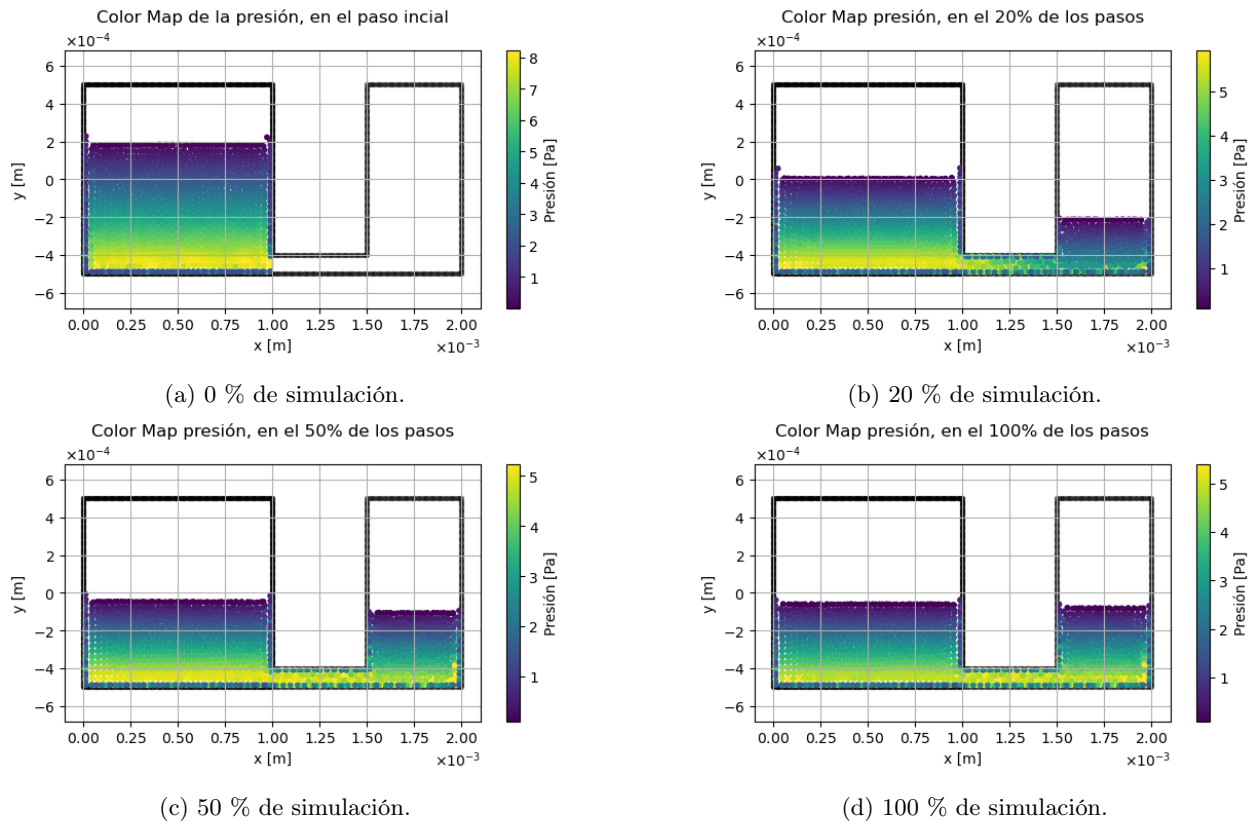


Figura 10.18: Mapas de color de la presión del fluido en distintos instantes de la simulación.

Discusión de los resultados

Las pruebas experimentales de los diferentes parámetros y modelos, especialmente asociados a la forma en la que la simulación trataba la presión dieron resultados concluyentes especialmente en el uso de la ecuación de Tait para sistemas en presencia de un campo gravitacional y superficies libres, sin embargo, cuando observamos los parámetros asociados a estas ecuaciones notamos que los valores de la literatura no son especialmente estables bajo estas condiciones, especialmente con el modelo de interacción de frontera en base a una sola línea de partículas de tipo I.

El modelo de repulsión de frontera tampoco resultó completamente funcional por sí solo y muestra que se requiere de un valor de presión adecuado en las fronteras para funcionar completamente, tal como se pudo observar en la *figura 10.5* aunque tampoco posee la capacidad de contención necesaria en base a una sola línea de partículas de tipo I. Ahora bien cuando analizamos el barrido de parámetros encontramos regiones que son estables al modelo de frontera con presión fija igual a 0, pero no que no presentan una región de parámetros estable muy grande, sin pérdida de por lo menos algunas partículas, tal como se presentó en la *figura 10.12*. El comportamiento de fluido con los parámetros de presión y velocidad del sonido adecuados, el número de partículas y la escala, presenta un comportamiento muy similar al que ocurre físicamente, este comportamiento funciona en una base de parámetros que no poseen los valores reales de la física, pero que sigue los lineamientos del modelo WCSPH en el cual la velocidad del sonido, la viscosidad y la presión, están representados por valores artificiales. Este comportamiento adecuado fue confirmado en las dos validaciones numéricas.

En la validación basada en la ley de Torricelli encontrar valores de velocidad inicial menores tiene sentido físico debido a la presencia de interacción con las fronteras que otorga velocidad en la componente y , además de la presencia de viscosidad que alejan el comportamiento real de la ley ideal de Torricelli. El ajuste de parámetros directamente con el polinomio de segundo grado permitió incluso obtener un valor de aceleración gravitacional promedio menor al real con un error del 7.75%, que aunque es alto no tuvo en cuenta las aceleraciones producidas por el campo de presión dentro de la caída.

Mientras, el comportamiento en la ley de vasos comunicantes presenta una tendencia a buscar la misma altura en ambos vasos, generada por la diferencia en el campo de presión inferior de cada vaso asociado a la altura del fluido y también que la variación espacial entre cada paso disminuye a medida que las presiones se igualan. Otros comportamientos numéricos que presenta este experimento y que resultan remarcables pues no ocurren solo en este pueden verse en la *figura 10.5b*, donde ocurre en la parte cercana a la frontera una caída notoria de los valores de presión incluso en el fondo del recipiente donde la presión debería ser máxima, esto debido al modelo de frontera implementado donde la presión de las partículas que la componen es de 0; mientras que en la parte superior algunas partículas que interactúan sobre la frontera presentan cohesión a la frontera aún teniendo el modelo de repulsión de Lennard-Jones activado, esto no corresponde a un error funcional sino que representa procesos reales que son simulados también en SPH asociados a procesos de tensión superficial y capilaridad, que son estudiados más a detalle en trabajos como el de Tartakovsky and Meakin (2005).

Perspectivas

En base a este estudio y como perspectivas de trabajos futuros se propone estudiar de mejor manera las interacciones fluido, frontera tanto de presión como un modelo de viscosidad entre ambos, apoyando la estabilidad de los campos cercanos a la frontera y con ello el modelo de repulsión de frontera que se escoja.

Asociado al mismo modelamiento de frontera y con intención de mantener un dominio más estable de la longitud de suavizado, se pueden implementar más líneas de frontera como se hizo por ejemplo en Korzani et al. (2014) haciendo también un estudio de la relación de la densidad de partículas con criterios de estabilidad y contención del fluido, este estudio puede también orientarse en la implementación de partículas de frontera fantasma (tipo II), entendiendo y estudiando el aumento de coste computacional que estas tienen.

Otro criterio que luego del estudio se considera necesario para determinar los mejores parámetros de simulación es su relación con la discretización del fluido y con ello el número de partículas y los cambios en la masa que esto implica, calculando los intervalos en los que ciertos parámetros resisten el aumento en la discretización.

Pruebas de comparación con valores realistas y comparaciones con experimentos reales pueden ser otro punto de estudio necesario para poder seguir escalando el *software* hasta ser aplicable a problemas reales, pensando desde el diseño desde un planteamiento de prototipos progresivos hasta alcanzar comparaciones con modelos realistas solo cuando las validaciones numéricas son lo suficientemente fuertes a cambios en la geometría o modelos que varíen parámetros como el número de partículas o incluso implementando modelos más fuertes en longitud de suavizado variable.

Conclusiones

El trabajo mostró la clara sensibilidad de SPH y especialmente del modelo WCSPH a parámetros asociados a la presión del sistema y a la velocidad del sonido artificial que el modelo implica, estos parámetros aunque referenciados y estudiados en la literatura requieren de interacciones adecuadas con la frontera, modelos que no son lo suficiente detallados en los estudios sobre todo en su relación con los parámetros de presión y velocidad del sonido; no considerar estas relaciones no solo genera inestabilidades numéricas fuertes sino que también afecta la utilidad de modelos de interacción independientes como sucedió con el potencial de repulsión de Lennard-Jones.

Se comprobó la habilidad de este modelo en representación del comportamiento de problemas físicos como lo fueron la ley de Torricelli, entendiendo que el modelo tiende a representar valores más asociados a la realidad que separan los resultados de la situación ideal que esta ley propone, mientras que el comportamiento del fluido en la validación por vasos comunicantes permitió no solo confirmar un comportamiento adecuado sino también las limitaciones que tiene asociadas tanto a la interacción con fronteras como a la discretización del fluido. El correcto funcionamiento del modelo en estas pruebas de validación permite revisiones posteriores de la inserción de modelos asociados a la cohesión, capilaridad y otras interacciones con las fronteras, no sin requerir validaciones adicionales de parámetros y modelos de presión que permitan regiones más estables de simulación tanto en dichos parámetros como en la discretización del fluido.

La metodología en base al diseño de *software* permitió mantener una idea global del *software* haciendo mucho más eficiente la construcción continua de las herramientas necesarias para realizar el estudio del problema, no sin evidenciar las limitaciones que tiene en cuanto al *software* científico, pues requiere de un marco teórico fuerte y definido específicamente al problema, donde solo el estudio de parámetros representa un trabajo completo y un fin en sí mismo. Los planteamientos de necesidades de entidades externas asociadas a capacidades de cómputo y almacenamiento, resultan claras para escalar el alcance del proyecto y la complejidad de los modelos. Donde en este caso resultó muy rentable la construcción del árbol cuaternario para ahorrar tiempo de cómputo.

Bibliografía

- Adami, S., Hu, X. Y., and Adams, N. A. (2012). A generalized wall boundary condition for smoothed particle hydrodynamics. *Journal of Computational Physics*, 231(21):7057–7075.
- Afanasiev, K. E., Makarchuk, R. S., and Popov, A. Y. (2008). Comparative analysis of the sph and isph methods. In Krause, E., Shokin, Y. I., Resch, M., and Shokina, N., editors, *Computational Science and High Performance Computing III*, volume 101 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, pages 191–198. Springer, Berlin, Heidelberg.
- Batchelor, G. K. (1967). *An Introduction to Fluid Dynamics*. Cambridge University Press.
- Becker, M. and Teschner, M. (2007). Weakly compressible sph for free surface flows. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '07, pages 209–217, Goslar, Germany. Eurographics Association.
- Belytschko, T., Krongauz, Y., Organ, D., Fleming, M., and Krysl, P. (1996). Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139(1–4):3–47.
- Cordero, P. (2013). *Métodos Numéricos*. Departamento de Ingeniería, Universidad de Chile, Santiago, Chile. Apuntes de clase, disponible en cec.uchile.cl.
- Dehnen, W. and Aly, H. (2012). Improving convergence in smoothed particle hydrodynamics simulations without pairing instability. *Monthly Notices of the Royal Astronomical Society*, 425(2):1068–1082.
- Eirís, A., Ramírez, L., Couceiro, I., and et al. (2023). Mls-sph-ale: A review of meshless-fv methods and a unifying formulation for particle discretizations. *Archives of Computational Methods in Engineering*, 30:4959–4981.
- Fraga Filho, C. A. D. (2019). *Smoothed Particle Hydrodynamics: Fundamentals and Basic Applications in Continuum Mechanics*. Springer Nature Switzerland AG, Cham.
- Gingold, R. A. and Monaghan, J. J. (1977). Smoothed particle hydrodynamics: Theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389.
- Guan, X., Sun, P., Lyu, H., Liu, N., Peng, Y., Huang, X., and Xu, Y. (2022). Research progress of sph simulations for complex multiphase flows in ocean engineering. *Energies*, 15(23):9000.
- Hairer, E., Lubich, C., and Wanner, G. (2006). *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, volume 31 of *Springer Series in Computational Mathematics*. Springer, Berlin, Heidelberg, 2nd edition.
- Hernquist, L. and Katz, N. (1989). Treesph: A unification of sph with the hierarchical tree method. *The Astrophysical Journal Supplement Series*, 70:419–446.
- Korzani, M. G., Galindo Torres, S., Williams, D., and Scheuermann, A. (2014). Numerical simulation of tank discharge using smoothed particle hydrodynamics. *Applied Mechanics and Materials*, 553:168–173.

- Landau, L. D. and Lifshitz, E. M. (1987). *Fluid Mechanics*. Pergamon Press.
- Lautrup, B. (2011). *Physics of Continuous Matter: Exotic and Everyday Phenomena in the Macroscopic World*. CRC Press, Boca Raton.
- Liu, G. R. and Liu, M. B. (2003). *Smoothed Particle Hydrodynamics: A Meshfree Particle Method*. World Scientific Publishing Company.
- Liu, G. R., Liu, M. B., and Li, Z. (n.d.). Lecture notes on the smoothed particle hydrodynamics (sph). Unpublished lecture notes, Centre for Advanced Computations in Engineering Science, National University of Singapore.
- Liu, M. B., Liu, G. R., and Zong, Z. (2008). An overview on smoothed particle hydrodynamics. *International Journal of Computational Methods*, 5(1):135–188.
- López, A. (2025). Waterdischarge. <https://github.com/AndresL7/WaterDischarge>.
- López Ramírez, C. and Quiroga Peláez, L. F. (2025). Guía instructiva y metodológica para el desarrollo de software científico intuitivo. Manuscrito no publicado.
- Marrone, S., Antuono, M., Colagrossi, A., Colicchio, G., Lugni, C., et al. (2011). -sph model for simulating violent impact flows. *Computer Methods in Applied Mechanics and Engineering*, 200:1526–1542.
- Monaghan, J. J. (1992). Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30:543–574.
- Monaghan, J. J. (1994). Simulating free surface flows with sph. *Journal of Computational Physics*, 110(2):399–406.
- Monaghan, J. J. (2005). Smoothed particle hydrodynamics. *Reports on Progress in Physics*, 68(8):1703–1759.
- Monaghan, J. J. and Gingold, R. A. (1983). Shock simulation by the particle method sph. *Journal of Computational Physics*, 52(2):374–389.
- Morris, J. P., Fox, P. J., and Zhu, Y. (1997). Modeling low reynolds number incompressible flows using SPH. *Journal of Computational Physics*, 136(1):214–226.
- Nguyen, V. P., Rabczuk, T., Bordas, S., and Duflo, M. (2008). Meshless methods: A review and computer implementation aspects. *Mathematics and Computers in Simulation*, 79:763–813.
- OpenAI (2025). Chatgpt. Versión gratuita, modelos GPT-4.2 y GPT-5.
- Patel, V. G. and Rachchh, N. V. (2020). Meshless method – review on recent developments. *Materials Today: Proceedings*, 26:1598–1603. 10th International Conference of Materials Processing and Characterization.
- Savitzky, A. and Golay, M. J. E. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36(8):1627–1639.
- Scardovelli, R. and Zaleski, S. (1999). Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, 31:567–603.

- SE, F. (2026). Viscosidad dinámica y cinemática: definición y ejemplos. <https://www.flottweg.com/es/wiki/tecnica-de-separacion/viscosidad-dinamica/>. Consultado el 11 de enero de 2026.
- Serway, R. A. and Jewett, J. W. (2008). *Physics for Scientists and Engineers*. Cengage Learning, 7 edition.
- Shadloo, M. S. et al. (2016). Smoothed particle hydrodynamics method for fluid flows, towards industrial applications. *Computers & Fluids*, 136:11–34.
- Shadloo, M. S., Zainali, A., Yildiz, M., and Suleman, A. (2011). A robust weakly compressible sph method and its comparison with an incompressible sph. *International Journal for Numerical Methods in Engineering*, 89:939–956.
- Springel, V. (2010). Smoothed particle hydrodynamics in astrophysics. *Annual Review of Astronomy and Astrophysics*, 48:391–430.
- Tartakovsky, A. M. and Meakin, P. (2005). Modeling of surface tension and contact angles with smoothed particle hydrodynamics. *Physical Review E*, 72(2):026301.
- Tryggvason, G., Scardovelli, R., and Zaleski, S. (2011). *Direct Numerical Simulations of Gas–Liquid Multiphase Flows*. Cambridge University Press.
- Vacondio, R., Altomare, C., De Leffe, M., Hu, X., et al. (2021). Grand challenges for smoothed particle hydrodynamics numerical schemes. *Computational Particle Mechanics*, 8:575–590.
- Violeau, D. (2012). *Fluid Mechanics and the SPH Method: Theory and Applications*. Oxford University Press, Oxford.
- Violeau, D. and Leroy, A. (2013). On the maximum time step in weakly compressible sph. *Journal of Computational Physics*, 256:388–415.
- Virtanen, P. et al. (2020). Scipy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods*, 17:261–272.
- White, F. M. (2017). *Fluid Mechanics*. McGraw-Hill, 7 edition.
- Ye, T., Pan, D., Huang, C., and Liu, M. B. (2019). Smoothed particle hydrodynamics (sph) for complex fluid flows: Recent developments in methodology and applications. *Physics of Fluids*, 31(1):011301.
- Yoo, H. S., Jo, Y. B., and Kim, E. S. (2024). Comparative study of wcsph, eisph and explicit incompressible-compressible sph (eicsph) for multi-phase flow with high density difference. *Journal of Computational Physics*, 506:112930.
- Zhao, C., Yu, Y., Haidn, O. J., and Hu, X. (2025). Physics-driven complex relaxation for multi-body systems of sph method. *Computer Physics Communications*, 313:109615.
- Çengel, Y. A. and Cimbala, J. M. (2018). *Mecánica de fluidos: Fundamentos y aplicaciones*. McGraw-Hill Interamericana, 4 edition. Traducción al español de *Fluid Mechanics: Fundamentals and Applications*.

Apéndice A: Perfil de usuarios

Usuario Investigador

Nombre del perfil: Estudiante de hidrodinámica.

Rol: Estudiante de pregrado en astronomía realizando la tesis en simulación hidrodinámica.

Objetivos:

- Realizar modelado de las condiciones de frontera físicas del sistema en formato de partículas con representación computacional.
- Visualizar la estructura de frontera generada.
- Agregar al sistema una estructura de partículas que simula la distribución inicial del fluido.
- Visualizar las condiciones iniciales de frontera y fluido.
- Simular fuentes y pérdidas de material.
- Simular el comportamiento del fluido dentro de la caja de simulación.
- Comprobar la consistencia física de la simulación.
- Realizar un análisis estadístico.

Necesidades:

- Archivos JSON editables para los parámetros de las condiciones iniciales del sistema.
- Archivo JSON de preprocesado sobre los algoritmos y modelos a usar en la simulación.
- Mantener un monitoreo constante del programa en ejecución mediante `console logging`.
- Exportar resultados en formato `.csv` y `.png` para cada estado de la simulación.
- Generar un archivo de resumen de la simulación.

Dificultades:

- Nivel medio de programación.
- Poca experiencia en diseño de *software* aplicado en C++.
- Necesidad de documentación clara y casos de uso.
- Dificultad en el planteamiento del *workflow* para generar informes del proceso.
- Poca experiencia en la creación de pruebas automáticas.

Nivel técnico: Medio.

Frecuencia de uso del sistema: Intensivo durante periodos académicos.

Apéndice B: Historias de usuario.

#	Cómo	Quiero	Para	Tipo de necesidad
Fase 1: Estructura base del simulador.				
1 (Global)	Usuario	Tener un programa de simulación de fluidos basado en el método SPH,	Poder ejecutar simulaciones hidrodinámicas en geometrías definidas, como el deprimido de la Terminal del Norte de Medellín.	RN
2	Usuario	Ejecutar la simulación desde un solo comando de terminal que se encuentra como un archivo dentro de la carpeta de la simulación	Ejecutar todo la simulación desde un solo comando inicial y que tengo un flujo predeterminado	RF
3	Usuario	Llenar un archivo de parámetros para las propiedades de las partículas de la simulación en formato JSON.	Pasar los parámetros de simulación y poder cambiarlos según corresponda	RF
4	Usuario	Llenar un archivo de preprocesado que me permita determinar los algoritmos y modelos a implementar en la simulación	Decidir cuáles de los modelos programados se van a implementar dependiendo del proceso a representar	RF
5	Usuario	Que el sistema genere un archivo de logs que muestre las variables cargadas desde el archivo JSON, y las variables derivadas a partir de estas o variables inicializadas en cero para modificarse durante ejecución.	Poder verificar que todos los valores fueron leídos y asignados correctamente.	RF
Fase 2: Condiciones iniciales y geometría.				
6	Usuario	Seleccionar entre componentes de geometrías regulares.	Estructurar las condiciones de frontera y del fluido en base a geometrías regulares.	RF
7	Usuario	El software debe poder construir cualquier geometría regular en 2D.	Adaptarse a todo tipo de condiciones iniciales, sin importar la dimensión teórica del sistema.	RF
8	Usuario	El software debe poder agregar, quitar y realizar agujeros dentro de las geometrías antes construidas.	Hacer particularidades sobre las geometrías para adaptarse a la solución de diferentes condiciones de vaciado y llenado del sistema.	RF
9	Usuario	Llenar un archivo de parámetros de la geometría inicial de la simulación en formato JSON.	Seleccionar la geometría del fluido y sus fronteras, además de modificar las dimensiones propias de cada una.	RF
10	Usuario	Validar que las condiciones iniciales hayan sido pasadas de su estructura en Python a su estructura en JSON	Verificar que el código en Python si reconoce la estructura de datos creada desde Python	RF
11	Usuario	Poder observar y validar gráficamente las condiciones iniciales generadas.	Observar la distribución del fluido y geometría antes de comenzar la simulación	RF
Fase 3: Algoritmia del modelo SPH.				
3.1. Kemels.				
12	Usuario	Poder seleccionar el tipo de kernel de interpolación a utilizar en la simulación,	Adaptar los cálculos del modelo SPH a diferentes condiciones físicas o experimentales.	RF
13	Usuario	Que el simulador incluya al menos los siguientes kernels de interpolación: Gaussiano, Cubic Spline, Quartic Spline y Wendland,	Asegurar que se cumplen los métodos estándar requeridos para los experimentos.	RF
14	Usuario	Validar visualmente que el kernel seleccionado	Validar que cumple con la estructura necesaria para ser utilizado en cálculos posteriores	RF

3.2. Búsqueda de Vecinos.				
15	Usuario	Seleccionar entre 2 métodos de búsqueda de vecinos	Poder realizar la búsqueda de vecinos según la eficiencia buscada	RF
16	Usuario	Tener los métodos de búsqueda de vecinos: Linked List y Árbol cuaternario.	Tener métodos de búsqueda de vecinos eficientes para simulaciones complejas	RF
17	Usuario	Validar gráficamente que la búsqueda de vecinos funciona correctamente	Observar las áreas de interacción de una partícula y las partículas con las cuales interactúa	RF
3.3. Tipos de Algoritmo SPH.				
18	Usuario	Tener un modelo de SPH de tipo XSPH con longitud de suavizado variable y que sea expandible.	Poder agregar otras interacciones a las ecuaciones de manera que se puedan representar los diferentes procesos de la simulación.	RF
19	Usuario	Tener un sistema de selección por preprocesado de los diferentes procesos adicionales al modelo base de SPH que serán incluido dentro de la simulación.	Añadir y quitar modelos de simulación adicionales al modelo base.	RF
3.4. Integradores.				
20	Usuario	Seleccionar entre 2 integradores del movimiento de las partículas que conserven la energía	Realizar los cálculos de las variables y su cambio temporal	RF
21	Usuario	Tener los integradores simbióticos: Leapfrog y Velocity Verlet	Tener varios integradores que conserven la energía y adaptarlos según la necesidad de la simulación	RF
Fase 4: Validación y Resultados.				
22	Usuario	Que el simulador permita validar los resultados numéricos mediante cálculos teóricos aplicados a geometrías con solución analítica como comparación o mediante visualización.	Asegurarme de que el comportamiento del fluido simulado sea físicamente correcto.	RNF
23	Usuario	Validar gráficamente el comportamiento de la simulación	Ver la representación física de los valores numéricos obtenidos y compararla con otro simulador conocido	RF
24	Usuario	Que el programa esté diseñado para escalar en términos de número de partículas y tamaño del dominio.	Poder ejecutar simulaciones complejas, variando la escala.	RNF
25	Usuario	Obtener los resultados de la simulación en formato JSON	Tener los resultados en un formato fácilmente accesible para hacer estadística y cálculos adicionales	RF
26	Usuario	Realizar estadística sobre valores importantes como densidad, energía y velocidad.	Revisar que las variables importantes de la simulación conserven sentido físico.	RNF
27	Usuario	Comparar los resultados con otro simulador hidrodinámico validado externamente.	Validar numéricamente los resultados por otro método ya verificado	RNF
28	Usuario	Obtener un archivo log de resumen de la simulación que cuente variables importantes y las selecciones anteriores	Observar datos importantes que resuman las características y funciones representativas de la simulación dependiendo de la entidad física a desarrollar	RF

Apéndice C: Requisitos funcionales.

Fase 4: Validación y Resultados.			
Código	Nombre	Fecha	Prioridad del requisito
RF_023	Visualización gráfica del comportamiento de la simulación.	23/06/2025	Alta
Descripción:	El sistema debe permitir al usuario validar gráficamente el comportamiento de la simulación, mostrando la representación física de los valores numéricos obtenidos y comparándolos con referenciales o resultados de otros simuladores.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Resultados numéricos de la simulación.	Sistema	Gráficos 2D/3D comparativos.	Debe poder superponer o visualizar lado a lado series de datos.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario invoca la función de validación gráfica tras completar la simulación. 2. El sistema carga los datos numéricos y los datos de referencia (teóricos o de otro simulador). 3. Genera gráficos comparativos (por ejemplo, perfiles de velocidad, densidad). 4. Muestra la visualización en la interfaz. 		
Post - condiciones	El usuario dispone de una vista gráfica que facilita la inspección y comparación de resultados.		
Código	Nombre	Fecha	Prioridad del requisito
RF_025	Exportación de resultados en formato JSON.	24/06/2025	Alta
Descripción:	El sistema debe poder generar un archivo JSON que contenga todos los resultados de la simulación (posiciones, velocidades, densidades, etc.), facilitando su uso para análisis estadísticos o herramientas externas.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Resultados de simulación en memoria.	Sistema	Carpeta results con archivo state_*.json con datos estructurados.	Debe seguir un esquema JSON predefinido y validable.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario solicita la exportación en JSON al terminar la simulación. 2. El sistema serializa las estructuras de datos internas al esquema JSON. 3. Guardo los archivos JSON en la carpeta results. 		
Post - condiciones	Existe un archivo JSON listo para consumirse por scripts o herramientas de post-procesado.		
Código	Nombre	Fecha	Prioridad del requisito
RF_027	Generación de archivo log de resumen de simulación.	24/06/2025	Alta
Descripción:	El sistema debe crear un archivo de log de resumen que incluya las variables y métricas clave de la simulación (tiempos, energía total, recuento de partículas, parámetros usados), para ofrecer un vistazo rápido de los resultados.		
Rol	Sistema / Usuario		
Entradas	Fuente	Salida	Restricciones
Eventos de ejecución y variables clave.	Sistema	Archivo summary.log con secciones organizadas.	Debe ser legible en texto plano y contener encabezados claros.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. Al finalizar la simulación, el sistema recolecta métricas (p.ej., tiempo total, promedio de densidad). 2. Redacta un log estructurado con encabezados y valores. 3. Guarda summary.log en el directorio de salida. 		
Post - condiciones	El usuario dispone de un resumen completo en texto, listo para revisión o adjuntar a informes.		

Fase 1			
Código	Nombre	Fecha	Prioridad del requisito
RF_002	Ejecución de simulación con un solo comando	23/06/2025	Alta
Descripción:	El sistema debe permitir ejecutar toda la simulación mediante un único comando de terminal localizado en la carpeta de la simulación, iniciando automáticamente el flujo predeterminado.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Comando de terminal inicial	Usuario	Iniciación y logs de estado de la simulación.	El archivo ejecutable ya debe existir dentro de la carpeta de la simulación y el comando debe existir dentro de la documentación.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario abre la terminal en la carpeta raíz de la simulación. 2. Escribe y ejecuta el comando predefinido. 3. El sistema valida la existencia y permisos del archivo. 4. Si es válido, lanza el flujo de simulación predeterminado. 		
Post - condiciones	La simulación se ejecuta en su totalidad y se almacenan los resultados en las rutas configuradas.		
Código	Nombre	Fecha	Prioridad del requisito
RF_003	Configuración de parámetros de partículas en JSON	23/06/2025	Alta
Descripción:	El sistema debe permitir al usuario definir y modificar, en un archivo JSON, las propiedades de las partículas de la simulación (masa, viscosidad, densidad, etc.), y cargar estos parámetros antes de la ejecución.JSON.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Archivo parameters.json con propiedades de las partículas	Usuario	Parámetros cargados en el sistema para su uso en la simulación.	El JSON debe estar bien formado y contener todos los campos requeridos por el esquema.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario edita el archivo parameters.json en la carpeta de simulación. 2. Completa los campos requeridos (masa, viscosidad, etc.). 3. Guarda el archivo. 4. El sistema valida la sintaxis y carga los parámetros en memoria. 		
Post - condiciones	Los parámetros de partículas quedan disponibles y validados para la simulación.		
Código	Nombre	Fecha	Prioridad del requisito
RF_004	Configuración de archivo de preprocesado para selección de modelos.	24/06/2025	Alta
Descripción:	El sistema debe permitir al usuario especificar, mediante un archivo de preprocesado en formato JSON, los algoritmos y modelos SPH que se van a emplear en la simulación, de acuerdo con el proceso físico a representar.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Archivo preprocess.json con selección de modelos y algoritmos.	Usuario	Modelos y algoritmos cargados para la fase de preprocesado.	El JSON ha de referenciar únicamente modelos implementados y cumplir con el esquema definido.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario abre preprocess.json. 2. Selecciona los modelos (kernels, fuerzas, condiciones) a emplear. 3. Guarda el archivo. 4. El sistema valida la estructura y carga las opciones seleccionadas. 		
Post - condiciones	Los modelos definidos quedan listos para su aplicación durante la simulación.		
Código	Nombre	Fecha	Prioridad del requisito
RF_005	Generación de archivo de logs de parámetros cargados	24/06/2025	Media
Descripción:	El sistema debe generar un archivo de logs que documente todas las variables inicializadas (cargadas desde JSON o inicializadas en cero) y sus valores durante la ejecución, para verificar que hayan sido leídas y asignadas correctamente.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Parámetros de simulación cargados.	Sistema	Archivo de logs (simulation.log) con valores de variables.	Debe incluir todas las variables requeridas y estar en un formato legible (texto plano).
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. Al inicio, el sistema carga parámetros desde JSON. 2. Crea simulation.log e imprime cada variable con su valor inicial. 3. Durante la ejecución, registra cambios o inicializaciones adicionales. 		
Post - condiciones	El archivo de logs contiene un histórico completo de los valores de las variables para su revisión.		

Fase 2: Condiciones Iniciales			
Código	Nombre	Fecha	Prioridad del requisito
RF_006	Selección de geometrías regulares predefinidas	23/06/2025	Alta
Descripción:	El sistema debe permitir al usuario seleccionar geometrías regulares básicas (e.g. cuadrados, círculos, cilindros) para definir el dominio de simulación, sin necesidad de cargar archivos externos.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Tipo de geometría seleccionada (e.g. "círculo", "cubo").	Usuario	Tipo de geometría aplicada al dominio.	Solo se permiten geometrías implementadas por el sistema.
Procesos / Pasos para ejecutar	1. El usuario selecciona una geometría regular desde el archivo de configuración. 2. El sistema interpreta la selección y activa el generador correspondiente.		
Post - condiciones	El sistema prepara el espacio de simulación con la geometría seleccionada.		
Código	Nombre	Fecha	Prioridad del requisito
RF_007	Generación de geometría regular en 2D/3D	24/06/2025	Alta
Descripción:	El sistema debe construir automáticamente geometrías regulares en 2D o 3D a partir de parámetros proporcionados (e.g. dimensiones, orientación), para facilitar la definición del dominio de simulación sin archivos externos.		
Rol	Sistema / Usuario		
Entradas	Fuente	Salida	Restricciones
Parámetros geométricos (radio, altura, ancho, etc.).	Usuario	Geometría generada con partículas	Los parámetros deben ser físicamente válidos y positivos.
Procesos / Pasos para ejecutar	1. El usuario define los parámetros en el archivo JSON. 2. El sistema construye la geometría usando distribución de partículas adecuada. 3. Se integra al dominio de simulación.		
Post - condiciones	La geometría regular es construida y lista para ser simulada.		
Código	Nombre	Fecha	Prioridad del requisito
RF_008	Modificación de geometrías con adición, sustracción o perforación.	24/06/2025	Alta
Descripción:	El sistema debe permitir agregar, quitar o realizar agujeros dentro de geometrías previamente construidas, para representar casos con vaciados, orificios o compartimentos internos.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Instrucciones de modificación geométrica.	Usuario	Geometría modificada.	Las operaciones deben ser consistentes topológicamente.
Procesos / Pasos para ejecutar	1. El usuario especifica adiciones o sustracciones de formas sobre la geometría base. 2. El usuario especifica la ubicación de agujeros en la geometría base. 3. El sistema realiza las operaciones booleanas sobre las regiones.		
Post - condiciones	La geometría queda ajustada a las condiciones deseadas de vaciado o compartimentación.		
Código	Nombre	Fecha	Prioridad del requisito
RF_009	Configuración de parámetros geométricos en JSON	24/06/2025	Alta
Descripción:	El sistema debe permitir configurar las propiedades geométricas iniciales del dominio y del fluido a través de un archivo en formato JSON editable por el usuario.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Archivo geometry_config.json.	Usuario	Parámetros geométricos cargados.	El archivo debe estar bien formado y cumplir con el esquema.
Procesos / Pasos para ejecutar	1. El usuario edita el JSON con dimensiones y tipos de geometría. 2. El sistema carga y valida los valores. 3. Se construyen las regiones definidas.		
Post - condiciones	El dominio y el fluido tienen la forma y dimensiones definidas en el archivo JSON.		
Código	Nombre	Fecha	Prioridad del requisito
RF_010	Validación de estructura de condiciones iniciales	24/06/2025	Media
Descripción:	El sistema debe validar que las condiciones iniciales escritas en Python fueron correctamente convertidas a JSON, asegurando la integridad de los datos al pasar entre lenguajes.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones

Estructura de datos Python → JSON.	Sistema / Usuario	Validación de equivalencia de estructura.	Debe cumplirse la correspondencia de campos y tipos de datos.
Procesos / Pasos para ejecutar	1. El sistema convierte la estructura Python a JSON. 2. Ejecuta validaciones de equivalencia. 3. Reporta errores o confirmación de éxito.		
Post - condiciones	Se garantiza que los datos en JSON son fieles a la intención inicial del código en Python.		
Código	Nombre	Fecha	Prioridad del requisito
RF_011	Visualización gráfica de condiciones iniciales.	24/06/2025	Alta
Descripción:	El sistema debe permitir al usuario visualizar de manera gráfica la geometría y el fluido antes de iniciar la simulación, para facilitar la validación visual.		
Rol	Sistema / Usuario		
Entradas	Fuente	Salida	Restricciones
Geometría y distribución inicial del fluido.	Sistema	Vista previa en 2D/3D de condiciones iniciales.	Debe permitir renderizado rápido y navegación básica.
Procesos / Pasos para ejecutar	1. Tras generar condiciones iniciales, el sistema genera una vista previa. 2. El usuario inspecciona visualmente desde una interfaz.		
Post - condiciones	El usuario puede corregir errores antes de ejecutar la simulación real.		

Fase 3: Algoritmia del modelo SPH			
3.1 Kernels			
Código	Nombre	Fecha	Prioridad del requisito
RF_012	Selección del tipo de kernel de interpolación.	24/06/2025	Alta
Descripción:	El sistema debe permitir al usuario seleccionar el tipo de kernel de interpolación a utilizar, ajustando los cálculos del modelo SPH a las necesidades del experimento o condición física.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Tipo de kernel (e.g. "Cubic Spline").	Usuario	Kernel activo en la simulación.	Solo se pueden usar kernels implementados.
Procesos / Pasos para ejecutar	1. El usuario elige el kernel desde el archivo de configuración o preprocesado. 2. El sistema carga el módulo correspondiente.		
Post - condiciones	La simulación se ejecuta utilizando el kernel seleccionado.		
Código	Nombre	Fecha	Prioridad del requisito
RF_013	Inclusión de kernels estándar (Gaussiano, Cubic Spline, etc.)	24/06/2025	Alta
Descripción:	El sistema debe incluir al menos los siguientes kernels de interpolación: Gaussiano, Cubic Spline, Quartic Spline y Wendland, garantizando opciones estándar para simulaciones SPH.		
Rol	Sistema / Usuario		
Entradas	Fuente	Salida	Restricciones
Lista de kernels disponibles.	Sistema	Disponibilidad de múltiples kernels para selección.	Deben cumplir los criterios matemáticos de suavizamiento.
Procesos / Pasos para ejecutar	1. El sistema contiene implementaciones de todos los kernels. 2. El usuario puede seleccionar cualquiera de ellos para la simulación.		
Post - condiciones	El simulador ofrece flexibilidad para elegir el kernel más adecuado.		
Código	Nombre	Fecha	Prioridad del requisito
RF_014	Validación visual del kernel seleccionado	24/06/2025	Media
Descripción:	El sistema debe permitir validar gráficamente el comportamiento del kernel seleccionado, asegurando que su forma sea la adecuada para los cálculos posteriores.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Kernel seleccionado.	Sistema / Usuario	Gráfica del kernel activo.	Debe reflejar correctamente la función definida.
Procesos / Pasos para ejecutar	1. El usuario selecciona un kernel. 2. El sistema genera una gráfica con su distribución.		
Post - condiciones	El usuario puede comprobar visualmente si el kernel es adecuado.		
3.2 Búsqueda de Vecinos			
Código	Nombre	Fecha	Prioridad del requisito
RF_015	Selección de método de búsqueda de vecinos	24/06/2025	Alta
Descripción:	El sistema debe permitir al usuario seleccionar entre al menos dos métodos de búsqueda de vecinos para optimizar la eficiencia de la simulación.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Método de búsqueda seleccionado.	Usuario	Método activado para ejecución.	Solo se permiten métodos previamente implementados.
Procesos / Pasos para ejecutar	1. El usuario indica el método deseado (e.g. Linked List, Árbol cuaternario (u octal)). 2. El sistema carga el módulo correspondiente.		
Post - condiciones	La búsqueda de vecinos se realiza con el método elegido.		
Código	Nombre	Fecha	Prioridad del requisito
RF_016	Inclusión de Linked List y Árbol cuaternario (u octa) como métodos de búsqueda	24/06/2025	Alta
Descripción:	El sistema debe contar con los métodos de búsqueda de vecinos: Linked List y Árbol cuaternario (u octal), ofreciendo opciones eficientes según la complejidad del caso.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
		71	

Listado de métodos de búsqueda disponibles.	Sistema	Acceso a métodos eficientes.	Deben estar correctamente implementados y probados.
Procesos / Pasos para ejecutar	1. El sistema incluye ambos métodos. 2. El usuario selecciona uno según sus necesidades.		
Post - condiciones	El sistema tiene disponibles dos opciones funcionales de búsqueda de vecinos.		
Código	Nombre	Fecha	Prioridad del requisito
RF_017	Validación visual del método de búsqueda de vecinos	24/06/2025	Media
Descripción:	El sistema debe permitir observar gráficamente las áreas de interacción de una partícula y sus vecinos, validando el correcto funcionamiento del método de búsqueda.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Posición de partículas y radio de interacción.	Sistema	Visualización de vecindades.	Debe reflejar el algoritmo de vecindad activo.
Procesos / Pasos para ejecutar	1. El sistema ejecuta el método de búsqueda. 2. Se genera una visualización con partículas y sus vecinos conectados.		
Post - condiciones	El usuario puede verificar que los vecinos se identifican correctamente.		
3.3 Tipos de Algoritmo SPH.			
Código	Nombre	Fecha	Prioridad del requisito
RF_018	Implementación de modelo SPH tipo XSPH con longitud de suavizado variable.	24/06/2025	Alta
Descripción:	El sistema debe incluir un modelo de simulación basado en SPH del tipo XSPH que permita longitud de suavizado variable y capacidad de extensión con nuevas interacciones.		
Rol	Sistema / Usuario		
Entradas	Fuente	Salida	Restricciones
Configuración del modelo base (XSPH).	Usuario	Modelo XSPH operativo con h variable.	La longitud de suavizado debe actualizarse adecuadamente en cada paso.
Procesos / Pasos para ejecutar	1. El sistema inicializa las variables del modelo, incluida h. 2. La simulación ajusta dinámicamente h durante la ejecución.		
Post - condiciones	El modelo SPH opera correctamente bajo el esquema XSPH con parámetros adaptativos.		
Código	Nombre	Fecha	Prioridad del requisito
RF_019	Selección de procesos adicionales desde archivo de preprocesado	24/06/2025	Alta
Descripción:	El sistema debe permitir seleccionar, añadir o quitar procesos adicionales al modelo base de SPH mediante un archivo de preprocesado, permitiendo una simulación modular.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Archivo preprocess. json con lista de procesos adicionales.	Usuario	Módulos adicionales activados o desactivados.	Los módulos deben estar implementados correctamente y registrarse en el sistema.
Procesos / Pasos para ejecutar	1. El usuario edita el archivo de preprocesado. 2. Selecciona qué procesos agregar (e.g. disipación artificial, fuerza externa). 3. El sistema interpreta y activa solo los procesos indicados.		
Post - condiciones	El modelo SPH base se ejecuta con los módulos adicionales definidos por el usuario.		
3.4 Integradores			
Código	Nombre	Fecha	Prioridad del requisito
RF_020	Selección de integradores del movimiento	24/06/2025	Alta
Descripción:	El sistema debe permitir al usuario seleccionar entre al menos tres integradores de movimiento que conserven la energía (Leapfrog, Velocity Verlet, etc.).		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Integrador seleccionado.	Usuario	Integrador aplicado en la simulación.	Debe ser un integrador con implementación comprobada.
Procesos / Pasos para ejecutar	1. El usuario selecciona un integrador. 2. El sistema lo aplica en los cálculos temporales de posición y velocidad.		
Post - condiciones	La evolución temporal del sistema se rige por el integrador seleccionado.		
Código	Nombre	Fecha	Prioridad del requisito

RF_021	Inclusión de Leapfrog y Velocity Verlet como integradores simbióticos	24/06/2025	Alta
Descripción:	El sistema debe contar con los integradores Leapfrog y Velocity Verlet, permitiendo al usuario seleccionar según las necesidades de conservación de energía.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Listado de integradores implementados.	Sistema	Opción de integración disponible.	Deben cumplir propiedades de estabilidad y conservación de energía.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El sistema proporciona ambos integradores. 2. El usuario lo selecciona desde el archivo de preprocesado. 3. El integrador se aplica dentro del modelo. 		
Post - condiciones	La simulación evoluciona temporalmente con un integrador fiable y eficiente.		

Apéndice D: Requisitos no funcionales.

Código	Nombre	Fecha	Prioridad del requisito
RNF_022	Validación numérica mediante soluciones analíticas	25/06/2025	Alta
Descripción:	El sistema debe permitir validar los resultados numéricos de la simulación comparándolos con soluciones teóricas analíticas (en geometrías con solución conocida) o datos de referencia de otros simuladores, garantizando la corrección física del comportamiento.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Resultados numéricos de la simulación; soluciones analíticas o datos de simulador de referencia.	Sistema / Usuario	Reporte de diferencias y métricas de error.	Las comparaciones deben realizarse sólo en geometrías con solución analítica válida.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario selecciona la geometría con solución analítica o carga datos de referencia. 2. El sistema calcula métricas de error (p.ej. RMSE, desviación máxima). 3. Genera un informe de validación numérica. 		
Post - condiciones	Se dispone de un informe cuantitativo que confirma o señala discrepancias en la precisión de la simulación.		
Código	Nombre	Fecha	Prioridad del requisito
RNF_024	Escalabilidad en partículas y tamaño de dominio	25/06/2025	Alta
Descripción:	El sistema debe estar diseñado para escalar eficientemente tanto en número de partículas como en tamaño del dominio, manteniendo un rendimiento aceptable y sin degradación excesiva al aumentar la escala de la simulación.		
Rol	Sistema / Usuario		
Entradas	Fuente	Salida	Restricciones
Parámetros de escala (número de partículas, dimensiones del dominio).	Usuario	Métricas de rendimiento (tiempo de cómputo, uso de memoria).	La degradación de rendimiento no debe exceder un factor lineal con respecto al aumento de escala.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario define una prueba de escala en el archivo de configuración. 2. El sistema ejecuta la simulación con diferentes tamaños. 3. Recopila tiempos y uso para cada caso. 		
Post - condiciones	Se dispone de datos que demuestran la capacidad del sistema para ejecutarse en escalas crecientes sin cuellos de botella.		
Código	Nombre	Fecha	Prioridad del requisito
RNF_026	Estadística de variables clave de simulación.	25/06/2025	Alta
Descripción:	El sistema debe proporcionar estadísticas descriptivas (media, desviación estándar, máximos, mínimos) de variables críticas como densidad, energía y velocidad, permitiendo verificar la coherencia física de los resultados.		
Rol	Usuario		
Entradas	Fuente	Salida	Restricciones
Resultados de simulación en memoria.	Sistema	Informe estadístico en formato legible (texto o JSON).	Debe calcularse sobre todo el dominio o en regiones definidas por el usuario.
Procesos / Pasos para ejecutar	<ol style="list-style-type: none"> 1. El usuario solicita el informe estadístico tras la simulación. 2. El sistema agrupa los datos y calcula las métricas. 3. Genera y guarda el informe. 		
Post - condiciones	El usuario cuenta con un resumen estadístico que facilita la interpretación y la detección de valores atípicos o errores.		