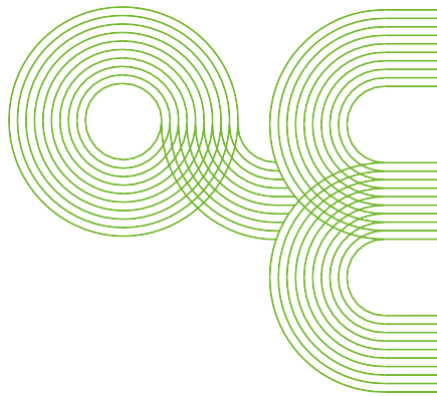


Universidade de Vigo

Memoria de traballo de fin de grao

Escola de Enxeñaría Aeronáutica e do Espazo



Autor: Jorge Ledo Rodríguez

Titulación: Enxeñaría Aeroespacial

Título do traballo: **Modelado físico y numérico de la interacción entre olas y objetos flotantes**

Titor: Alejandro Jacobo Cabrera Crespo

Co-titor: José Manuel Domínguez Alonso

Data: 15 de xuño de 2022

Agradecimientos

Me gustaría dar las gracias a todas las personas que han hecho posible este trabajo.

En especial, a mi tutor Álex por darme la oportunidad de realizar este trabajo con el grupo EPhysLab y ayudarme desde el primer día. Gracias por todos tus consejos y por hacer de esta experiencia algo agradable y entretenido.

Gracias también a José, mi cotutor, por resolver todas mis dudas durante este tiempo, dedicarme tantas horas y contribuir en este trabajo.

Por último, gracias a mis compañeros de laboratorio, Iván, Ángel, Bonaventura y Salvatore por la compañía durante estos meses y por toda la ayuda recibida.

Índice

1	Introducción	9
1.1	Caso de estudio	9
1.2	Modelado físico y numérico	10
1.3	Método SPH y aplicaciones en Ingeniería Aeroespacial	10
2	Objetivos	12
3	Teoría de las olas	13
3.1	Conceptos básicos	13
3.2	Clasificación de las olas	16
3.3	Generación de oleaje regular de segundo orden.....	20
4	Modelado físico: nanoCIEM	22
4.1	Dimensiones y características	22
4.2	Generación y absorción del oleaje	23
4.3	Diseño de la boya flotante	25
5	Modelado numérico: DualSPHysics.....	28
5.1	El método SPH.....	29
5.2	Código DualSPHysics	34
6	Validación	39
6.1	Validaciones previas en 2D y 3D	39
6.2	Propagación del oleaje regular.....	45
6.3	Test de decaimiento vertical	47
6.4	Objetos flotantes libres	51
6.5	Objetos flotantes con amarres.....	57
7	Conclusiones.....	64
	Referencias.....	65
	Apéndice.....	69

Lista de Figuras

Figura 3.1 – Onda sinusoidal.	14
Figura 3.2 – Ejemplo de superposición de olas en un estado de mar típico. Fuente: https://www.yumpu.com/user/comunidad.eduambiental.org	15
Figura 3.3 – Movimiento de las partículas de agua de las olas. Fuente: https://www.yumpu.com/user/comunidad.eduambiental.org	15
Figura 3.4 – Diagrama de aplicabilidad de las diferentes teorías de olas. Fuente: Le Méhauté (2013).	17
Figura 3.5 – Diferentes teorías de olas. Fuente: H. Karadeniz (2012).	18
Figura 3.6 – Comparación de las teorías de Airy, Stokes de 2º orden y ola solitaria (Raudkivi y Wittte, 1990).	20
Figura 4.1 – Imagen del canal de olas nanoCIEM en las instalaciones del grupo EPhysLab.	22
Figura 4.2 – Boya flotante utilizada en los ensayos físicos.	23
Figura 4.3 – Curva de capacidad máxima de generación de nanoCIEM.	24
Figura 4.4 – Software para el manejo del nanoCIEM.	24
Figura 4.5 – Plano y cotas de la boya.	25
Figura 4.6 – Vista previa del software Ultimaker Cura.	26
Figura 4.7 – Impresora 3D Anet ET4 X imprimiendo la boya flotante.	26
Figura 4.8 – Instantánea de Kinovea siguiendo la trayectoria de la boya.	27
Figura 5.1 – Esquema del dominio de influencia de la función kernel.	30
Figura 5.2 – Esquema de funcionamiento de las condiciones de contorno mDBC. Fuente: https://github.com/DualSPHysics/DualSPHysics/wiki/3.-SPH-formulation#38-boundary-conditions	33
Figura 5.3 – Principales instituciones académicas involucradas en el desarrollo de DualSPHysics. Fuente: https://dual.sphysics.org/	34
Figura 5.4 – Flujo de trabajo con DualSPHysics. Fuente: https://dual.sphysics.org/	35
Figura 5.5 – Esquema funcional de MoorDyn. Fuente: http://www.matt-hall.ca/moordyn.html	36
Figura 5.6 – Diagrama de flujo del acoplamiento de DualSPHysics con MoorDyn. Fuente: Domínguez et al. (2019).	37
Figura 5.7 – Discretización de la boya (diámetro=6.159 cm) diseñada utilizando diferentes resoluciones. Las partículas se representan con cubos en este caso para una mejor visualización.	38
Figura 6.1 – Esquema tanque experimental.	39
Figura 6.2 – Esquema tanque numérico 2D para generación y propagación de olas regulares.	40
Figura 6.3 – Series temporales teóricas y numéricas (2D) de la elevación de la superficie libre y velocidades orbitales para oleaje regular con $T=0.9$ s, $H=2.67$ cm, $d=13$ cm.	41
Figura 6.4 – Diferentes instantes de la simulación DualSPHysics para el oleaje regular con $T=0.9$ s, $H=2.67$ cm, $d=13$ cm. El color de las partículas corresponde a la velocidad longitudinal del fluido.	42
Figura 6.5 – Series temporales teóricas y numéricas (3D) de la elevación de la superficie libre y velocidades orbitales para oleaje regular con $T=0.9$ s, $H=2.67$ cm, $d=13$ cm.	44
Figura 6.6 – Condiciones de oleaje estudiadas representadas en el ábaco de Le Méhauté.	45
Figura 6.7 – Series temporales experimentales, numéricas y teóricas de la elevación de la superficie libre para las tres condiciones de oleaje regular: A) $H=4.22$ cm y $T=0.7$ s, B) $H=3.11$ cm y $T=0.9$ s, C) $H=2.67$ cm y $T=0.9$ s.	46
Figura 6.8 – Esquema de tanque numérico para el test de decaimiento vertical.	47

Figura 6.9 – Velocidad de las partículas fluidas durante el test de decaimiento.....	48
Figura 6.10 – Comparación del desplazamiento vertical teórico, numérico y experimental durante el test de decaimiento.....	49
Figura 6.11 – Esquema del tanque numérico para test con la boya flotante (en amarillo).....	51
Figura 6.12 – Diferentes instantes de la simulación con DualSPHysics para la configuración de ola C (T=0.9 s, H=2.67 cm, d=13 cm). El color de las partículas corresponde a la velocidad longitudinal del fluido y la boya se muestra en amarillo. La línea negra marca la posición de la boya en el instante t=8.10 s.	52
Figura 6.13 – Movimiento longitudinal (x) y vertical (z) de la boya usando la configuración de la ola A (T=0.7 s, H=4.22 cm).....	53
Figura 6.14 – Movimiento longitudinal y vertical de la boya usando la configuración de la ola B (T=0.9 s, H=3.11 cm).....	54
Figura 6.15 – Movimiento longitudinal y vertical de la boya usando la configuración de la ola C (T=0.9 s, H=2.67 cm).....	55
Figura 6.16 – Esquema del tanque numérico para test con la boya flotante amarrada al fondo.	57
Figura 6.17 – Esquema en detalle de la configuración de amarre en el experimento físico.....	58
Figura 6.18 – Diferentes instantes de la simulación con DualSPHysics de una boya amarrada al fondo para la configuración de ola C (T=0.9 s, H=2.67 cm). La boya se muestra en amarillo y el amarre en negro, mientras que el color del fluido indica su velocidad longitudinal. La línea negra marca la posición de la boya en el instante t=8.36 s.	59
Figura 6.19 – Movimiento longitudinal y vertical de la boya amarrada usando la configuración de la ola A (T=0.7 s, H=4.22 cm).....	60
Figura 6.20 – Movimiento longitudinal y vertical de la boya amarrada usando la configuración de la ola B (T=0.9 s, H=3.11 cm).	61
Figura 6.21 – Movimiento longitudinal y vertical de la boya amarrada usando la configuración de la ola C (T=0.9 s, H=2.67 cm).	62
Figura 6.22 – Tensiones medidas en la línea de amarre para las tres condiciones de oleaje. ...	63

Lista de Tablas

Tabla 3.1. Clasificación de olas según su periodo por Toffoli & Bitner-Gregersen (2015).....	13
Tabla 6.1 – Número de partículas y tiempo de ejecución según la resolución utilizada.	43
Tabla 6.2 – Condiciones de oleaje estudiadas.	45
Tabla 6.3 – Valores de período amortiguado mediante distintos métodos.	50
Tabla 6.4 – Propiedades de la línea de amarre utilizadas en el modelado numérico.	58

1 Introducción

Este Trabajo de Fin de Grado (TFG) se ha realizado con la colaboración y la supervisión de investigadores del grupo EPhysLab del Campus de Ourense, en donde se ha estudiado la interacción entre oleaje y dispositivos flotantes utilizando tanto el modelado físico (experimentos) como el modelado numérico (simulaciones).

Este documento está organizado de la siguiente manera: en este capítulo 1 se presenta el caso de estudio, y se introducen las principales características del modelado físico y numérico, el método SPH y sus aplicaciones en el campo de la Ingeniería Aeroespacial; el capítulo 2 enumera los objetivos principales de este TFG; el capítulo 3 explica los conceptos básicos de las olas, su clasificación y las principales teorías de generación de oleaje regular; el capítulo 4 se centra en el modelado físico, define el canal de oleaje nanoCIEM y explica el proceso de fabricación del objeto flotante utilizado; el capítulo 5 describe el modelado numérico, la metodología de SPH y el código DualSPHysics; el capítulo 6 presenta la validación de los resultados comparando los datos numéricos obtenidos en DualSPHysics con los resultados experimentales extraídos de nanoCIEM; y por último, el capítulo 7 presenta las conclusiones obtenidas.

1.1 Caso de estudio

El caso de estudio en este trabajo se centra en el análisis de la interacción entre olas y objetos flotantes. Se realizan varios experimentos en el canal de oleaje nanoCIEM y simulaciones numéricas gracias al software DualSPHysics, que ha sido desarrollado por el grupo EPhysLab.

A continuación, se resumen los diferentes ensayos realizados:

- i) En primer lugar, se va estudiar la generación, propagación y absorción de oleaje sin ningún objeto con el que interaccionar, lo que permite comprobar que la ola se genera y se propaga por el tanque de manera correcta y que el tanque no produce reflexión.
- ii) Después se realizará un caso de decaimiento, que consiste en desplazar verticalmente desde el punto de equilibrio un objeto flotante y medir su frecuencia natural de oscilación.
- iii) El siguiente paso es meter ese objeto flotante libre en el tanque de olas, estudiar su interacción con el oleaje regular y analizar sus movimientos longitudinales y verticales.
- iv) Y, por último, añadir a ese mismo objeto una línea de amarre conectada al fondo del tanque y estudiar cómo se restringen sus movimientos respecto al caso anterior.

1.2 Modelado físico y numérico

El modelado físico es una representación a escala, es decir, una maqueta del experimento real. Los ensayos en canales físicos suelen ser costosos tanto en coste como en tiempo y además hay que considerar los efectos de escala cuando se ensayan problemas a una escala diferente a la de la realidad. Por otro lado, el modelado numérico aplicado a la dinámica de fluidos, también llamado CFD (del inglés *Computational Fluid Dynamics*), es el resultado de utilizar métodos numéricos para resolver las soluciones de las ecuaciones que gobiernan el movimiento de un fluido. Durante las últimas décadas, el modelado numérico ha tenido una gran influencia en todos los campos de la ciencia. El gran desarrollo que han experimentado los equipos de cálculo en los últimos años permite a los científicos e ingenieros realizar simulaciones complejas con gran precisión y en tiempos razonables. Además, el modelado numérico puede proporcionar información adicional que no es posible obtener con la observación experimental. Sin embargo, el modelado numérico no sustituye al modelado físico, ya que éste sigue siendo usado como solución de referencia para confirmar que los resultados numéricos obtenidos son válidos. Los principales inconvenientes de los experimentos físicos, como ya se mencionó, son su elevado coste económico, la gran cantidad de tiempo necesario para su construcción y la necesidad de unas instalaciones adecuadas para el experimento ya que cada uno tiene sus peculiaridades. Una de las funciones más destacadas del modelado numérico, tras su validación, es reducir el número de pruebas experimentales, ya que algunos diseños pueden descartarse directamente con los resultados numéricos. Debido a esto, el modelado numérico permite reducir el número de ensayos físicos y ahorrar tiempo y dinero, por lo tanto, el modelado numérico complementa al modelado físico.

1.3 Método SPH y aplicaciones en Ingeniería Aeroespacial

Algunos de los métodos numéricos para dinámica de fluidos más populares y utilizados están basados en mallas, como son los métodos de volúmenes finitos. Sin embargo, y sólo en algunos casos, estos métodos tienen dificultades a la hora de modelar geometrías complejas en movimiento y casos con grandes deformaciones de la superficie libre. Un ejemplo de estos casos son las olas del mar impactando con objetos flotantes y en donde el flujo de agua se separa y se reconstruye. Las olas pueden romper, rebasar el objeto, etc. El caso de estudio de este trabajo consiste en modelar la interacción entre olas y objetos flotantes libres y amarrados, por lo que para el modelado numérico se utilizará un método libre de malla, en este caso un método de partículas llamado SPH (del inglés *Smoothed Particle Hydrodynamics*).

El método SPH es un modelo numérico Lagrangiano que discretiza el dominio del caso a estudiar en un conjunto de partículas. Las propiedades físicas de cada partícula se obtienen resolviendo las ecuaciones de Navier-Stokes para cada par de interacción entre partículas e interpolando las propiedades de las partículas circundantes dentro de un dominio de influencia dado por una función de suavizado llamada *kernel*.

El método SPH tiene aplicaciones en muchos campos de la ingeniería civil, mecánica, costera y también hay algunos problemas que tienen una mejor solución usando SPH dentro de la ingeniería aeroespacial.

A continuación, se incluyen algunos ejemplos de estudios utilizando modelos SPH para aplicaciones aeroespaciales:

- Simulación del aterrizaje del módulo de tripulación de la nave espacial Orion en una masa de agua: El trabajo presentado por Packard (2013) compara los resultados numéricos obtenidos con SPH con datos experimentales proporcionados por el NESC (Centro de Seguridad e Ingeniería de la NASA), que realizó 41 pruebas de caída del módulo de tripulación impactando con una masa de agua con distintas condiciones iniciales.
- Simulación del flujo de agua del sistema de boquillas *rainbird* utilizado para minimizar el sonido durante el lanzamiento de un cohete. Las simulaciones usando SPH presentadas en Vu et al. (2015) ayudan a determinar si el agua de las *rainbird* incidirá en las boquillas del cohete.
- Análisis numérico de la hidrodinámica de un submarino conceptual diseñado por la NASA para la navegación en la superficie libre de los mares de hidrocarburos líquidos de Titán, la mayor luna de Saturno. El trabajo de Mogan et al. (2018) compara resultados de SPH con los de métodos de volúmenes finitos.
- Simulación de un aterrizaje de emergencia de un avión en el agua. La interacción entre la estructura y el fluido se resuelve en Groenenboom et al. (2014) acoplado el método de los elementos finitos para la estructura con el método SPH para el agua.
- Simulación del flujo de agua en tanques parcialmente llenos que se encuentran en movimiento. Usando datos experimentales proporcionados por la ESA (*European Space Agency*), Green et al. (2021) compararon los resultados de sus simulaciones SPH con los datos de los ensayos físicos.
- Simulación del aterrizaje de un helicóptero en el agua con dispositivos de flotación de emergencia. El trabajo de Cartwright et al. (2010) estudió la interacción entre el helicóptero y el agua usando también un acoplamiento con SPH.

Otras aplicaciones interesantes relacionadas con la industria de la automoción o la ingeniería marítima son las siguientes:

- Estudio del patrón de olas generado por un buque con un movimiento estacionario, centrándose en la rotura de la ola en la proa presentado por Marrone et al. (2012).
- Análisis del rendimiento del casco de un barco en modo planeo. El trabajo de Tagliaferro et al. (2021) incluía además comparación con datos experimentales y con otros modelos.
- Simulación del flujo de combustible en el tanque de un vehículo de competición. Longshaw et al. (2015) presentaba las ventajas de SPH para este tipo de aplicaciones en donde hay que simular flujos muy violentos.

Por último, el artículo de Shabdloo et al. (2016) hace un resumen de las principales aplicaciones del método SPH, entre las que se encuentran la ingeniería aeroespacial, automoción, energías renovables, procesos industriales, ingeniería de costas, astrofísica, bioingeniería, flujos compresibles, etc.

2 Objetivos

El objetivo principal de este Trabajo de Fin de Grado es demostrar la capacidad que tiene el software DualSPHysics para simular la propagación de olas regulares y su interacción con estructuras flotantes libres y amarradas al fondo en un entorno cercano a la costa. Para ello será necesario adquirir nuevas habilidades e ir cumpliendo distintos objetivos parciales:

- Descubrir un nuevo software CFD Lagrangiano sin malla, ya que en este caso tiene varias ventajas a la hora de simular interacciones de olas con objetos flotantes respecto a los softwares CFD con malla utilizados a lo largo del grado.
- Comprender los fundamentos en los que se basa el método SPH y su utilidad para determinadas aplicaciones de ingeniería.
- Obtener los conocimientos básicos sobre la generación y propagación de olas, así como entender las distintas teorías existentes que describen este fenómeno.
- Aprendizaje en el manejo del software DualSPHysics y obtener los conocimientos necesarios para crear distintos casos de simulación mediante un archivo XML.
- Ejecutar simulaciones en CPU y GPU (la segunda opción reduce el tiempo de coste computacional de forma significativa).
- Analizar los resultados numéricos con la ayuda de las herramientas de postprocesado y crear imágenes y videos para visualizar los resultados obtenidos.
- Aprender a realizar ensayos físicos en el canal de oleaje nanoCIEM, que será de donde se obtendrán los datos experimentales.
- Realizar una validación comparando los resultados experimentales con los de SPH y razonar si los resultados son aceptables o no y encontrar los posibles problemas y/o soluciones.

3 Teoría de las olas

En este capítulo se explican los distintos tipos de olas, su formación, los parámetros que describen una ola, la dinámica del movimiento de las partículas de agua y los fenómenos que afectan a las olas. A continuación, se describen distintas teorías para el estudio analítico de oleaje regular y el rango de aplicación de cada teoría con la ayuda del ábaco de Le Méhauté. Por último, se desarrolla la teoría de generación de oleaje regular de primer y segundo orden para un generador de olas de tipo pistón.

3.1 Conceptos básicos

Una ola es una oscilación periódica que tiene lugar en la superficie del agua de océanos, mares, ríos y otras masas de agua. Los océanos ocupan la mayor parte de la superficie de nuestro planeta (entorno al 70%) por lo que las olas que se producen en ellos son fenómenos de gran importancia. Las olas en el océano se originan por diferentes procesos. Para que se forme una ola en una superficie de agua inicialmente en reposo, se necesita una perturbación inicial que estará afectada por una fuerza restauradora opuesta a dicha perturbación. Entre las fuerzas generadoras se encuentran el viento, las fuerzas gravitacionales del Sol y de la Luna, los terremotos, las tormentas y los gradientes de presión atmosférica. Y entre las fuerzas restauradoras están la gravedad, la tensión superficial y la fuerza de Coriolis.

Según Toffoli & Bitner-Gregersen (2015) podemos clasificar las olas en función de su período tal y como se puede apreciar en la Tabla 3.1. En ella se puede ver el tipo de olas, el intervalo de períodos al que pertenecen, las fuerzas que las generan y sus fuerzas restauradoras. En esta tabla, el gradiente de presión atmosférica se denota como Δp_{atm} , la tensión superficial como γ , la fuerza de la gravedad como F_g y la fuerza de Coriolis como F_c .

Tabla 3.1. Clasificación de olas según su periodo por Toffoli & Bitner-Gregersen (2015).

Tipo de olas	Intervalo de periodos	Fuerza generadora	Fuerza restauradora
Capilares	< 0.1 s	Viento	γ
De ultragravedad	0.1 - 1 s	Viento	F_g, γ
De gravedad	1 - 20 s	Viento	F_g
De infragravedad	20 s - 5 min	Viento, Δp_{atm}	F_g
De período largo	5 min - 12 h	Terremotos, Δp_{atm}	F_g
De mareas	12 - 24 h	Gravitacional	F_g, F_c
Transmareas	> 24 h	Gravitacional, tormentas	F_g, F_c

Como se puede ver en la Tabla 3.1, el viento es la principal fuerza generadora de las olas de periodos cortos. Y como el viento se origina debido al diferente calentamiento de la superficie terrestre por el Sol, la energía de olas también tiene su origen en el Sol. La formación de las olas depende principalmente de los siguientes factores:

- La velocidad del viento
- La duración del viento
- La superficie de agua afectada por el viento
- La profundidad de la superficie de agua

Cuando hay viento en la superficie del mar, las moléculas de aire interactúan con las moléculas de agua de la superficie libre con las que están en contacto. Debido a esto se modifica la superficie del océano y se forman olas de capilaridad. Estas olas provocan una mayor superficie de contacto, por lo que se incrementa el efecto de interacción entre el agua y el viento. Esto provoca una mayor superficie sobre la que el viento puede ejercer presión y causa un nuevo aumento de tamaño de la ola, formando olas de ultragravedad y de gravedad.

Una ola regular se puede describir como una onda sinusoidal (Figura 3.1) con una longitud de onda, λ , una amplitud, A , y un período, T .

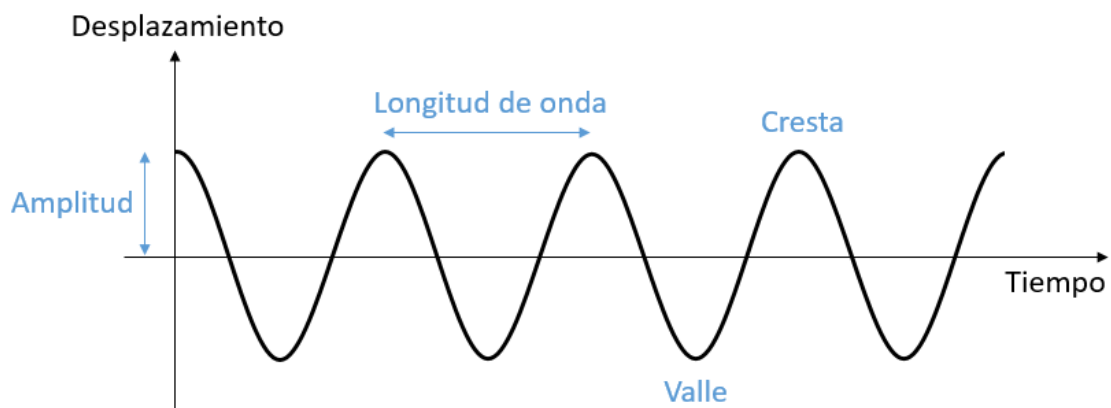


Figura 3.1 – Onda sinusoidal.

La cresta es la elevación más alta que puede alcanzar una onda y el valle es la elevación más baja, es decir, la amplitud máxima y mínima respectivamente. La longitud de onda es la distancia entre dos crestas o valles consecutivos. El período es el tiempo en que una cresta o valle recorre una longitud de onda. La frecuencia, f , es el número de oscilaciones de cresta a cresta o de valle a valle por segundo, es decir, la inversa del período. La diferencia de altura entre una cresta y un valle se denomina altura de ola, H . La velocidad de propagación, c , es el cociente de la longitud de onda (λ) y el período (T).

Sin embargo, las olas del océano no son olas sinusoidales puras como la de la figura anterior. El océano se compone de una superposición de olas sinusoidales de amplitud y período variable (Figura 3.2), en las que cada ola tendrá sus características propias, su velocidad de propagación, su dirección, su período y su altura de ola. La envolvente de todas esas olas tiene una velocidad distinta a las olas sinusoidales puras, a esta velocidad se le denomina velocidad de grupo, c_g .

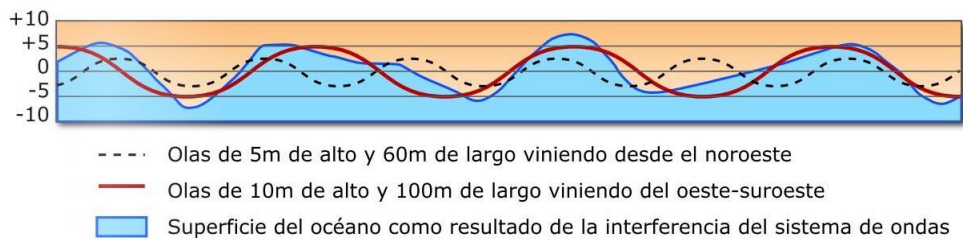


Figura 3.2 – Ejemplo de superposición de olas en un estado de mar típico. Fuente: <https://www.yumpu.com/user/comunidad.eduambiental.org>.

Como se ha dicho anteriormente, la superficie oceánica es enorme y muy desigual entre distintos puntos del planeta. Hay zonas más o menos profundas que otras. Hay barreras de coral que hacen que la profundidad en esa zona sea menor de lo esperado en un principio, como la Gran Barrera de Coral en Australia, y también existen depresiones del fondo marino que hacen que la profundidad en una pequeña zona sea mucho mayor que a su alrededor, como la Fosa de las Marianas en el Pacífico occidental que es el lugar más profundo del planeta, con una profundidad de 10994 m. Sin embargo, el océano se puede dividir en tres zonas bien diferenciadas entre sí:

- Aguas profundas.
- Aguas intermedias.
- Aguas someras.

En las aguas profundas, la profundidad es mayor que la mitad de la longitud de onda, λ . La principal característica de las aguas profundas es que el fondo marino tiene un efecto nulo sobre el oleaje. Las aguas intermedias abarcan desde una profundidad de $\lambda/2$ hasta $\lambda/4$. En estas aguas el fondo marino empieza a tener influencia en el perfil de la superficie libre. Las aguas someras son aquella cuya profundidad es inferior a un cuarto de la longitud de onda, λ . En las aguas someras las propiedades de la ola están influenciadas por el fondo marino.

A pesar de la opinión generalizada, una ola no representa el movimiento de un flujo de agua a través del océano, sino un movimiento de energía desde el origen de la ola hasta la rotura de la misma, que puede ocurrir en la costa o en el medio del océano. Las olas mueven la superficie del agua arriba y abajo y no en el sentido de propagación del oleaje.

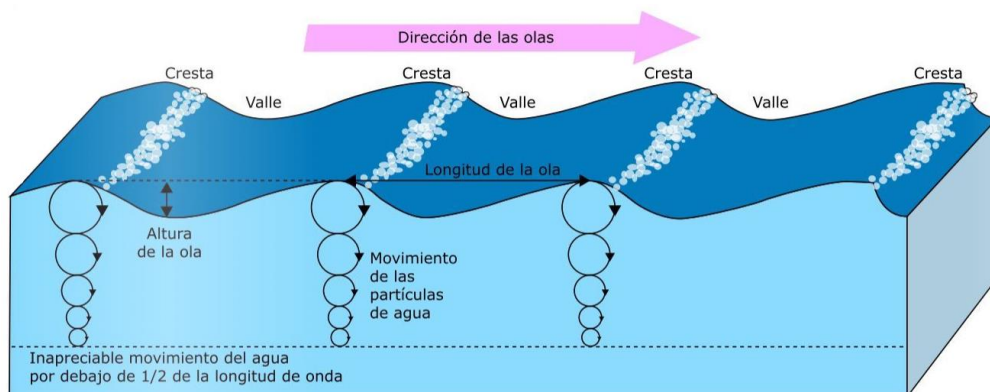


Figura 3.3 – Movimiento de las partículas de agua de las olas. Fuente: <https://www.yumpu.com/user/comunidad.eduambiental.org>.

Como se puede observar en la Figura 3.3, las olas en aguas profundas están formadas por partículas de agua que se mueven de forma circular. En las crestas, las partículas de agua se mueven en la dirección de la ola, mientras que en los valles las partículas de agua se mueven en la dirección contraria. En la superficie, el movimiento circular es del mismo tamaño que la altura de ola y disminuye exponencialmente con la profundidad, hasta llegar a una profundidad en la que el movimiento de partículas es despreciable. Cuando la ola se aproxima a la costa, y por lo tanto se encuentra en aguas someras, el movimiento de las partículas de agua pasa a ser elipsoidal y prevalece el movimiento horizontal. Además, la velocidad de avance y la longitud de onda disminuyen y la altura de ola aumenta hasta que la velocidad de las partículas de agua supera la velocidad de avance de la ola. En ese momento, la ola se vuelve inestable y rompe, esto ocurre cuando el cociente de la altura de ola, H , y la profundidad, d , es igual a 0.78, según el criterio de McCowan (1891), del que se hablará más adelante.

Otro fenómeno que puede ocurrir en lugar de que las olas rompan al llegar a la costa es la reflexión. La reflexión se produce cuando en la trayectoria de una ola hay un obstáculo que puede reflejar la energía de la ola. La cantidad de energía reflejada depende de la geometría del obstáculo. Si se trata de una estructura plana perpendicular a la dirección de propagación de la ola, la reflexión será muy importante y puede dar lugar a olas reflejadas que en combinación con las incidentes producen olas de gran altura. En cambio, una playa con una pendiente pequeña produce poca reflexión, ya que la mayor parte de la energía se disipará con la rotura de la ola. El coeficiente de reflexión, K_R , se define como el cociente de la altura de ola reflejada, H_r , y la altura de ola incidente, H_i .

$$K_R = \frac{H_r}{H_i} \quad (3.1)$$

Este coeficiente es muy importante cuando se generan olas en un dominio reducido porque muestra si las olas generadas se propagan correctamente sin tener influencia de las olas reflejadas por los límites del dominio.

3.2 Clasificación de las olas

La rotura de ola es una situación en la que hay una gran cantidad de turbulencia libre debido a una entrada de aire, lo que provoca una disipación de energía. Según Le Méhauté (2013), la rotura de una ola se produce cuando tiene lugar alguna de las siguientes circunstancias:

- La velocidad de las partículas de la cresta de la ola es mayor que la velocidad de la onda.
- La presión de la superficie libre de agua según la ecuación de Bernoulli no es compatible con la presión atmosférica.
- La aceleración de las partículas en la cresta de la ola tiene una tendencia a separarlas de la superficie del agua.
- La superficie libre se vuelve vertical.

El diagrama de Le Méhauté (Figura 3.4) es un instrumento de gran utilidad para determinar la teoría de olas más conveniente según ciertas condiciones de oleaje. También indica los límites de validez de cada teoría dependiendo del período, la profundidad y la altura de ola. Para representar una ola sobre este diagrama lo primero es conocer las características mencionadas anteriormente (H , T y d). Después hay que calcular su valor tanto en el eje x como en el eje y , que son dos números adimensionales. El eje x depende de la profundidad y del período y el eje y depende del período y de la altura de ola.

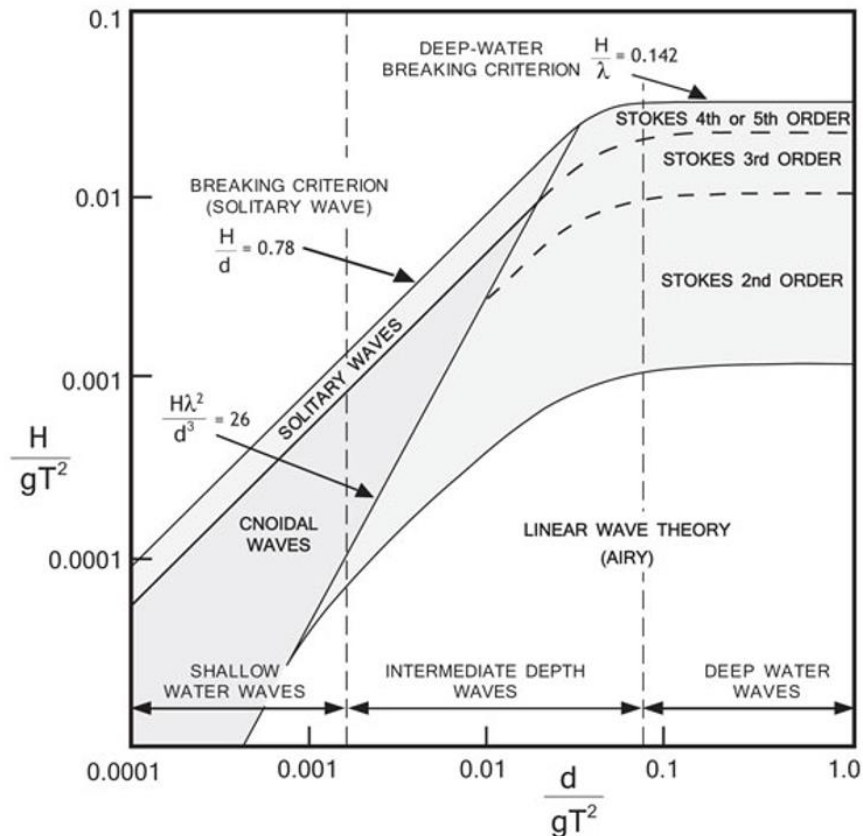


Figura 3.4 – Diagrama de aplicabilidad de las diferentes teorías de olas. Fuente: Le Méhauté (2013).

Como se puede ver en la Figura 3.4 para olas en aguas someras o poco profundas, en la mayor parte de los casos se usará la teoría Conoidal. En aguas intermedias, dependiendo de la altura de ola y del período de cada ola se utilizará mayoritariamente la teoría lineal de Airy, la teoría de Stokes de 2º orden o de orden superiores y en menor medida la teoría Cnoidal. En aguas profundas, dependiendo de los parámetros de cada ola se utilizará la teoría lineal de Airy o la teoría de Stokes de 2º orden o de orden superiores.

En el diagrama de Le Méhauté podemos ver tres números adimensionales que nos indican varios datos de interés. El número de Ursell, $Ur = \frac{H \times \lambda^2}{d^3}$, delimita la zona de aplicación de la teoría Conoidal, de la teoría lineal de Airy y de la teoría de Stokes. Cuando su valor es mayor que 26 la ola deberá ser estudiada por la teoría lineal o a la teoría de Stokes. Otro de los números adimensionales que aparece en el diagrama es el criterio de rotura para olas solitarias que fue desarrollado por McCowan (1891) y se calcula dividiendo la altura de ola entre la profundidad.

Cuando su valor es mayor que 0.78 la ola se encuentra en la zona de rotura. Y el último número adimensional es el criterio de rotura en aguas profundas que fue desarrollado por Miche (1944) y se calcula dividiendo la altura de ola entre la longitud de onda. Cuando su valor es mayor que 0.142 la ola se encuentra en la zona de rotura. Por lo tanto, la zona de rotura, que se encuentra principalmente en la parte superior izquierda del diagrama, queda delimitada por el criterio de rotura de ola solitaria y el criterio de rotura de aguas profundas. En la Figura 3.5 se puede ver una comparación de la superficie libre de agua para distintas teorías de olas.

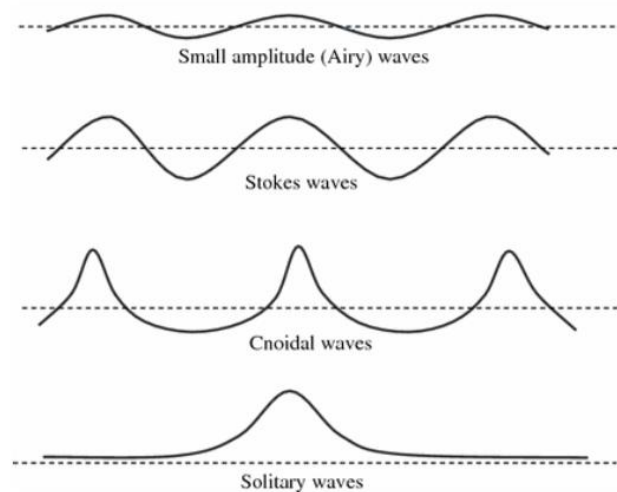


Figura 3.5 – Diferentes teorías de olas. Fuente: H. Karadeniz (2012).

El diagrama de Le Méhauté muestra hasta siete teorías diferentes para estudiar el oleaje regular. A continuación, se explicarán brevemente todas las teorías, haciendo especial hincapié en la teoría lineal de Airy y la teoría de Stokes de segundo orden que se usaron en este trabajo.

La teoría de oleaje más sencilla es la teoría lineal también llamada teoría de Stokes de primer orden. Según esta teoría, la formación del oleaje comienza en la superficie libre de agua por una fuerza externa que solo actúa como fuerza perturbadora y que deja de actuar cuando el oleaje se ha formado. Airy considera que el movimiento de las partículas de agua sigue una órbita cerrada, por lo que no hay un transporte de masa de agua en la dirección de propagación del oleaje. Para que esta teoría sea válida se tienen que cumplir las siguientes hipótesis:

- La ola es regular y bidimensional.
- El fluido es incompresible, es decir, la densidad es constante.
- El fluido es irrotacional y no viscoso, por lo tanto, se considera un fluido ideal y como consecuencia, existe un potencial de velocidad.
- La tensión superficial y la turbulencia son despreciables.
- El fondo marino es profundo e impermeable.
- La altura de ola es pequeña en comparación con la profundidad y con la longitud de onda.

Considerando un potencial de velocidad $\phi(x, z, t)$, donde x es la posición horizontal, z es la posición vertical y t es tiempo, la teoría lineal define la componente horizontal (u) y vertical (w) de la velocidad como

$$u(x, z, t) = -\frac{\partial \phi}{\partial x}, \quad (3.2)$$

$$w(x, z, t) = -\frac{\partial \phi}{\partial z} \quad (3.3)$$

y Bernoulli para flujos irrotacionales como

$$\frac{\partial \phi}{\partial t} + \frac{P}{\rho_{agua}} + gz = 0. \quad (3.4)$$

siendo P la presión, ρ la densidad y g la gravedad.

El movimiento de las olas está definido por la ecuación de Laplace, que es una ecuación diferencial de segundo orden con la siguiente forma

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial z^2} = 0. \quad (3.5)$$

Como la altura de ola es pequeña en relación a la longitud de onda, se puede relacionar la variación temporal de la superficie libre (η) con la componente vertical de la velocidad:

$$\frac{\partial \eta}{\partial t} = -\frac{\partial \phi}{\partial z}. \quad (3.6)$$

Las condiciones de contorno en la superficie libre, es decir, $z = 0$ son las siguientes: la presión atmosférica es cero, por lo tanto

$$\frac{\partial \phi}{\partial t} + gz = 0 \quad (3.7)$$

y en la superficie libre, la velocidad solo tiene componente vertical definida por

$$w = \frac{\partial \eta}{\partial t} + u \frac{\partial \eta}{\partial x}. \quad (3.8)$$

La condición de contorno en el fondo marino, $z = -h$, siendo h la profundidad, establece que la velocidad solo tiene componente horizontal, $w = 0$ siendo

$$-\frac{\partial \phi}{\partial z} = 0. \quad (3.9)$$

Según Airy la superficie libre tiene la misma fórmula tanto para aguas someras como para aguas intermedias y aguas profundas, definida como

$$\eta(x, t) = \frac{H}{2} \cos\left(\frac{2\pi x}{\lambda} - \frac{2\pi t}{T}\right). \quad (3.10)$$

La teoría de Stokes (1880) de segundo orden describe el oleaje de forma no lineal, añadiendo un término a la teoría de Airy. El efecto de este término es aumentar la amplitud de las crestas y disminuir la amplitud de los valles. Esta teoría supone un gran avance respecto a la teoría lineal, ya que, a pesar de ser muy parecidas en aguas profundas, en aguas someras e intermedias sí hay grandes diferencias entre las dos teorías. Estas diferencias ocurren debido a la limitación de la teoría de Airy, por la cual la altura de ola (H) tiene que ser pequeña en comparación con la longitud de onda (λ) y con la profundidad. Esta hipótesis presenta errores cuando la ola se encuentra en aguas someras e intermedias, debido a que el efecto del fondo marino hace que la altura de ola aumente cuando la profundidad disminuye. Según la teoría de Stokes, el perfil de la superficie libre tiene crestas más altas y puntiagudas y valles más planos que la teoría de Airy como se puede ver en la Figura 3.6.

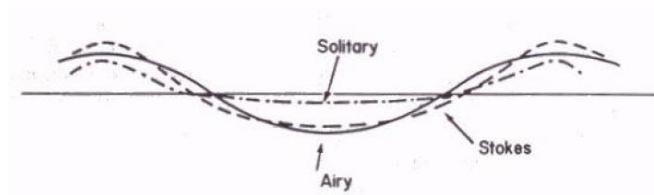


Figura 3.6 – Comparación de las teorías de Airy, Stokes de 2º orden y ola solitaria (Raudkivi y Wittte, 1990).

Además del distinto perfil de la superficie libre de agua, la otra principal diferencia entre la teoría de Airy y la de Stokes tiene que ver con las velocidades orbitales. Según Stokes las partículas de agua no tienen una trayectoria cerrada, debido a esto, se da un transporte de masa de agua en la dirección de propagación del oleaje, conocido como deriva de Stokes.

La expresión de la superficie libre de Stokes de segundo orden se aproxima mucho más a la forma real que tienen las olas en aguas intermedias y someras:

$$\eta(x, t) = \frac{H}{2} \cos(kx - wt) + \frac{\pi H^2}{2\lambda} \frac{\cosh(kd)[2 + \cosh(2kd)]}{(\sinh(kd))^3} \cos(kx - wt) . \quad (3.11)$$

donde k es el número de onda definido como $2\pi/\lambda$ y d es la profundidad.

Las teorías de Stokes de tercer, cuarto y quinto orden añaden términos extra correspondientes a cada teoría. En comparación con las olas de Stokes de segundo orden, las olas de Stokes de quinto orden tienen una mayor amplitud en las crestas y una menor amplitud en los valles.

La teoría Conoidal fue empezada por Boussinesq (1872) y Korteweg y De Vries (1895) y tiene contribuciones posteriores de Keulegan y Patterson (1940) y Keller (1948). El nombre Conoidal es debido a que el perfil de la superficie libre está descrito por la función coseno elíptico Jacobiano, cuya abreviatura es cn . Esta teoría tiene el rango de aplicación más grande, pero debido a su complejidad matemática su uso está muy limitado.

La teoría de ola solitaria representa una onda con solo una cresta que se propaga en agua en reposo, por lo que la onda no tiene valles y por lo tanto no es una onda oscilatoria como las descritas anteriormente en este apartado. Como la onda no oscila, no tiene ni período ni longitud de onda. Esta teoría es útil cuando las olas se encuentran en aguas someras, ya que se incrementa la altura de ola, haciendo que las crestas sean más grandes y los valles más planos. Bagnold (1974) y Munk (1949) recomiendan el uso de esta teoría en aguas someras cuando la ola está cerca de la rotura.

3.3 Generación de oleaje regular de segundo orden

En primer lugar, se explicará brevemente la generación de oleaje regular de primer orden, debido a que el resultado de primer orden es uno de los dos términos que componen la generación de oleaje regular de segundo orden.

La función de transferencia de Biesel expresa la relación entre la amplitud de onda y el desplazamiento del generador de oleaje (Biesel y Suquet, 1951) suponiendo que el fluido es irrotacional e incompresible y que la presión es constante en la superficie libre. La función de transferencia relaciona el desplazamiento del generador de oleaje tipo pistón con la elevación

de la superficie del agua, bajo la hipótesis de ondas monocromáticas sinusoidales unidimensionales en la dirección X :

$$\eta(x, t) = \frac{H}{2} \cos(\omega t - kx + \delta) . \quad (3.12)$$

donde, recordemos que, H es la altura de ola, T es el período, λ la longitud de onda, d la profundidad de agua, x la distancia al generador y δ la fase inicial de la onda, que viene dada por un número aleatorio entre 0 y 2π . La velocidad angular ω viene dada por $\omega = \frac{2\pi}{T}$ y el número de onda $k = \frac{2\pi}{\lambda}$.

La ecuación anterior expresa la elevación de la superficie en el infinito que Biesel definió como la solución de campo lejano. La función de Biesel puede derivarse para la solución de campo lejano y para un generador de ondas tipo pistón como

$$\frac{H}{S_0} = \frac{2\sinh^2(kd)}{\sinh(kd) \cosh(kd) + kd} . \quad (3.13)$$

donde S_0 es la carrera del pistón o *stroke*, que es igual a dos veces la amplitud. Una vez definida la carrera del pistón, la serie temporal del movimiento del pistón (e_1) viene dada por

$$e_1(t) = \frac{S_0}{2} \sin(\omega t + \delta) . \quad (3.14)$$

La implementación de una teoría de generación de ondas de segundo orden evitará la generación de ondas secundarias falsas. La teoría propuesta por Madsen (1971) es sencilla y precisa para ondas de primer y segundo orden.

La carrera del pistón S_0 se puede definir como $S_0 = \frac{H}{m_1}$ donde m_1 es

$$m_1 = \frac{2\sinh^2(kd)}{\sinh(kd) \cosh(kd) + kd} . \quad (3.15)$$

Según Madsen (1971) para generar una onda de segundo orden hay que añadir un término adicional al resultado de primer orden. Para un generador de ondas tipo pistón, este término se define como

$$e_2(t) = \left[\left(\frac{H^2}{32d} \right) \left(\frac{3\cosh(kd)}{\sinh^3(kd)} - \frac{2}{m_1} \right) \right] \sin(2\omega t + 2\delta) . \quad (3.16)$$

Por lo tanto, la serie temporal del desplazamiento del pistón para olas regulares de segundo orden es la suma del término $e_1(t)$ y del término $e_2(t)$:

$$e(t) = \frac{S_0}{2} \sin(\omega t + \delta) + \left[\left(\frac{H^2}{32d} \right) \left(\frac{3\cosh(kd)}{\sinh^3(kd)} - \frac{2}{m_1} \right) \right] \sin(2\omega t + 2\delta) . \quad (3.17)$$

Sin embargo, Madsen limitó la aplicación de su teoría de segundo orden a olas que cumplan la siguiente condición

$$\frac{H\lambda^2}{d^3} < \frac{8\pi^2}{3} . \quad (3.18)$$

4 Modelado físico: nanoCIEM

Este capítulo incluye información de las dimensiones y características del tanque físico empleado para realizar los experimentos. Además, se explica la generación de oleaje en el tanque y la zona de disipación, cuyo objetivo es disminuir la influencia de las olas reflejadas por la pared final del tanque. También se hace hincapié en las dos magnitudes de interés que es capaz de proporcionar el tanque: el movimiento del pistón y la elevación de la superficie libre. Por último, se describe la boya utilizada, su diseño y todas las herramientas utilizadas en su proceso de fabricación.

4.1 Dimensiones y características

Los ensayos físicos o experimentos se han llevado a cabo en el canal de oleaje de muy pequeña escala llamado nanoCIEM (Figura 4.1). Este canal fue construido por el Laboratorio de Ingeniería Marítima (LIM) de la Universidad Politécnica de Cataluña (UPC). El nanoCIEM tiene una longitud total de 200 cm, con una sección útil de 20 cm de ancho y 35 cm de alto. Las paredes del tanque son de metacrilato transparente y tienen un espesor de 1 cm. Para el correcto funcionamiento del tanque, la mínima y la máxima profundidad de agua permitidas son 12 y 15 cm respectivamente. El tanque tiene un orificio de 2 cm de diámetro en ambos extremos para su llenado y vaciado. Para su funcionamiento el nanoCIEM necesita una alimentación de corriente continua a 12 V y una conexión a un ordenador portátil para controlar el sistema de generación. Como se puede ver en la Figura 4.1 el canal de oleaje tiene tres partes básicas:

- La zona de generación: situada en el extremo derecho del tanque y que contiene el sistema de generación de oleaje, que en este caso es un pistón.
- La zona de disipación: situada en el extremo opuesto a la zona de generación. Es una zona llena de virutas de plástico que cuenta con una rejilla para evitar que las virutas salgan de esa zona.
- La zona de ensayos: parte que se encuentra entre la zona de generación y la zona de disipación y donde tiene lugar la interacción entre las olas y los objetos.



Figura 4.1 – Imagen del canal de olas nanoCIEM en las instalaciones del grupo EPhysLab.

El tanque tiene un sensor de ultrasonidos de pequeño formato que se puede colocar a lo largo del canal. El sensor es el modelo XX918A3F1M12 de la empresa Schneider Electric, con un rango máximo de 50 cm y un tiempo de respuesta de 25 ms. Su finalidad es medir la variación de la superficie libre, para ello el sensor suministra al software de generación una señal de voltaje (0-10 V) proporcional a la distancia entre él y la superficie libre de agua.

Además, el tanque tiene dos soportes para el sensor de ultrasonidos, que también actúan como rigidizadores de la estructura, y dos soportes en forma de U, que facilitan un punto de anclaje central en el fondo del canal.

El objeto flotante que se simula en estos experimentos es una boya esférica (Figura 4.2). Tiene un radio de 3.0795 cm (diámetro de 6.159 cm), una masa de 60.78 g y una densidad de 505 kg/m³, que es aproximadamente la mitad que la del agua, por lo tanto, con el agua en reposo, la boya se encontrará semisumergida. Debido a la diferencia entre el ancho del canal y el diámetro de la boya y a su forma esférica, cuando la boya esté en el tanque los experimentos tendrán una naturaleza tridimensional.



Figura 4.2 – Boya flotante utilizada en los ensayos físicos.

4.2 Generación y absorción del oleaje

La generación de olas se lleva a cabo mediante una pala del tipo pistón, que está situada en uno de los extremos del tanque. El movimiento del pistón lo realiza un actuador lineal con una carrera máxima de 10 cm y una velocidad de respuesta de 1.6 m/s. Para controlar el movimiento del pistón se utiliza un software instalado en un portátil que se conecta al tanque mediante USB. El software instalado en el portátil nos permite configurar la carrera del pistón o *stroke* (S_0) (llamado amplitud en el software) y el período de movimiento del pistón (T) para la generación de oleaje regular. S_0 puede tener un valor comprendido entre 0.3 y 10 cm en función del período seleccionado y T puede tener un valor comprendido entre 0.3 y 2 s en función del valor de S_0 seleccionado. Además, también permite seleccionar el número de olas que se van a generar. En el caso de no seleccionar ningún número, el pistón generará olas indefinidamente hasta que se le indique que pare. Para restringir las condiciones de oleaje del canal y que éstas no supongan un problema para el canal se ha introducido una restricción en los valores de amplitud y período (ver Figura 4.3). De este modo, para un período dado comprendido entre el mínimo posible (0.3 s) y 1.1 s, la máxima carrera permitida sigue la función $S_0 = 0.1T - 0.01$. Mientras que a partir de un período de 1.1 s y hasta el máximo período posible (2 s) la máxima carrera permitida está definida por la carrera máxima del pistón (10 cm).

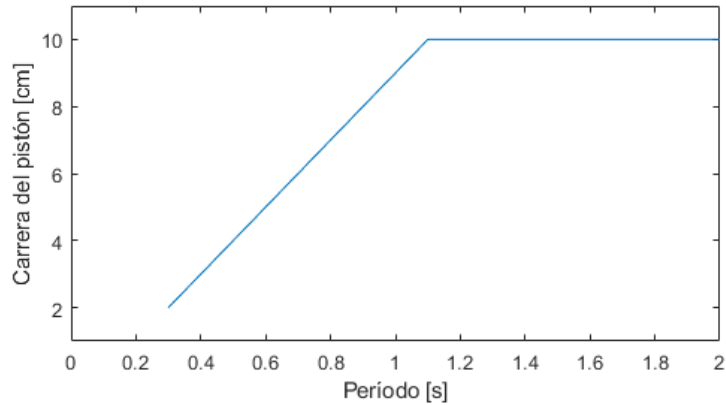


Figura 4.3 – Curva de capacidad máxima de generación de nanoCIEM.

Como ya se mencionó anteriormente, en el extremo opuesto al pistón, hay una zona de disipación o amortiguamiento compuesta por virutas de plástico que mide 33.5 cm, cuya finalidad es evitar o minimizar la reflexión de las olas incidentes.

El software del nanoCIEM nos muestra en pantalla dos gráficas (ver Figura 4.4), una de la variación temporal del movimiento del pistón (*Pala* en cm) y otra de la variación temporal de la superficie libre en la posición del sensor acústico (*Nivel* en cm). También muestra el valor del nivel de profundidad en todo momento. En la Figura 4.4 se puede apreciar en la esquina superior derecha de la segunda gráfica que la profundidad era de 13.01 cm en el momento de sacar la fotografía.

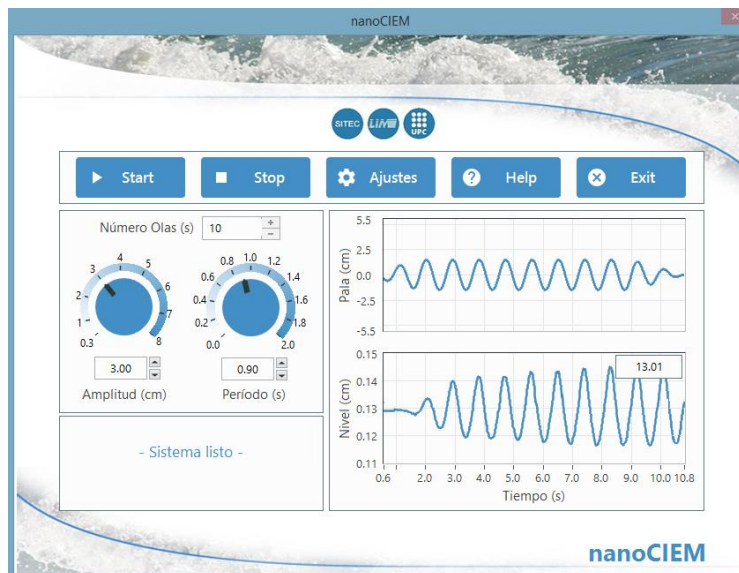


Figura 4.4 – Software para el manejo del nanoCIEM.

En la Figura 4.4, también se puede apreciar que, en la serie temporal del movimiento del pistón, el primer movimiento del pistón es más pequeño que los siguientes. Esto es debido a que utiliza una rampa de generación a lo largo del primer período y no alcanza la amplitud completa hasta el segundo periodo. De esta forma, se consigue que los movimientos mecánicos de la pala no sean tan bruscos y la transición hacia la amplitud deseada sea más suave. Como consecuencia de esto la primera elevación medida por el sensor acústico es sensiblemente inferior a las siguientes. Además de mostrar las gráficas en pantalla, la aplicación guarda los datos del movimiento del pistón y del nivel de la superficie libre de agua en un archivo de texto con extensión dat.

4.3 Diseño de la boya flotante

Aparte del canal de oleaje se han empleado otras herramientas para llevar a cabo estos experimentos. En particular se han utilizado diferentes softwares que nos permitirán diseñar e imprimir nuestra propia boya flotante.

En primer lugar, se usa SolidWorks, que es un software CAD, para diseñar una boya cuya densidad estuviera cercana a la mitad de la densidad del agua. Para calcular la densidad que tendrá la boya se ha tenido en cuenta el material con el que estará fabricada, PETG o poliéster de glicol (de densidad 1270 kg/m^3), y el radio interno y externo de la boya, ya que será una esfera hueca (Figura 4.5). Sabiendo estos datos, se pueden calcular tanto la masa como el volumen de la boya y por lo tanto saber su densidad. Hay que tener en cuenta que la distancia entre el radio interno y el radio externo debe ser lo suficientemente grande para que la boya no sufra deformaciones y para que sea impermeable. Para el cálculo de la densidad no se ha tenido en cuenta una pequeña argolla que tiene la boya, ya que debido a su pequeño tamaño no afectará demasiado al valor final de la densidad. La finalidad de la argolla es la de poder amarrar la boya con una cuerda.

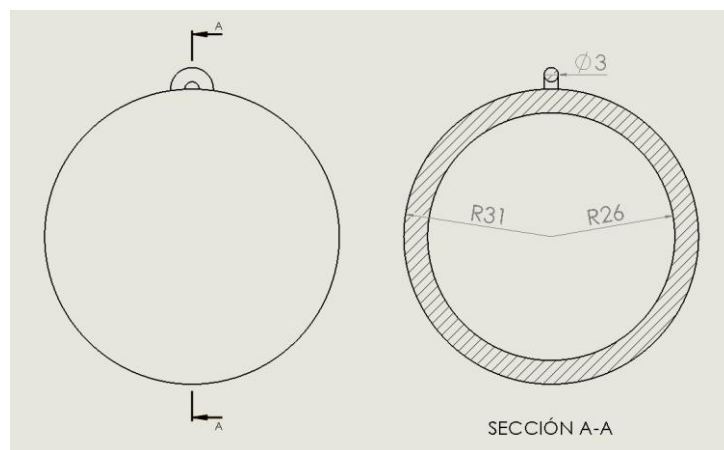


Figura 4.5 – Plano y cotas de la boya.

Una vez tenemos el diseño definitivo de la boya en SolidWorks, se guarda el archivo como STL y lo cargamos en el software Ultimaker Cura. En este nuevo software se ajustan los parámetros de impresión como la altura de capa y la velocidad de impresión y se crea un soporte para la

boya, ya que la impresora 3D no puede imprimir en el aire. El programa también nos permite ver una vista previa de nuestra pieza 3D capa a capa, la cantidad de material que necesitará para su impresión y el tiempo de impresión. Por último, se genera un código G, que es el lenguaje de programación más usado en máquinas que trabajan por control numérico computarizado (CNC), que se guarda en una tarjeta micro SD. En la Figura 4.6 podemos observar la vista previa para una altura de capa que se encuentra aproximadamente por la mitad de la boya. El material que aparece en la parte inferior de la boya en color azul es el soporte, que una vez terminada la pieza se retira con la ayuda de unos alicates.

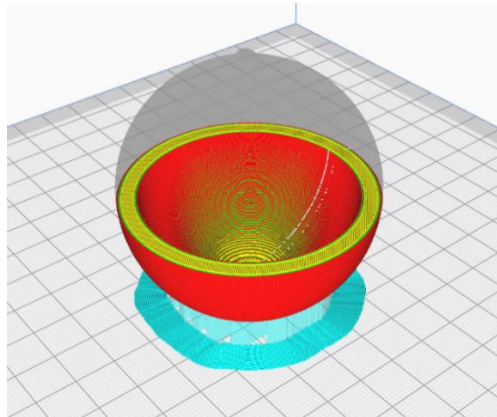


Figura 4.6 – Vista previa del software Ultimaker Cura.

El siguiente paso es meter la tarjeta micro SD en la impresora 3D (un modelo Anet ET4 X en este caso) y ajustar algunos parámetros como la temperatura de extrusión y la temperatura de la cama, que dependen del material utilizado (PETG). Cuando se han ajustado todos los parámetros necesarios la impresora está lista para empezar a imprimir nuestra boya tal y como se muestra en la Figura 4.7.

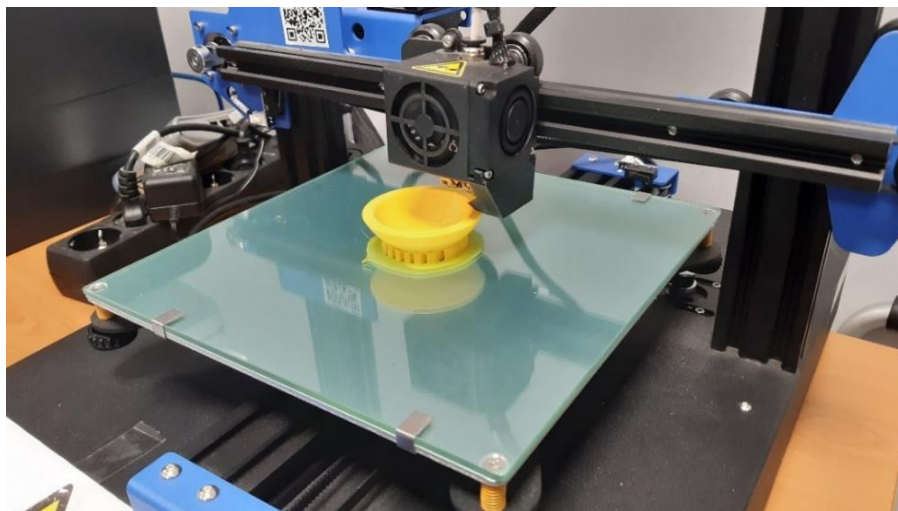


Figura 4.7 – Impresora 3D Anet ET4 X imprimiendo la boya flotante.

Por último, es importante mencionar que durante los ensayos físicos también se ha utilizado un trípode TripodPro168 de Phoenix y la cámara de un Samsung Galaxy A40 para grabar los movimientos de traslación de la boya en el tanque experimental. Los videos grabados han sido analizados en el software Kinovea para obtener el desplazamiento longitudinal (*surge* en inglés) y vertical (*heave* en inglés) de la boya, con el objetivo de poder comparar los datos experimentales con los resultados del modelo numérico utilizado en este trabajo. En la Figura 4.8 podemos ver un instante del video grabado y procesado con Kinovea, que introduce un marcador cuadrado rojo para el seguimiento de la boya. También se muestra la trayectoria que ha seguido el centro del cuadrado (línea roja), y por lo tanto la trayectoria de la boya, con anterioridad a este instante.

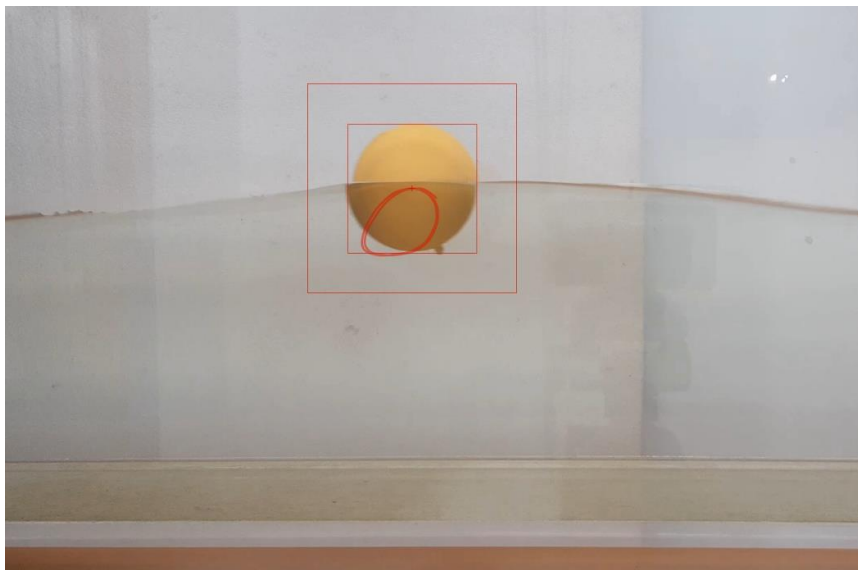


Figura 4.8 – Instantánea de Kinovea siguiendo la trayectoria de la boya.

5 Modelado numérico: DualSPHysics

En este capítulo se describen las bases del método SPH y se presenta DualSPHysics. El software SPH empleado en el desarrollo de este trabajo.

La interacción entre el oleaje y un objeto flotante es un problema complejo no lineal que se puede resolver utilizando la Dinámica de Fluidos Computacional (CFD). Los métodos CFD tienen en cuenta las fuerzas viscosas y las características no lineales de la interacción entre las olas del mar y un objeto. Los códigos CFD se basan en resolver las ecuaciones de Navier-Stokes. Estas ecuaciones se pueden resolver utilizando una aproximación Euleriana (generalmente asociado al uso de una malla) o una aproximación Lagrangiana (usando métodos sin malla). Los métodos basados en mallas (diferencias finitas, elementos finitos y volúmenes finitos) gozan de una elevada madurez y sus resultados son muy precisos en diversas aplicaciones. Los métodos con malla son perfectos para casos de estudio cuyo dominio esté bien definido y cuyos contornos sean fijos. Sin embargo, la correcta generación de la malla inicial es muy costosa y su modificación durante la simulación en entornos cambiantes puede dar lugar a importantes problemas.

Tradicionalmente los métodos CFD han usado métodos con malla. Sin embargo, los métodos sin malla se encuentran en continuo desarrollo y han despertado un gran interés en los últimos años, ya que se pueden aplicar a problemas no lineales con geometrías complejas, grandes deformaciones y discontinuidades en el flujo del fluido de una forma más natural. La malla se sustituye por un conjunto de nodos que pueden evolucionar de forma independiente. Además, los métodos sin malla no necesitan un algoritmo para detectar continuamente la superficie libre de fluido, ya que ésta se representa directamente por los nodos superficiales.

Uno de los métodos sin malla más populares es el método *Smoothed Particle Hydrodynamics* - SPH (Violeau, 2012). El método SPH es una buena opción para simular nuestro caso de estudio, donde un tren de olas interactúa con un objeto flotante y la superficie libre del agua sufre deformaciones significativas.

Entre las principales ventajas del método SPH destacan las siguientes: es posible discretizar flujos complejos de forma natural y no tiene limitaciones de la geometría del caso ni de la evolución de las condiciones iniciales. Por otro lado, los principales inconvenientes son: un coste computacional más alto que los métodos con malla debido a que cada partícula tiene que interactuar con un número elevado de partículas vecinas, que además cambian a lo largo de la simulación; algunas partículas pueden atravesar las condiciones de contorno establecidas y el desorden de las partículas afectará a la interpolación para hallar el valor de las propiedades de una partícula de interés.

En particular, en este trabajo se ha utilizado el código SPH llamado DualSPHysics, desarrollado, entre otros, por la Universidad de Vigo.

5.1 El método SPH

El método SPH es un modelo numérico Lagrangiano desarrollado durante la década de 1970 para resolver problemas de astrofísica y cosmología (Gingold & Monaghan, 1977; Lucy, 1977). En las últimas décadas, se ha utilizado con éxito para estudiar diferentes situaciones de la dinámica de fluidos como la propagación del oleaje y la interacción entre olas y estructuras costeras u objetos flotantes (Gotoh y Khayyer, 2018; Luo et al., 2021).

El método SPH discretiza el dominio en un conjunto de nodos o partículas. Las propiedades físicas de cada partícula (posición, velocidad, densidad y presión) se pueden obtener por interpolación local de las propiedades de las partículas circundantes. Las ecuaciones de Navier-Stokes se resuelven en cada par de interacción entre partículas. La tasa de la variación temporal de la densidad se calcula utilizando la ecuación de continuidad y la tasa de variación temporal de la velocidad con la ecuación del momento.

El método SPH está basado en la teoría de los interpolandos integrales. El principio básico es aproximar cualquier función $A(\mathbf{r})$ por:

$$A(\mathbf{r}) = \int_{\Omega} A(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' \quad (5.1)$$

siendo \mathbf{r} el vector de posición de una partícula, W la función *kernel*, que depende de la distancia entre partículas y del radio de interacción, h , que es la distancia de interacción entre partículas en un dominio cerrado.

La aproximación integral anterior se puede reescribir en notación discreta por medio del siguiente sumatorio:

$$A_i = \sum_{j=1}^N A_j W(\mathbf{r}_i - \mathbf{r}_j, h) \frac{m_j}{\rho_j} \quad (5.2)$$

donde el sumatorio se lleva a cabo sobre las partículas vecinas j de la partícula de interés i que se encuentran en el radio de interacción del *kernel*. Además, el volumen de una partícula se expresa como su masa (m_j) dividida por su densidad (ρ_j).

Para pasar de un dominio continuo, como un fluido, a un dominio discreto, partículas, se utiliza la función *kernel*. El *kernel* es una función que interpola los valores de cualquier propiedad del fluido en función del valor de las partículas vecinas que se encuentran dentro del radio de interacción (h). Esta función *kernel* cumple las siguientes propiedades:

- Positiva: $W(\mathbf{r} - \mathbf{r}', h) \geq 0$ dentro del dominio
- Soporte compacto: $W(\mathbf{r} - \mathbf{r}', h) = 0$ fuera del dominio
- Normalizada: $\int W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' = 1$
- Comportamiento de función delta: $\lim_{h \rightarrow 0} W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' = \delta(\mathbf{r} - \mathbf{r}')$
- Monótona decreciente

La Figura 5.1 muestra el dominio de influencia de la función *kernel* (W). Las partículas que se encuentran a una distancia menor que kh , en nuestro caso, $2h$, es decir, dos veces la longitud de suavizado h , son las que interactúan con la partícula de interés. La función *kernel* tiene forma de Campana de Gauss, por lo que la contribución de las partículas cercanas será mucho mayor que las partículas que se encuentran más alejadas de la partícula de interés.

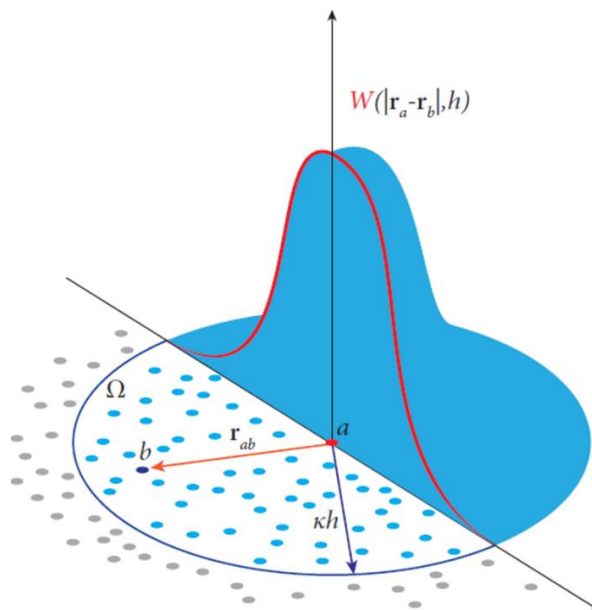


Figura 5.1 – Esquema del dominio de influencia de la función *kernel*.

El rendimiento de un modelo SPH depende en gran medida de la elección de la función *kernel* utilizada. Estos *kernel* se expresan en función de la distancia adimensional entre partículas (q), dada por $q = \frac{r}{h}$, donde r es la distancia entre dos partículas y h es el radio de interacción. En este trabajo se ha empleado la función *kernel* de quinto orden Wendland (Wendland, 1995):

$$W(r, h) = \begin{cases} \alpha_D \left(1 - \frac{q}{2}\right)^4 (2q + 1) & \text{si } 0 \leq q \leq 2 \\ 0 & \text{si } q > 2 \end{cases} \quad (5.3)$$

siendo α_D una constante que vale $\frac{7}{4}\pi h^2$ en 2D y $\frac{21}{16}\pi h^3$ en 3D.

Ecuaciones fundamentales

La formulación del método SPH se presenta de forma detallada en Monaghan (1992), Liu (2003) y Monaghan (2005). A continuación, se describen brevemente las principales ecuaciones utilizadas en SPH.

La ecuación de continuidad (o conservación de la masa) de Navier-Stokes para un dominio continuo puede escribirse de la siguiente forma:

$$\frac{1}{\rho} \frac{d\rho}{dt} + \nabla \mathbf{v} = 0 \quad (5.4)$$

Donde ρ es la densidad, t es el tiempo y \mathbf{v} velocidad del fluido.

Utilizando la función *kernel* para pasar de un dominio continuo a un dominio discreto, la ecuación anterior se reescribe como:

$$\left(\frac{d\rho}{dt}\right)_i = \sum_{j=1}^N m_j (\mathbf{v}_i - \mathbf{v}_j) \nabla_i W_{ij} \quad (5.5)$$

La ecuación de conservación del momento de Navier-Stokes para un dominio continuo puede escribirse de la siguiente forma:

$$\rho \left(\frac{d\mathbf{v}}{dt} + \mathbf{v} \nabla \mathbf{v}\right) = -\nabla p + \nabla f_{viscosas} + \mathbf{f}_{ext} \quad (5.6)$$

donde ahora p es presión.

Utilizando la función de *kernel*, la ecuación (5.6) se reescribe como:

$$\left(\frac{d\mathbf{v}}{dt}\right)_i = - \sum_{j=1}^N m_j \left(\frac{p_j}{\rho_j^2} + \frac{p_i}{\rho_i^2} + \Pi_{ij} \right) \nabla_i W_{ij} + \mathbf{g} \quad (5.7)$$

Siendo Π_{ij} el término de la viscosidad y \mathbf{g} la fuerza de la gravedad.

La viscosidad (Π_{ij}) tiene un papel importante en la simulación del movimiento del fluido, ya que representa la fricción interna entre las moléculas del fluido, es decir, la resistencia del fluido al flujo. La manera más simple de representar el término de la viscosidad es con la viscosidad artificial de Monaghan (1992), siendo ésta la formulación utilizada en las simulaciones de este trabajo. Su uso es común en SPH debido a su simplicidad y resulta adecuada para modelar la generación, propagación e interacción de olas con estructuras costeras obteniendo buenos resultados (Altomare et al., 2017).

Además de las ecuaciones de Navier-Stokes, se utiliza la ecuación de estado para relacionar la densidad y la presión, ya que se usa una aproximación débilmente compresible del fluido. Esta ecuación ha sido desarrollada por Monaghan y Kos (1999) y Batchelor (1974).

$$p = b \left[\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right] \quad (5.8)$$

siendo ρ_0 la densidad de referencia y b un factor que depende de la densidad de referencia, de la velocidad del sonido y del índice politrópico (γ). En aplicaciones marítimas el índice politrópico es igual a 7, por lo que pequeñas variaciones en la densidad provocarán un cambio considerable en la presión, siendo débilmente compresible. Según Monaghan (1994), la compresibilidad se ajusta en el modelo para que la velocidad del sonido pueda reducirse artificialmente, lo que establece un valor razonable para el tamaño del paso de tiempo. Sin embargo, este ajuste obliga a que la velocidad del sonido sea al menos 10 veces mayor que la velocidad máxima del fluido, manteniendo las variaciones de densidad en un margen inferior al 1%. Esto es necesario porque si se utilizase la velocidad del sonido real, se necesitaría un paso de tiempo muy pequeño para

asegurar la convergencia de la solución numérica, lo que aumentaría significativamente el tiempo de cálculo. Debido a esto, se define el factor b en la ecuación como:

$$b = \frac{\rho_0 c_0^2}{\gamma} \quad (5.9)$$

Siendo $\gamma=7$ y c_0 la velocidad del sonido cuando la densidad es igual a $\rho_0 = 1000 \text{ kg/m}^3$.

Condiciones de contorno

El objetivo principal de las condiciones de contorno es contener el fluido y modelar los elementos sólidos con los que debe interactuar el fluido sin atravesarlos. Por otro lado, también deben contribuir a completar el soporte del *kernel* de las partículas de fluido próximas al límite porque en estos casos la función *kernel* no sería una función normalizada y proporcionaría resultados incorrectos. Para abordar este aspecto, hay diferentes aproximaciones para implementar las condiciones de contorno.

Las Condiciones de Contorno Dinámicas (DBC), cuya descripción completa se encuentra en Crespo et al. (2007), es un método en el que las partículas de contorno se rigen por las mismas ecuaciones que las partículas de fluido, con la diferencia de que no se mueven de acuerdo a las fuerzas ejercidas sobre ellas, si no que permanecen fijas en su posición (contorno fijo) o se mueven según un movimiento impuesto (pistón, flap, compuerta...). Cuando una partícula de fluido se aproxima a una partícula contorno y su distancia es menor que la distancia de interacción entre las partículas, la densidad de las partículas de contorno aumenta y provoca un aumento de la presión. Debido a esto, se origina una fuerza de repulsión sobre las partículas de fluido. Las DBC son muy versátiles y permiten modelar geometrías muy complejas sin mayores dificultades (Zhang et al., 2018). Sin embargo, estas condiciones tienen algunas desventajas como el exceso de repulsión que ejercen sobre el fluido generando un hueco entre fluido y contorno. Estos inconvenientes obligan a usar una resolución elevada para simular correctamente la capa límite en flujos 2D estacionarios, como los flujos de Poiseuille y Couette.

Para solucionar estos problemas de las DBC se desarrollaron las DBC modificadas (mDBC), descritas en profundidad en English et al. (2019) y que se han usado en este trabajo. En este caso, las partículas de contorno se colocan de la misma manera que en las DBC, con la interfaz limítrofe situada a medio espacio entre las partículas de fluido y de frontera. Para cada partícula contorno se proyecta un nodo fantasma en el fluido respecto la interfaz limítrofe usando un método semejante al de Marrone et al. (2011). Para una superficie plana, el nodo fantasma se refleja a través de la interfaz limítrofe en la dirección de la normal que apunta al fluido. Las propiedades del fluido se encuentran en este nodo fantasma a través de una suma SPH corregida de las partículas de fluido circundantes. Como se puede ver en la Figura 5.2, para una partícula contorno (naranja) se crea un nodo fantasma (cruz naranja) en el fluido (partículas azules). Las partículas rosas son las partículas circundantes del nodo fantasma y con ellas se calculan las magnitudes físicas de la partícula contorno de interés. La línea negra representa el límite entre contorno (partículas verdes) y fluido (partículas azules y rosas).

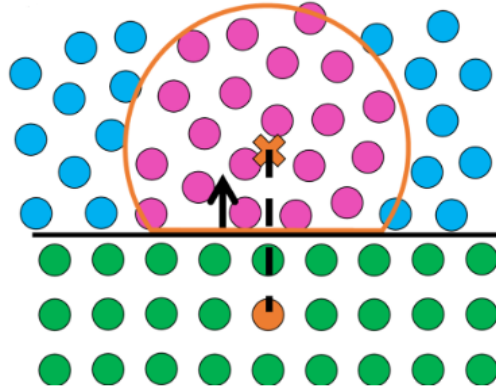


Figura 5.2 – Esquema de funcionamiento de las condiciones de contorno mDBC. Fuente: <https://github.com/DualSPHysics/DualSPHysics/wiki/3.-SPH-formulation#38-boundary-conditions>

Este método para hallar la densidad de las partículas contorno, proporciona campos de presión y densidad más suaves en la frontera y reduce considerablemente la brecha entre el contorno y el fluido que es uno de los principales problemas de las DBC. Las partículas de frontera obtienen las propiedades del fluido extrapolando los valores calculados en el nodo fantasma de una manera similar a la utilizada para las fronteras abiertas en Tafuni et al. (2018).

Objetos flotantes

Un objeto flota en un fluido debido al principio de Arquímedes, que dice que todo cuerpo sumergido en un líquido sufre un empuje hacia arriba igual peso del volumen de líquido desalojado. Los objetos flotantes en SPH son objetos rígidos que se mueven libremente en función de fluido que los rodea. Para simular el movimiento de un objeto flotante hay que tener en cuenta su interacción con las partículas del fluido y las fuerzas que estas ejercen sobre él y luego añadirle la fuerza de su peso. Suponiendo que el objeto flotante es un cuerpo rígido, la fuerza neta sobre cada partícula se calcula sumando las contribuciones de cada partícula de fluido según la función *kernel* y la longitud de suavizado. Por lo tanto, cada partícula contorno k del objeto experimenta una fuerza por unidad de masa dada por:

$$f_k = \sum f_{ka} \quad (5.10)$$

siendo f_{ka} la fuerza por unidad de masa ejercida por la partícula de fluido sobre la partícula contorno k y cumple la siguiente igualdad:

$$m_k f_{ka} = -m_a f_{ak} \quad (5.11)$$

Se utilizan las ecuaciones básicas de la dinámica de un sólido rígido:

$$M \frac{dV}{dt} = \sum m_k f_k \quad (5.12)$$

$$I \frac{d\Omega}{dt} = \sum m_k (\mathbf{r}_k - \mathbf{R}_0) \times f_k \quad (5.13)$$

siendo M la masa del objeto, I su momento de inercia, \mathbf{V} su velocidad, $\boldsymbol{\Omega}$ su velocidad de rotación y \mathbf{R}_0 su centro de masas. Integrando respecto al tiempo estas ecuaciones se hallan la velocidad lineal y la velocidad de rotación para el siguiente paso de tiempo. Por lo tanto, la velocidad de cada partícula contorno k viene dada por:

$$\mathbf{u}_k = \mathbf{V} + \boldsymbol{\Omega} \times (\mathbf{r}_k - \mathbf{R}_0) \quad (5.14)$$

Este método conserva el momento lineal y angular de las partículas tal como demuestran Monaghan et al. (2003) y Monaghan (2005). Por otro lado, Bouscasse et al. (2013) mostraron validaciones exitosas de la interacción no lineal del oleaje con objetos flotantes en SPH comparando sus resultados con datos experimentales de superficie libre y movimiento de objetos (Hadzić et al., 2005).

5.2 Código DualSPHysics

El código DualSPHysics (Domínguez et al., 2021) es un software basado en el método SPH y desarrollado por el grupo EPhysLab de la Universidad de Vigo, en colaboración con la Universidad de Manchester (Reino Unido), la Universidad de Parma (Italia), el Instituto Tecnológico de Lisboa (Portugal), la Universidad de Gante (Bélgica) y el Instituto Tecnológico de Nueva Jersey (Estados Unidos) como se puede ver en la Figura 5.3.



Figura 5.3 – Principales instituciones académicas involucradas en el desarrollo de DualSPHysics.

Fuente: <https://dual.sphysics.org/>.

DualSPHysics se ha desarrollado especialmente para simular problemas de la superficie libre de agua. El grupo EPhysLab se centra sobre todo en la propagación del oleaje y la interacción entre olas y estructuras flotantes o fijas tanto en 2D como en 3D (Crespo et al., 2011; Altomare et al., 2015; Domínguez et al., 2019).

DualSPHysics es un código libre y abierto que puede descargarse gratuitamente desde <https://dual.sphysics.org/>. Este código se publica bajo la licencia GNU Lesser General Public License, lo que significa que el código puede incorporarse tanto a *software* libre como a *software* privativo. Uno de los propósitos de que el código sea libre es animar a otros investigadores al uso del método SPH y a participar activamente en el desarrollo del código.

La primera versión de DualSPHysics se publicó en 2011 y hasta el día de hoy se han registrado más de 100.000 descargas. Los códigos SPH dan una descripción precisa del fluido, pero su principal desventaja es su elevado coste computacional, lo que puede limitar su uso en el estudio de problemas de ingeniería real. El tiempo de cálculo se incrementa rápidamente al aumentar el número de partículas y las aplicaciones reales de ingeniería requieren un número muy alto de partículas. Por ello, se utilizan técnicas de computación de altas prestaciones para reducir el tiempo de ejecución y así poder abordar problemas de ingeniería reales. Una de estas técnicas consiste en el uso de tarjetas gráficas para ejecutar el método SPH. Estos dispositivos tienen una gran potencia de cálculo en paralelo y permiten acelerar la ejecución de SPH de forma significativa (Domínguez et al., 2013).

DualSPHysics puede ejecutarse tanto en la CPU (*Central Processing Unit* en inglés) ya que está implementado en lenguaje C++, como en la GPU (*Graphic Processing Unit* en inglés) gracias a su programación en lenguaje CUDA. Las ejecuciones en la GPU pueden ser hasta 100 veces más rápidas que las de la CPU (Crespo et al., 2011; Domínguez et al., 2013). Esto permite al software simular millones de partículas para realizar simulaciones de resolución alta con un tiempo computacional razonable.

Además del código fuente de DualSPHysics, el paquete de *software* disponible en la página web incluye herramientas de pre-procesado y post-procesado. Estas herramientas de pre-procesado permiten al usuario definir la geometría y la configuración del caso a simular. Para ello, el usuario creará un archivo XML de entrada o puede utilizar la interfaz gráfica de usuario de DualSPHysics basada en FreeCad. El post-procesado permite al usuario analizar los resultados obtenidos como: las fuerzas ejercidas sobre los objetos, la elevación de la superficie libre en distintos puntos, la presión o la velocidad de las partículas de agua. Además, también se pueden crear archivos para visualizar la simulación mediante programas gratuitos como Paraview (<https://www.paraview.org/>). La Figura 5.4 muestra un esquema con los pasos necesarios para crear un caso, ejecutar la simulación y obtener los resultados para su posterior análisis.

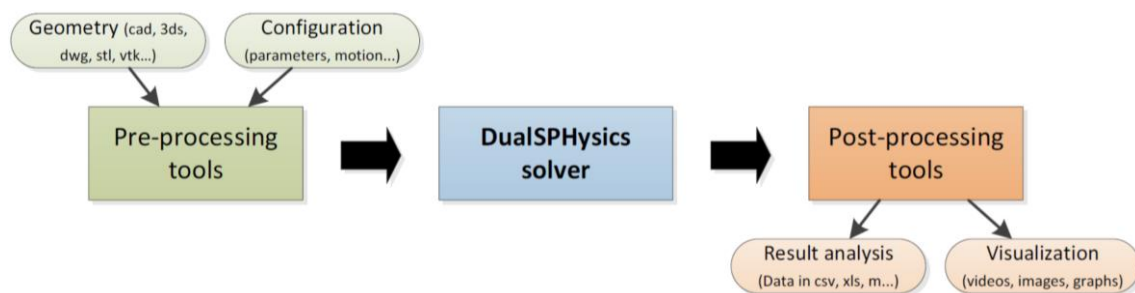


Figura 5.4 – Flujo de trabajo con DualSPHysics. Fuente: <https://dual.sphysics.org/>

Generación de oleaje

El generador de oleaje en DualSPHysics puede ser del tipo pistón o del tipo flap y permite generar tanto olas regulares como irregulares. La teoría implementada en DualSPHysics para la generación de oleaje está basada en Madsen (1971), que desarrolló una teoría simple de generación de olas de segundo orden de Stokes que no cambian su forma al propagarse. La formulación detallada se describe en el Capítulo 3.

Además de la teoría de Madsen para determinar el movimiento del pistón o flap, DualSPHysics incluye la opción de leer un fichero de texto con los datos del movimiento del pistón experimental, lo que permite reproducir en el modelado numérico exactamente el mismo movimiento del experimento.

Acoplamiento con MoorDyn

MoorDyn es un código abierto y gratuito que puede descargarse en <http://www.matt-hall.ca/moordyn.html>. Esta librería resuelve la dinámica de un sistema de amarres y ha sido desarrollada para poder ser acoplada con otros modelos. MoorDyn resuelve las líneas de amarre como masas puntuales unidas por modelos lineales de amortiguación para modelar la elasticidad en la dirección axial del amarre. La amortiguación hidrodinámica y la masa añadida se caracterizan mediante la ecuación de Morrison aplicada a cada masa puntual. Además, también resuelve la fricción de los amarres con el fondo y permite unir varias líneas de amarre y conectar varias estructuras flotantes. En la Figura 5.5 se muestra un esquema de como MoorDyn discretiza las líneas de amarre, así como las distintas funcionalidades que proporciona esta librería de amarres.

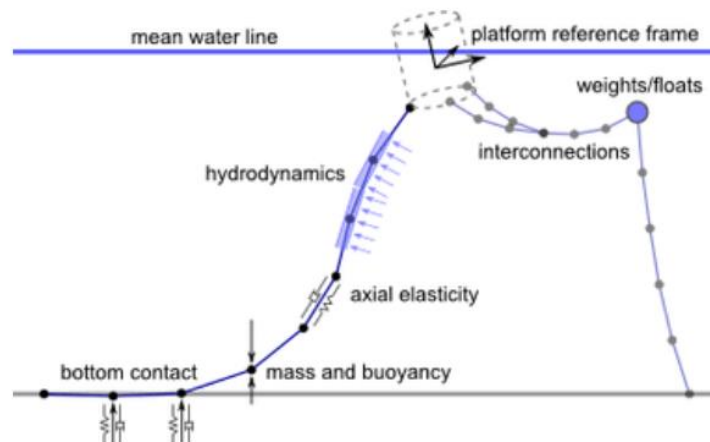


Figura 5.5 – Esquema funcional de MoorDyn. Fuente: <http://www.matt-hall.ca/moordyn.html>.

MoorDyn ha demostrado, con una formulación muy simple, que es computacionalmente eficiente y preciso para simular objetos flotantes con amarres en condiciones de oleaje regular (Domínguez et al., 2019).

Este trabajo incluye un caso de estudio donde se modela el comportamiento de una boya flotante amarrada al fondo para limitar su movimiento. Por tanto, los grados de libertad que se encuentran afectados son: el movimiento de traslación en la dirección de propagación de las olas (*surge*), el movimiento de traslación vertical (*heave*) y el movimiento de rotación de cabeceo (*pitch*). De esta forma, el amarre ejerce una fuerza adicional sobre la boya que debe ser simulada de manera precisa para reproducir numéricamente su movimiento en el experimento. Esta nueva fuerza adicional se resuelve en DualSPHysics utilizando la librería externa MoorDyn.

La Figura 5.6 muestra gráficamente el acoplamiento entre DualSPHysics y la biblioteca MoorDyn. Este acoplamiento puede dividirse en tres pasos:

- 1) Los movimientos y rotaciones resueltos inicialmente en DualSPHysics (\mathbf{V} , $\mathbf{\Omega}$, \mathbf{R}_0) son los datos de entrada de MoorDyn.
- 2) MoorDyn resuelve el comportamiento de la línea de amarre durante el paso de tiempo utilizado en DualSPHysics, es decir, actualiza la posición de los segmentos de la línea de amarre y calcula las fuerzas en los nodos de la línea ($d\mathbf{V}/dt$, $d\mathbf{\Omega}/dt$).
- 3) Las fuerzas adicionales calculadas en MoorDyn se pasan a DualSPHysics para obtener la fuerza resultante que actúa sobre el objeto flotante y que se utiliza para calcular los movimientos y rotaciones finales del objeto flotante.

Cabe destacar que el coste computacional de MoorDyn en cada paso de tiempo suele ser insignificante comparado con el tiempo de ejecución necesario para resolver un paso de tiempo en el modelo SPH.

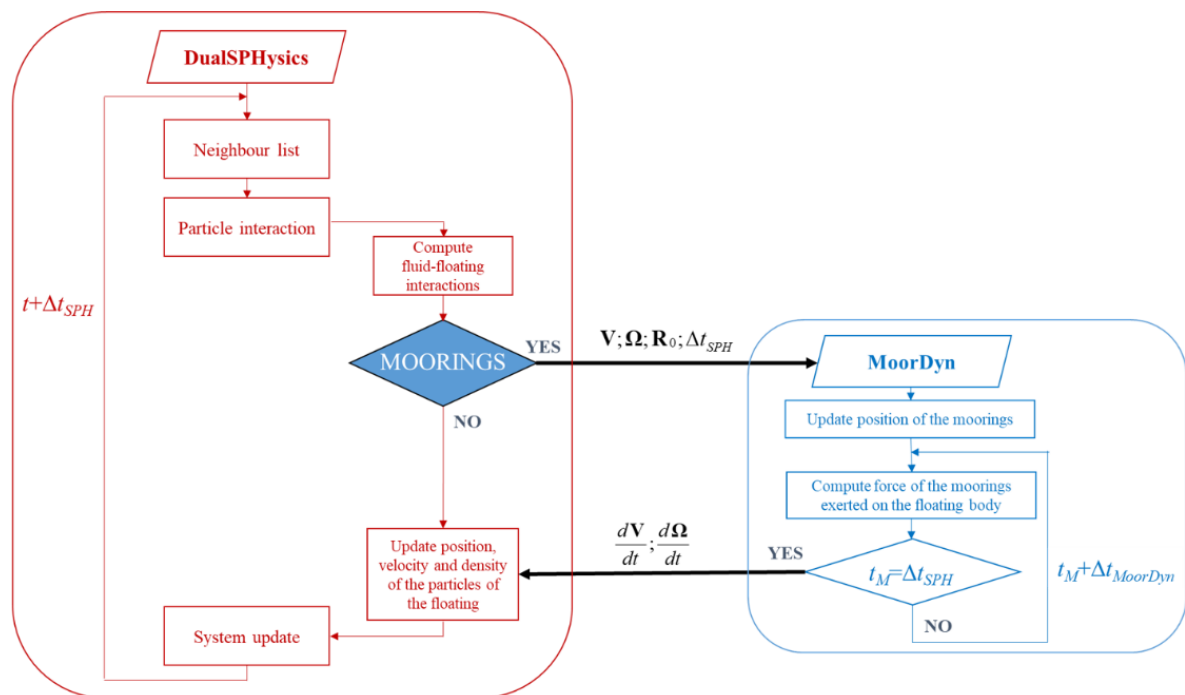


Figura 5.6 – Diagrama de flujo del acoplamiento de DualSPHysics con MoorDyn. Fuente: Domínguez et al. (2019).

Generación de condición inicial en DualSPHysics

La condición inicial de un caso de simulación en DualSPHysics se crea con la herramienta de pre-procesado llamada GenCase. Este programa discretiza la geometría del contorno y el fluido en partículas. Como se mencionó anteriormente, SPH es un método Lagrangiano sin malla, pero para definir la posición inicial de las partículas sí se utiliza una malla. Sin embargo, una vez creadas las partículas, estas se moverán libremente por el dominio según las ecuaciones de SPH. GenCase utiliza una malla tridimensional con un tamaño de celda dp (distancia inicial entre partículas) para ubicar las partículas en los nodos de esta malla, las partículas únicamente se crean en los nodos necesarios para diseñar la geometría deseada. Este método permite definir el caso para la resolución deseada (mediante la configuración de dp) independientemente de lo compleja que sea la geometría utilizada. El resultado como es de esperar depende de la resolución empleada tal como se muestra en la Figura 5.7, al disminuir la distancia entre partículas, aumenta el número de partículas y, por lo tanto, también aumenta la resolución.

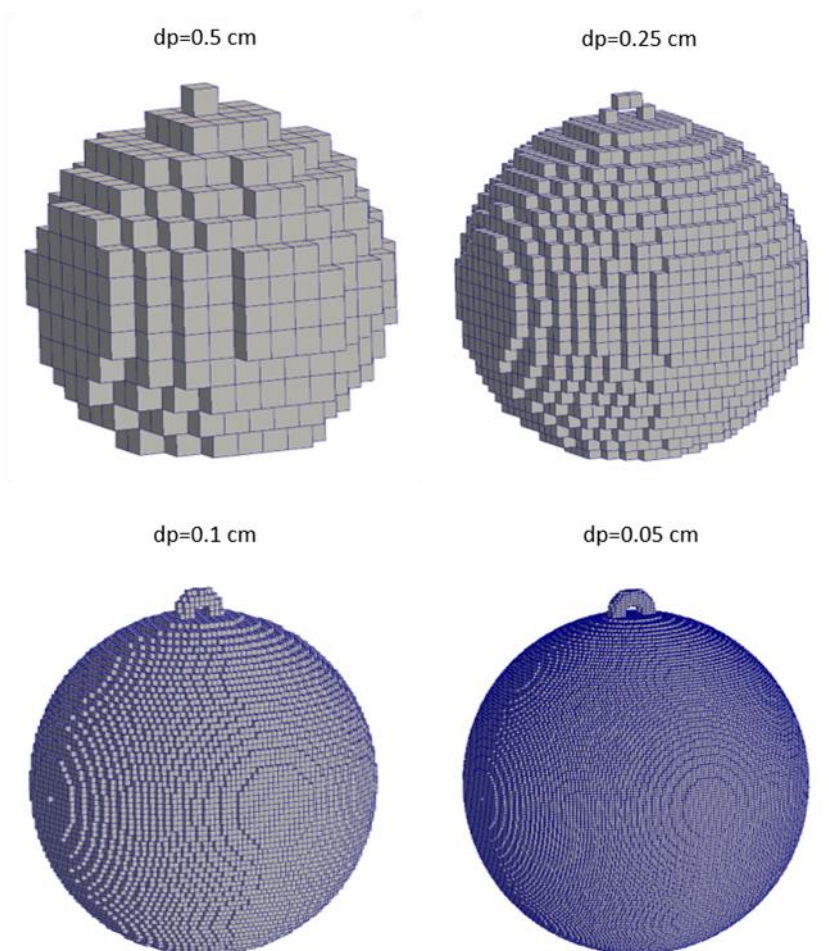


Figura 5.7 – Discretización de la boya (diámetro=6.159 cm) diseñada utilizando diferentes resoluciones. Las partículas se representan con cubos en este caso para una mejor visualización.

6 Validación

La finalidad de este capítulo es demostrar la capacidad de DualSPHysics para recrear la interacción entre olas y objetos flotantes. Para realizar la validación, se ha recreado el tanque experimental con el modelo numérico DualSPHysics. En primer lugar, se ha comprobado que el tanque numérico puede generar, propagar y absorber las diferentes olas simuladas tanto en 2D como en 3D, contrastando los resultados numéricos obtenidos con los experimentales. A continuación, se ha realizado un test de decaimiento para ver la interacción del objeto flotante con el agua en calma. Después se estudia ese objeto flotante bajo la acción del oleaje en el tanque numérico. Por último, se simula el objeto flotante pero esta vez amarrado al fondo. En todos los casos se comparan los resultados numéricos de DualSPHysics con los datos experimentales.

6.1 Validaciones previas en 2D y 3D

En la Figura 6.1 se puede ver la configuración del tanque físico en el que se han llevado a cabo las mediciones experimentales. Como podremos ver más adelante, la principal diferencia con el tanque numérico será la zona de amortiguamiento, que es la encargada de evitar la reflexión de las olas. La posición del sensor acústico (WG) para medir la elevación de la superficie del agua está a 103.5 cm del generador y permitirá verificar la correcta generación y propagación del oleaje. La profundidad inicial del agua en todos los experimentos es de 13 cm.

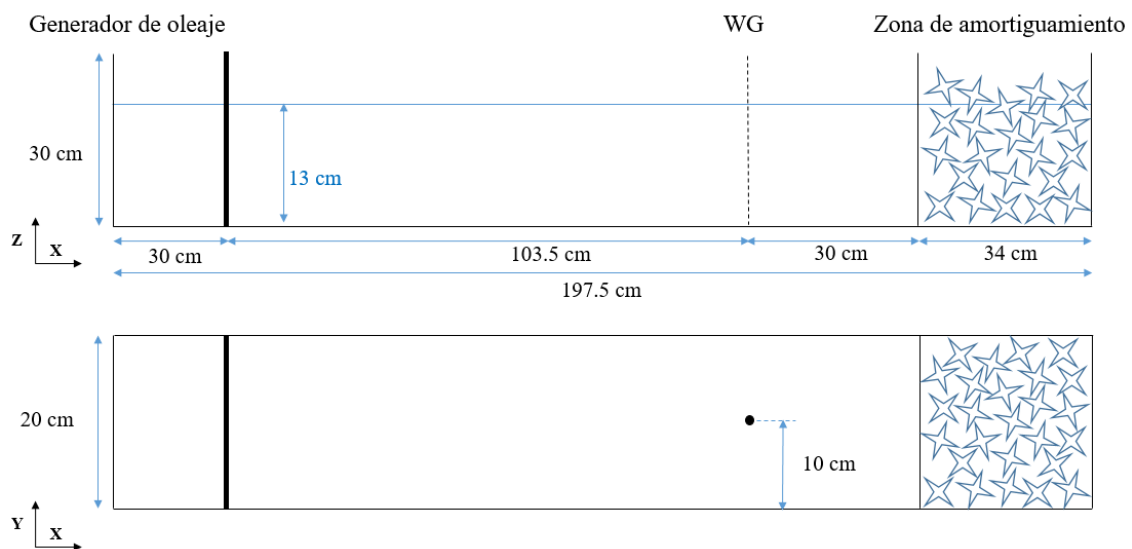


Figura 6.1 – Esquema tanque experimental.

La Figura 6.2 muestra un esquema del tanque numérico utilizado en DualSPHysics para validar la generación y propagación del oleaje. Tiene un pistón en uno de sus extremos, que es el encargado de generar el oleaje, y una playa de disipación con una pendiente (α) de 1:15.68 en el otro extremo, cuya finalidad es absorber el oleaje para evitar que se produzcan olas reflejadas. En este caso el tanque tiene una naturaleza bidimensional, ya que la interacción de las partículas fluidas con las partículas de las paredes laterales del tanque no altera su propagación, lo que supone un gran ahorro de tiempo computacional sin prácticamente afectar a los resultados. Todas las simulaciones realizadas en este trabajo se han realizado para una profundidad de 13 cm (igual a la del experimento). El punto de medida para la elevación de la superficie está situado a 103.5 cm de la posición inicial del pistón (igual que el sensor experimental), lo que significa que hay más de una longitud de onda para las tres olas utilizadas en este estudio, esto es necesario para que la ola generada por el pistón se forme correctamente y los valores medidos sean válidos.

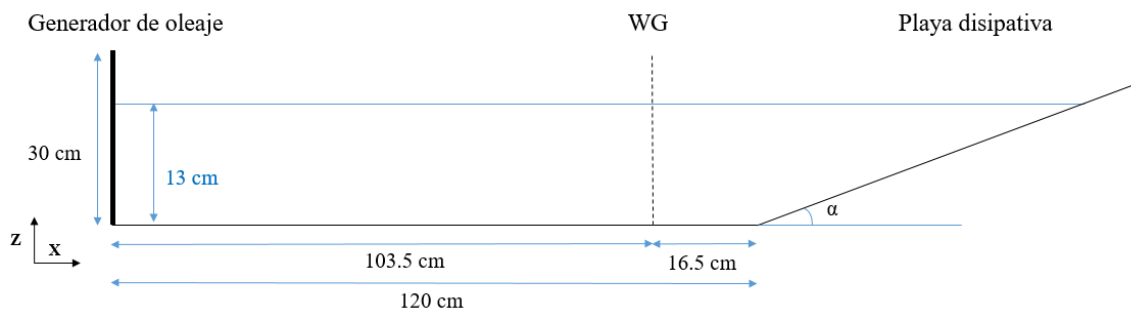


Figura 6.2 – Esquema tanque numérico 2D para generación y propagación de olas regulares.

El primer paso es comprobar que el oleaje se genera, propaga y absorbe de forma correcta en nuestro tanque numérico. Para esto, se reproduce una condición de oleaje regular usando DualSPHysics en el tanque numérico expuesto en la Figura 6.2. Se seleccionó para esta primera prueba olas regulares de período $T=0.9$ s, carrera del pistón $S_0=3$ cm (ver ecuación 3.13), altura de ola $H=2.67$ cm y longitud de onda $\lambda=90.65$ cm, realizando varias simulaciones con diferentes resoluciones. La resolución está definida por la distancia inicial entre partículas, llamada dp . Altomare et al. (2017) recomiendan usar una resolución en la que haya 8-10 partículas por altura de ola, ya que con esos valores se obtiene una buena precisión con unos tiempos de cálculo razonables. Los valores de dp utilizados en este caso han sido de 0.01 m, 0.005 m y 0.002 m. En esta primera validación se comparan resultados numéricos con la solución teórica y las variables analizadas han sido la superficie libre de agua (η) y la velocidad orbital (horizontal y vertical) en una posición determinada del tanque. En concreto esta posición está a 103.5 cm del pistón y 5 cm por debajo de la superficie libre inicial, o lo que es lo mismo, 8 cm por encima del fondo del tanque.

La variación temporal de la superficie libre y la velocidad orbital en un punto se puede obtener gracias a la teoría de Stokes de segundo orden, explicada en el capítulo 3.2. En la Figura 6.3 se incluye una comparación de los valores teóricos con los resultados numéricos para las tres resoluciones. Puede observarse que para un dp de 0.01 m la velocidad orbital vertical tiene grandes fluctuaciones, además tanto los valores de la velocidad orbital horizontal como la elevación de la superficie libre distan bastante de los valores teóricos, por lo que los resultados con este dp no son precisos. En cambio, los resultados numéricos para un dp de 0.002 m y 0.005 m muestran una buena correspondencia con los resultados teóricos, siendo mejores para el dp menor, que equivale a una mayor resolución. Por lo tanto, de ahora en adelante nuestras simulaciones numéricas tendrán un dp de 0.002 m. Los resultados con esta resolución confirman la correcta generación y propagación de oleaje en nuestro tanque numérico con DualSPHysics.

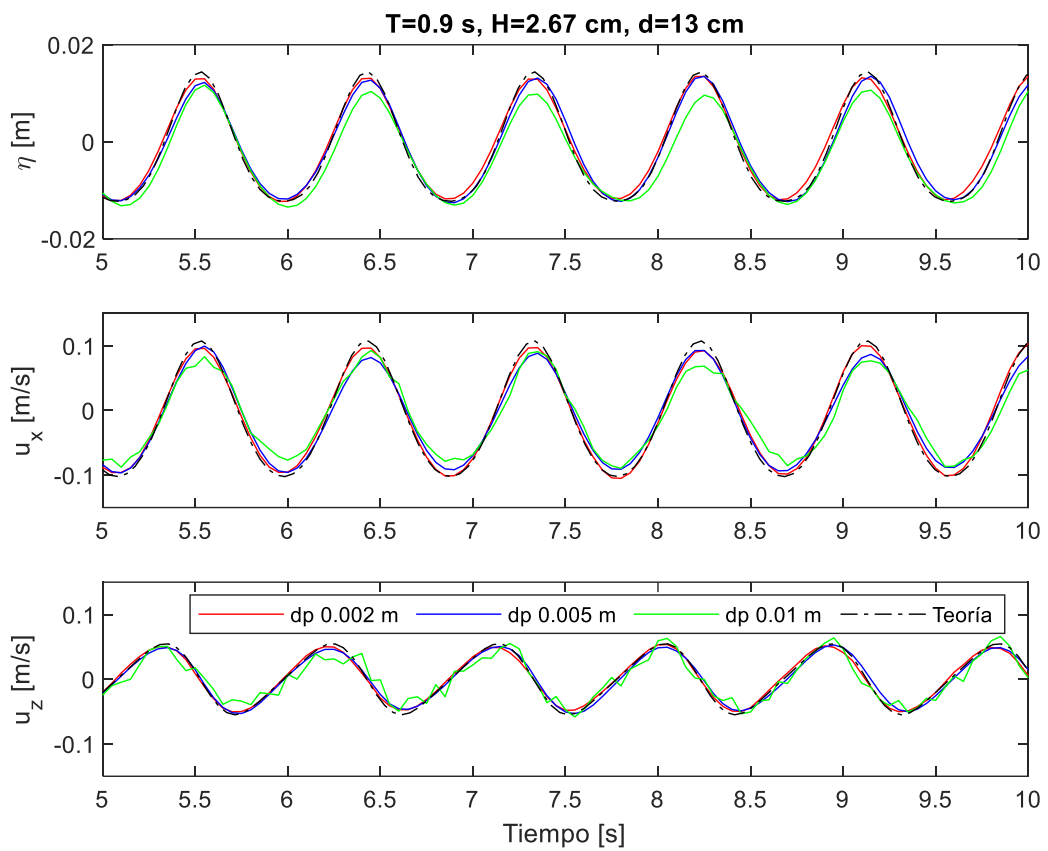


Figura 6.3 – Series temporales teóricas y numéricas (2D) de la elevación de la superficie libre y velocidades orbitales para oleaje regular con $T=0.9$ s, $H=2.67$ cm, $d=13$ cm.

En la Figura 6.4 se muestran 5 instantes de la simulación con DualSPHysics de un período completo. Se representan los valores de la velocidad longitudinal de las partículas fluidas y la posición de la superficie libre para el caso con mayor resolución ($dp=0.002$ m). Se puede observar como el primer instante (8.10 s) y el último (9.00 s) tienen los mismos valores de velocidad longitudinal y los dos están en contrafase respecto al instante intermedio (8.55 s). Igualmente se puede observar como la velocidad de las crestas en la playa disipativa es mayor que en la zona plana, esto es debido al efecto del fondo.

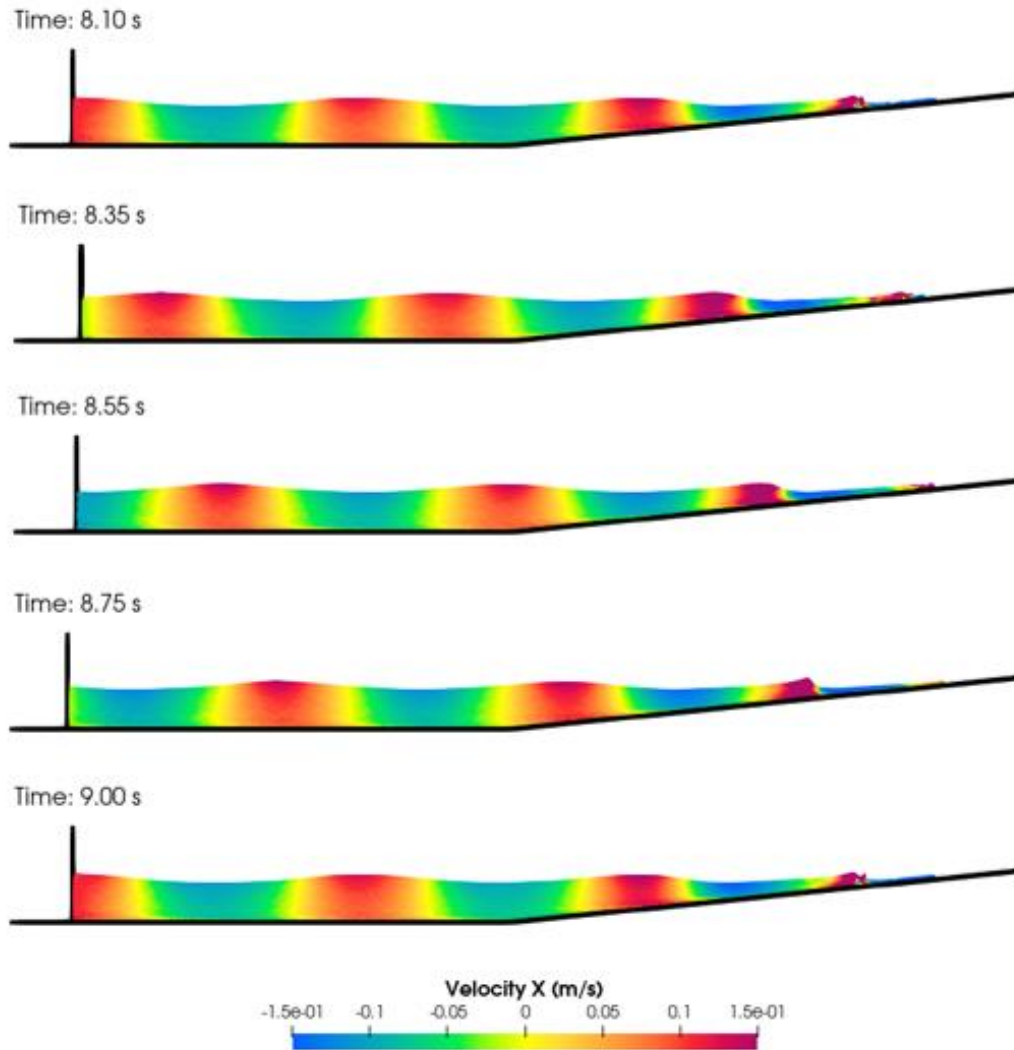


Figura 6.4 – Diferentes instantes de la simulación DualSPHysics para el oleaje regular con $T=0.9$ s, $H=2.67$ cm, $d=13$ cm. El color de las partículas corresponde a la velocidad longitudinal del fluido.

Por otra parte, para validar la capacidad de absorción del oleaje de la playa disipativa del tanque numérico, se utiliza al método de Healy (Eagleson y Dean, 1966), con el que se puede obtener el coeficiente de reflexión, K_R , cuyo valor se calcula con la siguiente fórmula:

$$K_R = \frac{H_{max} - H_{min}}{H_{max} + H_{min}} \quad (6.1)$$

siendo H_{max} la altura máxima y H_{min} la altura mínima. En este caso K_R es inferior al 4.5%, por lo que la reflexión será mínima y prácticamente no afectará a las olas incidentes, lo que significa que las olas se propagan de forma correcta y no existe reflexión al final del tanque numérico. Esto es importante cuando se introduzcan objetos flotantes puesto que asegura que la ola incidente que llegue a la posición del objeto es la deseada.

La Tabla 6.1 incluye el número de partículas y el tiempo de ejecución para cada resolución requerido para simular 12 segundos de tiempo físico usando una tarjeta gráfica Nvidia GeForce RTX 2080 Ti. Como se puede ver a menor distancia inicial entre partículas, mayor número de partículas, y por lo tanto mayor resolución, lo que conlleva a un mayor tiempo de cálculo.

Tabla 6.1 – Número de partículas y tiempo de ejecución según la resolución utilizada.

dp [m]	Partículas	Tiempo de cálculo [min]
0.010	4,209	2.27
0.005	13,736	5.11
0.002	75,335	14.27

Una vez validado nuestro tanque numérico en 2D, el siguiente paso es asegurar la misma precisión de los resultados en 3D ya que las simulaciones con objetos flotantes que se presentarán más adelante tienen una naturaleza tridimensional. Para esto, se realiza otra simulación en el tanque numérico expuesto en la Figura 6.2, pero esta vez con una anchura de canal de 20 cm. La simulación se ha realizado para el mismo oleaje regular ($T=0.9$ s, $S_0=3$ cm y $H=2.67$ cm) con un $dp=0.002$ m y se han comparado los resultados numéricos con los resultados teóricos. Como se puede ver en la Figura 6.5 la comparación entre los resultados numéricos en 3D y la teoría es realmente buena tanto para la variación temporal de la superficie libre como para la velocidad orbital de un punto (a 103.5 cm del generador y 3 cm bajo la superficie). Además, si se vuelve a calcular el coeficiente de reflexión (Ecuación 6.1) para este tanque 3D se obtiene un valor de 6.65%, que sigue siendo un valor muy bajo. De acuerdo a esto quedan validadas tanto la generación como la propagación de oleaje en nuestro tanque numérico 3D.

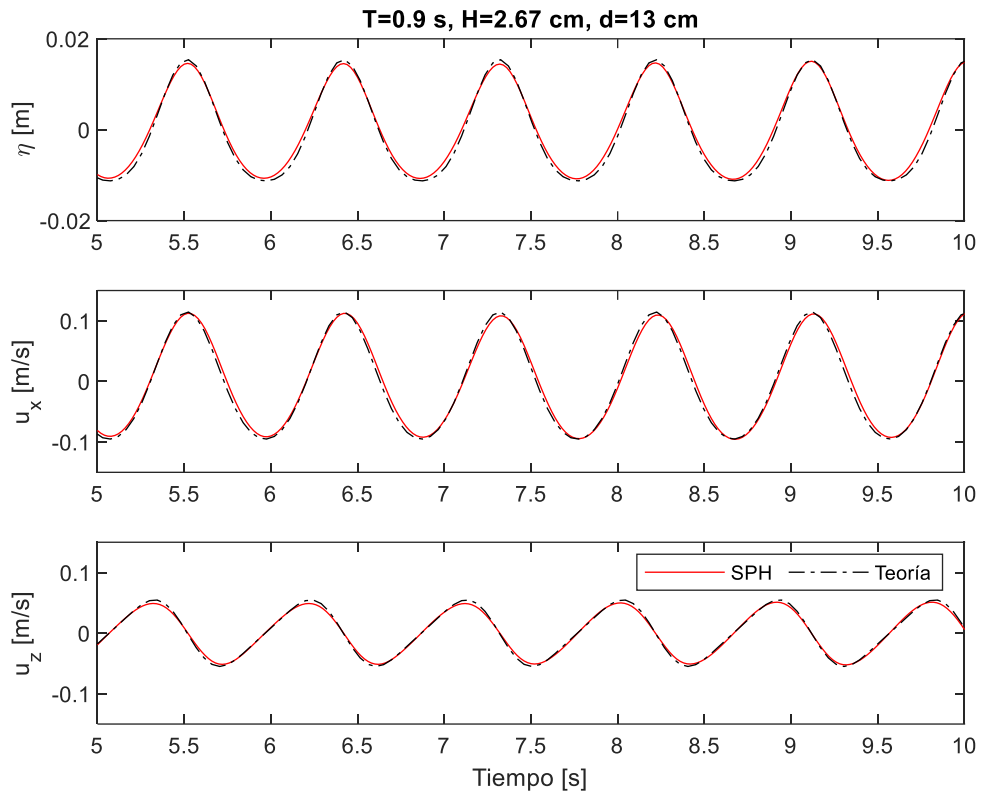


Figura 6.5 – Series temporales teóricas y numéricas (3D) de la elevación de la superficie libre y velocidades orbitales para oleaje regular con $T=0.9$ s, $H=2.67$ cm, $d=13$ cm.

Por último, es importante mencionar que, para simular 12 s de generación y propagación de esta condición de oleaje en la misma tarjeta gráfica, el tiempo de cálculo ha sido de 1892.31 minutos (31.53 h). Este elevado tiempo de ejecución se debe a la gran cantidad de partículas simuladas en el caso 3D (6,833,500).

Una vez validado nuestro tanque numérico tanto en 2D como en 3D, ya podemos llevar a cabo simulaciones más complejas y comparar los resultados numéricos obtenidos por DualSPHysics con los resultados experimentales obtenidos en nanoCIEM.

6.2 Propagación del oleaje regular

Los primeros test en donde se comparan datos experimentales con datos del modelo numérico tienen como objetivo validar sólo la generación y propagación para tres condiciones de oleaje distintas. Son tres olas que siguen la teoría de Stokes, de las cuales dos tienen el mismo período y distinta carrera del pistón (*stroke*) y dos tienen el mismo *stroke* y distinto período tal y como se puede observar en la Tabla 6.2.

Tabla 6.2 – Condiciones de oleaje estudiadas.

Ola	Stroke S_0 [cm]	Período T [s]	Altura H [m]	Longitud de onda λ [m]
A	3.5	0.7	0.0422	0.6503
B	3.5	0.9	0.0311	0.9065
C	3.0	0.9	0.0267	0.9065

La Figura 6.6 presenta las tres olas que se van a validar situadas en el ábaco de Le Méhauté. Como se aprecia, las tres olas se encuentran en la zona de profundidad intermedia. También se observa como las olas B y C se encuentran dentro de la teoría de Stokes de segundo orden y como la ola A pertenece a la teoría de Stokes de tercer orden, pero debido a su proximidad a la teoría de segundo orden se puede aproximar por esta última teoría, ya que las diferencias serán mínimas.

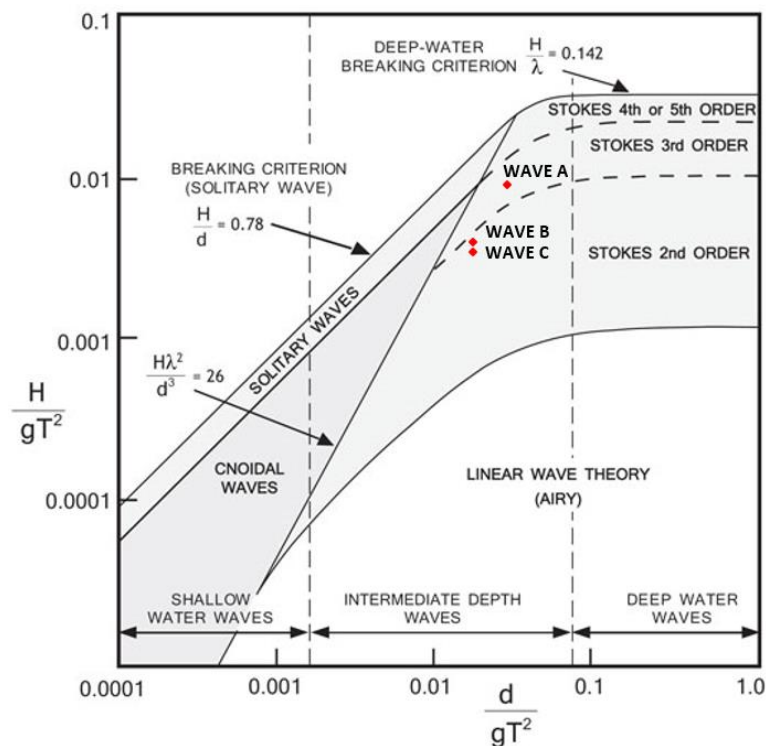


Figura 6.6 – Condiciones de oleaje estudiadas representadas en el ábaco de Le Méhauté.

A continuación, se analizan las series temporales de la elevación de la superficie libre tanto experimentales como numéricas y teóricas para las tres condiciones de oleaje escogidas con anterioridad. Estas simulaciones se pueden considerar en 2D, ya que, al no haber ningún objeto flotante en el tanque, los resultados serán prácticamente los mismos que si la simulación fuera tridimensional (tal y como se demostró en el apartado anterior).

Las series temporales de la superficie libre a 103.5 cm del pistón generador se muestran en la Figura 6.7, incluyendo los datos experimentales, los resultados numéricos y las soluciones teóricas. Como se puede ver en la figura, los resultados numéricos y teóricos para las tres condiciones de oleaje son prácticamente idénticos, algo que ya se había comprobado para el caso de $T=0.9$ s, $H=2.67$ cm y $d=13$ cm. Sin embargo, ahora también se incluyen magnitudes medidas durante los ensayos en el canal nanoCIEM para poder compararlos. Los resultados numéricos obtenidos con el código DualSPHysics son muy parecidos a los datos experimentales. Existen algunas pequeñas desviaciones o ruido en los datos experimentales debidas a pequeñas vibraciones que afectan a la sensibilidad del sensor acústico del experimento.

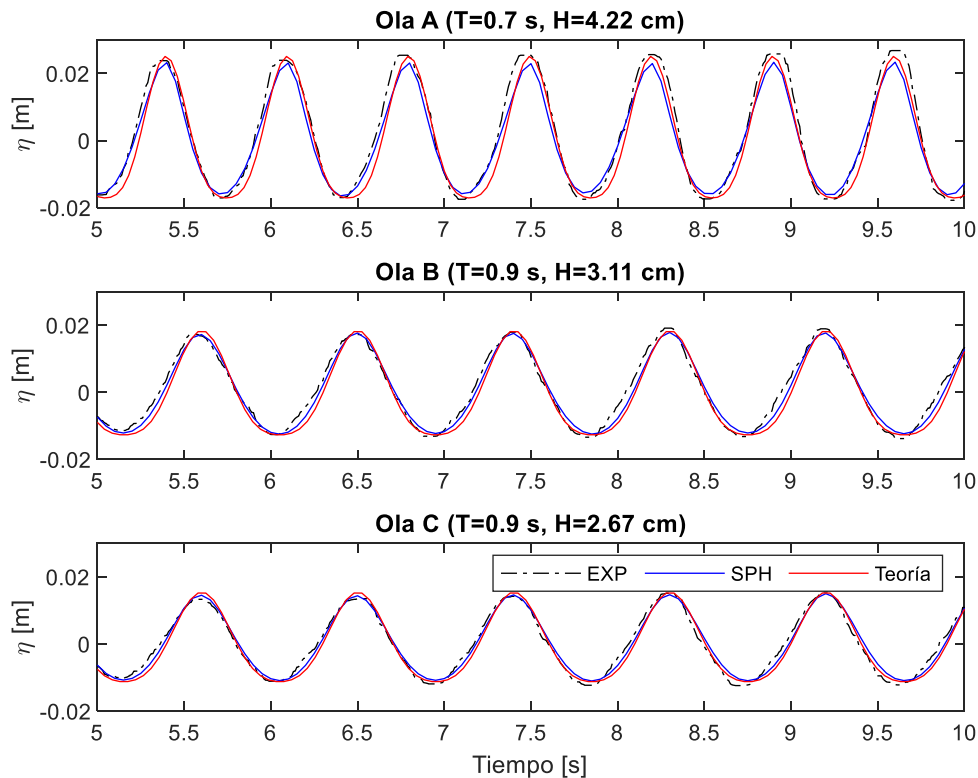


Figura 6.7 – Series temporales experimentales, numéricas y teóricas de la elevación de la superficie libre para las tres condiciones de oleaje regular: A) $H=4.22$ cm y $T=0.7$ s, B) $H=3.11$ cm y $T=0.9$ s, C) $H=2.67$ cm y $T=0.9$ s.

6.3 Test de decaimiento vertical

Antes de introducir el objeto flotante en el tanque numérico bajo la acción de las distintas condiciones de oleaje se realiza un test de decaimiento para confirmar que se ha definido correctamente el objeto flotante en DualSPHysics. Un test de decaimiento vertical consiste en desplazar el objeto flotante de su posición de equilibrio y dejarlo oscilar libremente en el eje vertical (eje z) hasta alcanzar de nuevo su posición de equilibrio. Para realizar este test, el agua del tanque tiene que estar en reposo.

Recordemos las características principales de la boya flotante: esfera de radio 3.0795 cm, masa 60.78 g y densidad 505 kg/m³. Todos estos valores se deben definir en la configuración de la simulación con DualSPHysics.

Como se puede ver en la Figura 6.8, el tanque numérico utilizado para simular el test de decaimiento es un cuadrado de 64 cm de lado y con una profundidad inicial de agua de 15 cm. La boya se encuentra inicialmente situada en el centro del cuadrado y 2 cm por encima de la superficie libre de agua, es decir, 2 cm por encima de su posición de equilibrio. El tanque numérico tiene una zona de amortiguamiento en sus paredes laterales de 7 cm, lo que permite disminuir la velocidad de las partículas fluidas y evitar la reflexión de las olas que produce la oscilación de la boya.

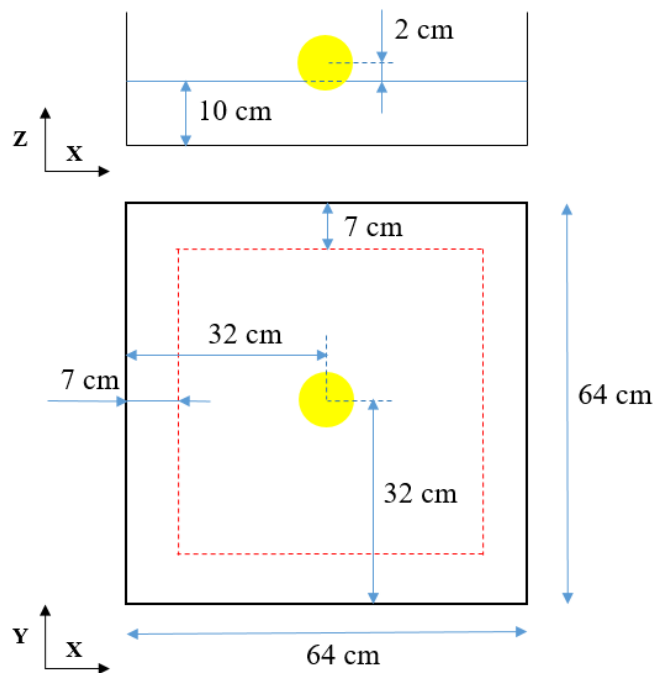


Figura 6.8 – Esquema de tanque numérico para el test de decaimiento vertical.

En la Figura 6.9 se pueden observar varios instantes de la simulación numérica del test de decaimiento. Como es lógico, en los primeros instantes la velocidad del fluido tiene un valor mayor debido a la transferencia de energía mecánica. Al soltarse desde una cierta altura, la boya tiene una determinada energía potencial. Esta energía potencial se transformará en energía cinética y la boya se moverá interactuando con el fluido y generando pequeñas ondas. La

energía cinética del fluido se disipará gracias al amortiguamiento de las paredes del tanque numérico. Al pasar el tiempo, la velocidad es menor ya que la boya va disminuyendo su oscilación y por lo tanto la energía potencial de la boya va disminuyendo. Como consecuencia de esto, la transferencia de energía es menor, hasta llegar a un momento en que es nula. En ese momento la boya se volverá a encontrar en su posición de equilibrio y el agua del tanque volverá a estar en reposo. También se puede apreciar como el perfil de velocidad no es circular en los últimos tres instantes que se muestran. Esto es debido a que el tanque es cuadrado y a que la zona de amortiguamiento no es lo suficientemente amplia para evitar toda la reflexión que producen las paredes del tanque, aunque es suficiente para mostrar unos buenos resultados comparados con la solución teórica.

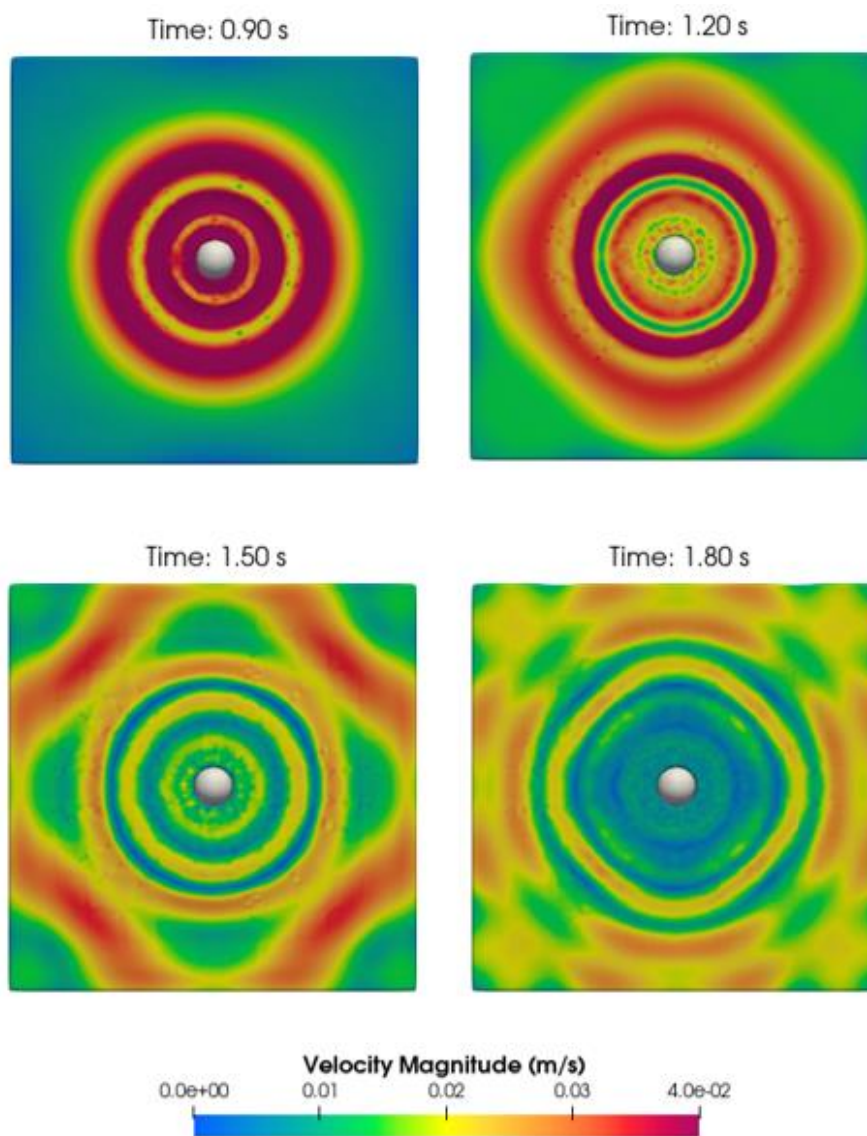


Figura 6.9 – Velocidad de las partículas fluidas durante el test de decaimiento.

Las series temporales del desplazamiento vertical (z) se muestran en la Figura 6.10. Se incluyen la solución teórica, los datos experimentales y los resultados numéricos de la simulación con DualSPHysics. Se puede apreciar cómo, en todos los casos, la boya oscila con una frecuencia constante mientras su amplitud disminuye hasta llegar a su posición de equilibrio. Esto ocurre debido a que la boya no se encuentra sometida a ninguna fuerza externa. Los resultados que se obtienen numéricamente se parecen mucho a los teóricos y también a los experimentales (mientras no hay reflexión en el tanque físico). A partir del instante en el que la boya del tanque experimental sufre la influencia de las olas reflejadas por las paredes laterales del tanque nanoCIEM (alrededor del segundo 0.8), se puede apreciar como el valor experimental permanece con una amplitud constante en lugar de ir disminuyendo hasta cero. Esto ocurre debido a que el tanque experimental no es lo suficientemente grande y las paredes laterales producen reflexión pronto. Además, también influye la calidad con la que se ha grabado el vídeo del decaimiento experimental para luego poder analizarlo en Kinovea. Este software estima el centro de la boya para poder obtener el desplazamiento vertical de la boya y la calidad del vídeo puede dar lugar a pequeños errores. El vídeo se ha grabado con la cámara de un teléfono móvil Samsung A40 y está limitado a 30 frames por segundo.

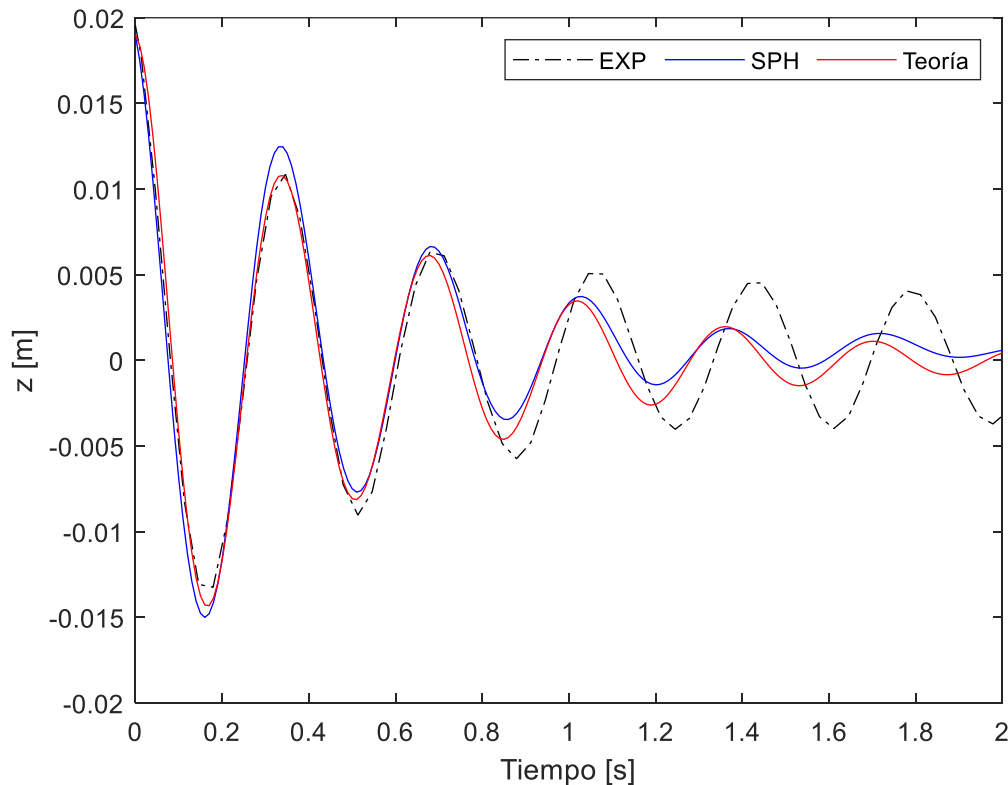


Figura 6.10 – Comparación del desplazamiento vertical teórico, numérico y experimental durante el test de decaimiento.

Las series temporales de la Figura 6.10 permitirán calcular las frecuencias (o periodos) naturales de oscilación, para así comparar el valor teórico, experimental y numérico. Para ello, en primer lugar, se explicarán brevemente algunos conceptos como vibración libre, vibración amortiguada, frecuencia natural, frecuencia amortiguada y amortiguamiento.

Una vibración libre se produce cuando un sistema vibra por sí mismo después de una perturbación inicial sin que ninguna fuerza externa actúe sobre el sistema. Una vibración amortiguada tiene lugar cuando se pierde o disipa energía por fricción durante la oscilación del sistema. La frecuencia natural es la frecuencia de un sistema que tiene vibración libre sin fricción y la frecuencia amortiguada es la frecuencia de un sistema que tiene vibración libre con fricción. El mecanismo que hace que se disipe la energía se llama amortiguamiento. Cuando un sistema vibra en un fluido, como en nuestro caso agua, la fuerza de amortiguamiento es proporcional a la velocidad del cuerpo vibratorio, $F = -c_a \dot{z}$, siendo c_a la constante de amortiguamiento y \dot{z} la velocidad. La relación de amortiguamiento (ζ) se define como la relación entre la constante de amortiguamiento y la constante de amortiguamiento crítico.

$$\zeta = \frac{c_a}{2m\omega_n} \quad (6.2)$$

siendo m la masa del cuerpo vibratorio y ω_n su frecuencia natural.

Si el valor de la relación de amortiguamiento es cero, la vibración es no amortiguada y su amplitud permanece constante. Si el valor es distinto de cero, hay tres posibles casos de amortiguamiento. En nuestro caso nos interesan los sistemas subamortiguados ($\zeta < 1$) que son aquellos en los que el sistema oscila alrededor de su posición de equilibrio con una amplitud que disminuye progresivamente. La frecuencia amortiguada para sistemas subamortiguados viene dada por la siguiente ecuación:

$$\omega_d = \sqrt{1 - \zeta^2} \omega_n \quad (6.3)$$

Por lo tanto, para calcular la frecuencia amortiguada teórica se necesita la frecuencia natural teórica, que se calcula añadiendo una masa a una semiesfera sumergida como explica Storti y D'Elía (2004). La frecuencia natural teórica obtenida para este test de decaimiento es de 18.4864 Hz. Una vez conocida la frecuencia natural, se calcula su frecuencia amortiguada mediante prueba y error variando la relación de amortiguamiento en el intervalo abierto entre 0 y 1 y se escoge la que más se parezca al decaimiento numérico.

Una vez explicado esto, a partir de la Figura 6.10, se puede obtener el período amortiguado teórico, numérico y experimental. Como se puede apreciar en la Tabla 6.3, los tres valores anteriores son muy parecidos, con lo que se puede afirmar que el test de decaimiento se ha realizado correctamente.

Tabla 6.3 – Valores de período amortiguado mediante distintos métodos.

Método de obtención	Valor
Numérico	0.3500 s
Teórico	0.3413 s
Experimental	0.3340 s

6.4 Objetos flotantes libres

Ahora que ya está validada la generación, propagación y absorción del oleaje en nuestro tanque numérico 3D (según resultados de sección 6.2) y nuestro objeto flotante está bien caracterizado numéricamente (sección 6.3), es el momento de empezar a realizar experimentos y simulaciones con el objeto flotante bajo las condiciones de oleaje mencionadas en la Tabla 6.2.

Las simulaciones numéricas llevadas a cabo en este apartado han sido realizadas en el tanque numérico mostrado en la Figura 6.11. Como se puede observar la boya se ha colocado prácticamente a una longitud de onda (λ) del pistón para las olas B y C y casi a 1.5λ para la ola A, con lo que la ola se habrá formado correctamente cuando llegue al objeto flotante. Además, en el plano X-Y se puede apreciar claramente la naturaleza tridimensional de este experimento, ya que, debido a la influencia del objeto flotante, se creará una componente transversal de la velocidad de las partículas fluidas cuando éstas interactúen con la boya. Debido a esto, las paredes laterales producirán cierta reflexión que afectará a la propagación del oleaje regular.

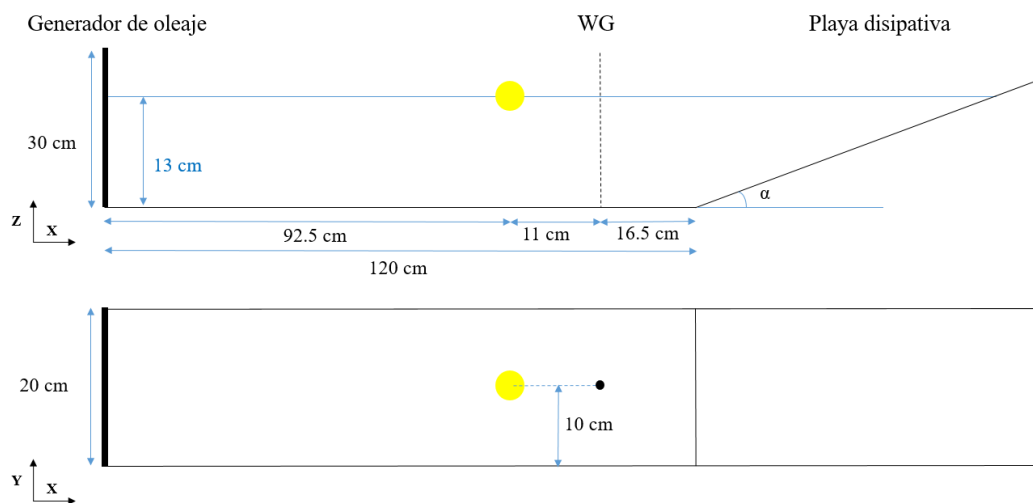


Figura 6.11 – Esquema del tanque numérico para test con la boya flotante (en amarillo).

En la Figura 6.12, se muestran 5 instantes de la simulación de DualSPHysics durante un período completo. Se representan los valores de la velocidad longitudinal de las partículas fluidas y la posición de la superficie libre para las condiciones de oleaje correspondientes a la ola C ($T=0.9$ s, $H=2.67$ cm). Se ha añadido una línea negra de referencia para comparar en cada instante la posición de la boya respecto al instante inicial (8.10 s). Esta línea de referencia permite apreciar como la boya va avanzando lentamente. En el instante intermedio (8.56 s) tiene lugar la mayor distancia respecto a la línea negra de referencia y en el instante final (9.0 s) retrocede un poco respecto al instante intermedio, pero no llega a la posición del instante inicial, por lo que con cada período la boya avanza ligeramente en la dirección longitudinal (eje x). Esto se podrá ver con más claridad en las gráficas del desplazamiento longitudinal (*surge*).

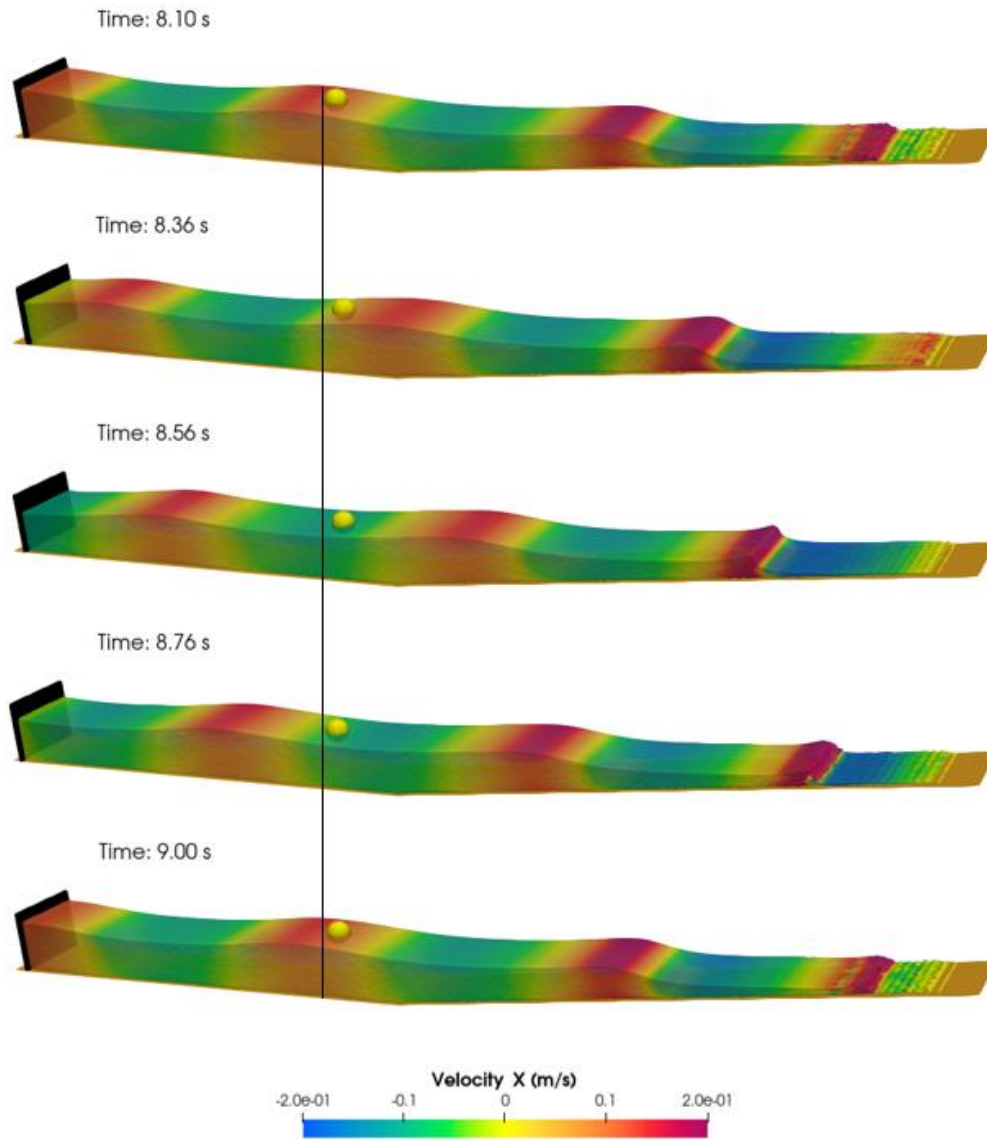


Figura 6.12 – Diferentes instantes de la simulación con DualSPHysics para la configuración de ola C ($T=0.9$ s, $H=2.67$ cm, $d=13$ cm). El color de las partículas corresponde a la velocidad longitudinal del fluido y la boya se muestra en amarillo. La línea negra marca la posición de la boya en el instante $t=8.10$ s.

Se realizó el mismo ensayo (experimentos en nanoCIEM y simulaciones con DualSPHysics) para las tres condiciones de oleaje seleccionadas anteriormente (Tabla 6.2). El experimento consiste en colocar un objeto flotante, en nuestro caso la boya descrita en el Capítulo 4, y medir su desplazamiento horizontal (*surge*) y su desplazamiento vertical (*heave*) respecto a su posición inicial con el agua en reposo.

El primer caso con la boya flotando libremente utiliza las condiciones de oleaje de la ola A ($T=0.7$ s, $H=4.22$ cm) y los desplazamientos de la boya numérica y experimental se muestran en la Figura 6.13. Como se puede observar en el desplazamiento longitudinal, los resultados numéricos y experimentales son prácticamente iguales para esta ventana temporal de 5 segundos. A partir del segundo 8, el desplazamiento experimental disminuye debido a la reflexión que existe en el tanque experimental ya que la zona de amortiguamiento de la parte final del nanoCIEM no es perfecta y para algunas condiciones de oleaje se puede registrar cierta reflexión al cabo de un tiempo. Sin embargo, la reflexión del tanque numérico es muy baja (valores menores al 10% en todos los casos), por lo que no afectará a la propagación de las olas ni al desplazamiento de la boya. En cuanto al desplazamiento vertical, se puede observar como la amplitud de los resultados experimentales y numéricos se mantiene constante, pero hay una ligera diferencia entre ellos. Más adelante se discutirá sobre la posible razón de estas discrepancias.

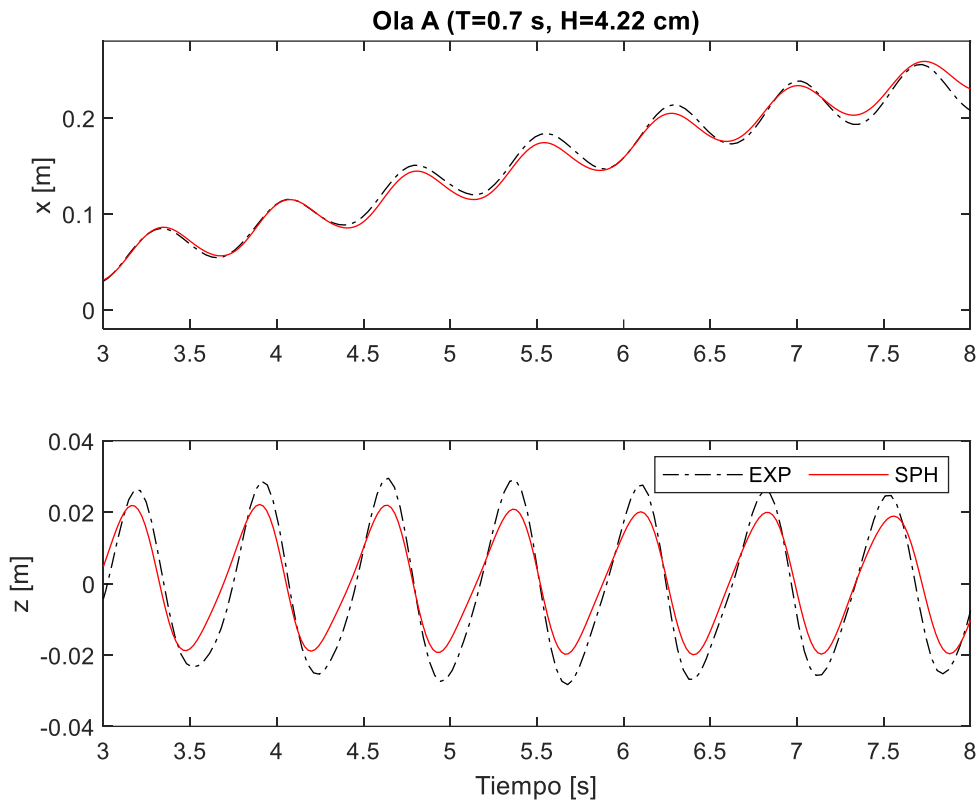


Figura 6.13 – Movimiento longitudinal (x) y vertical (z) de la boya usando la configuración de la ola A ($T=0.7$ s, $H=4.22$ cm).

El segundo caso estudia el movimiento de la boya bajo las condiciones de oleaje correspondientes a la ola B ($T=0.9$ s, $H=3.11$ cm). Los resultados numéricos y experimentales del movimiento de la boya con la ola B se muestran en la Figura 6.14. Como se puede ver en el desplazamiento longitudinal, los resultados numéricos y experimentales son muy parecidos desde el segundo 3 hasta el segundo 6, momento en el que la boya durante el experimento avanzó menos debido a que empieza a afectar la reflexión del final del tanque físico. En el desplazamiento vertical se puede ver como con estas condiciones de oleaje, la amplitud de los resultados numéricos se mantiene constante pero la de los resultados experimentales crece con el paso del tiempo.

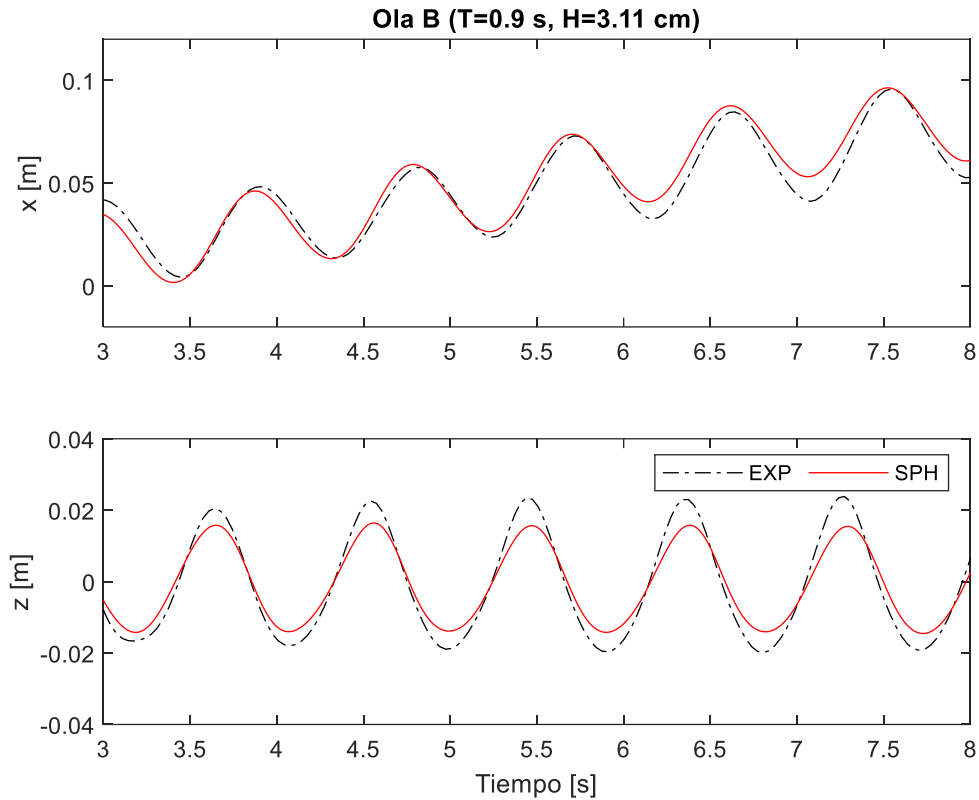


Figura 6.14 – Movimiento longitudinal y vertical de la boya usando la configuración de la ola B ($T=0.9$ s, $H=3.11$ cm).

El último caso con la boya flotando libremente se realiza bajo las condiciones de oleaje de la ola C ($T=0.9$ s, $H=2.67$ cm). La Figura 6.15 compara los desplazamientos horizontal y vertical obtenidos durante el experimento en nanoCIEM y los obtenidos usando el código DualSPHysics. Se observa una mayor diferencia entre el desplazamiento longitudinal numérico y experimental. Del segundo 3 al segundo 6 el desplazamiento es prácticamente el mismo, pero a partir de este momento la boya experimental avanza menos que la boya numérica ya que la reflexión en el tanque experimental no es tan eficiente como la implementada en la simulación numérica. En cuanto al desplazamiento vertical, igual que sucede en el caso anterior la amplitud de los resultados numéricos se mantiene constante pero la de los resultados experimentales crece con el paso del tiempo, por lo que se observa una diferencia importante entre ambos.

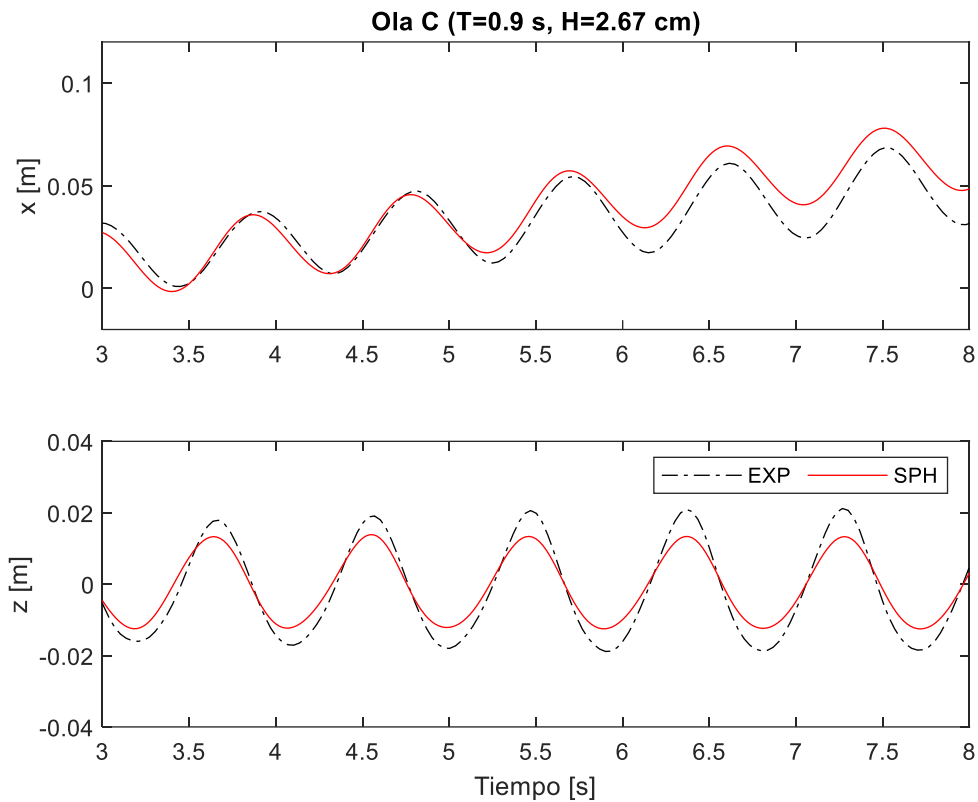


Figura 6.15 – Movimiento longitudinal y vertical de la boya usando la configuración de la ola C ($T=0.9$ s, $H=2.67$ cm).

Para las tres condiciones de oleaje se puede apreciar que los resultados numéricos y los resultados experimentales son muy parecidos para el desplazamiento longitudinal. Las comparaciones del desplazamiento vertical son las que presentan una mayor discrepancia. Esto puede ser debido a que la boya genera una componente transversal de la velocidad de las partículas fluidas que provoca reflexión con las paredes laterales del tanque. En el caso del tanque numérico esto se ha solucionado con una zona de amortiguamiento lateral a lo largo de las dos paredes laterales, pero en el tanque experimental no se puede solucionar. Tendría que utilizarse otro tanque experimental donde la distancia entre sus paredes laterales sea lo suficientemente grande para que la reflexión lateral no tenga tanta influencia en el movimiento de la boya. Esta reflexión lateral afecta al campo de velocidades tras la boya y por lo tanto a las olas incidentes que llegan a la zona de amortiguación al final del tanque, por lo que la eficiencia de la zona de absorción de energía del tanque experimental puede funcionar aún peor.

Los resultados mostrados en este apartado comparando los resultados numéricos y experimentales de los desplazamientos muestran la capacidad de DualSPHysics para simular objetos flotantes libres bajo condiciones de oleaje regular, en donde las zonas de amortiguamiento son más eficientes que en el tanque físico nanoCIEM.

6.5 Objetos flotantes con amarres

Una vez que se ha validado la capacidad de DualSPHysics para simular objetos flotantes libres, el siguiente paso es simular objetos flotantes con amarres, lo que afectará y limitará su movimiento. La simulación de objetos flotantes con amarres utiliza la librería MoorDyn acoplada a DualSPHysics, como se ha explicado en el Capítulo 5, para calcular y aplicar la tensión ejercida por el amarre. En nuestro caso no tenemos datos experimentales de la tensión de amarres, ya que el tanque experimental nanoCIEM no dispone de ningún sensor que permita medirla. Por lo tanto, se compararán los mismos datos numéricos y experimentales que en el caso sin amarres: desplazamiento horizontal (*surge*) y desplazamiento vertical (*heave*) de la boya. Además, se mostrarán las tensiones producidas en el amarre únicamente para las simulaciones numéricas realizadas con DualSPHysics acoplado con MoorDyn.

La Figura 6.16 muestra la configuración del tanque numérico empleado para este caso. Como se puede observar el amarre de la boya al fondo del tanque está situado a 92.5 cm del pistón, siendo esta distancia mayor de una longitud de onda para las tres configuraciones de ola (Tabla 6.2).

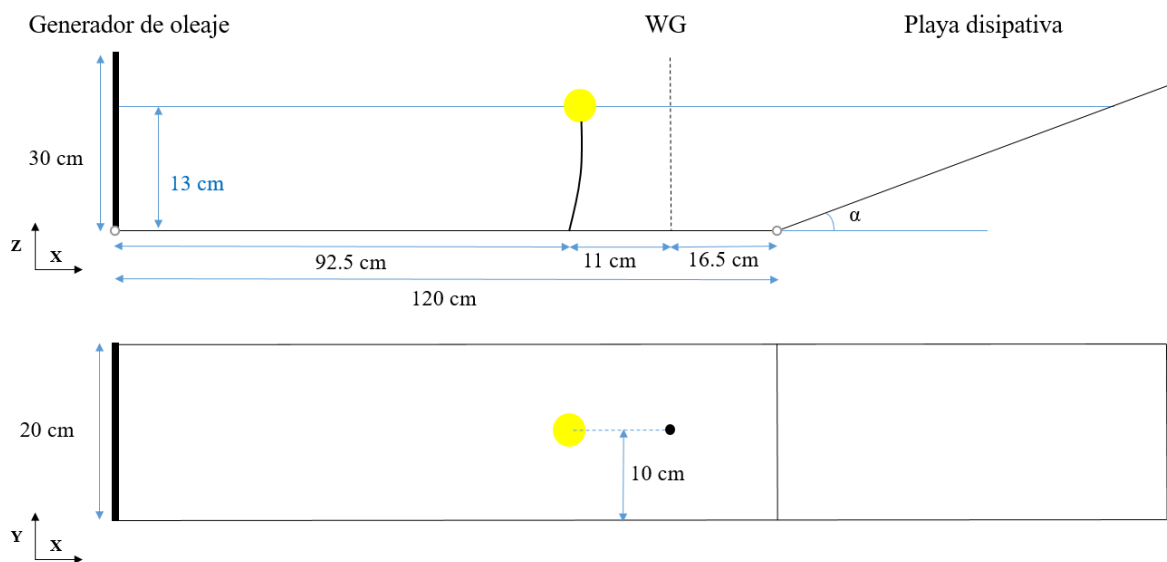


Figura 6.16 – Esquema del tanque numérico para test con la boya flotante amarrada al fondo.

En la configuración experimental, la línea de amarre no está sujeta al fondo del canal de oleaje ni tampoco se encuentra a una distancia de 3.0795 cm (*radio de la boya*) del centro de la boya, sino que está atada a una pequeña argolla diseñada precisamente para poder atar un cabo o una cadena a la boya y así permitir experimentos con una boya sujeta por una línea de amarre. La Figura 6.17 muestra un esquema detallado de la configuración de la boya y de la línea de amarre para la situación inicial con el agua en reposo.

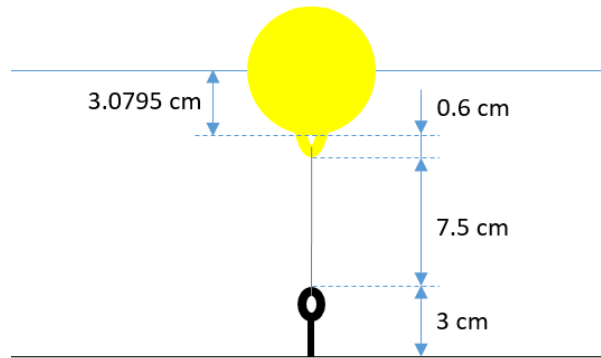


Figura 6.17 – Esquema en detalle de la configuración de amarre en el experimento físico.

La Tabla 6.4 contiene los principales parámetros de configuración de la línea de amarre, una cuerda de nylon en el experimento físico, para su simulación numérica con Moordyn.

Tabla 6.4 – Propiedades de la línea de amarre utilizadas en el modelado numérico.

Parámetro	Valor
Longitud	0.075 m
Densidad longitudinal	0.00077 kg/m
Diámetro	0.002 m
Módulo de elasticidad	1.59×10^9 N

En la Figura 6.18, se muestran 5 instantes de la simulación de DualSPHysics durante un período completo. Se representan los valores de la velocidad longitudinal de las partículas fluidas y la posición de la superficie libre para las condiciones de oleaje correspondientes a la ola C ($T=0.9$ s, $H=2.67$ cm). Igual que en la figura de la simulación sin amarres (Figura 6.12), se ha añadido una línea negra para comparar la posición de la boya en distintos instantes de un periodo con respecto al primer instante (8.10 s). En este caso, se puede observar como es el segundo instante (8.36 s), el momento en el que la boya se encuentra más alejada de la posición de referencia. Mientras que en el instante final (9.0 s) la posición de la boya casi no se ha desplazado de la posición de referencia. La boya nunca se aleja mucho de su posición inicial debido al amarre y mantiene un movimiento oscilatorio como muestran las siguientes gráficas de desplazamiento longitudinal (*surge*).

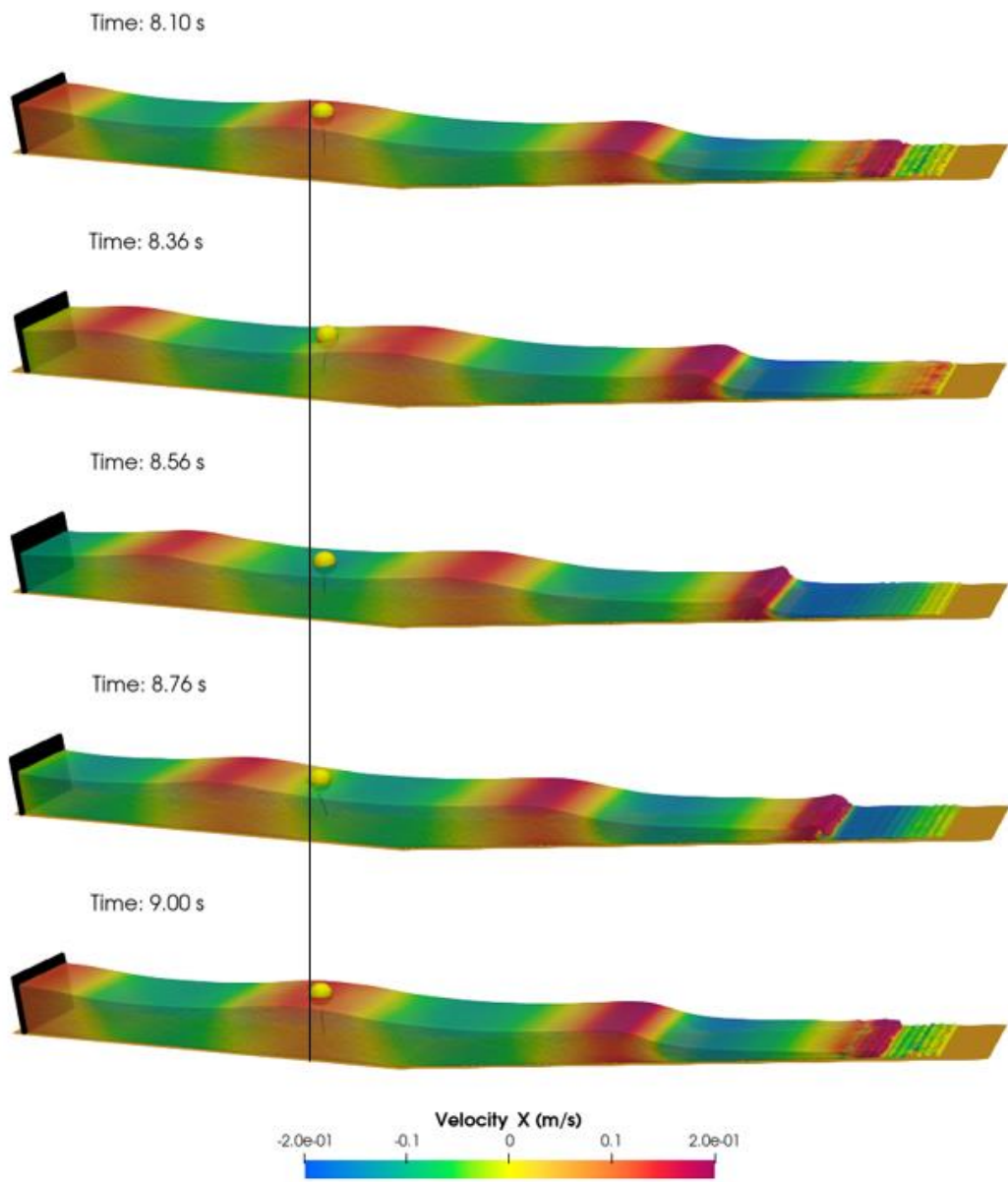


Figura 6.18 – Diferentes instantes de la simulación con DualSPHysics de una boya amarrada al fondo para la configuración de ola C ($T=0.9$ s, $H=2.67$ cm). La boya se muestra en amarillo y el amarre en negro, mientras que el color del fluido indica su velocidad longitudinal. La línea negra marca la posición de la boya en el instante $t=8.36$ s.

A continuación, se muestran las gráficas comparando los resultados experimentales y numéricos de la boya amarrada al fondo para diferentes condiciones de oleaje (Tabla 6.2). La Figura 6.19 muestra los resultados del primer caso con la boya amarrada bajo las condiciones de oleaje regular de la ola A ($T=0.7$ s, $H=4.22$ cm). Como se puede observar en el desplazamiento longitudinal, los resultados numéricos y experimentales son prácticamente iguales. Las oscilaciones son constantes en ambos casos lo que refleja el efecto del amarre. Se observa una amplitud menor en los resultados numéricos en comparación con los experimentales. En cuanto al desplazamiento vertical, se puede apreciar que los resultados numéricos presentan una correspondencia muy buena con los experimentales durante toda la simulación.

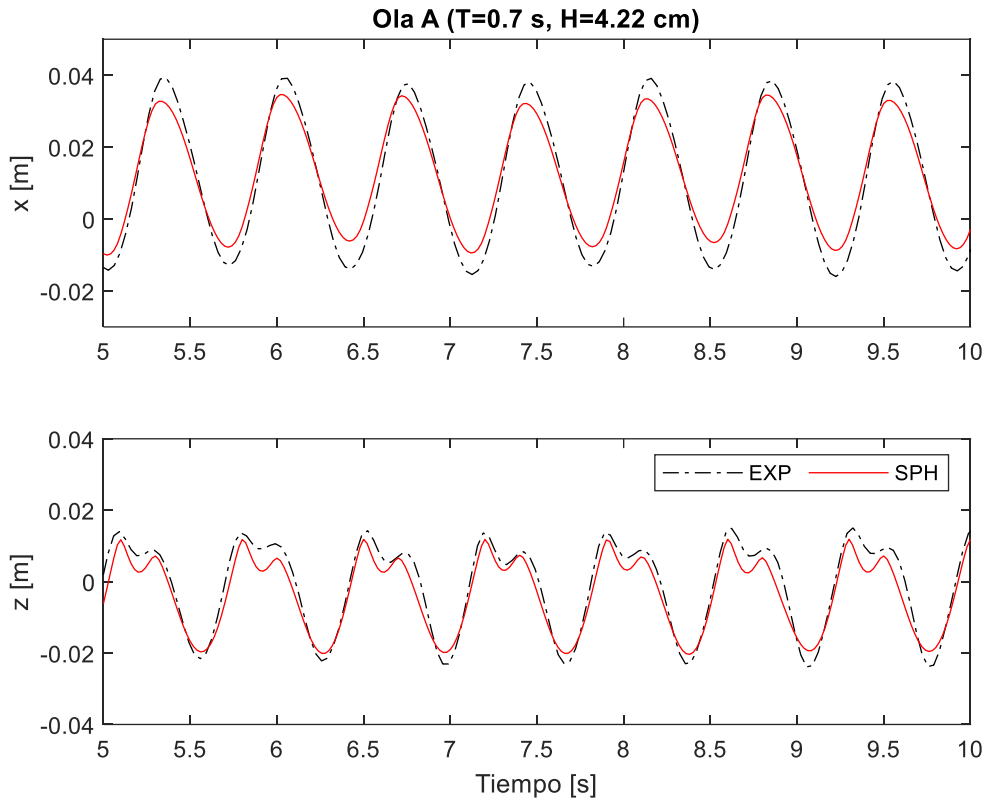


Figura 6.19 – Movimiento longitudinal y vertical de la boya amarrada usando la configuración de la ola A ($T=0.7$ s, $H=4.22$ cm).

Los resultados del segundo caso con el objeto flotante amarrado bajo las condiciones de oleaje correspondientes a la ola B ($T=0.9$ s, $H=3.11$ cm) se muestran en la Figura 6.20. En esta figura se puede observar que los resultados numéricos y experimentales del desplazamiento longitudinal muestran una ligera diferencia, del segundo 5.5 al segundo 7 los resultados numéricos se aproximan más a las crestas de los resultados experimentales y del segundo 7.5 al segundo 9 se aproximan más a los valles de los resultados experimentales. En cuanto al desplazamiento vertical, se puede apreciar que los resultados numéricos se ajustan muy bien con los experimentales en todo el dominio temporal.

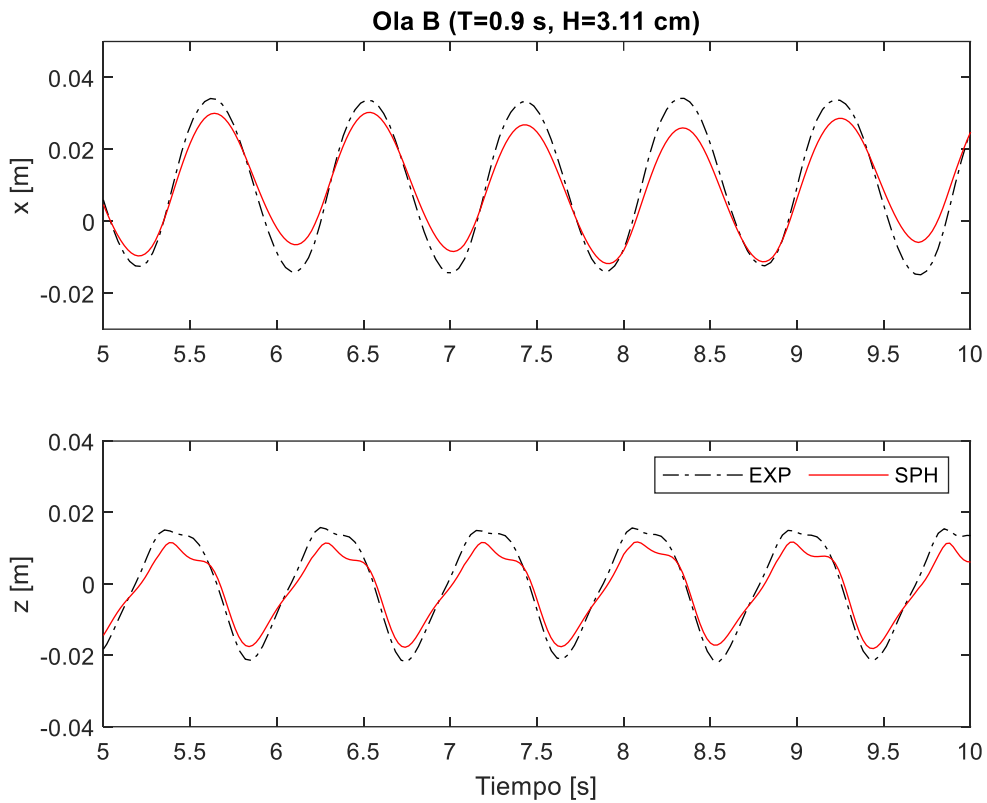


Figura 6.20 – Movimiento longitudinal y vertical de la boya amarrada usando la configuración de la ola B ($T=0.9$ s, $H=3.11$ cm).

La Figura 6.21 muestra los desplazamientos numéricos y experimentales de la boya utilizando la configuración de oleaje regular de la ola C ($T=0.9$ s, $H=2.67$ cm). El desplazamiento longitudinal de la boya muestra un buen ajuste entre resultados numéricos y experimentales en el período de la señal. Se puede ver como la amplitud del desplazamiento experimental se mantiene constante desde los 5 hasta los 10 segundos del experimento. En cuanto al desplazamiento vertical, los resultados numéricos y experimentales coinciden durante toda la simulación.

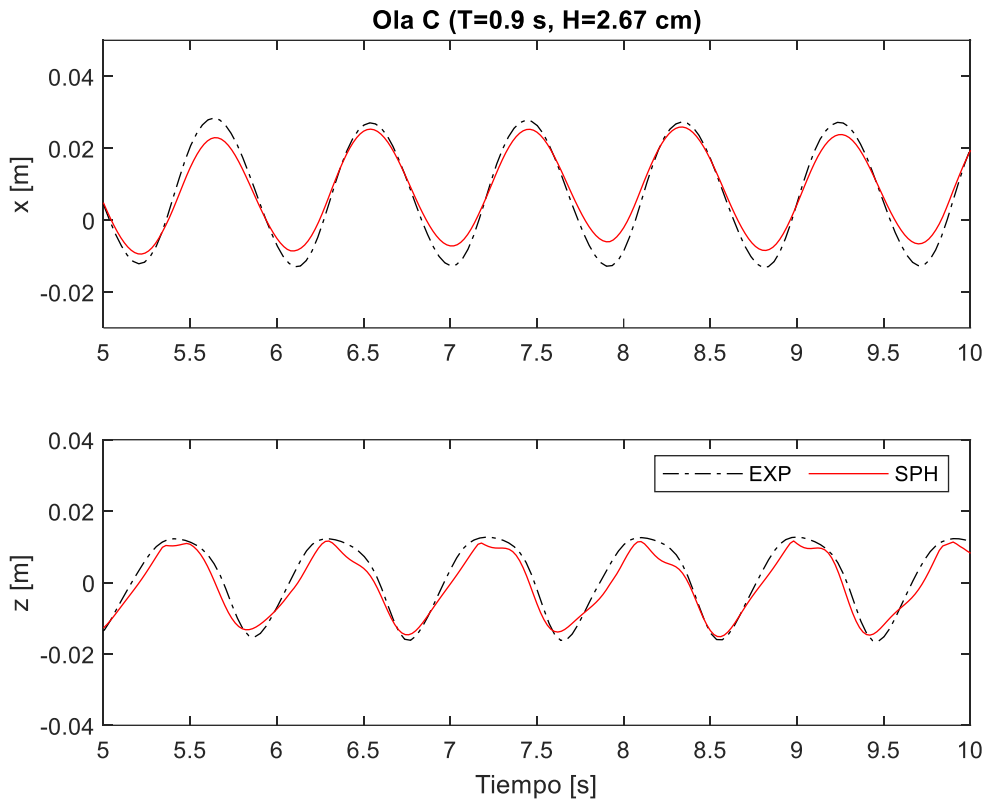


Figura 6.21 – Movimiento longitudinal y vertical de la boya amarrada usando la configuración de la ola C ($T=0.9$ s, $H=2.67$ cm).

La principal diferencia entre el comportamiento de la boya libre y cuando se sujeta con un amarre al fondo del canal se observa en el desplazamiento longitudinal. La boya amarrada no puede desplazarse más de 4 cm desde su posición original, pasando a tener un movimiento oscilatorio. La boya avanza y retrocede con la acción del oleaje, pero en la medida que lo permite la longitud de la línea de amarre. El amarre también afecta al desplazamiento vertical al reducir significativamente su amplitud. Estos resultados confirman que la línea de amarre limita los grados de libertad del objeto flotante, tal como se había dicho anteriormente.

Por último, en la Figura 6.22 se puede ver la variación temporal de las tensiones registradas en el punto de amarre para las simulaciones numéricas de las tres condiciones de oleaje. Se puede observar como las tensiones bajo la acción de la ola A tienen unos picos máximos alrededor de 1 N seguidos de otro menor de 0.5 N que se repiten de forma periódica con la llegada de las olas. Las tensiones para la ola B son menores y muestran picos de tensión más o menos periódicos con valores próximos a 0.35 N. Para la ola C la variación de la tensión en el amarre sigue siendo periódica pero los valores ya son mucho más bajos, siendo menores a 0.2 N durante toda la señal. Por lo tanto, y como era de esperar, los valores máximos de las tensiones en la línea de amarre aumentan con la altura de la ola.

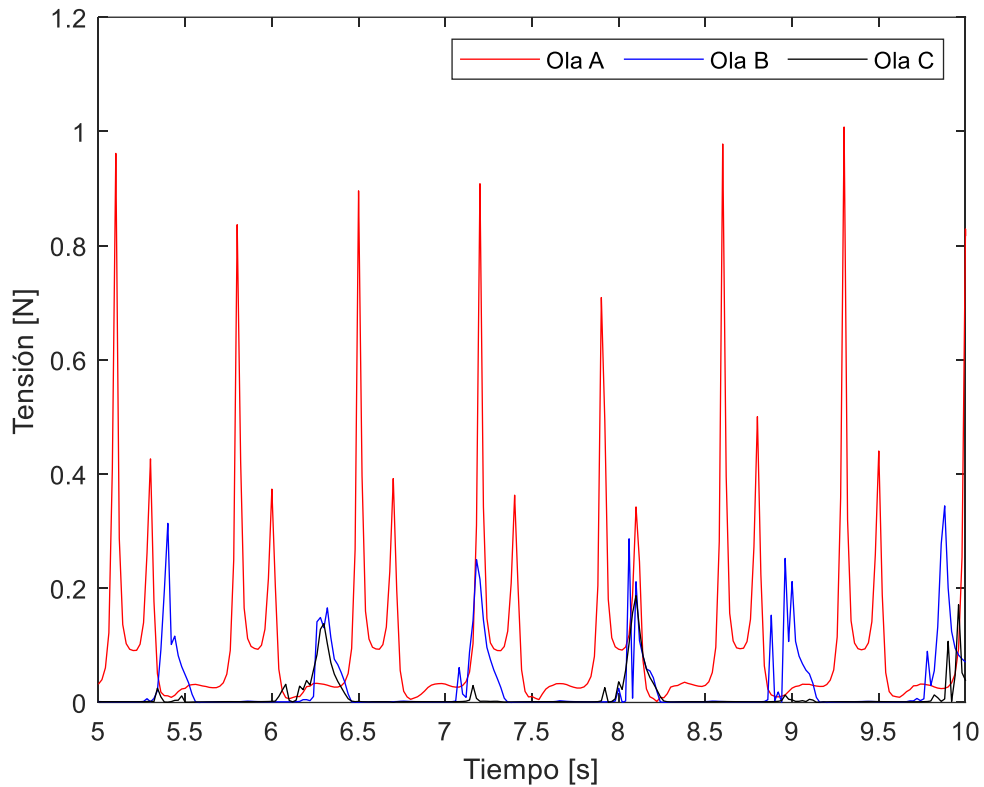


Figura 6.22 – Tensiones medidas en la línea de amarre para las tres condiciones de oleaje.

Los resultados mostrados en este apartado comparando los resultados numéricos con los experimentales muestran la capacidad de DualSPHysics y MoorDyn para simular objetos flotantes amarrados bajo condiciones de oleaje regular.

7 Conclusiones

Este trabajo intenta probar que el método numérico SPH es una alternativa viable a los métodos basados en mallas para simular un objeto flotante bajo condiciones de oleaje regular. En particular, se ha demostrado que DualSPHysics es capaz de generar, propagar y absorber las olas, y de reproducir la interacción entre estas olas y un objeto flotante libre o amarrado. Además, también es posible reproducir el comportamiento de las líneas de amarre con el acoplamiento de DualSPHysics a la librería MoorDyn.

Se ha realizado, además del modelado numérico, un trabajo experimental en el canal de oleaje nanoCIEM para poder así comparar los datos experimentales con los resultados de las simulaciones numéricas y validar el software DualSPHysics. Se han realizados varios experimentos distintos para tres condiciones de oleaje regular diferentes. Para el primer experimento, en el que la boya está libre, los desplazamientos longitudinales de la boya tanto numéricos como experimentales son realmente similares. Sin embargo, existen discrepancias en relación al desplazamiento vertical debido a la reflexión lateral y en la parte final del tanque experimental. Es decir, el canal nanoCIEM debe ser mejorado, en términos de reflexión, para estudiar la interacción entre olas y objetos flotantes libres. En el segundo experimento, con la boya atada a una línea de amarre, tanto los desplazamientos longitudinales como los verticales son precisos y muy similares comparando experimento y simulaciones con DualSPHysics. En este caso, la línea de amarre limita el movimiento de la boya y esto provoca que la componente transversal (perpendicular a la dirección de propagación) de la velocidad del fluido sea menor, lo que hace que la reflexión lateral sea más pequeña e influya menos. Además, se ha demostrado que aumentando la resolución en las simulaciones numéricas se obtienen unos resultados más precisos, muy próximos a los datos experimentales.

Por lo tanto, se ha alcanzado el objetivo planteado inicialmente, que era reproducir numéricamente el experimento físico y conseguir una buena correlación entre los resultados numéricos y experimentales. En cuanto a los problemas surgidos, la reflexión tanto longitudinal como lateral es una limitación del tanque experimental que en las simulaciones numéricas se puede solucionar fácilmente gracias al uso de zonas de amortiguamiento. También hay que comentar las limitaciones del modelado numérico, y es que para tener una resolución aceptable en 3D los tiempos de ejecución son elevados, ya que es necesario simular varios millones de partículas.

A lo largo de este estudio sólo se han simulado olas regulares, sin embargo, con DualSPHysics también es posible reproducir olas irregulares para reproducir unas condiciones oceánicas más realistas. En un futuro trabajo, será de interés el estudio de oleaje irregular y de unas condiciones de oleaje más extremas para ver el comportamiento del objeto flotante y de la línea de amarre, ya que los modelos SPH tienen un valor añadido para estudios de supervivencia.

Referencias

- Altomare C., Crespo A.J.C., Domínguez J.M., Gómez-Gesteira M., Suzuki T., Verwaest T. 2015. Applicability of Smoothed Particle Hydrodynamics for estimation of sea wave impact on coastal structures. *Coastal Engineering*, 96, 1-12. [DOI:10.1016/j.coastaleng.2014.11.001](https://doi.org/10.1016/j.coastaleng.2014.11.001).
- Altomare C., Domínguez J.M., Crespo A.J.C., González-Cao J., Suzuki T., Gómez-Gesteira M., Troch P. 2017. Long-crested wave generation and absorption for SPH-based DualSPHysics model. *Coastal Engineering*, 127, 37-54. [DOI:10.1016/j.coastaleng.2017.06.004](https://doi.org/10.1016/j.coastaleng.2017.06.004).
- Batchelor G.K. 1974. *Introduction to fluid dynamics*. Cambridge University Press. [DOI:10.1017/CBO9780511800955](https://doi.org/10.1017/CBO9780511800955).
- Bagnold R.A. 1974. Fluid forces on a body in shear-flow; experimental use of 'stationary flow'. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 340(1621), 147-171. [DOI:10.1098/rspa.1974.0145](https://doi.org/10.1098/rspa.1974.0145).
- Biesel F. and Suquet F. 1951. Les Appareils Générateurs de Houle en Laboratoire. *Houille Blanche-revue Internationale De L Eau*, 147-165. [DOI:10.1051/LHB%2F1951033](https://doi.org/10.1051/LHB%2F1951033).
- Bouscasse B., Colagrossi A., Marrone S., Antuono M. 2013. Nonlinear water wave interaction with floating bodies in SPH. *Journal of Fluids and Structures*, 42, 112-129. [DOI:10.1016/j.jfluidstructs.2013.05.010](https://doi.org/10.1016/j.jfluidstructs.2013.05.010).
- Boussinesq J. 1872. Théorie des ondes et des remous qui se propagent le long d'un canal rectangulaire horizontal, en communiquant au liquide contenu dans ce canal des vitesses sensiblement pareilles de la surface au fond. *Journal de Mathématiques Pures et Appliquées. Deuxième Série* 17, 55-108.
- Cartwright B.K., Chhor A., Groenenboom P.H. 2010. Numerical simulation of a helicopter ditching with emergency flotation devices. *Proceedings of 10th SPHERIC international workshop*, 98-105.
- Crespo A.J.C., Gómez-Gesteira M., Dalrymple R.A. 2007. Boundary Conditions Generated by Dynamic Particles in SPH Methods. *CMC: Computers, Materials, & Continua*, 5(3), 173-184. [DOI:10.3970/cmc.2007.005.173](https://doi.org/10.3970/cmc.2007.005.173).
- Crespo A.J.C., Domínguez J.M., Barreiro A., Gómez-Gesteira M. and Rogers B.D. 2011. GPUs, a new tool of acceleration in CFD: Efficiency and reliability on Smoothed Particle Hydrodynamics methods. *PLoS ONE* 6 (6), e20685. [DOI:10.1371/journal.pone.0020685](https://doi.org/10.1371/journal.pone.0020685).
- Domínguez J.M., Crespo A.J.C., Gómez-Gesteira M. 2013. Optimization strategies for CPU and GPU implementations of a smoothed particle hydrodynamics method. *Computer Physics Communications*, 184(3), 617-627. [DOI:10.1016/j.cpc.2012.10.015](https://doi.org/10.1016/j.cpc.2012.10.015).

Domínguez J.M., Crespo A.J.C., Hall M., Altomare C., Wu M., Stratigaki V., Troch P., Capietti L., Gómez-Gesteira M. 2019. SPH simulation of floating structures with moorings. *Coastal Engineering*, 153, 103560. [DOI:10.1016/j.coastaleng.2019.103560](https://doi.org/10.1016/j.coastaleng.2019.103560).

Domínguez J.M., Fourtakas G., Altomare C., Canelas R.B., Tafuni A., García-Feal O., Martínez-Estévez I., Mokos A., Vacondio R., Crespo A.J.C., Rogers B.D., Stansby P.K., Gómez-Gesteira M. 2021. DualSPHysics: from fluid dynamics to multiphysics problems. *Computational Particle Mechanics*. [DOI:10.1007/s40571-021-00404-2](https://doi.org/10.1007/s40571-021-00404-2).

Eagleson P., Dean R. 1966. *Small Amplitude Wave Theory*. McGraw-Hill, New York.

English A., Domínguez J.M., Vacondio R., Crespo A.J.C., Stansby P.K., Lind S.J., Chiapponi L., Gómez-Gesteira M. 2021. Modified dynamic boundary conditions (mDBC) for general-purpose smoothed particle hydrodynamics (SPH): application to tank sloshing, dam break and fish pass problems. *Computational Particle Mechanics*. [DOI:10.1007/s40571-021-00403-3](https://doi.org/10.1007/s40571-021-00403-3).

Gingold R. and Monaghan J. 1977. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181 (3), 375–389. [DOI:10.1093/mnras/181.3.375](https://doi.org/10.1093/mnras/181.3.375).

Green M.D., Zhou Y., Dominguez J.M., Gesteira M.G., Peiró J. 2021. Smooth particle hydrodynamics simulations of long-duration violent three-dimensional sloshing in tanks. *Ocean Engineering*, 229, 108925. [DOI:10.1016/j.oceaneng.2021.108925](https://doi.org/10.1016/j.oceaneng.2021.108925).

Groenenboom P., Campbell J., Benítez Montañés L., Siemann, M. 2014. Innovative SPH methods for aircraft ditching. *Proceedings of WCCM XI–ECCM V–ECFD VI*, 4555-4566.

Härdzić I., Hennig J., Peric M., Xing-Kaeding Y. 2005. Computation of flow-induced motion of floating bodies. *Applied Mathematical Modelling*, 29, 1196–1210. [DOI:10.1016/j.apm.2005.02.014](https://doi.org/10.1016/j.apm.2005.02.014).

Keller J.B. 1948. The solitary wave and periodic waves in shallow water. *Annals of the New York Academy of Sciences*, 51(3), 345-350. [DOI:10.1002/cpa.3160010402](https://doi.org/10.1002/cpa.3160010402).

Keulegan G.H., Patterson G.W. 1940. Mathematical theory of irrotational translation waves. *J. Res. Natl. Bureau Stand.*, 24, 47-101

Korteweg D.J. and De Vries G. 1895. On the Change of Form of Long Waves Advancing in a Rectangular Canal and a New Type of Long Stationary Waves. *Philosophical Magazine Series*, 39, 422-443. [DOI:10.1080/14786449508620739](https://doi.org/10.1080/14786449508620739).

Le Méhauté, B. 2013. *An introduction to hydrodynamics and water waves*. Springer Science & Business Media. [DOI:10.1007/978-3-642-85567-2](https://doi.org/10.1007/978-3-642-85567-2).

Liu G.R. 2003. *Mesh Free methods: Moving beyond the finite element method*, CRC Press. [DOI:10.1115/1.1553432](https://doi.org/10.1115/1.1553432).

- Longshaw S.M. and Rogers B.D. 2015. Automotive fuel cell sloshing under temporally and spatially varying high acceleration using GPU-based Smoothed Particle Hydrodynamics (SPH). *Advances in Engineering Software*, 83, 31-44. [DOI:10.1016/j.advengsoft.2015.01.008](https://doi.org/10.1016/j.advengsoft.2015.01.008).
- Lucy L. 1977. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal*, 82, 1013–1024. [DOI:10.1086/112164](https://doi.org/10.1086/112164).
- Madsen O.S. 1971. On the generation of long waves. *J. Geophys. Res.* 76, 8672–8683. [DOI:10.1029/JC076i036p08672](https://doi.org/10.1029/JC076i036p08672).
- Marrone S., Antuono M.A.G.D., Colagrossi A., Colicchio G., Le Touzé D., Graziani G. 2011. δ -SPH model for simulating violent impact flows. *Computer Methods in Applied Mechanics and Engineering*, 200(13-16), 1526-1542. [DOI:10.1016/j.cma.2010.12.016](https://doi.org/10.1016/j.cma.2010.12.016).
- Marrone S., Bouscasse B., Colagrossi A., Antuono M. 2012. Study of ship wave breaking patterns using 3D parallel SPH simulations. *Computers & Fluids*, 69, 54-66. [DOI:10.1016/j.compfluid.2012.08.008](https://doi.org/10.1016/j.compfluid.2012.08.008).
- McCowan J. 1891. VII. On the solitary wave. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 32(194), 45-58.
- Miche R. 1944. Mouvements ondulatoires des mers en profondeur constante ou décroissant. *Annals des Points et Chaussées*, 164-270.
- Mogan S.C., Chen D., Hartwig J.W., Sahin I., Tafuni A. 2018. Hydrodynamic analysis and optimization of the Titan submarine via the SPH and Finite-Volume methods. *Computers & Fluids*, 74, 271-282. [DOI:10.1016/j.compfluid.2018.08.014](https://doi.org/10.1016/j.compfluid.2018.08.014).
- Monaghan J. 1992. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics* 30, 543- 574. [DOI:10.1146/annurev.aa.30.090192.002551](https://doi.org/10.1146/annurev.aa.30.090192.002551).
- Monaghan J. 1994. Simulating Free Surface Flows with SPH. *Journal of Computational Physics*, 110(2), 399–406. [DOI:10.1006/jcph.1994.1034](https://doi.org/10.1006/jcph.1994.1034).
- Monaghan J., Kos A. 1999. Solitary waves on a Cretan beach. *Journal of Waterway, Port, Coastal and Ocean Engineering*, 125, 145-154. [DOI:10.1061/\(ASCE\)0733-950X\(1999\)125:3\(145\)](https://doi.org/10.1061/(ASCE)0733-950X(1999)125:3(145)).
- Monaghan J., Kos A., Issa N. 2003. Fluid motion generated by impact. *Journal of Waterway, Port, Coastal and Ocean Engineering* 129, 250-259. [DOI:10.1061/\(ASCE\)0733-950X\(2003\)129:6\(250\)](https://doi.org/10.1061/(ASCE)0733-950X(2003)129:6(250)).
- Monaghan J. 2005. Smoothed Particle Hydrodynamics. *Rep. Prog. Phys*, 68, 1703-1759. [DOI:10.1088/0034-4885/68/8/R01](https://doi.org/10.1088/0034-4885/68/8/R01).
- Munk W.H. 1949. The solitary wave theory and its application to surf problems. *Annals of the New York Academy of Sciences*, 51(3), 376-424. [DOI:10.1111/j.1749-6632.1949.tb27281.x](https://doi.org/10.1111/j.1749-6632.1949.tb27281.x).
- Packard M.M. 2013. Modeling floating bodies and fluid flow interactions using Smooth Particle Hydrodynamics. San Antonio College, San Antonio, Texas 78212.

Raudkivi A.J. and Witte H.H. 1990. Development of bed features. Journal of Hydraulic Engineering (ASCE), 116 (9), 1063-1079. [DOI:10.1061/\(ASCE\)0733-9429\(1990\)116:9\(1063\)](https://doi.org/10.1061/(ASCE)0733-9429(1990)116:9(1063)).

Shadloo M.S., Oger G., Le Touzé D. 2016. Smoothed particle hydrodynamics method for fluid flows, towards industrial applications: Motivations, current state, and challenges. Computers & Fluids, 136, 11-34. [DOI:10.1016/j.compfluid.2016.05.029](https://doi.org/10.1016/j.compfluid.2016.05.029).

Stokes G.G. 1880. On the theory of oscillatory waves. Transactions of the Cambridge philosophical society.

Storti M.A., D'Elía J. 2004 Added Mass of an Oscillating Hemisphere at Very-Low and Very-High Frequencies. Journal of Fluids Engineering 126. [DOI:10.1115/1.1839932](https://doi.org/10.1115/1.1839932).

Tafuni A., Domínguez J.M., Vacondio R., Crespo, A.J.C. 2018. A versatile algorithm for the treatment of open boundary conditions in Smoothed particle hydrodynamics GPU models. Computer Methods in Applied Mechanics and Engineering, 342, 604-624. [DOI:10.1016/j.cma.2018.08.004](https://doi.org/10.1016/j.cma.2018.08.004).

Tagliafierro B., Mancini S., Roperio-Giralda P., Domínguez J.M., Crespo A.J.C., Viccione G. 2021. Performance Assessment of a Planing Hull Using the Smoothed Particle Hydrodynamics Method. Journal of Marine Science and Engineering, 9(3), 244. [DOI:10.3390/jmse9030244](https://doi.org/10.3390/jmse9030244).

Toffoli A., Bitner-Gregersen E. 2015. Types of ocean surface waves, wave classification. Encyclopedia of Maritime and Offshore Engineering. 2017 John Wiley & Sons, Ltd. [DOI:10.1002/9781118476406.emoe077](https://doi.org/10.1002/9781118476406.emoe077).

Violeau D. 2012. Fluid Mechanics and the SPH Method: Theory and Applications. Oxford University Press. [DOI:10.1093/acprof:oso/9780199655526.001.0001](https://doi.org/10.1093/acprof:oso/9780199655526.001.0001).

Vu B.T., Berg J.J., Harris M.F., Crespo A.J.C. 2015. Launch Environment Water Flow Simulations Using Smoothed Particle Hydrodynamics. Proceedings of the 53rd AIAA Aerospace Sciences Meeting. [DOI:10.2514/6.2015-0049](https://doi.org/10.2514/6.2015-0049).

Wendland H. 1995. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. Advanced in Computational Mathematics 4, 389-396. [DOI:10.1007/BF02123482](https://doi.org/10.1007/BF02123482).

Zhang F., Crespo A.J.C., Altomare C., Domínguez J.M., Marzeddu A., Shang S., Gómez-Gesteira M. 2018. DualSPHysics: a numerical tool to simulate real breakwaters. Journal of Hydrodynamics, 30(1), 99-105. [DOI:10.1007/s42241-018-0010-0](https://doi.org/10.1007/s42241-018-0010-0).

H. Karadeniz. 2012. Stochastic analysis of offshore steel structures: an analytical appraisal.

Apéndice

Este apéndice incluye el archivo de configuración XML utilizado para ejecutar en DualSPHysics el caso del objeto flotante amarrado correspondiente a la ola C (T=0.9 s y H=2.67 cm).

```
<?xml version="1.0" encoding="UTF-8" ?>
<case>
  <casedef>
    <constantsdef>
      <mkconfig boundcount="230" fluidcount="9">
        <geometry>
        <floatings>
        <normals>
        <motion>
      </casedef>
    <execution>
      <special>
        <wavepaddles>
        <initialize>
        <moorings>
        <damping>
      </special>
      <parameters>
    </execution>
  </case>
```

1. Definición de constantes y configuración del mk

```
<constantsdef>
  <gravity x="0" y="0" z="-9.81" comment="Gravitational acceleration" units_comment="m/s^2" />
  <rho0 value="1000" comment="Reference density of the fluid" units_comment="kg/m^3" />
  <hswl value="0" auto="true" comment="Maximum still water level to calculate speedofsound using coefsound"
  units_comment="metres (m)" />
  <gamma value="7" comment="Polytropic constant for water used in the state equation" />
  <speedsystem value="0" auto="true" comment="Maximum system speed (by default the dam-break propagation is used)" />
  <coefsound value="20" comment="Coefficient to multiply speedsystem" />
  <speedsound value="0" auto="true" comment="Speed of sound to use in the simulation
  (by default speedofsound=coefsound*speedsystem)" />
  <coefh value="1.5" comment="Coefficient to calculate the smoothing length (h=coefh*sqrt(3*dp^2) in 3D)" />
  <cflnumber value="0.2" comment="Coefficient to multiply dt" />
</constantsdef>
<mkconfig boundcount="230" fluidcount="9">
  <mkorientfluid mk="0" orient="Xyz" />
</mkconfig>
```

2. Definición de la geometría

```

<geometry>
  <definition dp="0.002">
    <pointmin x="-1" y="-0.1" z="-1" />
    <pointmax x="4" y="0.1" z="2" />
  </definition>
  <commands>
    <list name="GeometryForNormals">
      <setactive drawpoints="0" drawshapes="1"/>
      <setshapemode>actual | bound</setshapemode>
      <setmkbound mk="0" />
      <drawextrude closed="false">
        <extrude x="0" y="2" z="0" />
        <point x="-0.2" y="-1" z="-0.132" />
        <point x="1.2" y="-1" z="-0.132" />
        <point x="2.8" y="-1" z="0.03" />
        <layers vdp="0.5" />
      </drawextrude>
      <shapeout file="hdp" />
      <!-- Sphere (only for visualisation) -->
      <setmkbound mk="15" />
      <setfrdrawmode auto="true" />
      <drawsphere radius="#0.0616/2">
        <point x="0.925" y="0" z="0" />
      </drawsphere>
      <setfrdrawmode auto="false" />
      <shapeout file="VIS_SPHERE" />
      <resetdraw />
    </list>
    <mainlist>
      <!-- Actual geometry at dp/2 -->
      <runlist name="GeometryForNormals" />
      <!-- Particle generation -->
      <setshapemode>actual | dp | bound</setshapemode>
      <setdrawmode mode="full" />

      <!-- Piston -->
      <setmkbound mk="10" />
      <drawbox>
        <boxfill>solid</boxfill>
        <point x="#0-5*Dp" y="-1" z="-0.132" />
        <size x="#5*Dp" y="2" z="0.3" />
      </drawbox>
      <!-- Tank -->
      <setmkbound mk="0" />
      <setfrdrawmode auto="false" />
      <drawextrude closed="false">
        <extrude x="0" y="2" z="0" />
        <point x="-0.2" y="-1" z="-0.132" />
        <point x="1.2" y="-1" z="-0.132" />
        <point x="2.8" y="-1" z="0.03" />
        <layers vdp="0,-1,-2,-3,-4" />
      </drawextrude>
      <setfrdrawmode auto="false" />
      <!-- Sphere -->
      <setmkbound mk="15" />
      <setfrdrawmode auto="true" />
      <drawsphere radius="#0.0616/2-Dp/2">
        <point x="0.925" y="0" z="0" />
      </drawsphere>
      <setfrdrawmode auto="false" />
      <!-- Fluid -->
      <setmkfluid mk="0" />
      <fillbox x="0.5" y="0" z="-0.1">
        <modefill>void</modefill>
        <point x="-1" y="-1" z="-1" />
        <size x="5" y="2" z="1" />
      </fillbox>
      <shapeout file="" reset="true" />
    </mainlist>
  </commands>
</geometry>

```

3. Objeto flotante

```
<floatings>
  <floating mkbound="15">
    <massbody value="0.06078" />
  </floating>
</floatings>
```

4. Configuración condiciones de contorno mDBC

```
<normals>
  <distanceh value="2.0" comment="Maximum distance (H*distanceh) to compute normals data (default=2)" />
  <ftsupport value="1" />
  <geometryfile file="[CaseName]_hdp_Actual.vtk" comment="File with boundary geometry (VTK format)" />
</normals>
```

5. Movimiento del pistón

```
<motion>
  <objreal ref="10">
    <begin mov="1" start="0" />
    <mvnull id="1" />
  </objreal>
</motion>
```

6. Generador de olas

```
<wavepaddles>
  <piston>
    <mkbound value="10" comment="Mk-Bound of selected particles" />
    <waveorder value="2" comment="Order wave generation 1:1st order, 2:2nd order (def=1)" />
    <start value="0" comment="Start time (def=0)" />
    <duration value="0" comment="Movement duration, Zero is the end of simulation (def=0)" />
    <depth value="0.130" comment="Fluid depth (def=0)" />
    <pistondir x="1" y="0" z="0" comment="Movement direction (def=(1,0,0))" />
    <waveheight value="0.0267" comment="Wave height" />
    <waveperiod value="0.9" comment="Wave period" />
    <phase value="0" comment="Initial wave phase in function of PI (def=0)" />
    <ramp value="1" comment="Periods of ramp (def=0)" />
    <savemotion periods="20" periodsteps="20" xpos="1.03" zpos="-0.05" comment="Saves motion data.
    xpos and zpos are optional. zpos=-depth of the measuring point" />
  </piston>
</wavepaddles>
```

7. Configuración condiciones de contorno mDBC

```
<initialize>
  <boundnormal_plane mkbound="10">
    <point auto="true" comment="Point is calculated automatically according to normal configuration." />
    <normal x="1" y="0" z="0" />
    <maxdisth v="0" comment="Maximum distance to boundary limit. It uses H*maxdisth (default=2)" />
  </boundnormal_plane>
  <boundnormal_sphere mkbound="15">
    <center x="0.925" y="0" z="0" />
    <radius v="#0.0616/2" />
    <inside v="true" comment="Boundary particles inside the cylinder" />
    <maxdisth v="2.0" comment="Maximum distance to boundary limit. It uses H*maxdisth (default=2)" />
  </boundnormal_sphere>
</initialize>
```

8. Configuración de la línea de amarre (librería MoorDyn)

```

<moorings>
  <savevtk_moorings value="true" comment="Saves vtk with moorings (default=true)" />
  <savecsv_points value="true" comment="Saves csv with link points (default=true)" />
  <savevtk_points value="false" comment="Saves vtk with link points (default=false)" />
  <mooredfloatings>
    <floating mkbound="15" comment="Mkbound of the Floating body the mooring is linked to" />
  </mooredfloatings>
  <!-- <moordyn file="moordyn.xml" /> % Loads the configuration of the file indicates in attribute "file" -->
  <moordyn comment="MoorDyn configuration"> %Loads the next configuration
    <solverOptions>
      <waterDepth value="0.130" comment="Water depth" units_comment="m"/>
      <freesurface value="0" comment="Z position of the water free surface.(default=0)" />
      <kBot value="3.0e6" comment="Bottom stiffness constant. (default=3.0e6)" units_comment="Pa/m"/>
      <cBot value="3.0e5" comment="Bottom damping constant. (default=3.0e5)" units_comment="Pa*s/m" />
      <dtM value="0.000001" auto="false" comment="Desired mooring model time step. (default=0.0001)"/>
      <waveKin value="0" comment="Wave kinematics flag ( 0: neglect [the only option currently supported] ). (default=0)"/>
      <writeUnits value="yes" comment="Write units line. value=[yes|no]. (default=yes)"/>
      <frictionCoefficient value="0" comment="General bottom friction coefficient, as a start. (default=0.0)"/>
      <fricDamp value="200" comment="Damping coefficient used to model the friction with speeds near zero. (default=200.0)"/>
      <statDynFricScale value="1.0" comment="Ratio between static and dynamic friction (mu_static/mu_dynamic). (default=1.0)"/>
      <dtIC value="1.0" comment="Period to analyse convergence of dynamic relaxation for initial conditions. (default=1.0)" units_comment="s"/>
      <cdScaleIC value="2" comment="Factor to scale drag coefficients during dynamic relaxation for initial conditions. (default=5)"/>
      <threshIC value="0.001" comment="Convergence threshold for for initial conditions. (default=0.001)"/>
      <tmaxIC value="1" comment="Maximum time for initial conditions without convergence. (default=0)" units_comment="s"/>
    </solverOptions>
    <bodies>
      <body ref="15" comment="Floating driven structure to attach mooring lines."/>
    </bodies>

    <lines>
      <linedefault comment="Shared properties for each line.">
        <e value="1.59e9" comment="Line stiffness, product of elasticity modulus and cross-sectional area." units_comment="N"/>
        <diameter value="2e-3" comment="Volume-equivalent diameter of the line." units_comment="m"/>
        <massDenInAir value="0.00077" comment="Mass per unit length of the line." units_comment="kg/m"/>
        <ba value="-0.8" comment="Line internal damping (BA/-zeta). (default=-0.8)" units_comment="Ns"/>
        <can value="1.0" comment="Transverse added mass coefficient (with respect to line displacement). (default=1.0)"/>
        <cat value="0.0" comment="Tangential added mass coefficient (with respect to line displacement). (default=0.0)"/>
        <cdn value="1.6" comment="Transverse drag coefficient (with respect to frontal area, d*1). (default=1.6)"/>
        <cdt value="0.05" comment="Tangential drag coefficient (with respect to surface area, n*d*1). (default=0.05)"/>
        <outputFlags value="p" comment="Node output properties. (default=-) [-:None|p:Positions|v:Velocities|U:Wave Velocities|t:Tension|D:Hydrodynamic Drag Force|c=Internal Damping|s:Strain of each segment|d:rate of strain of each segment]"/>
      </linedefault>
      <line> %line 0
        <vesselconnection bodyref="15" x="0.925" y="0.0" z="-0.0368" />
        <fixconnection x="0.925" y="0.0" z="-0.1" />
        <length value="0.075" comment="(m)"/>
        <segments value="5" />
      </line>
    </lines>
    <savedata comment="Saves CSV with the connection properties." >
      <tension value="true" comment="Stores tensions at the connections.(default: value=false)"/>
      <force value="true" comment="Stores forces at the connections.(default: value=false)"/>
      <velocity value="true" comment="Stores velocities at the connections.(default: value=false)"/>
      <position value="true" comment="Stores positions at the connections.(default: value=false)"/>
    </savedata>
  </moordyn>
</moorings>

```


9. Amortiguamiento lateral

```
<damping>
  <!-- lateral limits -->
  <dampingzone>
    <limitmin x="0" y="-0.095" z="0" />
    <limitmax x="0" y="-0.1" z="0" />
    <overlimit value="1" />
    <redumax value="10" />
    <factorxyz x="0" y="1" z="0" comment="Application factor in components (def x=1,y=1,z=1)" />
  </dampingzone>
  <!-- lateral limits -->
  <dampingzone>
    <limitmin x="0" y="0.095" z="0" />
    <limitmax x="0" y="0.1" z="0" />
    <overlimit value="1" />
    <redumax value="10" />
    <factorxyz x="0" y="1" z="0" comment="Application factor in components (def x=1,y=1,z=1)" />
  </dampingzone>
</damping>
```

10. Parámetros de simulación SPH

```
<parameters>
  <parameter key="Boundary" value="2" comment="Boundary method 1:DBC, 2:mDBC (default=1)" />
  <parameter key="SavePosDouble" value="1" comment="Saves particle position using double precision (default=0)" />
  <parameter key="StepAlgorithm" value="2" comment="Step Algorithm 1:Verlet, 2:Symplectic (default=1)" />
  <parameter key="VerletSteps" value="40" comment="Verlet only: Number of steps to apply Euler timestepping (default=40)" />
  <parameter key="Kernel" value="2" comment="Interaction Kernel 1:Cubic Spline, 2:Wendland (default=2)" />
  <parameter key="ViscoTreatment" value="1" comment="Viscosity formulation 1:Artificial, 2:Laminar+SPS (default=1)" />
  <parameter key="Visco" value="0.01" comment="Viscosity value" /> % Note alpha can depend on the resolution. A value of 0.01 is recommended for near irrotational flows.
  <parameter key="ViscoBoundFactor" value="1" comment="Multiply viscosity value with boundary (default=1)" />
  <parameter key="DensityDT" value="3" comment="Density Diffusion Term 0:None, 1:Molteni, 2:Fourtakas, 3:Fourtakas (full) (default=0)" />
  <parameter key="DensityDTvalue" value="0.1" comment="DDT value (default=0.1)" />
  <parameter key="Shifting" value="0" comment="Shifting mode 0:None, 1:Ignore bound, 2:Ignore fixed, 3:Full (default=0)" />
  <parameter key="ShiftCoef" value="-2" comment="Coefficient for shifting computation (default=-2)" />
  <parameter key="ShiftTFS" value="0" comment="Threshold to detect free surface. Typically 1.5 for 2D and 2.75 for 3D (default=0)" />
  <parameter key="RigidAlgorithm" value="1" comment="Rigid Algorithm 0:collision-free, 1:SPH, 2:DEM, 3:Chrono (default=1)" />
  <parameter key="FtPause" value="0.0" comment="Time to freeze the floatings at simulation start (warmup) (default=0)" units_comment="seconds" />
  <parameter key="CoefDtMin" value="0.05" comment="Coefficient to calculate minimum time step dtmin=coefdtmin*h/speedsound (default=0.05)" />
  <parameter key="DtIni" value="0" comment="Initial time step. Use 0 to default use (default=h/speedsound)" units_comment="seconds" />
  <parameter key="DtMin" value="0" comment="Minimum time step. Use 0 to default use (default=coefdtmin*h/speedsound)" units_comment="seconds" />
  <parameter key="DtFixed" value="0" comment="Fixed Dt value. Use 0 to disable (default=disabled)" units_comment="seconds" />
  <parameter key="DtFixedFile" value="NONE" comment="Dt values are loaded from file. Use NONE to disable (default=disabled)" units_comment="milliseconds (ms)" />
  <parameter key="DtAllParticles" value="0" comment="Velocity of particles used to calculate DT. 1:All, 0:Only fluid/floatings (default=0)" />
  <parameter key="TimeMax" value="15" comment="Time of simulation" units_comment="seconds" />
  <parameter key="TimeOut" value="0.02" comment="Time out data" units_comment="seconds" />
  <parameter key="PartsOutMax" value="1" comment="%/100 of fluid particles allowed to be excluded from domain (default=1)" units_comment="decimal" />
  <parameter key="RhopOutMin" value="700" comment="Minimum rhop valid (default=700)" units_comment="kg/m^3" />
  <parameter key="RhopOutMax" value="1300" comment="Maximum rhop valid (default=1300)" units_comment="kg/m^3" />
  <parameter key="YPeriodicIncZ" value="0.0" comment="Increase of Z with periodic BC" units_comment="metres (m)" />
  <simulationdomain comment="Defines domain of simulation (default=Uses minimum and maximum position of the generated particles)">
    <posmin x="default" y="default" z="default" comment="e.g.: x=0.5, y=default-1, z=default-10%" />
    <posmax x="default" y="default" z="default" />
  </simulationdomain>
</parameters>
```