



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE CIENCIAS

DETECCIÓN DE VÓRTICES POR DEFORMACIÓN
DE LA SUPERFICIE LIBRE

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

FÍSICO

P R E S E N T A :

IÑAKI SÁNCHEZ CALVO GONZÁLEZ

DR. GERARDO RUÍZ CHAVARRÍA

CIUDAD UNIVERSITARIA, CD. MX.

2018



FACULTAD DE CIENCIAS
UNAM



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*Dedicado a mis padres por su apoyo incondicional, a mi hermana y
sus hijos Catilucas y MarianoBot.*

Agradecimientos

Deseo agradecer al gran inventor por hacer todo tan interesante y tan extenso, nunca me aburriré ni dejaré de asombrarme. A mis padres por patrocinar mis sueños y presionarme para no abandonarlos. Al Dr. Gerardo Ruíz Chavarría por aceptar dirigir mi tesis, por sus enseñanzas y paciencia en las clases que tomé con él. A la Universidad Nacional Autónoma de México por la gran educación de calidad. A la Facultad de Ciencias por reunir a tantos científicos de gran talla que ayudaron a forjarme. Al Dr. Érick Javier López Sánchez, por su orientación, ideas y trabajo en el que se basa esta tesis. A mis compañeros de laboratorio: M. en C. Sergio Hernández Zapata, Pamela Villamil Sapién, José Antonio Trejo Gutiérrez, Luis Alejandro Urizar Valle y Tomás Rodríguez Luna, quienes fueron añadiendo ideas. A DGAPA-UNAM proyecto PAPIIT IN116312 "Vorticidad y ondas no lineales en fluidos". A Google por ayudarme a encontrar y Wikipedia por calmar mi curiosidad.

Índice general

| | |
|---|-----------|
| 1. Fundamentos de la Dinámica de Fluidos | 1 |
| 1.1. Transporte | 1 |
| 1.2. Número de Reynolds y de Strouhal | 2 |
| 1.3. Descripción de los fluidos | 3 |
| 1.4. Componentes locales del gradiente campo de velocidades | 4 |
| 1.5. Ecuaciones locales de la dinámica de fluidos | 5 |
| 1.6. Ecuaciones del movimiento para un fluido | 6 |
| 1.7. Ecuación de flujo de Navier-Stokes para un fluido Newtoniano | 7 |
| 1.8. Ecuación de Bernoulli | 7 |
| 1.9. Capa límite y formación de vorticidad | 9 |
| 2. Dinámica de vórtices | 12 |
| 2.1. Análogo electrodinámico | 12 |
| 2.2. Energía cinética del vórtice | 15 |
| 2.3. La ecuación de transporte de vorticidad | 17 |
| 2.4. Teorema de Kelvin: la conservación de la circulación | 17 |
| 2.5. Dinámica de un sistema de vórtices lineares paralelos | 18 |
| 3. Método de schlieren sintético | 20 |
| 3.1. Imagen refractada a través de la superficie plana | 22 |
| 3.2. Imagen refractada a través de la superficie deformada | 23 |

| | |
|---|-----------|
| 4. Desarrollo experimental | 28 |
| 4.1. Dispositivo experimental | 28 |
| 4.2. Descripción de programas | 31 |
| 5. Resultados y Conclusiones | 34 |
| 5.1. Resultados | 34 |
| 5.2. Análisis de resultados | 41 |
| 5.3. Conclusiones | 43 |
| A. Script de linux | 46 |
| B. Programa en Matlab buscador de vórtices | 48 |
| C. Programa en Matlab filtrador de centros de vórtices | 51 |
| D. Programa en Matlab promediador | 56 |
| E. Programa en Matlab graficador | 65 |

Introducción

En este trabajo se estudian vórtices que tienen contacto con una superficie libre. Los vórtices están presentes en los flujos geofísicos en una gran variedad de escalas. Ejemplos de esto son los vórtices formados cuando el agua fluye entre Bahías y el mar, conectados por un canal¹, son vórtices formados por agua cálida y salada que sale del mar Mediterráneo hacia el Atlántico a través del estrecho de Gibraltar, de estructura aplanada que giran en sentido de las manecillas del reloj (anticiclónicos) y se ubican a profundidades medias del océano. Otros ejemplos de vórtices geofísicos son los de Naruto en Japón, Old Sow en Deer Island de New Brunswick, Canadá, los cuales se forman por los flujos de marea. Para detectarlos se usa la propiedad que el núcleo del vórtice es una región de baja presión, por lo que se produce una deformación de la superficie libre. El método que se utilizó para detectar un vórtice es conocido como schlieren sintético, desarrollado recientemente para el estudio de ondas de superficie de pequeña amplitud. En esta tesis se hizo la medición de posiciones y velocidades de traslación de los vórtices del dipolo formado frente a un canal que une dos reservorios. El método de schlieren sintético usa los avances tecnológicos recientes, principalmente la aparición de cámaras digitales de alta resolución. Este procedimiento fue desarrollado en el Laboratorio FAST (Fluides, Automatique et Systèmes Thermiques) [5], el método permite reconstruir la topografía de una superficie libre por medio de la refracción de la luz, al colocar un patrón de puntos al azar en el fondo del líquido. La deformación de la superficie da lugar a un movimiento aparente de los puntos, éste desplazamiento aparente es proporcional al gradiente de la superficie. Para medir los desplazamientos aparentes se utiliza un procedimiento desarrollado para el estudio del campo de velocidades en

¹meddies o Mediterranean Eddies

flujos, que lleva el nombre de Velocimetría por Imágenes de Partículas, se eligió el programa DPIVsoft [9]. Para reconstruir la forma de la superficie libre se debe integrar el gradiente de esa superficie. Una vez obtenida se buscaron los mínimos y se registraron las posiciones, para finalmente comparar los resultados de la posición de los vórtices con el modelo teórico [8].

Objetivos

Los objetivos del presente trabajo son:

- Obtener las posiciones en función del tiempo de los vórtices, así como la profundidad de éstos usando el método de schlieren sintético.
- Comparar el desplazamiento del dipolo experimental con el modelo teórico.

Organización

En el primer capítulo se da una descripción de los principios de la dinámica de fluidos, entre ellos, la deducción de las ecuaciones de Navier-Stokes y la ecuación de Bernoulli. Posteriormente en el segundo capítulo se revisa la dinámica de vórtices; la vorticidad, circulación del fluido, forma de la superficie libre en el vórtice de Rankine, el análogo electrodinámico, el teorema de Kelvin y la dinámica de sistemas de vórtices lineales paralelos. El tercer capítulo hace una deducción del método de schlieren sintético y habla del patrón aleatorio de puntos. El cuarto capítulo muestra el desarrollo experimental y explica los programas creados para el análisis de los resultados. El quinto capítulo presenta los resultados de los experimentos, el análisis de resultados y las conclusiones.

Capítulo 1

Fundamentos de la Dinámica de Fluidos

1.1. Transporte

El fenómeno de transporte aparece en los fluidos debido a pequeñas desviaciones del equilibrio de manera tal que la respuesta del sistema es aproximadamente lineal. Muchas cantidades físicas pueden ser transportadas, como el calor, la materia, el momento, etc. pero solo de dos formas se pueden transportar: transporte difusivo y transporte convectivo. En particular, la ecuación de transporte de calor está dada por

$$\frac{\partial T(r, t)}{\partial t} + (\mathbf{v} \cdot \nabla) T = k \nabla^2 T(r, t) \quad (1.1)$$

donde la función $T(r, t)$ es la temperatura y k el coeficiente de difusividad térmica, el cual mide la razón de transferencia de calor de un material. En el caso estático, la ecuación 1.1 pierde el término $(\mathbf{v} \cdot \nabla)$ y queda la ecuación conocida como de Fourier que es característica en todos los fenómenos de difusión. Un rasgo esencial en todos los fenómenos difusivos es la proporcionalidad entre la distancia media que se ha extendido la difusión y la raíz cuadrada del tiempo transcurrido; esta

relación también se puede deducir de la ecuación 1.1, observando que dimensionalmente el término de la izquierda debe ser igual al de la derecha, lo que conduce a la relación $\frac{\Theta}{t} + \frac{l\Theta}{lt} = k\frac{\Theta}{l^2}$ y al despejar l del lado derecho resulta la misma relación. Si se compara esta ecuación con la de propagación de onda $\frac{\partial^2 A}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 A}{\partial t^2}$, se obtiene que la distancia l recorrida por la onda es proporcional al tiempo t .

El transporte por convección de momento de un fluido con velocidad local U uniforme, es igual al producto de U por la cantidad transportada (el momento) $\rho_f U$ donde ρ_f es la densidad del fluido. Resultando $\rho_f U^2$, con dimensiones de presión, usualmente llamada presión dinámica del fluido. El significado de transporte convectivo es básicamente lo que se transporta por medio del movimiento colectivo de los elementos de volumen. A diferencia con el transporte por difusión donde los elementos de volumen no se mueven colectivamente, sin embargo, algunas cantidades físicas son transferidas hacia los elementos vecinos; el calor es un ejemplo claro.

1.2. Número de Reynolds y de Strouhal

En un flujo arbitrario los mecanismos difusivos y convectivos aparecen simultáneamente, pero dependiendo de la geometría y velocidad del fluido, uno tendrá más relevancia cuantitativa que el otro. El flujo de momento convectivo es del orden de $\rho_f U^2$, mientras que el difusivo para el caso de un flujo paralelo, asociado a la viscosidad es $\eta \frac{\partial v_x}{\partial y}$, por lo que es del orden de $\eta U/L$ donde L es la longitud característica del fluido. La razón adimensional formada por éstas dos se conoce como el número de Reynolds.

$$Re = \frac{\rho_f U^2}{\eta U/L} = \frac{UL}{\nu} \quad (1.2)$$

el cual puede obtenerse si se escribe la ecuación de Navier Stokes de forma adimensional.

En valores críticos del número de Reynolds, el flujo deja de ser estacionario y la velocidad del fluido depende explícitamente del tiempo. Por ejemplo, Osborne Reynolds (1842-1912) realizó experimentos con agua en un tubo al cual añadía un chorro de agua con color y encontró que el flujo de dicha línea se mantenía diferenciada a lo largo de todo el tubo si las velocidades eran bajas, pero a velocidades mayores la línea se rompía en un punto específico (punto de transición) y el color se difundía con el resto del fluido en toda la sección transversal del tubo. Para un flujo en un tubo de diámetro D los datos experimentales muestran que si $Re_D < 2000$ tendremos flujo laminar o estacionario, para valores comprendidos en $2000 < Re_D < 4000$ el flujo presenta inestabilidades, i. e. puede ser laminar o turbulento dependiendo de otros factores como la superficie del tubo o uniformidad del flujo y para valores más grandes $Re_D > 4000$ se vuelve turbulento. Un caso interesante aparece cuando el flujo es dependiente del tiempo debido a la interacción con un objeto y se tiene un valor de $Re_d > 45$ (para un cilindro de diámetro d); entonces se forman vórtices alternados en la estela del objeto, por desprendimiento de la capa límite de cada lado del cilindro; se les llama Calle de von Karman y la frecuencia de emisión es proporcional al número de Reynolds [1]. En el sistema estudiado se generan un par de vórtices simultáneamente (dipolo) con cada período de forzamiento del flujo. Debido a que tenemos un fenómeno periódico es conveniente emplear el número adimensional de Strouhal, el cual está dado por

$$S_r = \frac{L}{U\tau} \quad (1.3)$$

L corresponde al ancho del canal, U la velocidad máxima del fluido y τ el período de forzamiento.

1.3. Descripción de los fluidos

En la mecánica de fluidos existen dos formas de estudiar el flujo. Una de ellas es la descripción euleriana. En ella las propiedades depen-

den de las coordenadas espaciales y del tiempo; la velocidad adquiere un carácter de campo vectorial y es una variable dependiente, i.e. $\mathbf{V}(\mathbf{r}, t)$, es la representación más comúnmente utilizada en la dinámica de fluidos, estos campos son obtenidos al introducir sondas fijas y obteniendo las velocidades de diferentes partículas del fluido conforme pasan la sonda. Otra forma de medir los campos es por medio de la velocimetría de partículas que consiste en la introducción de partículas al fluido que viajan en trayectorias paralelas al flujo. Con ayuda de los medios digitales actuales es posible obtener velocidades instantaneas para cada tiempo. La descripción lagrangiana, en ella se sigue a una partícula en su movimiento. Es la representación usada en la mecánica clásica de partículas puntuales y del cuerpo rígido. La velocidad del fluido está caracterizada por el vector $\mathbf{V}(\mathbf{r}_0, t)$ que depende de dos variables. Este punto de vista consiste en hacer las mediciones con instrumentos que se mueven con el fluido, lo cual es más complicado de realizar.

Para poder establecer una ecuación del movimiento en los fluidos, se debe introducir la aceleración de una partícula de fluido, que por un lado se debe a la variación explícita con el tiempo del campo de velocidades $v(r_1, t') - v(r_1, t)$ y por otro, del cambio de la velocidad con respecto a la posición, $v(r_2, t') - v(r_1, t')$. El cambio resultante da la velocidad conocida como la derivada total o convectiva.

$$\frac{d\mathbf{v}}{dt} = \frac{\partial\mathbf{v}}{\partial t} + v_x \frac{\partial\mathbf{v}}{\partial x} + v_y \frac{\partial\mathbf{v}}{\partial y} + v_z \frac{\partial\mathbf{v}}{\partial z} \quad (1.4)$$

Mas adelante se introduce la relación de ésta con las fuerzas presentes en los fluidos.

1.4. Componentes locales del gradiente campo de velocidades

La variación espacial de la velocidad a primer orden es representada por $dv_i = \sum_{j=1}^3 \left(\frac{\partial v_i}{\partial x_j} \right) dx_j$ donde las cantidades $G_{ij} = \frac{\partial v_i}{\partial x_j}$ son los

elementos del tensor de segundo orden conocido como el tensor gradiente de velocidad, cuya representación en parte simétrica y antisimétrica es

$$\begin{aligned} G_{ij} &= \frac{\partial v_i}{\partial x_j} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) + \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i} \right) \\ &= e_{ij} + \omega_{ij}; \quad i, j = 1, 2, 3 \end{aligned}$$

Los términos diagonales del tensor gradiente de velocidad $G_{ii} \equiv \frac{\partial v_i}{\partial x_i}$ representan la razón de elongación o compresión en la dirección i -ésima; la suma de los elementos de la diagonal $\sum_{i=1}^3 \frac{\partial v_i}{\partial x_i} = \nabla \cdot \mathbf{v}$ es la razón de expansión del elemento de fluido, que para un fluido incompresible es $\nabla \cdot \mathbf{v} = 0$.

Para los términos fuera de la diagonal i.e. $G_{ij} = \frac{\partial v_i}{\partial x_j}$ para $i \neq j$ se tiene dos casos:

- La parte simétrica $e_{ij} \equiv \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)$ de estos elementos representa la deformación cortante de un elemento de volumen, o la deformación del elemento sin cambio de volumen.
- La parte antisimétrica $\omega_{ij} \equiv \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i} \right)$ representa una rotación local sin deformación alguna del campo de velocidades. Se expresa al tensor antisimétrico ω_{ij} como un pseudovector con componentes $\omega_k = -\varepsilon_{ijk} \omega_{ij}$, este es conocido como la vorticidad del flujo.

1.5. Ecuaciones locales de la dinámica de fluidos

En cualquier material el tensor de esfuerzo total σ está definido como la suma del tensor de esfuerzo viscoso σ' , el tensor de esfuerzo elástico ε y la presión hidrostática p . En materiales completamente fluidos i.e. que no pueden tener esfuerzos cortantes estáticos, se tiene $\varepsilon = 0$, quedando entonces $\sigma_{ij} = \sigma'_{ij} - p\delta_{ij}$; la presión hidrostática corresponde a la presión

ejercida por un elemento de fluido sobre el elemento adyacente y actúa normal a los elementos de superficie, la magnitud es independiente de la orientación por tanto un sólo número es suficiente para caracterizarlo.

El tensor de esfuerzo σ cuyos elementos σ_{ij} representan el componente en la dirección i del esfuerzo ejercido sobre la superficie con normal orientada en la dirección j , i.e. σ_{yx} es la componente y de la fuerza ejercida sobre un área unitaria con normal apuntando en la dirección x , a esto se le conoce como esfuerzo tangencial o cortante. En un fluido en movimiento por lo general los esfuerzos no son perpendiculares a las superficies de los elementos de volumen.

En un fluido Newtoniano el esfuerzo viscoso es proporcional al tensor de rapidez de deformaciones.

$$\sigma_{ij} = 2Ae_{ij} + B\delta_{ij}e_{ll}; \quad i, j = 1, 2, 3 \quad (1.5)$$

donde A y B son constantes reales que caracterizan el fluido y los índices repetidos implican una suma, de acuerdo a la convención de Einstein.

1.6. Ecuaciones del movimiento para un fluido

La aplicación de la segunda ley de Newton a un fluido conduce a la siguiente ecuación

$$\rho_f \frac{\partial \mathbf{v}}{\partial t} + \rho_f (\mathbf{v} \cdot \nabla) \mathbf{v} = \rho_f \mathbf{f} - \nabla p + \nabla \cdot [\sigma'] \quad (1.6)$$

- El primer término del lado izquierdo de 1.6 representa la aceleración de la partícula debido a la dependencia temporal explícita de la velocidad en un marco de referencia fijo (euleriano).
- El segundo término corresponde a los cambios de velocidad conforme una partícula de fluido es transportada por convección. Esto conduce a un término de aceleración aún si el campo de velocidades $\mathbf{v}(\mathbf{r})$ es independiente del tiempo.

- Del lado derecho de (1.6), el término $\varrho_f \mathbf{f}$ representa las fuerzas de volumen aplicadas al fluido; que pueden ser la gravedad, fuerzas electromagnéticas, etc.
- El siguiente término, $-\nabla p$ indica el efecto de la presión, corresponde al esfuerzo normal que está presente, incluso en ausencia de movimiento del fluido (presión hidrostática). Para un fluido estático ($\mathbf{v} = 0$) la ecuación (1.6) se simplifica a la ley fundamental de la hidrostática.

$$\varrho_f \mathbf{f} + \nabla p = 0 \quad (1.7)$$

- El último término $\nabla \cdot [\sigma']$ representa las fuerzas viscosas debido a las deformaciones de los elementos de fluido.

1.7. Ecuación de flujo de Navier-Stokes para un fluido Newtoniano

Al sustituir al tensor $[\sigma']$ con la forma explícita; se obtiene las fuerzas viscosas escritas en su forma vectorial.

$$\nabla \cdot [\sigma'] = \eta \nabla^2 \mathbf{v} + \left(\zeta + \frac{\eta}{3} \right) \nabla (\nabla \cdot \mathbf{v}) \quad (1.8)$$

donde al sustituir ésta en la ecuación (1.6) se obtiene la ecuación del movimiento para fluidos Newtonianos compresibles e incompresibles. En el caso de fluidos incompresibles $\nabla \cdot \mathbf{v} = 0$ se llega a la ecuación de Navier-Stokes.

$$\varrho_f \frac{\partial \mathbf{v}}{\partial t} + \varrho_f (\mathbf{v} \cdot \nabla) \mathbf{v} = \varrho_f \mathbf{f} - \nabla p + \eta \nabla^2 \mathbf{v} \quad (1.9)$$

1.8. Ecuación de Bernoulli

La ecuación de Bernoulli se puede obtener de la ecuación 1.9 al imponer ciertas condiciones en el sistema, se supone flujo independiente del tiempo, irrotacional y se desprecia la viscosidad. De manera que

$$\frac{\partial \mathbf{y}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\frac{1}{\rho_f} \nabla p + \mathbf{f} + \nu \nabla^2 \mathbf{v} \quad (1.10)$$

sustituyendo el lado derecho de la ecuación 1.10 con la identidad vectorial

$$(\mathbf{v} \cdot \nabla) \mathbf{v} = \frac{1}{2} \nabla v^2 - \mathbf{v} \times (\nabla \times \mathbf{v}) \quad (1.11)$$

se obtiene

$$\frac{1}{2} \nabla v^2 - \mathbf{v} \times (\nabla \times \mathbf{v}) = -\frac{1}{\rho_f} \nabla p + \mathbf{f} + \nu \nabla^2 \mathbf{v} \quad (1.12)$$

Cuando las fuerzas de volumen \mathbf{f} resultan de un potencial φ ; $\mathbf{f} = -\nabla\varphi$, la ecuación 1.12 queda

$$\frac{1}{2} \nabla v^2 = -\frac{1}{\rho_f} \nabla p - \nabla\varphi \quad (1.13)$$

pasando todos los términos del lado derecho, tenemos

$$\nabla \left(\frac{1}{2} \rho_f v^2 + p + \rho_f \varphi \right) = 0 \quad (1.14)$$

lo que implica que es igual a una constante, obteniendo la primer forma de la ecuación de Bernoulli

$$\rho_f \frac{v^2}{2} + p + \rho_f \varphi = cte \quad (1.15)$$

la cantidad $\rho_f \frac{v^2}{2}$ que tiene dimensiones de presión, es llamada presión dinámica.

La ecuación de Bernoulli para el flujo potencial; se da cuando el campo de velocidades puede ser derivado de un potencial Φ de manera que $\mathbf{v} = \nabla\Phi$ por lo que es irrotacional, se sigue suponiendo que el flujo es incompresible con densidad constante y que sobre éste solo actúan fuerzas de volumen que pueden ser derivadas de un potencial ϕ , pero ya

no se supone que el campo de velocidades sea estacionario, por lo que la ecuación queda

$$\varrho_f \frac{\partial \mathbf{v}}{\partial t} = \varrho_f \frac{\partial (\nabla \Phi)}{\partial t} = \nabla \left(\varrho_f \frac{\partial \Phi}{\partial t} \right) = -\varrho_f (\mathbf{v} \cdot \nabla) \mathbf{v} - \nabla p - \nabla (\varrho_f \phi) \quad (1.16)$$

usando la identidad vectorial 1.11 en el primer termino del lado derecho de la ecuación 1.16 se obtiene

$$\nabla \left(\varrho_f \frac{\partial \Phi}{\partial t} + \varrho_f \frac{v^2}{2} + \varrho_f \phi \right) = 0 \quad (1.17)$$

entonces

$$\varrho_f \frac{\partial \Phi}{\partial t} + \varrho_f \frac{v^2}{2} + p + \varrho_f \phi = cte \quad (1.18)$$

si el flujo es estacionario, i.e. $\varrho_f \frac{\partial \Phi}{\partial t} = 0$ la ecuación se reduce a la 1.15 pero esta ecuación es válida para todo el volumen del flujo en lugar de para una línea de flujo.

1.9. Capa límite y formación de vorticidad

La capa límite es la capa de fluido en la vecindad inmediata de las superficies que delimitan al fluido; en esta zona los efectos de vorticidad son muy importantes. Se pueden clasificar vagamente de acuerdo a su estructura y a las circunstancias bajo las cuales son creadas; la capa delgada que se desarrolla sobre un cuerpo oscilante o similarmente cuando el cuerpo esta fijo y el flujo oscila, son ejemplos de capa límite de Stokes, mientras que la capa límite de Blasius es la formada cerca de superficies planas bajo un flujo unidireccional, la capa de Ekman producida en fluidos rotando al balancearse las fuerzas viscosas con las de Coriolis, estas son algunas de las clasificaciones de las capas límites. Dentro de cada tipo de capa límite también se puede clasificarlas por el tipo de flujo; laminar y turbulenta. La capa límite laminar tiene un flujo suave, crea menos fricción superficial de arrastre que la turbulenta pero

es menos estable y tienden a ensancharse conforme avanzan sobre la superficie. La capa límite turbulenta contiene remolinos y contra flujos, se forman a cierta distancia en la dirección del flujo desde el borde de ataque, es el resultado del rompimiento del flujo laminar y tiene mayor arrastre que la capa límite laminar. El espesor de la capa límite se define normalmente como la distancia desde el cuerpo sólido hasta donde tenemos una velocidad de flujo del 99 % de aquél que hay en la corriente libre. En la imagen 1.1 se observa el campo de velocidades de una capa límite laminar.

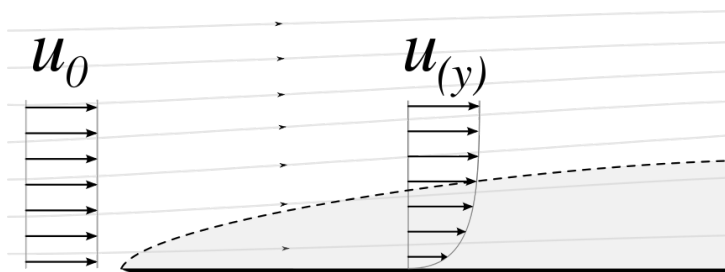


Figura 1.1: Campo de velocidades de flujo laminar sobre superficie plana. Imagen obtenida de Flanker-Own work, CC BY-SA 3.0, [https:// commons.wikimedia.org/w/index.php?curid=3602085](https://commons.wikimedia.org/w/index.php?curid=3602085)

La separación de la capa límite ocurre cuando la porción de esta, más pegada al cuerpo invierte la dirección de flujo, el punto donde pasa (determinado por la geometría del objeto y las características del fluido) es llamado punto de separación. La causa de la separación es porque la capa límite avanza lo suficientemente lejos en contra de un gradiente de presión adverso $dp/ds > 0$ en dirección de la línea de corriente s , se puede ver en la ecuación de momento en dirección de la línea de corriente, que el gradiente de presión hará disminuir la velocidad u

$$u \frac{\partial u}{\partial s} = -\frac{1}{\rho_f} \frac{dp}{ds} + \nu \frac{\partial^2 u}{\partial y^2} \quad (1.19)$$

y eventualmente cambiar la de dirección, como se observa en la imagen 1.2.

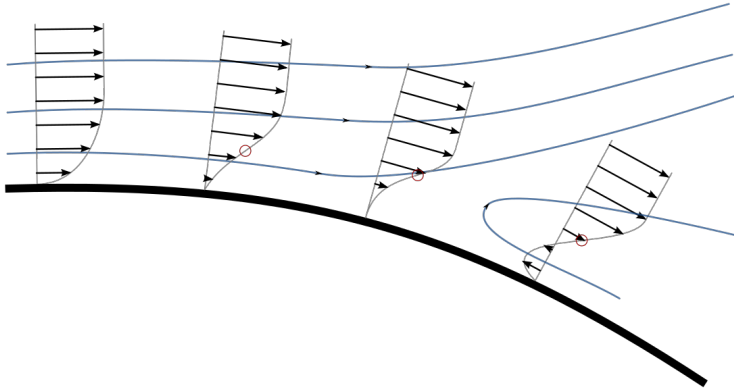


Figura 1.2: Campo de velocidades del fluido en la capa límite, se observa el punto de rompimiento donde aparece el contra flujo. Imagen de Ariadacapo - Own work, CC BY 3.0, [https:// commons.wikimedia.org/w/index.php?curid=42807305](https://commons.wikimedia.org/w/index.php?curid=42807305)

En el siguiente capítulo se definirá la circulación del fluido y su relación con la vorticidad; pero se menciona brevemente que si se evalúa la circulación en un circuito de la capa límite en la figura 1.1 o 1.2 serán diferentes de cero, lo que implica la existencia de vorticidad.

Capítulo 2

Dinámica de vórtices

Se define al vector vorticidad como

$$\vec{\omega} = \vec{\nabla} \times \vec{v}(\vec{r}) \quad (2.1)$$

donde $\vec{v}(r)$ es el campo de velocidades del fluido. La vorticidad corresponde a la parte antisimétrica del tensor gradiente de velocidad $G_{ij} = \frac{\partial v_i}{\partial x_j} = e_{ij} + w_{ij}$.

La vorticidad aparece siempre que un fluido no es puramente potencial y por tanto lo hace en fluidos viscosos [1]. En algunos casos el fluido es potencial fuera de un núcleo de diámetro ξ que es pequeño en comparación con las dimensiones macroscópicas del fluido.

2.1. Análogo electrodinámico

La ecuación 2.1 que relaciona a la velocidad del fluido con la vorticidad es análoga a la ecuación electromagnética $\vec{j}(r) = \frac{1}{\mu_0} \vec{\nabla} \times \vec{B}(r)$ que da la dependencia del caso estacionario de la densidad de corriente con el campo magnético en el vacío. Se elabora la analogía para un fluido incompresible libre de fuentes o sumideros de flujo (como en el caso electromagnético) [1] ya que ellos satisfacen las siguientes ecuaciones.

$$\vec{\nabla} \cdot \vec{B} = 0 \quad y \quad \vec{\nabla} \cdot \vec{v}(\vec{r}) = 0 \quad (2.2)$$

Por otra parte la ley de Ampere establece que

$$I = \iint_{\zeta} \vec{j} \cdot d\vec{S} = \frac{1}{\mu_0} \iint_{\zeta} (\vec{\nabla} \times \vec{B}) \cdot d\vec{S} = \frac{1}{\mu_0} \int_C \vec{B} \cdot d\vec{l} \quad (2.3)$$

donde la tercer igualdad se obtiene aplicando el Teorema de Stokes. Análogamente se tiene lo mismo para la circulación; definida como la integral de línea de la velocidad del fluido alrededor de una curva cerrada

$$\Gamma = \int_C \vec{v} \cdot d\vec{l} = \iint_{\zeta} (\vec{\nabla} \times \vec{v}) \cdot d\vec{S} = \iint_{\zeta} \vec{\omega} \cdot d\vec{S} \quad (2.4)$$

Al aplicar la ecuación de la vorticidad al caso del vórtice de tubo recto, el cual es análogo al alambre recto que transporta una densidad uniforme de corriente j se llega a la ecuación

$$\Gamma = \omega \rho_0^2 \pi \quad (2.5)$$

que representa la intensidad del vórtice, donde ω es la vorticidad y ρ_0 es el radio del vórtice. Igualmente al aplicar las condiciones del caso, para $\rho > \rho_0$ donde $\omega = (\nabla \times \vec{v})_z = 0$, se obtiene

$$v_{\varphi}(\rho) = \frac{\Gamma}{2\pi\rho} \quad (2.6)$$

El decrecimiento de v_{φ} como ρ^{-1} es característico de cualquier campo de velocidades azimutales con cero vorticidad. La variación de la presión P alrededor del vórtice para cualquier profundidad se obtiene al aplicar la ecuación de Bernoulli 1.16 en la zona irrotacional del campo de velocidades, donde $v = \nabla\Phi$ la velocidad es la derivada de un campo potencial, al igual que las fuerzas de volumen derivadas de un potencial φ .

$$P + \frac{1}{2}\varrho_f v^2 = P + \frac{1}{2}\varrho_f \frac{\Gamma^2}{4\pi^2 \rho^2} = cte \quad (2.7)$$

donde ϱ_f es la densidad del fluido, ρ la coordenada cilíndrica radial. Puede verse en la ecuación 2.7 que la presión decrece conforme decrece ρ para una profundidad fija en el fluido (pues el término $\varrho_f gh$ se cancela); este hecho se observa al producir vórtices en agua carbonatada, obteniendo una mayor formación de burbujas en el centro del vórtice que en otras zonas. En particular aplicando la ecuación 2.7 al modelo de aproximación del remolino formado al vaciar un contenedor (figura 2.1), conocido como vórtice de Rankine en la superficie $P = P_0$ y para $\rho > \rho_0$ (zona irrotacional) se tiene



Figura 2.1: Vórtice de Rankine al vaciar contenedor, imagen tomada en el Laboratorio de Fluidos de la Facultad de Ciencias de la UNAM.

$$\frac{P}{\varrho_f} + \frac{1}{2}v_\varphi^2 + gh(\rho) = \frac{P_0}{\varrho_f} + gh_0 \quad (2.8)$$

pues $v_\varphi \rightarrow 0$. Con lo que se obtiene una expresión para la altura $h(\rho)$ en la zona irrotacional del fluido

$$h(\rho) = \frac{gh_0 - \frac{1}{2}v_\varphi^2}{g} = h_0 - \frac{\Gamma^2}{8\pi^2 \rho^2 g} \quad (2.9)$$

el cual es un perfil hiperbólico cuando $\rho > \rho_0$. Cuando $\rho < \xi$ sólo es posible la rotación de cuerpo rígido, claramente $\xi < \rho_0$. La velocidad

angular Ω es constante y $v_\varphi = \Omega\rho$. Esta región central es el núcleo del vórtice. Luego entonces,

$$\omega = (\nabla \times \vec{v})_z = \frac{1}{\rho} \frac{\partial}{\partial \rho} (\rho v_\varphi) = \frac{1}{\rho} \frac{\partial}{\partial \rho} (\Omega\rho^2) = 2\Omega \quad (2.10)$$

la vorticidad es finita y constante dentro del núcleo. El fluido se encuentra en reposo en un marco de referencia que rota con el fluido con velocidad angular Ω , entonces para cualquier punto en el fluido el gradiente de presión contrarresta las fuerzas reales aplicadas.

$$\frac{\partial P}{\partial \rho} = \varrho_f \Omega^2 \rho \quad \frac{\partial P}{\partial z} = -\varrho_f g \quad (2.11)$$

integrando resulta

$$P(\rho, z) = \frac{1}{2} \varrho_f \Omega^2 \rho^2 - \varrho_f g (h(\rho) - h_1) \quad (2.12)$$

tomando a P como una constante en la superficie y despejando $h(\rho)$ tendríamos

$$h(\rho) = \frac{\Omega^2 \rho^2}{2g} + cte \quad (2.13)$$

lo que implica que $h(\rho)$ tiene un perfil parabólico en el núcleo. Esta deformación (depresión) en la superficie es explotada con el método de schlieren sintético en el presente trabajo. La hipótesis de que la vorticidad es constante en el núcleo es solo una primera aproximación, un modelo más realista es suponer que la vorticidad decrece a partir del centro como un función gaussiana.

2.2. Energía cinética del vórtice

La energía por unidad de longitud del vórtice está dada por la siguiente ecuación

$$e_k = \frac{1}{2} \iint \varrho_f v_\varphi^2 dS = \frac{\varrho_f}{2} \int \left(\frac{\Gamma}{2\pi\rho} \right)^2 2\pi\rho d\rho \quad (2.14)$$

La energía cinética en el interior del núcleo $\rho < \xi$ es igual a $\frac{1}{2} J \Omega_\xi^2$ donde $J = \pi\rho\xi^4/2$ es el momento de inercia y $\Omega_\xi = |\omega|/2 = \Gamma/2\pi\xi^2$ es la velocidad angular, con lo que obtenemos la energía total por unidad de longitud.

$$ek = \varrho_f \frac{\Gamma^2}{4\pi} \left(\log \frac{L}{\xi} + \frac{1}{8} \right) \quad (2.15)$$

La analogía electromagnética al tratar distribuciones arbitrarias de vorticidad se obtiene de la ecuación de Biot Savart

$$d\vec{B} = -\frac{\mu_0}{4\pi} I \frac{d\vec{l} \times \vec{r}}{|\vec{r}|^3} \quad (2.16)$$

que se generaliza en

$$\vec{B}(\vec{r}) = -\frac{\mu_0}{4\pi} \int_V \frac{\vec{j}(\vec{r}') \times (\vec{r} - \vec{r}')}{|\vec{r} - \vec{r}'|^3} dV' \quad (2.17)$$

lo que permite obtener análogamente para el campo de velocidades

$$d\vec{v} = \frac{1}{4\pi} \Gamma \frac{d\vec{l} \times \vec{r}}{|\vec{r}|^3} \quad , \quad \vec{v}(\vec{r}) = -\frac{1}{4\pi} \int_V \frac{\vec{\omega}(\vec{r}') \times (\vec{r} - \vec{r}')}{|\vec{r} - \vec{r}'|^3} dV' \quad (2.18)$$

para el caso particular de un vórtice curvo con radio R la velocidad autoinducida es

$$|u_1| \approx \frac{|\Gamma|}{4\pi R} \ln \left(\frac{R}{\xi} \right) \quad (2.19)$$

2.3. La ecuación de transporte de vorticidad

La ecuación de Navier Stokes se puede escribir de la siguiente manera

$$\frac{\partial \vec{v}}{\partial t} - \vec{v} \times \vec{\omega} + \nabla \left(\frac{1}{2} v^2 \right) = \vec{f} - \frac{1}{\rho_f} \nabla P + \nu \nabla^2 \vec{v} \quad (2.20)$$

donde $(\vec{v} \cdot \nabla) \vec{v} = -\vec{v} \times \vec{\omega} + \nabla \left(\frac{1}{2} v^2 \right)$. Aplicando el rotacional a la ecuación 2.20

$$\frac{\partial}{\partial t} (\nabla \times \vec{v}) - \nabla \times (\vec{v} \times \vec{\omega}) = \nabla \times \left(f - \frac{1}{\rho_f} \nabla P \right) + \nu \nabla \times (\nabla^2 \vec{v}) \quad (2.21)$$

suponiendo que las fuerzas son conservativas, i.e. $f = -\nabla \varphi$, que $\rho_f = cte$, se tendrá

$$\begin{aligned} \frac{\partial}{\partial t} (\nabla \times \vec{v}) - \nabla \times (\vec{v} \times \vec{\omega}) &= \nu \nabla \times (\nabla^2 \vec{v}) \\ \frac{\partial \vec{\omega}}{\partial t} + (\vec{v} \cdot \nabla) \vec{\omega} &= (\vec{\omega} \cdot \nabla) \vec{v} + \nu \nabla^2 \vec{\omega} \end{aligned}$$

esta ecuación juega el papel para $\vec{\omega}(r, t)$ equivalente al que la ecuación 2.20 juega para $\vec{v}(r, t)$ reconociendo que

$$\frac{\partial \vec{\omega}}{\partial t} + (\vec{v} \cdot \nabla) \vec{\omega} = \frac{d\vec{\omega}}{dt} \quad (2.22)$$

es la derivada Lagrangiana.

2.4. Teorema de Kelvin: la conservación de la circulación

El teorema establece que para cada contorno cerrado que se desplaza con el flujo, la circulación es constante si se cumplen las siguientes condiciones.

- Fluido no viscoso $\eta = 0$
- Fuerzas externas derivadas de una función potencial $\varphi : f = -\nabla\varphi$
- La densidad del fluido ϱ_f es constante o más general; función de la presión: $\varrho_f = f(P)$

entonces

$$\frac{d}{dt} \left[\int_C \vec{v} \cdot \delta\vec{l} \right] = 0 \quad (2.23)$$

$\frac{d}{dt}$ es la derivada convectiva o Lagrangiana calculada a lo largo de las partículas.

Sustituyendo en la ecuación 2.23 la ecuación 2.4

$$\frac{d}{dt} \left(\iint_{\zeta} \nabla \times \vec{v} \cdot \delta\vec{S} \right) = \frac{d}{dt} \left(\iint_{\zeta} \vec{\omega} \cdot d\vec{S} \right) = 0 \quad (2.24)$$

i.e. el flujo de la vorticidad ω a través de una superficie ζ anclada por el contorno C se mantiene constante. Se tiene entonces que $\Gamma = \pi\rho^2\omega$ es constante por lo que al reescribirlo de la siguiente manera

$$\pi\rho^2\omega = \frac{\delta m\rho^2}{2} \frac{\omega}{\delta m} \frac{4\pi}{2} = J\Omega k \quad (2.25)$$

donde $\delta m = \varrho_f\rho^2\pi\delta l$ es la masa del fluido, $J = \delta m\rho^2/2$ el momento de inercia y $k = 4\pi/\delta m$ una constante. Es evidente que $J\Omega$ es constante, por lo tanto el momento angular se conserva.

2.5. Dinámica de un sistema de vórtices lineares paralelos

Como consecuencia del teorema de Kelvin tenemos que los núcleos de los vórtices se mueven con la velocidad del fluido local en dicho punto, esta velocidad es el resultado de la suma de los campos de velocidades

de cada vórtice. Para el caso de un par de vórtices con $\Gamma_1 + \Gamma_2 = 0$ lo que significa que tienen signos opuestos o que giran en direcciones opuestas. En este caso los vórtices se moverán con una velocidad

$$v = \Gamma/2\pi d \tag{2.26}$$

perpendicular a la línea que los une [1], Γ es la magnitud de la circulación y d es la distancia entre ellos. Cuando la circulación de uno es más grande, adicional a la traslación el dipolo tendrá una rotación en la dirección del vórtice con mayor circulación. Si los vórtices fueran del mismo signo estos girarían alrededor del otro.

Capítulo 3

Método de schlieren sintético

El método consiste en obtener la forma de la superficie libre $h(x, y)$ (en mecánica de fluidos se llama superficie libre a una interfase entre líquido y un gas) a partir del campo de desplazamientos aparentes $\delta r(x, y)$ inducido por la refracción de la luz dispersada desde un patrón de puntos situado en el fondo del fluido $z = 0$.

Para poder determinar la dirección y magnitud del desplazamiento, las imágenes obtenidas son subdivididas en particiones llamadas ventanas de interrogación; éstas pueden traslaparse y deben ser del mismo tamaño. El objetivo de esto es obtener el vector de desplazamiento del contenido de cada ventana, tomando en cuenta que no se puede hacer la ventana tan pequeña como se quiera pues a partir de un valor mínimo se pierde información del conjunto i. e. la confianza estadística del vector se reduce drásticamente; para la distribución de puntos empleada, ese valor mínimo corresponde a cajas de 26×26 pixeles, sin embargo debido al tamaño de las particiones posibles impuesta por el programa DPIVsoft la medida debe ser potencia de 2. Debido a que la división de la imagen del patrón en ventanas contiguas nos proporciona una cantidad reducida de vectores de desplazamiento, se permite la utilización

de ventanas superpuestas, pues con ellas se obtiene más información, haciendo al campo vectorial mas denso y resultando en una mejor superficie al integrar. Para que el análisis de las imágenes sea concluyente se debe adaptar el patrón a la resolución de la cámara empleada y a la distancia de colocación. El patrón debe consistir en puntos aleatorios con una densidad tal que se obtenga al menos 10 puntos por celda de interrogación (32 pixeles por lado) y que tengan un radio de 2 pixeles ($0.27mm$) aproximadamente. En la imagen 3.1 se observa una zona ampliada del patrón, 65×65 pixeles. La razón por la que se utiliza un patrón aleatorio de puntos, es para evitar que los desplazamientos aparentes producidos por deformaciones periódicas den como resultado deformaciones nulas, i.e. si se tiene una deformación periódica con patrón de puntos periódicos existirían deformaciones de la superficie cuya imagen de puntos es igual al de la superficie plana localmente, por lo que al comparar las ventanas de interrogación de estas zonas específicas se obtendría un desplazamiento aparente nulo. El desarrollo teórico presentado a continuación fue obtenido del trabajo realizado por Moisy [5].



Figura 3.1: Sección de 65×65 pixeles del patrón de puntos aleatorios pegado al fondo del tanque.

Para poder determinar el campo $\delta r(x, y)$ se considera en la imagen 3.3a un punto M del patrón y se determinan los puntos virtuales M' y M'' ; correspondientes a los de la superficies plana 3.3b y deformada

3.4a del fluido y definir a $MM'' = \delta r$, para posteriormente relacionar δr con ∇h el gradiente de la superficie deformada.

Se Supondrá lo siguiente:

- Aproximación paraxial .- $H \gg L$ donde L es el tamaño del campo dado un ángulo paraxial máximo $\beta_{max} \simeq L/\sqrt{2}H \ll 1$. en la imagen 3.2 se muestra como se considera L .
- Aproximación de pendiente pequeña.- El ángulo θ entre el vector normal \hat{n} a la interfase y el vector vertical \hat{z} es pequeña (figura 3.4b). Como consecuencia la pendiente γ de la superficie, medida en el plano de incidencia (plano CAM figura 3.4a) también es pequeño.
- Aproximación de amplitud pequeña.- Tomando $h(x, y) = h_p + \eta(x, y)$ como la altura de la superficie en el punto (x, y) , la amplitud $|\eta|$ es pequeña comparada con la altura media h_p (figura 3.3a).

3.1. Imagen refractada a través de la superficie plana

El objeto virtual B' ubicado sobre el patrón en $(x_M, y_M, \alpha h_P)$ (figura 3.3b) con $\alpha = 1 - n/n'$. El desplazamiento de M a M' es hacia afuera en la dirección radial $MM' = h_p(\tan i - \tan i')\hat{r}$, donde la relación entre los ángulos de los rayos incidentes y refractados están dados por la ley de Snell $n \sin i = n' \sin i'$. Para la superficie plana el ángulo incidente es el ángulo paraxial β con la aproximación paraxial y las ecuaciones antes mencionadas, se obtiene

$$MM' = \alpha h_p i \hat{r} \quad (3.1)$$

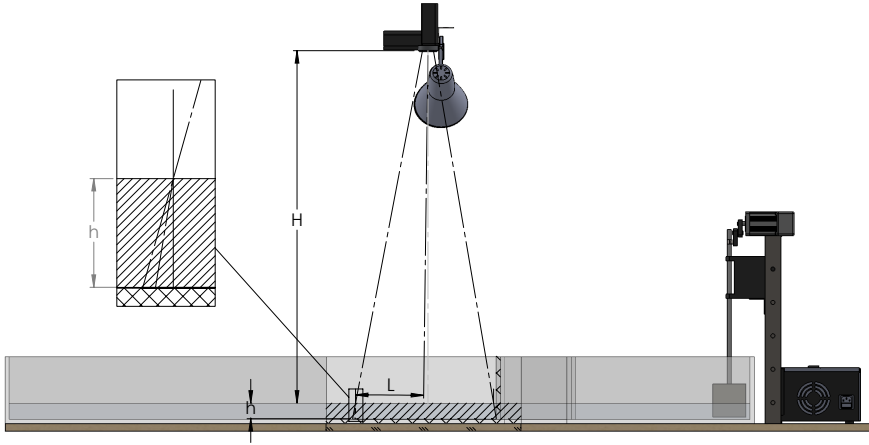


Figura 3.2: Posición de la cámara sobre el dispositivo experimental.

3.2. Imagen refractada a través de la superficie deformada

Considerando el caso de la superficie deformada, se define el vector normal a la superficie \hat{n} como

$$\hat{n} = \frac{\hat{z} - \nabla h}{\sqrt{1 + |\nabla h|^2}} \quad (3.2)$$

aplicando la aproximación de pendiente pequeña $|\nabla h|^2 \ll 1$ (cercana a cero); la ec. 3.2 se simplifica en $\nabla h = \hat{z} - \hat{n}$. Como \hat{n} se encuentra en el mismo plano que CM'' y \hat{S} (figura 3.4a) se le puede representar como la combinación lineal de éstos.

$$\hat{n} = a\hat{s} + b\frac{CM''}{|CM''|} \quad (3.3)$$

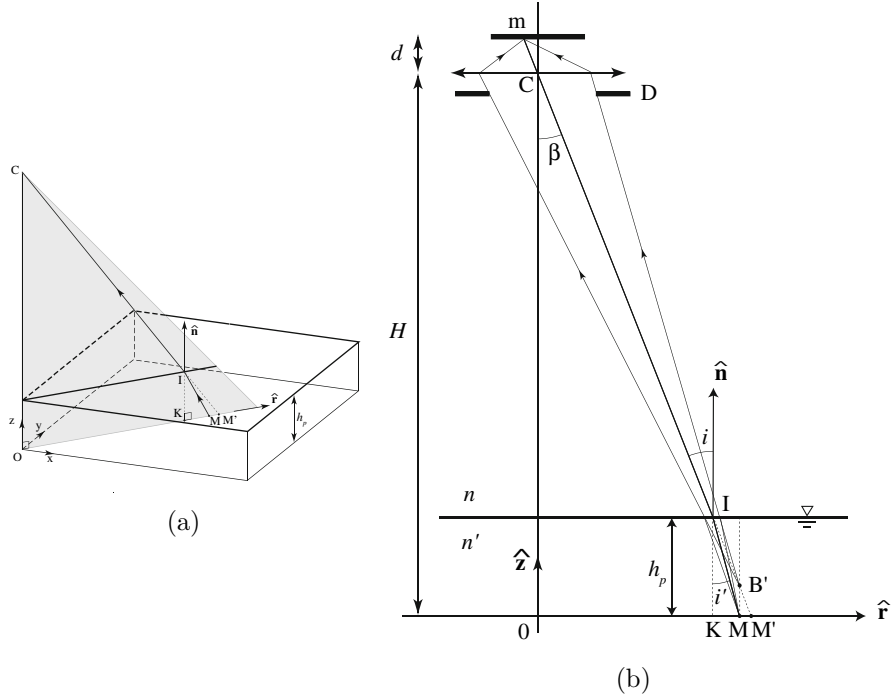


Figura 3.3: a) Geometría tridimensional de los rayos sobre la superficie plana. b) Vista bidimensional de los rayos en el plano incidente COM. Diagramas obtenidos de [5]

usando el proceso de ortonormalización de Gram-Schmidt se obtiene las proyecciones de \hat{n} sobre los vectores base $\frac{CM''}{|CM''|}$ y \hat{S} y se resta la aportación del vector $\frac{CM''}{|CM''|}$ sobre \hat{S} quedando

$$\begin{aligned}
 \hat{n} &= \left(\hat{n} \cdot \hat{S} - \frac{CM''}{|CM''|} \cdot \hat{S} \right) \hat{S} + \left(\hat{n} \cdot \frac{CM''}{|CM''|} \right) \frac{CM''}{|CM''|} \\
 &= \left(-\cos\left(\frac{\pi}{2} - \theta\right) + \cos\left(\frac{\pi}{2} - (i + \theta)\right) \right) \hat{S} + (-\cos(i)) \frac{CM''}{|CM''|}
 \end{aligned}$$

$$\hat{n} = (-\sin \theta - (-\cos i - \sin \theta)) \hat{S} - \cos i \frac{CM''}{|CM''|}$$

después de considerar las aproximaciones paraxiales de primer orden; se obtiene para la ecuación 3.3 $a = i$ y $b = -1$, por lo que

$$\hat{n} = i\hat{s} - \frac{CM''}{|CM''|} \quad (3.4)$$

sustituyendo \hat{n} en la ecuación de ∇h y reemplazando $\hat{z} = -\frac{CO}{|CO|} = -\frac{CO}{|H|}$ y $\frac{CM''}{|CM''|} \approx \frac{CO}{H} + \frac{OM''}{H}$ se obtiene

$$\nabla h = \frac{OM''}{H} - i \frac{MM''}{|MM''|} = \frac{OM''}{H} - i\hat{s} \quad (3.5)$$

de la figura 3.4a se tiene que $MM'' = IK(\tan(\theta + i) - \tan(\theta + i'))\hat{s}$ donde K es la proyección de I sobre $z = 0$, $IK = II_0 \cos \theta$ con I_0 a lo largo del vector normal \hat{n} e $II_0 = h(I)/\cos \gamma$ entonces.

$$MM'' = h(I) \frac{\cos \theta}{\cos \gamma} (\tan(\theta + i) - \tan(\theta + i')) \hat{s} \approx h_p i \alpha \hat{s} \quad (3.6)$$

considerando ángulos y deformaciones pequeñas $h(I) \approx h_p$, $\cos \theta \approx 1$, $\cos \gamma \approx 1$, $\tan(i + \theta) \approx i + \theta$, $\tan(\theta + i') \approx \theta + i \frac{n}{n'}$; por lo tanto $i\hat{s} = \frac{MM''}{\alpha h_p}$ y al sustituir en la ecuación 3.5 e introduciendo $OM'' = OM' + \delta r$ y $MM'' = MM' + \delta r$ (figura 3.5) resulta

$$\nabla h = \frac{OM'}{H} + \frac{\delta r}{H} - \left(\frac{MM'}{\alpha h_p} + \frac{\delta r}{\alpha h_p} \right) \quad (3.7)$$

$$\nabla h = -\delta r \left(\frac{1}{\alpha h_p} - \frac{1}{H} \right) + \frac{OM'}{H} - \frac{MM'}{\alpha h_p} \quad (3.8)$$

Los últimos dos términos de la ecuación 3.8 son idénticos por semejanza de triángulos que se puede ver en la figura 3.3b, por lo tanto se

tiene que el gradiente de la superficie posee una relación lineal simple con δr el desplazamiento.

$$\nabla h = -\frac{\delta r}{h^*} \quad (3.9)$$

$$\text{con } \frac{1}{h^*} = \left(\frac{1}{\alpha h_p} - \frac{1}{H} \right) > 0$$

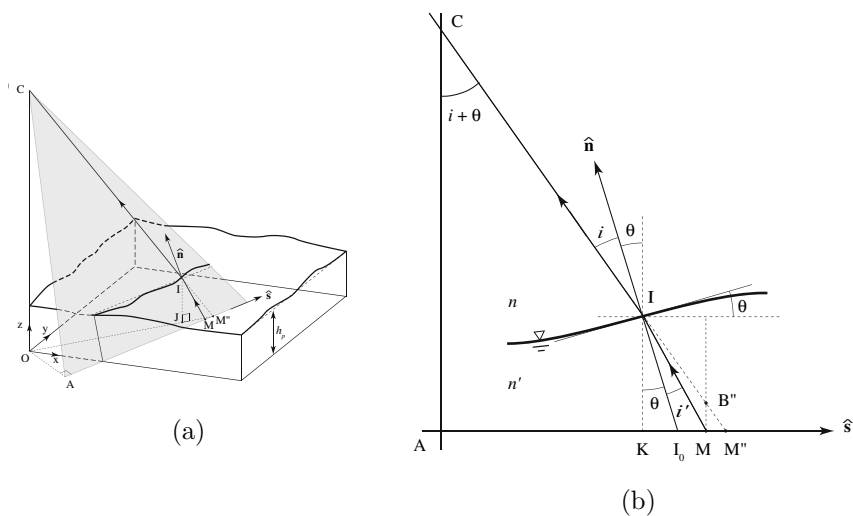


Figura 3.4: a) Geometría tridimensional de los rayos sobre la superficie deformada. b) Vista bidimensional de los rayos en el plano incidente CAM, donde C corresponde a la cámara, A es el punto de intersección entre la línea con dirección \hat{s} que pasa por M y la línea perpendicular a \hat{s} que pasa por el origen O. Diagramas obtenidos de [5]

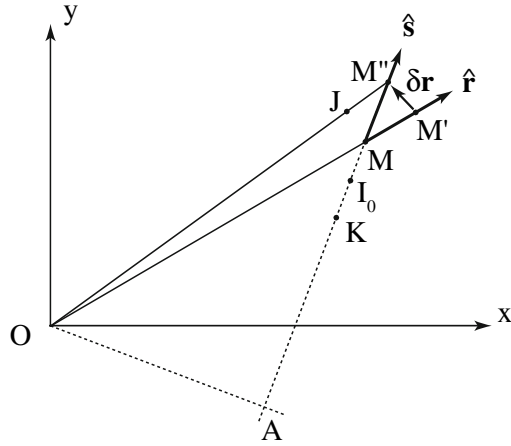


Figura 3.5: Vista superior del plano del patrón, se muestra el punto M y sus dos objetos virtuales M' y M'' para los casos de las superficies plana y deformada respectivamente. La línea AMM'' es la intersección del plano incidente CAM con el plano del patrón Oxy . J es la proyección vertical del punto I donde el rayo de luz MIC intercepta a la interfase. $M'M'' = \delta r$ es desplazamiento medido por la correlación digital de las imágenes. Diagrama obtenido de [5]

Capítulo 4

Desarrollo experimental

4.1. Dispositivo experimental

Para poder medir el fenómeno se utilizó un tanque acrílico, el cual es una maqueta simplificada de una bahía conectada por un canal con el mar, por lo que éste consta de dos espacios; uno de mayor tamaño que el otro y están conectados entre sí. Para controlar el ancho del canal se fabricaron unas compuertas de acrílico que se ubican dentro del canal y se sujetan con tornillos, los cuales permiten variar el ancho del canal, como se muestra en la figura 4.1. El tanque era llenado con agua a una profundidad aproximada de 3.8 cm. Se utilizó un par de lámparas montadas sobre el tanque en un armazón que también sostenía la cámara. La relación de posición entre la iluminación y el dispositivo de captura evitaba grabar el reflejo de las lámparas sobre la superficie, pues ello incrementaba el ruido en los análisis posteriores efectuados con el programa buscador de mínimos apéndice B.

Para crear el dipolo se produjo un flujo periódico con un sistema oscilante que simula las mareas de forma senoidal. Este dispositivo está formado por un controlador de motor Kollmorgen P70530, motor de pasos (T23NRLH-LNN-NS-00) de 200 por revolución y una biela manivela de 1.5 cm; lo que proporciona un movimiento recíprocante casi

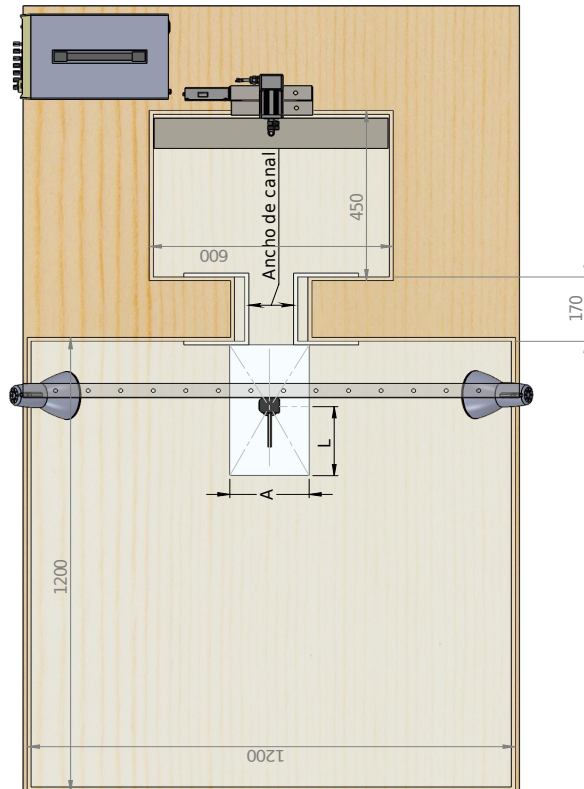


Figura 4.1: Vista superior del arreglo; se observan las compuertas del tanque de acrílico y el dispositivo recíprocante en la zona menor, que representaría la bahía, también se observa el montaje de la cámara con es sistema de iluminación sobre el patrón de puntos.

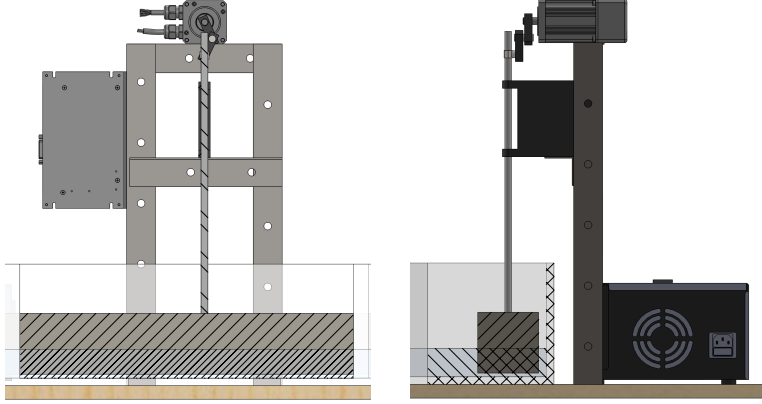


Figura 4.2: Sistema reciprocante

senoidal. A la biela se unieron 3 bloques de unicel (uno diferente en cada ocasión), con diferentes anchos para variar el gasto de agua y modificar así el número de Strouhal. El mecanismo se muestra en la imagen 4.2.

Se hizo funcionar al mecanismo a bajas frecuencias para obtener un gasto en el canal igual al generado por los bloques al ser sumergidos. El gasto es el volumen del líquido que fluye a través de una sección transversal de área A a una velocidad U , mientras que el gasto producido al sumergir el bloque está dado por

$$Q = \frac{dV(t)}{dt} = la_i r \omega \cos(\omega t) \quad (4.1)$$

donde l es la longitud, a_i el ancho del bloque y r el radio de la manivela, ω la frecuencia angular. La velocidad máxima en el canal se da cuando $\cos(\omega t)$ toma su máximo valor, i.e. $\omega t = 2\pi n$, por lo que para este sistema la ecuación de la velocidad máxima del fluido está dada por

$$U = \frac{Q_{max}}{A} = \frac{la_i r 2\pi}{\tau (w_{ch} d)} \quad (4.2)$$

con w_{ch} el ancho del canal, d la profundidad del agua y τ el período de

forzamiento.

Sistema de captura; está formado por una Cámara JVC everio modelo GZ-HM845BE para la captura de los videos y un montaje para sujetar la cámara con luces de apoyo para mejorar la iluminación del patrón de puntos. Para simplificar el análisis de los videos se filmaron inicialmente en reposo y se mantienen así por 10 segundos, pasados los cuales se inicia el movimiento del motor, esto con el propósito de obtener una imagen con la superficie del fluido plana en el segundo 8 e intercalarlo con las imágenes de la superficie deformada con ayuda del programa linux script apéndice A. Después de diferentes intentos con los videos se decidió dejar correr el experimento por un tiempo de 10 veces el período de forzamiento, con el fin de poder obtener el promedio de los recorridos.

4.2. Descripción de programas

Se desarrolló un conjunto de programas computacionales mostrados en los apéndices para el análisis de los videos. Los programas consisten en una automatización del procedimiento de preparación para Matlab; el primer script (apéndice A) es para la extracción de las imágenes con el uso del programa avconv de libre distribución. Con éstas imágenes el programa selecciona una de ellas (al inicio del video donde el experimento está detenido y el agua está en reposo), ésta imagen es copiada sobre cada imagen impar del conjunto total, con la finalidad de tener una imagen de referencia para las pares, donde la superficie del agua está deformada por los vórtices, a continuación se borran las imágenes de los primeros 10 segundos y renombran para el análisis con DPIV-soft [9] (ocupado posteriormente) pues éste ignoraba las primeras 100 imágenes. Finalmente empaqueta las imágenes en un archivo zip y repite el procedimiento para cada video de la carpeta. Dentro de la carpeta se añadió un archivo de texto con los datos del experimento para cada video, y el script lo lee para obtener los datos del ancho del canal, la frecuencia de inmersión del bloque, la distancia desde la boca del ca-

nal, altura de la cámara, y el área cubierta por ésta. Estos datos son usados igualmente por el programa modificado DPIVsoft, el cual es llamado dentro de ésta rutina sin interfaz gráfica. El programa DPIVsoft [9] creado para Velocimetría de imagen de partículas, obtiene el desplazamiento entre dos imágenes al analizar la correlación de fases de las transformadas de Fourier de las imágenes [13], pero permite calcular los desplazamientos aparentes de los puntos del patrón en el fondo del tanque de igual forma. El programa fue diseñado por Patrice Meunier en Matlab [9], para funcionar con una interfaz gráfica e interacción del usuario, pregunta el número de imágenes a tratar, el nombre que se les asignará a los campos de desplazamientos obtenidos y la ubicación de éstos. Debido al número de videos a analizar se modificó el programa para que no usara la interfaz ni requiriera los datos al usuario, pasando toda la información por medio del script. Al terminar el cálculo de los campos de desplazamiento se empaquetan los resultados en un zip.

El resto de los programas se ejecutan en Matlab; el primero (apéndice B) es una modificación de un programa proporcionado por mi asesor para integrar el campo de desplazamientos usando mínimos cuadrados en las posiciones para obtener la superficie que mejor se aproxima al gradiente. Este programa se modificó para añadirle automatización, haciendo que tome los datos del experimento del archivo de texto antes mencionado, y ejecute sobre todos los archivos comprimidos de los campos de desplazamiento, adicionalmente se agregó el código para la obtención de los centroides de los vórtices. Debido a la forma como se captan las posiciones de los centroides, que consisten en buscar los mínimos de la superficie obtenida proporcionando posiciones no nada mas de los centroides, sino también de impurezas flotando en la superficie del fluido que la deforman, produciendo mucho ruido, por lo que se elaboró el programa del apéndice C para filtrar los datos y preservar únicamente los pertenecientes a los vórtices, los cuales debían cumplir lo siguiente:

- Encontrarse en posiciones posteriores al punto de partida, i. e. que se alejen del canal.
- Que no se alejen demasiado de las posiciones anteriores, pues eso

implicaría que sufre una aceleración repentina; que no está presente.

- Que no se alejen demasiado en el tiempo.

El tercer programa (apéndice D) analiza el número de veces que se repitió el fenómeno; 10 en los últimos videos y si los recorridos de los vórtices fueron completos, i.e. que los vórtices cruzaron el umbral especificado. En este caso se seleccionó 2.5 veces el ancho del canal y una vez identificados los índices de la matriz donde se ubican los “regresos”, i. e. donde se repite la formación del dipolo a la salida del canal y por tanto el reinicio de las posiciones, se almacenan en una lista para retroceder el tiempo del recorrido i -ésimo en i períodos, interpolar datos y finalmente realizar un promedio de éstos y obtener una posición más precisa. Finalmente el programa grafica las posiciones promediadas, las experimentales y una curva teórica de la posición del dipolo.

El último programa (apéndice E) calcula la velocidad de los vórtices y la distancia entre ellos para comparar como varían las velocidades en función de la distancia.

Capítulo 5

Resultados y Conclusiones

5.1. Resultados

Con el programa B sin modificar se obtiene la forma de la superficie del fluido; en la figura 5.1 se muestra el resultado de un instante cuando el dipolo está pasando por la zona de análisis.

Los datos de los últimos 6 experimentos se muestran en la tabla 5.1, ésta contiene el número de video, el ancho de canal, la frecuencia de forzamiento, el ancho del área cubierta por la cámara (usado para darle escala a los datos) y los números de Reynolds y Strouhal calculados con Ec. 1.2, Ec. 1.3 usando la velocidad máxima del canal obtenida de la ecuación 4.2; se colocó la cámara a una altura sobre el patrón de 91.6 cm, usando el bloque de 57.4 cm de largo por 8.0 cm de ancho y profundidad del agua de 3.75 cm.

Los primeros videos grabaron sólo un período de forzamiento, esto producía un dipolo irregular que llegaba más lejos. Cuando se decidió grabar varios períodos para promediar las trayectorias, se observó que los dipolos subsecuentes no llegaban tan lejos, de manera que fue imposible promediarlos en la zona donde los otros no llegaron.

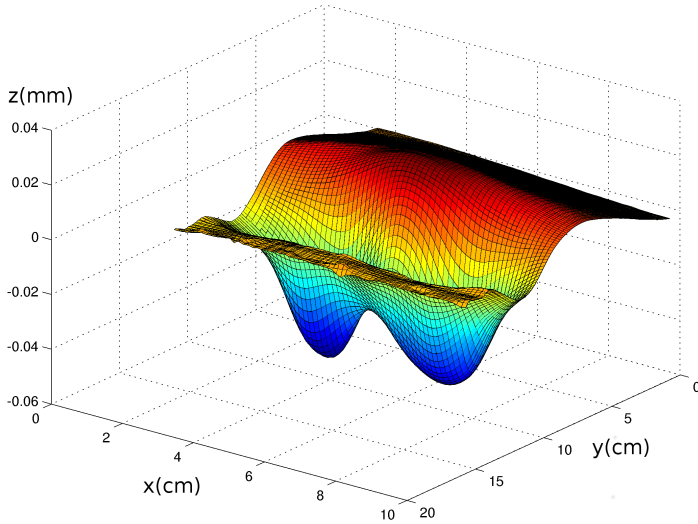


Figura 5.1: Superficie del fluido obtenida al usar el programa `intgrad2` (D'Errico, <http://www.mathworks.com/matlabcentral>)

| # video | w_{ch} (cm) | ν (Hz) | A_a (cm) | Re | St |
|---------|---------------|------------|------------|-------|-------|
| 24 | 2.0 | 0.05 | 8.9 | 5770 | 0.003 |
| 25 | 2.0 | 0.08 | 8.9 | 9232 | 0.003 |
| 26 | 2.0 | 0.1 | 8.9 | 11540 | 0.003 |
| 27 | 3.7 | 0.05 | 15.0 | 5770 | 0.012 |
| 28 | 3.7 | 0.08 | 15.0 | 9232 | 0.012 |
| 30 | 3.7 | 0.1 | 15.0 | 11540 | 0.012 |

Tabla 5.1: Datos experimentales

Esto hizo que los primeros videos no fueran concluyentes. Posteriormente y basados en los resultados de las simulaciones numéricas [6], [10] que reportan la destrucción de los vórtices a distancias mayores de 2.5 veces el ancho del canal, se obtuvieron 6 videos cuyos resultados fueron más

consistentes.

Otro problema encontrado en los videos fue que el patrón se levantaba levemente al paso del dipolo, modificando la distancia de la superficie al patrón e introduciendo más ruido, por lo que en los últimos videos se adhirió el patrón a todo lo largo, frente al canal con cinta adhesiva doble cara.

En las gráficas 5.2 se muestran los datos obtenidos sin el uso de los programas de filtrado, únicamente el programa de búsqueda apéndice (B) que encontraba los mínimos. Como ya se explicó no todos los mínimos corresponden a vórtices, ni todos los vórtices forman parte del dipolo, pues se observó que después del dipolo aparecían vórtices de una forma alternada similar a la calle de Von Karman, pero después de la ejecución del programa antes mencionado éstos eran ignorados. En la gráfica 5.2 se muestran los mínimos encontrados, donde el cambio de color esta relacionado con la variable temporal, i.e. puntos del mismo color o de color similar sucedieron al mismo tiempo o en tiempos muy cercanos. Se puede observar que los mínimos correspondientes a los vórtices del dipolo aparecen simultáneamente, en forma simétrica con respecto al centro del canal y alejándose de este. Los vórtices que no pertenecían al dipolo eran de menor intensidad y aparecían después del dipolo, generalmente de forma alternada. Los mínimos de los dipolos corresponden a los puntos que se encuentran por encima 0.6 y por debajo de -0.6 (i.e. los vórtices del dipolo distan entre ellos más que el ancho del canal).

Las gráficas 5.3, 5.4 y 5.5 se obtuvieron con datos filtrados y promediados por los programas de los apéndices C y D. En la gráfica 5.3 se muestra la posición de los vórtices del dipolo, cada línea corresponde a un recorrido del vórtice, el cambio de color en las líneas está relacionado con la variable temporal, el motivo por el que la mayoría de los vórtices aparecen con la misma variación de color es porque el tiempo fue desplazada en cada periodo por el programa D, de forma que se tengan todos los recorridos partiendo del tiempo cero y se pueda comparar con facilidad. La línea negra es la trayectoria promediada, para

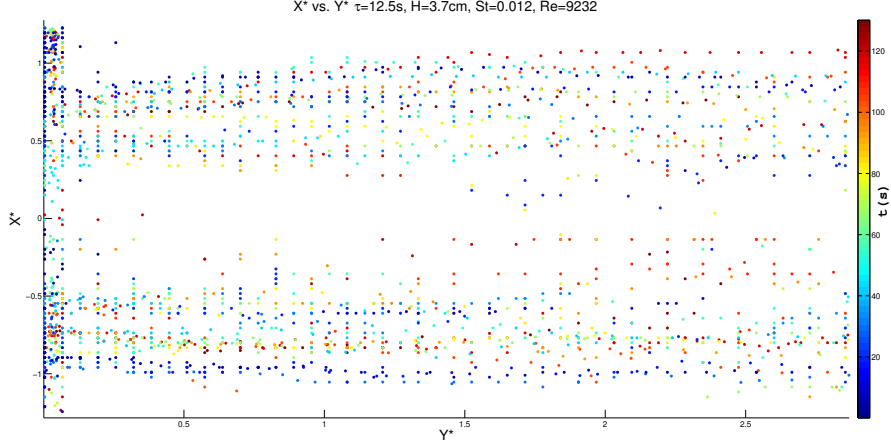


Figura 5.2: Gráfica de la posición de los mínimos de la superficie. Ejes adimensionalizados

la cual se interpolaron linealmente los datos de cada trayectoria experimental y así garantizar la existencia de los datos en cada instante de tiempo, al momento de calcular el promedio. En la gráfica 5.4 está la posición contra el tiempo para los recorridos del vórtice derecho e izquierdo respectivamente, los promedios y la curva teórica dada por la ecuación

$$Y(t) = \frac{1}{4\pi S^2} \left(\frac{1}{2}t^2 + \frac{1}{16\pi^2} [\cos(2\pi t) - 1] \right) \quad (5.1)$$

donde $t = t/\tau$, t es el tiempo en segundos y τ el período de forzamiento; obtenida de [7]. En la gráfica 5.6 se muestra la profundidad de uno de los vórtices del dipolo, de igual forma que en las otras, se grafican cada período y el promedio. En la última gráfica 5.7 aparecen las velocidades de cada vórtice en el tiempo y la distancia entre vórtices.

Del total de los videos capturados (más de 70), la mayoría de ellos presentaron diferentes dificultades que fueron resueltas en los subsecuentes. Los primeros 50 sólo tomaban uno o dos períodos a lo más,

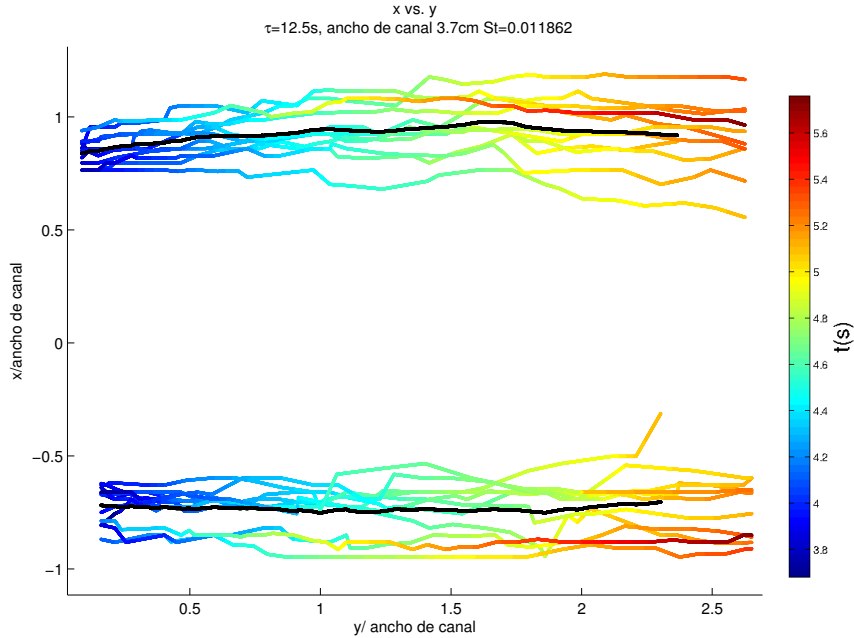


Figura 5.3: Gráfica de la posición de los vórtices filtrados. Las líneas negras corresponden al promedio de los recorridos de cada vórtice, el cambio de color en las otras líneas corresponde al tiempo; nótese que los 10 períodos que se dejó correr el experimento aparecen con las misma variación de coloración, esto es debido a que el tiempo fue recorrido para cada período de manera que se pudiera hacer el promedio temporal igualmente. Ejes adimensionalizados

introduciendo errores en las velocidades al iniciar y detener el motor; para solucionar esto se decidió grabar el fenómeno 10 veces y así obtener un promedio. Otro problema fue que captaban un área demasiado grande, perdiendo resolución y tomando zonas donde los vórtices eran muy débiles e inestables, pues aunque no necesariamente duraban poco, la profundidad de los vórtices se reducía hasta el grado donde el sistema ya no podía discriminar con claridad. Las simulaciones numéricas realizadas por E. López y G. Ruíz [11] y las observaciones sugerían que los

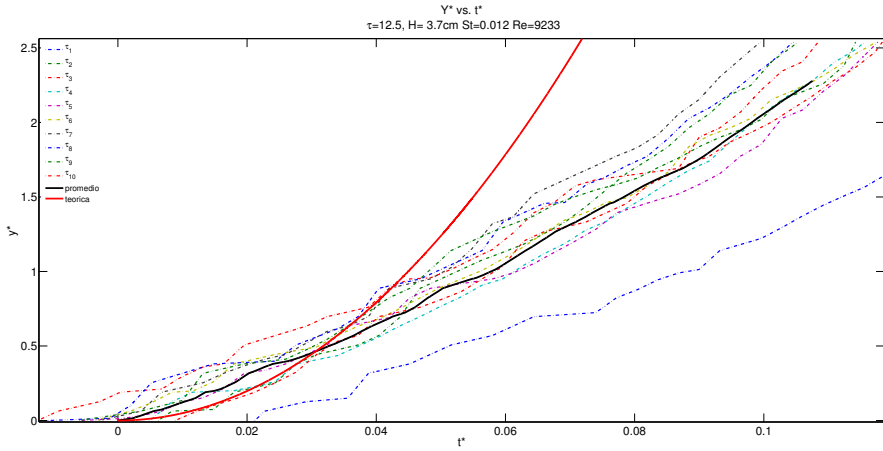


Figura 5.4: Vórtice derecho. Ejes adimensionalizados $y^* = y/w_{ch}$ con w_{ch} es el ancho del canal y $t^* = t/\tau$.

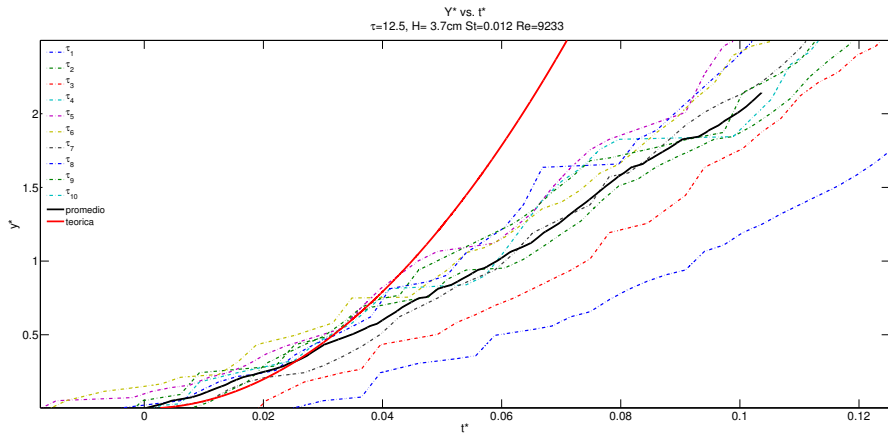


Figura 5.5: Vórtice izquierdo. Ejes adimensionalizados $y^* = y/w_{ch}$ con w_{ch} es el ancho del canal y $t^* = t/\tau$.

vórtices se debilitaban demasiado como para ser detectados si se alejaban mas de 2.5 veces el ancho del canal, lo que ayudó a tomar la decisión

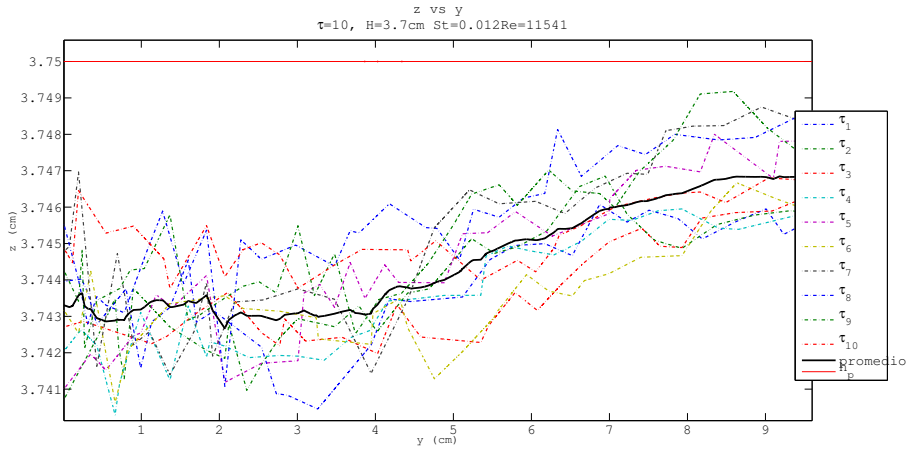


Figura 5.6: Gráfica de la distancia entre el fondo del tanque y la superficie libre en el núcleo del vórtice contra distancia desde el canal. Las líneas punteadas representan los diferentes períodos. h_p es la distancia desde el fondo hasta la capa de líquido

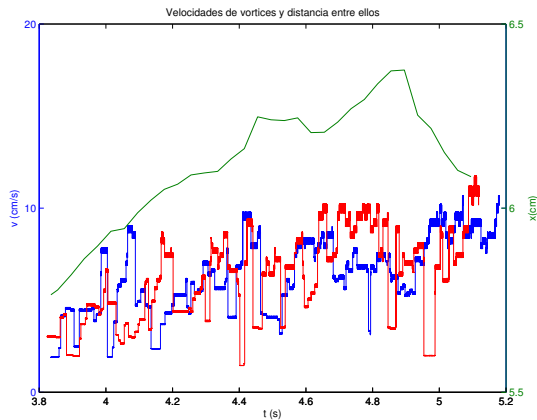


Figura 5.7: Gráficas de las velocidades contra tiempo para cada vórtice (rojo y azul), el verde muestra la distancia entre vórtices

de considerar únicamente dicha zona; situación que redujo la cantidad de datos espurios obtenidos en los primeros videos; otro problema fue

la cantidad de ruido introducido principalmente por impurezas flotando en la superficie del fluido, pero posterior al filtrado de datos se pudieron analizar los resultados.

5.2. Análisis de resultados

El método de schlieren sintético mostró ser muy preciso para medir las pequeñas deformaciones de la superficie; del orden de $30\mu m$. Se observó a la salida del canal que los vórtices se aceleran, esto es debido a que después de la creación de los vórtices y mientras éstos se encuentran justo fuera del canal, el flujo sigue incrementando la circulación, por tanto se aumenta la velocidad de traslación y profundidad de la superficie en el núcleo del vórtice, como se observa en las gráficas 5.4 y 5.5.

Se ve en las gráficas 5.4 y 5.5 que la curva teórica se parece sólo al principio con la experimental, los vórtices experimentales avanzaron más lentamente que lo predicho en el modelo Wells-Heijst [8]; debido a que el modelo no considera la viscosidad del fluido. Conforme los vórtices se alejan del canal se observa que alcanzan una velocidad casi constante (se van frenando ligeramente) y se mantienen así aproximadamente hasta 2.5 veces el ancho del canal, donde generalmente comienzan a destruirse o al menos a no ser detectables, ya que se observa en la gráfica antes mencionada que la incertidumbre es aproximadamente de $20\mu m$, y en $y = 8cm$ se tiene una profundidad promedio del núcleo de $30\mu m \pm 20\mu m$, por lo que al alejarse el dipolo un poco más, la resolución del método experimental ya no permite seguir al núcleo del vórtice de forma continua y debido al programa de filtrado de ruido (apéndice C) cualquier segmento de puntos (mínimos de la superficie libre) lo suficientemente separados eran desechados. Igualmente en la gráfica 5.6 conforme el dipolo se aleja la profundidad de la superficie libre se reduce, resultado de la disminución de la circulación por la viscosidad.

Se propone para trabajos posteriores mejorar el sistema de forzamiento para reducir vibraciones; una opción sería sustituir el sistema

actual por un sistema recirculatorio de agua que puede ser implementado fácilmente con bombas de agua donde podamos controlar la velocidad del motor. Esto ayudaría a reducir la incertidumbre en la profundidad de los núcleos de los vórtices, pues los impulsos producidos por los pasos del motor generaban ondas que hacían variar la profundidad de la superficie; mejorar las compuertas ya que al no ser perfectamente paralelas se introduce una componente de velocidad perpendicular al desplazamiento del dipolo, todo esto con el objetivo de poder rastrear los núcleos con mayor precisión y a mayor distancia del canal, para enfocarse en cómo afecta el hecho de que la velocidad del flujo forzante no incrementa la vorticidad al dipolo indefinidamente, i.e. la velocidad del dipolo no puede ser creciente siempre, pero el modelo teórico predice tal cosa pues al tomar la derivada temporal de la ec. 5.1 se obtiene

$$\frac{dY(t)}{dt} = \frac{1}{4\pi S^2} \left(t - \frac{1}{8\pi} \sin(2\pi t) \right) \quad (5.2)$$

y claramente es una velocidad creciente. Además de lo anterior hay que considerar que el dipolo se aleja del canal y esto implica que la circulación deja de ser alimentada en alguna distancia determinada, por lo que es de interés encontrar dicha posición y velocidad máxima. Igualmente se propone analizar el fenómeno con el método de PIV para compararlo con el presente trabajo.

5.3. Conclusiones

Se obtuvieron las posiciones de los vórtices con el método de schlieren sintético con muy buena precisión, de no ser por el ruido que introduce el motor de pasos se habría podido captar la posición del dipolo a distancias mayores y comparar los resultados con los obtenidos en [6], ya que en éstos los dipolos avanzan mucho más lejos y para ciertos valores del número de Strouhal llegan a interactuar entre si.

En lo que respecta a la comparación con el modelo de Wells-Heijst, claramente la falta de viscosidad en éste hace que la predicción se aleje del experimento conforme evoluciona el tiempo, que se acentúa al aumentar la velocidad del fluido.

Bibliografía

- [1] Etienne Guyon, Jean-Pierre Hulin, Luc Petit, *Hydrodynamique physique* SAVOIRS ACTUELS, EDP Sciences/CNRS Editions, nd Edition, 2001.
- [2] Arnold Sommerfeld, *Mechanics of Deformable Bodies* Lectures on Theoretical Physics, Volume 2, Academic Press, Sep 3, 2013.
- [3] Currie, I. G., *Fundamental Mechanics of Fluids* McGraw-Hill (1974)
- [4] Batchelor, G. K., (1967), *An Introduction to Fluid Dynamics* Cambridge University Press, ISBN 0-521-66396-2
- [5] F. Moisy, M. Rabaud, K. Salsac, *A Synthetic schlieren method for the measurement of the topography of a liquid interface* Experiments in Fluids (2009), Springer-Verlag.
- [6] Erik Javier López Sánchez, *Vorticidad y transporte de partículas en un flujo periódico a la salida de un canal* Tesis Dr. en Ciencias de la Tierra, Universidad Nacional Autónoma de México (2013).
- [7] Gerardo Ruiz-Chavarría, Erick Javier López-Sánchez, *Formation and Interaction of Multiple Dipoles in a Periodic Driving Flow* enviado a Physical Review Fluids
- [8] Mathew G. Wells, Gert-Jan F. van Heijst, *A model of tidal flushing of an estuary by dipole formation* Dynamics of Atmospheres and Oceans 37 (2003) 223-244, Elsevier.

- [9] Patrice Meunier, *Velocimétrie par Images de Particules (PIV)* <https://www.irphe.fr/~meunier/>.
- [10] E. J. López Sánchez, P. Villamil Sapien, G. Ruíz Chavarria and I. Sánchez Calvo González, *Erosion and Deposition of Particles in Periodic Forced Flow* Selected Topics of Computational and Experimental Fluid Mechanics (2015) 447-453, Springer-Verlag.
- [11] E. J. López Sánchez, G. Ruíz Chavarria, *Numerical Simulation of a Spanwise Vortex in a Periodic Forced Flow* Selected Topics of Computational and Experimental Fluid Mechanics (2015) 439-445, Springer-Verlag.
- [12] Gerardo Ruiz-Chavarría, Michael Le Bars and Patrice Le Gal, *Focusing of Surface Waves* Experimental and Computational Fluid Mechanics (2014) 315-325, Springer-Verlag.
- [13] Hassan Foroosh (Shekarforoush), Josiane B. Zerubia, Marc Bertod, *Extension of Phase Correlation to Subpixel Registration* IEEE Transactions on Image Processing, V. 11, No. 3, Mar. 2002, 188-200.

Apéndice A

Script de linux

```
1  #!/bin/bash
2  ##Programa que toma todos los archivos del directorio para extraer imagenes de video con ffmpeg,
   selecciona
3  ##una de las ultimas imagenes y la intercala con todas las otras que se consideran en movimiento i.e. la
   mitad de
4  ##ellas. Finalmente comprime todas las imagenes intercaladas en movimiento creando zip y borra el resto
   de las imagenes.
5  for fn in *.MTS; do      #ciclo para todas las peliculas
6      n=$(echo "$fn" | sed -e 's/0*\(.*\).MTS/\1/g') #variable con el numero(nombre) de pelicula
7      #llamado al programa avconv para pelicula $fn y nombrar imagenes jpg
8      avconv -i "$fn" -qscale 1 -r 50 mov"$n" -%d.jpg
9      j=$(ls *.jpg | wc -l)      #variable con numero de imagenes extraidas.
10     l=400      #numero de la imagen fija seleccionada, esta al principio del video 8s.
11     #ciclo para intercalar imagen fija entre cada una de movimiento
12     echo "Intercalando_imagen_fija_de_"$fn"
13     for (( i=501; i<=j; i++))
14     do
15         cp mov"$n" -"$i".jpg mov"$n" -"$i".jpg
16         i=$((i+1))
17     done
18     #ciclo para borrar el resto de imagenes fijas
19     echo "Borrando_imagenes_fijas"      #correspondiente a los primeros 10s i.e. a 50fps 500f
20     for (( k=1; k<=499; k++))
21     do
22         rm mov"$n" -"$k".jpg
23     done
24     #Renombrando para que la numeracion empiece en 101.jpg
25     echo "Renombrando_imagenes_para_DPIVsoft"
26     o=0
27     for ((i=500; i<=j; i++))
28     do
```



```

29     o=$(i-399)
30     cp mov"$n"-"$i".jpg mov"$n"-"$o".jpg
31 done
32 rm mov"$n"-.?.jpg
33 rm mov"$n"-??.jpg
34 for ((i=o; i<=j; i++))
35 do
36     rm mov"$n"-"$i".jpg
37 done
38 #creacion de archivo zip con todas las imagenes intercaladas
39 echo "Empaquetando_imagenes_en_zip"
40 zip mov"$n".zip mov"$n"-*.*jpg
41 echo "Eliminando_imagenes_de_mov"$n"_"
42 rm mov"$n"-*.*jpg
43 done
44 #
45 #lectura de la lista con datos de los experimentos debemos proporcionar
46 #en la carpeta de ejecucion de DPIVsoft lista con datos de experimento
47 filelist=$(readlink -f explist1.txt)
48 #ciclo con el que abrimos matlab para cada archivo .zip de imagenes, ejecutamos los programas
49 #Automaticprocess.2sa.m -> Automaticprocess.3.m -> Automaticprocess.4.m y despues ejecutamos
50 #con la misma serie de comandos a lectura4.m -> cclr.m
51 for zp in *.zip; do #ciclo para cada archivo .zip
52 unzip $zp #desempaquetado del archivo en variable zp
53 n=$(echo "$zp" | sed -e 's/0*(.*)\.zip/1/g') #obtencion del nombre de zp sin extension zip
54 m=$(echo "$zp" | sed -e 's/mov*(.*)\.zip/1/g') #obtencion del numero de pelicula para la creacion
de archivo .mat con vel??.mat
55 channel_wide=$(awk '1 ~ /'"$m"'$/ {print $2}' explist1.txt)
56 frequency=$(awk '1 ~ /'"$m"'$/ {print $3}' explist1.txt)
57 distance_chan=$(awk '1 ~ /'"$m"'$/ {print $4}' explist1.txt)
58 filename=$(ls *.jpg | sort -t- -n -k2.1 | head -1) #variable para guardar el nombre de la primer
imagen a tratar
59 simpname=$(echo $filename | sed -e 's/0*(.*)\.jpg/1/g')
60 compppath_filename=$(readlink -f $filename)
61 path_filename=$(echo $comppath_filename | sed -e 's/0*(.*)'"$filename"'/1/g')
62 j=$(ls *.jpg | wc -l) #variable con el numero total de imagenes para el analisis de campos = j/2
63 matlab -nodisplay -nosplash -r " _pathname_lout=['$path_filename'],_filename_lout=['$filename'],_
num_of_vel=$m,_n_field_ans2=$j,_chawide=['$channel_wide'],_frecuencia=['$frequency'],_
distanchan=['$distance_chan'],_Automaticprocess.2sa,_quit"
64 rm $n*.jpg #borrado de imagenes .jpg despues del analisis con DPIVsoft para evitar recuento en s
65 velname='vel'$m'w'$channel_wide'f'$frequency'd'$distance_chan'.zip'
66 zip $velname vel$m*.mat
67 rm vel$m*.mat
68 done #sig. valor de zp

```

Apéndice B

Programa en Matlab buscador de vórtices

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  %Programa para integrar automaticamente todos los archivos de una carpeta
3  %con archivos zip que contienen los campos vectoriales proporcionados por
4  %DPIVsoft
5  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6  clear;
7  %hold off;
8  cpl=importdata('cmplist2.txt');
9
10 archs = dir('*.*zip');
11 for arch = archs'
12     arctr = arch.name;
13     nvid=(arctr(strfind(arctr,'l')+1:strfind(arctr,'w')-1));
14     wd=(arctr(strfind(arctr,'w')+1:strfind(arctr,'f')-1));
15     command=sprintf('%s_%s', 'unzip', arctr);
16     unix(command);
17     vfield = dir('*.*piv');
18     arctrnzp=(arctr(1:strfind(arctr,'.')-1));
19     arctrnzpvn=(arctr(4:strfind(arctr,'.')-1));
20     nini=100;
21     indarray=[];
22     for vfl = vfield'
23         nomvfl=vfl.name;
24         nomvfl1=(nomvfl(1:strfind(nomvfl,'-')));
25         command=sprintf('%s %s', 'sscanf(nomvfl,','',nomvfl1,' %d''));
26         indarray=[indarray;eval(command)];
27     end
```

```

28 .
29 ...
30 .
31 %%%%%%%%%%%
32 % Converison a imagenes para obtener los centroides.
33 %%%%%%%%%%%
34 % divido el campo en dos partes para buscar el minimo con mayor
35 % precision en cada una .
36     for i=1:nx
37         for j=1:ny
38             if j<=32
39                 fh1l(i,j)=fh1(i,j);
40             else
41                 fh1r(i,j-32)=fh1(i,j);
42             end
43         end
44     end
45 % creacion de imagen en escala de grises para cada campo, posteriormente
46 % creacion de imagenes blanco y negro para para buscar los voritces
47 % en cada lado con mayor precision pues no tienen la misma profundidad.
48 % finalmente creacion de imagen BW con la union de ambos.
49     I1=mat2gray(fh1l);
50     Ir=mat2gray(fh1r);
51     I2=[I1 Ir];
52     BW1=I1<0.009;
53     BWr=Ir<0.009;
54     I=mat2gray(fh1);
55     BW=[BW1 BWr];
56 % obtencion de centroides para los vecindades de los vortices.
57     s=regionprops(BW, I, {'Centroid', 'WeightedCentroid'});
58
59     numObj=numel(s);
60     for k=1:numObj
61         rcy=(round(s(k).Centroid(1)));
62         rcx=(round(s(k).Centroid(2)));
63         %condicion para guardar en arch. si estan dentro de limites %%% modificado de rcx<=118
64         if (rcy>20 & rcy<44) & (rcx>6 & rcx<=118)
65             %definicion de matriz vecindad del punto dado por Regionprops
66             pnhm=fh(rcx-2:rcx+2,rcy-2:rcy+2);
67             %busqueda de minimo de la matriz pnhm por columnas
68             [mincol,indrow]=min(pnhm);
69             %busqueda de minimo del conjunto de minimos anterior
70             [minmtx,indcol]=min(mincol);
71             %indices en donde se encuentra el minimo en la matriz pnhm
72             indmin=[indrow(indcol),indcol];
73             %posicion real (indices de la matriz fh)
74             Absmin=[rcx-4+indmin(1),rcy-4+indmin(2)];
75             fprintf(Cpos,' %f_ %f_ %f_\n', s(k).Centroid(1),s(k).Centroid(2),fh(Absmin(1),
76                 Absmin(2)), t);

```

```

77         end
78         if k == numObj & flag
79             fprintf(Cpos, '\n');
80             flag=0;
81         end
82     end
83 end
84 %concatenamos el nombre de las imagenes .piv
85     nindex=[arctrnzip '*.*piv'];
86 %definimos el comando rm y el nombre para pasarlo a unix
87     command=sprintf('%s_%s', 'rm', nindex);
88 %ejecutamos command en la shell de Unix
89     unix(command);
90     toc(t0);
91     fclose(Cpos);
92 end
93 cclr2

```

Apéndice C

Programa en Matlab filtrador de centros de vórtices

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  % cclr.m
3  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4  %programa para filtrar los centroides correspondientes a vortices
5  %y remover los falsos positivos debido al ruido; producidos en la
6  %salida del programa lectura3.m
7  clear;
8  cpl=importdata('cmplist3.txt');
9  files = dir('centroides*.txt');
10 %files=dir('centil*.txt');
11 for file = files'
12     workfile = file.name;
13     wf = importdata(workfile);
14     [wfr, wfc]=size(wf);
15     %Obtencion de datos de experimento
16     nvid=workfile(strfind(workfile,'des')+3:strfind(workfile,'w')-1);
17     wd=workfile(strfind(workfile,'w')+1:strfind(workfile,'f')-1);
18     Cam_a=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),10);
19     Cam_l=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),9);
20     Ch_w=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),2)/10;
21     frec=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),3);
22     Tau=1/frec;
23     Delta_tau=0.9* Tau;
24     %creacion de archivos .txt con las listas limpias; uno para cada vort.
```

```

25 Cll1=['cl1sc' workfile];
26 Cll2=['cl2sc' workfile];
27 %limites
28 nyf=round(Cam.l/(Cam.a/64));
29 nyi=nyf-3*round(Ch.w/(Cam.a/64));
30 %busqueda de los centros de vort. entre los intervalos.
31 nyff=nyf-(nyi-1);
32 y1=nyff-8; y2=nyff-8;
33 Ti1=wf(1,4); Tf1=wf(1,4);
34 Ti2=wf(1,4); Tf2=wf(1,4);
35 flag1r=0; flag2r=0;
36 Mcclr1=[]; Mcclr2=[];
37 i.c1=1; i.c2=1;
38 k1=0; k2=0;
39 j1=i.c1+k1; j2=i.c2+k2;
40 ndx.rt1=[0];ndx.ncpl1=[];
41 ndx.rt2=[0];ndx.ncpl2=[];
42 for i=1:wfr
43     x=wf(i,2);
44     %scatter(wf(i,1),wf(i,2),[],wf(i,4));
45     if (x >= 4) & (x<29)
46         ytemp1=wf(i,1);
47         Tf1=wf(i,4);
48         %condicion para resetear el limite superior para buscar
49         %otra vez. Y mover indice de completez; indice que marca
50         %donde empezar la sustitucin en caso no no tener recorrido
51         %completo.
52         if (y1<6 & ytemp1<6) | (Tf1-Ti1>0.6)
53             %si entr en condicin por el tiempo no muevas el indice
54             y1=nyff-1;
55             if Tf1-Ti1>0.2 & ~(y1<6 & ytemp1<6)
56                 k1=0;
57             end
58             if y1<6 & ytemp1<6
59                 %si lleg al final mueve el indice de completez.
60                 i.c1=i;
61             end
62         end
63         if y1-ytemp1>0 & y1-ytemp1<6
64             Mcclr1(i,:)=wf(i,:);
65             y1=ytemp1;
66             Ti1=Tf1;
67         end
68     end
69 end
70 Mcclr1(~any(Mcclr1,2),:)=[];
71 %dlmwrite(Cll1,Mcclr1);
72 [Mcclr1r,Mcclr1c]=size(Mcclr1);
73 %listas con indices de retrocesos e incompletos
74 for i=1:Mcclr1r-1

```

```

75 %si retrocedes guarda indice
76     if (Mcclr1(i+1,1)-Mcclr1(i,1))> 0
77         ndx_rt1=[ndx_rt1, i+1];
78 %si la posicion anterior no llego al "final"
79     if Mcclr1(i,1)>7
80 %guarda indice, junto con indice de inicio de recorrido
81         ndx_ncpl1=[ndx_ncpl1, ndx_rt1(end-1), i];
82     end
83 end
84 end
85 %crecion de lista vacia para crear lista con indices completos a
86 %remove ndx_ncpl1=(...,Ji,Nf,...)-->ndx_ncplint=(...,Ji,K,L,M,Nf,...)
87 ndx_ncplint=[];
88 [ndx1r,ndx1c]=size(ndx_ncpl1');
89 for j=1:2:ndx1r
90 %k numero de indices entre j y j+1
91     k=ndx_ncpl1(j+1)-ndx_ncpl1(j);
92 %incersion de k indices entre j y j +1
93     for l=0:k
94         ndx_ncplint=[ndx_ncplint, ndx_ncpl1(j)+l];
95     end
96 end
97 %remocion de indice cero
98 ndx_ncplint(:,~any(ndx_ncplint,1))=[];
99 if ~isempty(ndx_ncplint)
100     Mcclr1ni=[];
101 %remocion de renglones marcados por lista ndx_ncplint
102     Mcclr1ni=removerows(Mcclr1,'ind',ndx_ncplint);
103 %dlmwrite(Cll1,Mcclr1ni);
104 end
105 %remocion de todos los datos que se
106 %encuentren despues del recorrido
107 %principal y antes del periodo del exp.
108 [Mcclr1nir,Mcclr1nic]=size(Mcclr1ni);
109 flag1=0;
110 ti=Mcclr1ni(1,4);
111 Mcclr1nisc=[Mcclr1ni(1,:)];
112 for i=1+1:Mcclr1nir
113 %matriz para almacenar datos
114     Mcclr1nisc=[Mcclr1nisc; Mcclr1ni(i,:)];
115 %si retrocedes marcalo (primer retroceso)
116     if Mcclr1ni(i,1)-Mcclr1ni(i-1,1)>0
117         flag1=1;
118     end
119 %si el dato es Delta_tau despues entonces
120 %reinicia marcador y mueve inicio de periodo
121     if Mcclr1ni(i,4)-ti>Delta_tau
122         flag1=0;
123         ti=Mcclr1ni(i,4);
124     end

```

```

125 %si marcador de primer retroceso esta activo
126 %y el dato esta antes de Delta_tau+ti borra
127 %ultimo renglon aadido
128     if flag1 & Mcclr1ni(i,4)-ti<Delta_tau
129         Mcclr1nisc(end,:)=[];
130     end
131 end
132 dlmwrite(Cll1,Mcclr1nisc);
133 for i=1:wfr
134     x=wf(i,2);
135     if (x >= 30) & (x<55)
136         ytemp2=wf(i,1);
137         Tf2=wf(i,4);
138         %condicion para resetear el limite superior para buscar
139         %otra vez. Y mover indice de completez; indice que marca
140         %donde empezar la sustitucin en caso no no tener recorrido
141         %completo.
142         if (y2<6 & ytemp2<6) | (Tf2-Ti2>0.6)
143             %si entr en condicin por el tiempo no muevas el indice
144             y2=nyff-1;
145             if Tf2-Ti2>0.2 & ~(y2<6 & ytemp2<6)
146                 k2=0;
147             end
148             if y2<6 & ytemp2<6
149                 %si lleg al final mueve el indice de completez.
150                 i.c2=i;
151             end
152         end
153         if y2-ytemp2>0 & y2-ytemp2<6
154             Mcclr2(i,:)=wf(i,:);
155             y2=ytemp2;
156             Ti2=Tf2;
157             %scatter(wf(i,1),wf(i,2),[],wf(i,4),'*');
158         end
159     end
160 end
161 Mcclr2(~any(Mcclr2,,:))=[];
162 %dlmwrite(Cll2,Mcclr2);
163 [Mcclr2r,Mcclr2c]=size(Mcclr2);
164 %listas con indices de retrocesos e incompletos
165 for i=1:Mcclr2r-1
166     if Mcclr2(i+1,1)-Mcclr2(i,1) > 0
167         ndx_rt2=[ndx_rt2, i+1];
168         if Mcclr2(i,1)>8
169             ndx_ncpl2=[ndx_ncpl2, ndx_rt2(end)-1, i];
170         end
171     end
172 end
173 ndx_ncpl2int=[];
174 [ndx_r,ndxc]=size(ndx_ncpl2');

```



```

175     for j=1:2:ndxr
176         k=ndx_ncpl2(j+1)-ndx_ncpl2(j);
177         for l=0:k
178             ndx_ncpl2int=[ndx_ncpl2int, ndx_ncpl2(j)+1];
179         end
180     end
181     ndx_ncpl2int(:,~any(ndx_ncpl2int,1))=[];
182     if ~isempty(ndx_ncpl2int)
183         Mcclr2ni=[];
184         Mcclr2ni=removerows(Mcclr2,'ind',ndx_ncpl2int);
185         %dlmwrite(Cll2,Mcclr2ni);
186     end
187     %remocion de todos los datos que se
188     %encuentren despues del recorrido
189     %principal y antes del periodo del exp.
190     [Mcclr2nir,Mcclr2nic]=size(Mcclr2ni);
191     flag2=0;
192     ti=Mcclr2ni(1,4);
193     Mcclr2nisc=[Mcclr2ni(1,:)];
194     for i=1+1:Mcclr2nir
195         %matriz para almacenar datos
196         Mcclr2nisc=[Mcclr2nisc; Mcclr2ni(i,:)];
197         %si retrocedes marcalo (primer retroceso)
198         if Mcclr2ni(i,1)-Mcclr2ni(i-1,1)>0
199             flag2=1;
200         end
201         %si el dato es Delta_tau despues entonces
202         %reinicia marcador y mueve inicio de periodo
203         if Mcclr2ni(i,4)-ti>Delta_tau
204             flag2=0;
205             ti=Mcclr2ni(i,4);
206         end
207         %si marcador de primer retroceso esta activo
208         %y el dato esta antes de Delta_tau+ti borra
209         %ultimo renglon aadido
210         if flag2 & Mcclr2ni(i,4)-ti<Delta_tau
211             Mcclr2nisc(end,:)=[];
212         end
213     end
214     dlmwrite(Cll2,Mcclr2nisc);
215 end
216 clearvars -except file files
217 fprintf('cclr_terminado,_empezando_trprom\n');
218 trprom

```

Apéndice D

Programa en Matlab promediador

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 % trprom2.m
3 % promediador de trayectorias, este programa toma los archivos cl?.txt y
4 % obtiene un promedio de las deferentes repeticiones del fenomeno.
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6 clear;
7 cpl=importdata('cmplist3.txt');
8 cl_files = dir('cl*sc*.txt');
9 for cl_file = cl_files'
10     fid_cl=fopen(cl_file.name,'rt');
11     nLines=0;
12     while (fgets(fid_cl) ~= -1)
13         nLines=nLines+1;
14     end
15     if cl_file.bytes==0 | nLines < 20
16         continue;
17     end
18     workfile = cl_file.name;
19     cl_wf = importdata(workfile);
20     [cl_wfr, cl_wfc]=size(cl_wf);
21 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
22 % extraccion de datos en nombre de archivo identificacion de experimento
23 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
24     numcl=(workfile(strfind(workfile,'l')+1:strfind(workfile,'cent')-1));
25     nvid=(workfile(strfind(workfile,'es')+2:strfind(workfile,'w')-1));
26     wd=(workfile(strfind(workfile,'w')+1:strfind(workfile,'f')-1));
27 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```

28 % extraccion de datos del experimento indicado por nombre de archivo
29 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
30 deep=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),5);
31 frec_exp=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),3);
32 T=1/frec_exp;
33 dist_chn=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),4);
34 block_w=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),7);
35 block_l=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),6);
36 Cam_a=cpl.data(find(cpl.data(:,1)==str2num(nvid) & cpl.data(:,2)==str2num(wd)),10);
37 factor=Cam_a/64;
38 chn_w=str2num(wd)/10; %ancho del canal en cm.
39 l.br.m=1.50; %brazo reciprocante??
40 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
41 %apertura de archivo para guardar trayectoria promediada.
42 Promname=['c' numcl 'pr' nvid 'w' wd 'f' num2str(frec_exp,2) 'd' num2str(dist_chn) '.txt'];
43 Clpr=fopen(Promname, 'w');
44 Errpromname=['Err' numcl 'pr' nvid 'w' wd 'f' num2str(frec_exp,2) 'd' num2str(dist_chn) '.txt'];
45 CLerrpr=fopen(Errpromname, 'w');
46 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
47 % calculo teorico del numero de Strouhal y Reynolds
48 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
49 U=(2*pi*frec_exp*block_l*block_w*l.br.m)/(chn_w*deep);
50 Num.Strhl=chn_w/(U*T);
51 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
52 % % ciclo para remover valores iguales en t, y evitar problemas en la
53 % interpolacion de los datos.
54 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
55 fprintf('remocion_de_y_repetidas.\n');
56 while i<cl_wfr-1
57     for i=1:(cl_wfr-1)
58         if cl_wf(i+1,1)-cl_wf(i,1)==0
59             cl_wf(i,:)=[];
60             [cl_wfr,cl_wfc]=size(cl_wf);
61         break
62     end
63 end
64 end
65 [cl_wfr, cl_wfc]=size(cl_wf);
66 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
67 % % Interpolacion de las series de datos
68 % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
69 % creacion de particion fina para todo el conjunto sin incluir
70 % retrocesos
71 yi=cl_wf(1,1);
72 cl_wftemp=cl_wf;
73 cl_wftemp(:,1)=[];
74 cl_wfi=[];
75 Y_i=[];
76 m=1;
77 fprintf('particion_fina_en_y\n');

```

```

78   for i=2:cl_wfr
79       if (cl_wf(i,1)-cl_wf(i-1,1))<=0 %E (cl_wf(i+1,2)-cl_wf(i-1,2))<=0
80           % tiempo final para intervalo de y a particionar
81           yf=cl_wf(i,1);
82           % vector renglon con los valores finos solo entre m - i
83           Y_f=[yi:-0.001:yf];
84           % vector renglon con todos los valores finos de posicion
85           Y_i=[Y_i Y_f];
86           % interpolacion para cada coordenada en funcion de y
87           [ysize,gbg]=size(Y_i');
88           %fprintf(int2str(ysize));
89           %fprintf('\n');
90           cl_wfi=[cl_wfi;interp1(cl_wf(m:i,1),cl_wf(m:i,:),Y_f)];
91           yi=cl_wf(i,1);
92           m=i;
93       end
94       if cl_wf(i,1)-cl_wf(i-1,1)>0
95           yi=cl_wf(i,1);
96       end
97   end
98   cl_wf1=[Y_i' cl_wfi];
99   [cl_wfr1, cl_wfc1]=size(cl_wf1);
100  % % % %
101  %recorte de las y's extremas para que todas midan lo mismo
102  % % % %
103  [cl_wfr1,cl_wfc1]=size(cl_wf1);
104  lst_min=[];
105  lst_max=[cl_wf1(1,1)];
106  for i=1:cl_wfr1-1
107      if cl_wf1(i+1,1)-cl_wf1(i,1)>0
108          lst_min=[lst_min,cl_wf1(i,1)];
109          lst_max=[lst_max,cl_wf1(i+1,1)];
110      end
111  end
112  Sm_max=min(lst_max);
113  Bg_min=max(lst_min);
114  lst_rm_y=[];
115  for i=1:cl_wfr1
116      if cl_wf1(i,1)>Sm_max
117          lst_rm_y=[lst_rm_y, i];
118      end
119      if cl_wf1(i,1)<Bg_min
120          lst_rm_y=[lst_rm_y, i];
121      end
122  end
123  cl_wf1rem=removerows(cl_wf1,'ind',lst_rm_y);
124  cl_wf1=cl_wf1rem;
125  [cl_wfr1,cl_wfc1]=size(cl_wf1);
126  % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
127  % busqueda del numero de veces e indice donde suceden las repeticiones

```

```

128 % o numero de periodos para cl_wfl
129 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
130 % lista con los indices donde suceden los cambios de periodo.
131 p_nr=[1] ;
132 % numero de periodos que corrio el experimento
133 k=1;
134 flag1ciclo=1;
135 for i=2:cl_wfl
136     if (cl_wfl(i,1)-cl_wfl(i-1,1))>0
137         p_nr=[p_nr,i];
138         k=k+1;
139     end
140     % aado el ultimo indice a la lista para tener cota inf.
141     if i == cl_wfl
142         p_nr=[p_nr,i];
143     end
144 end
145 % control para evitar errores si solo se film un ciclo
146 if k==1
147     flag1ciclo=0;
148 end
149 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
150 % matriz con tiempos retrocedidos el numero de periodos correspondientes
151 % y flag_incmp de remocion de ultimo periodo en caso de ser incompleto.
152 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
153 % vecindad donde se considera periodo completo
154 delta_y=0.01*10;
155 flag_incmp=0;
156 if flag1ciclo
157     for i=1:cl_wfl
158         for l=1:k
159             % desplazamiento de t para el ultimo periodo
160             if i==p_nr(k+1)&l==k
161                 cl_wfl(i,4)=cl_wfl(i,4)-(l-1)*T;
162                 %cl_wfl(i,4);
163                 % si el ultimo elemento del ultimo periodo es
164                 % delta-menor que ultimo elemento del periodo
165                 % anterior, levanto bandera flag_incmp
166                 if (cl_wfl(i,4) <= (cl_wfl(p_nr(k)-1,4)-delta_y))
167                     %flag_incmp=1;
168                 end
169             end
170             % desplazamiento de t para los peirodos k-1
171             if (i>=p_nr(l) & i<p_nr(l+1))
172                 cl_wfl(i,4)=cl_wfl(i,4)-(l-1)*T;
173                 %break
174             end
175         end
176     end
177 end

```

```

178 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
179 % matriz para realizar promedio de los valores en diferentes periodos
180 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
181 fprintf('matriz_de_promedio_\n');
182 Y_max=max(cl.wfl(:,1));
183 Y_min=min(cl.wfl(:,1));
184 % numero de renglones de matriz Mf_rp
185 p=round((Y_max-Y_min)/0.001);
186 % numero de columnas de matriz Mf_rp es de
187 % 3 dimensiones por cada periodo mas el tiempo
188 % a menos que flag-incmp este activa
189 if flag-incmp
190     %un periodo menos.
191     k=k-1;
192 end
193 q=(3*k)+1;
194 % creacion de matriz de nan's de prq
195 if flag1ciclo
196     Mf_rp=nan(p,q);
197     for i=1:cl.wf1
198         for j=1:cl.wf1
199             % ciclo para identificar a que periodo corresponde el renglon
200             % en funcion de la lista p_nr
201             for l=1:k
202                 % si j es de las primeras tres columnas (X, Y, Z)
203                 if (j>=1 & j<=3)
204                     % si i esta entre inicio y fin de periodo l-esimo
205                     if (i>=p_nr(l) & i<p_nr(l+1))
206                         Mf_rp(round((cl.wfl(i,1)-Y_min)/0.001)+1,k*(j-1)+1)=cl.wfl(i,j+1);
207                         break
208                     end
209                 end
210                 % si j no es de la primera 3 columnas ent. es la ultima (que va en q).
211                 if j==4
212                     Mf_rp(round((cl.wfl(i,1)-Y_min)/0.001)+1,q)=cl.wfl(i,j-3);
213                 end
214             end
215         end
216     end
217 end
218 if ~ flag1ciclo
219     Mf_rp=cl.wf1;
220 end
221 %busqueda de renglones de la matriz Mf_rp que contienen
222 %nan, para evitar dislocaciones en la grafica promediada.
223 fprintf('matriz_nan_\n');
224 [Mf_rpr,Mf_rpc]=size(Mf_rp);
225 ind_nan=0;
226 lst_rm=[];
227 for i=1:Mf_rpr

```

```

228     for j=1:Mf_rpc
229         % si encuentras nan en (i,j)
230         if isnan(Mf_rp(i,j))
231             % guarda la el renglon (i) donde sucede
232             ind_nan_temp=i;
233             % compara valor de renglon para solo apuntar
234             % una vez el indice de renglon en la lista lst_rm
235             if ind_nan_temp ~= ind_nan
236                 % si es renglon distinto aadelo a lst_rm
237                 lst_rm=[lst_rm, ind_nan_temp];
238                 % cambia el renglon
239                 ind_nan=ind_nan_temp;
240             end
241         end
242     end
243 end
244 [Mf_rpnns,PS]=removerows(Mf_rp,lst_rm);
245 Mf_rp=Mf_rpnns;
246 [Mf_rpr,Mf_rpc]=size(Mf_rp);
247 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
248 % Promedio de los diferentes recorridos. Guardandolos convertidos a cm (X,Y)
249 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
250 fprintf('promedio_de_recorridos_\n');
251 cl_wfp=[];
252 for i=1:Mf_rpr
253     for j=1:3
254         if flag1ciclo
255             cl_wfp(i,j)=nanmean(Mf_rp(i,k*(j-1)+1:k*(j-1)+k));
256         end
257         if ~ flag1ciclo
258             cl_wfp(i,j)=Mf_rp(i,j);
259         end
260     end
261     cl_wfp(i,4)=Mf_rp(i,Mf_rpc);
262 end
263 cl_wfp(:,[1,2,3,4])=cl_wfp(:,[4,1,2,3]);
264 [cl_wfpr,cl_wfpc]=size(cl_wfp);
265 for i=1:cl_wfpr
266     fprintf(Clpr,'%f_%f_%f_%f\n',cl_wfp(i,1)*factor,cl_wfp(i,2)*factor,cl_wfp(i,3),cl_wfp(i,4));
267 end
268 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
269 % conversion de la subdivision 64/Cam_a
270 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
271 % para el conjunto de datos
272 for i=1:cl_wf1
273     for j=1:2
274         if j==1
275             cl_wf1(i,j)=cl_wf1(i,j)*factor;
276         end
277     if j==2

```

```

278         cl_wfl(i,j)=cl_wfl(i,j)*factor;
279     end
280 end
281 end
282 % para los datos promediados
283 for i=1:cl_wfpr
284     for j=1:2
285         if j==1
286             cl_wfp(i,j)=cl_wfp(i,j)*factor;
287         end
288         if j==2
289             cl_wfp(i,j)=cl_wfp(i,j)*factor;
290         end
291     end
292 end
293 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
294 % generacion de graficas.
295 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
296 f=figure('visible','off');
297 % Grafica Posicion 'Y' vs t
298 MxY=max(cl_wfl(:,1));
299 for i=1:k
300     if i>1
301         hold on;
302         hold all;
303     end
304     if i<k
305         plot(cl_wfl(p_nr(i):p_nr(i+1)-1,4),MxY-cl_wfl(p_nr(i):p_nr(i+1)-1,1),'');
306     else
307         plot(cl_wfl(p_nr(i):cl_wfr1,4),MxY-cl_wfl(p_nr(i):cl_wfr1,1),'');
308     end
309 end
310 plot(cl_wfp(:,4),MxY-cl_wfp(:,1),'k');
311 title(['y_vs_t'];['_\tau=',num2str(T),'_cancho_de_canal_',num2str(chn_w),'cm'],'_St=',num2str(
    Num_Strhl]));
312 xlabel('t_(s)');
313 ylabel('y_(cm)');
314 Pgraph_name=sprintf('%s %s %0.0f %s %0.2f %s %0.0f %s',num2str(numcl),'yt_w',chn_w*10,'F',
    frec_exp,'d',dist_chn,'.jpg');
315 command1=['print_-depsec_', Pgraph_name,'.eps'];
316 eval(command1);
317 hold off;
318 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
319 % Grafica posiocion x vs y
320 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
321 for i=1:k
322     if i>1
323         hold on;
324         hold all;
325     end
326     if i<k

```



```

327     plot(MxY-cl_wfl(p_nr(i):p_nr(i+1)-1,1),cl_wfl(p_nr(i):p_nr(i+1)-1,2),'t');
328     else
329     plot(MxY-cl_wfl(p_nr(i):cl_wfr1,1),cl_wfl(p_nr(i):cl_wfr1,2),'t');
330     end
331     end
332     plot(MxY-cl_wfp(:,1),cl_wfp(:,2),'k');
333     title(['x_vs_y'];['_\tau=' num2str(T), '_ancho_de_canal_' num2str(chn_w),'cm', '_distancia_del_
canal_' num2str(dist_chn), '_St=' num2str(Num_Strhl)]];
334     xlabel('y_(cm)');
335     ylabel('x_(cm)');
336     Dgraph_name=sprintf('%s %s %0.0f %s %0.2f %s %0.0f %s' ,num2str(numcl),'xy_w',chn_w*10,'f',
freq_exp,'d',dist_chn,'jpg');
337     command2=['print_-depesc_', Dgraph_name, '.eps'];
338     eval(command2);
339     hold off;
340     % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
341     % % Grafica posiocion z vs t
342     % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
343     for i=1:k
344         if i>1
345             hold on;
346             hold all;
347         end
348         if i<k
349             plot(cl_wfl(p_nr(i):p_nr(i+1)-1,4),deep-cl_wfl(p_nr(i):p_nr(i+1)-1,3),'t');
350         else
351             plot(cl_wfl(p_nr(i):cl_wfr1,4),deep-cl_wfl(p_nr(i):cl_wfr1,3),'t');
352         end
353     end
354     plot(cl_wfp(:,4),deep-cl_wfp(:,3),'k');
355     title(['z_vs_t'];['_\tau=' num2str(T), '_ancho_de_canal_' num2str(chn_w),'cm', '_distancia_del_
canal_' num2str(dist_chn), '_St=' num2str(Num_Strhl)]];
356     xlabel('t_(s)');
357     ylabel('z_(cm)');
358     Dgraph_name=sprintf('%s %s %0.0f %s %0.2f %s %0.0f %s' ,num2str(numcl),'zt_w',chn_w*10,'f',
freq_exp,'d',dist_chn,'jpg');
359     command3=['print_-depesc_', Dgraph_name, '.eps'];
360     eval(command3);
361     hold off;
362     % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
363     % % Grafica posiocion z vs y
364     % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
365     for i=1:k
366         if i>1
367             hold on;
368             hold all;
369         end
370         if i<k
371             plot(cl_wfl(p_nr(i):p_nr(i+1)-1,1),deep-cl_wfl(p_nr(i):p_nr(i+1)-1,3),'t');
372         else
373             plot(cl_wfl(p_nr(i):cl_wfr1,1),deep-cl_wfl(p_nr(i):cl_wfr1,3),'t');

```

```

374     end
375 end
376 plot(cl_wfp(:,1),deep-cl_wfp(:,3),'k');
377 title({'z_vs_y';['_\tau=' num2str(T), '_ancho_de_canal_' num2str(chn_w), 'cm', '_distancia_del_'
canal_' num2str(dist_chn), '_St=' num2str(Num_Strhl)]});
378 xlabel('y(cm)');
379 ylabel('z(cm)');
380 Dgraph_name=sprintf('%s %s %0.0f %s %0.2f %s %0.0f %s', num2str(numcl), 'zy_w', chn_w*10, 'f',
frec_exp, 'd', dist_chn, 'jpg');
381 command4=['print--depsec_', Dgraph_name, '.eps'];
382 eval(command4);
383 hold off;
384 fclose(Clpr);
385 fclose(Clerrpr);
386 clearvars -except cpl cl_files cl_file
387 end
388 fprintf('trprom_terminado_\n');
389 fprintf('comenzando_vortex_vel_dist_\n');
390 vortex_vel_dist
391 f=figure('visible','on');

```

Apéndice E

Programa en Matlab graficador

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  %calculador de velocidad de los centroides. vortex_vel_dist.m
3  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4  clear;
5  c_files = dir('c\scpr*w*.txt');
6  err_file=fopen('errorfile.txt','w');
7  fprintf(err_file,'%s\n','archivos_inexistentes_por_no_haber_obtenido_promedio');
8  for c_file = c_files'
9      fid_c=fopen(c_file.name,'rt');
10     nLines=0;
11     while (fgets(fid_c) ~= -1)
12         nLines=nLines+1;
13     end
14     if c_file.bytes==0 | nLines < 20
15         fprintf(err_file,'%s\n', c_file.name);
16         continue;
17     end
18     % importando datos del par de vortices del experimento
19     workfile=c_file.name;
20     workfile2=['c2',(workfile(strfind(workfile,'p')-2:end))];
21     if ~exist(sprintf('%s %s %s',' ',workfile2,''))
22         fprintf(err_file, '%s\n', workfile2);
23     continue;
24     end
25     vel_chart=workfile(strfind(workfile,'r'):strfind(workfile, '.t')-1);
26     c1_wf=importdata(workfile);
27     c2_wf=importdata(workfile2);
```



```

78     for i=1:c2_wfr
79         if c2_wf(i,4)>=Tmin && c2_wf(i,4)<=Tmax
80             j=round((c2_wf(i,4)-Tmin)/0.04)+1;
81             Dt(j,2)=c2_wf(i,2);
82         end
83     end
84     %distancias entre vortices
85     [Dtr,Dtc]=size(Dt);
86     DsT=nan(Dtr,2);
87     for i=1:Dtr
88         DsT(i,1)=abs(Dt(i,1)-Dt(i,2));
89         DsT(i,2)=Dt(i,3);
90     end
91     f=figure('visible','off');
92     hold on
93     hold all
94     plot(DsT(:,2),DsT(:,1),'r');
95     plot(fmvel1,'b');
96     plot(fmvel2,'g');
97     title(workfile);
98     xlabel('t_s');
99     ylabel('v_(cm/s)');
100    Dgraph_name=sprintf('%s %s %s','velocity_chart','vel_chart','.jpg');
101    hgexport(gcf, Dgraph_name, hgexport('factorystyle'),'Format','jpeg');
102    hold off
103    fclose(vfile1);
104    fclose(vfile2);
105    clearvars -except c_files c_file err_file
106 end
107 fclose(err_file);
108 fprintf('velocidad_de_vortices_y_distancia_entre_vortices_obtenidos.\n');

```