

**An Experimental Testbed for Creating Controlled Lagrangian Coherent  
Structures**

A Thesis

Submitted to the Faculty

of

Drexel University

by

Dennis J. Larkin

in partial fulfillment of the

requirements for the degree

of

Master of Science in Mechanical Engineering and Mechanics

September 2015

© Copyright September 2015  
Dennis J. Larkin. All Rights Reserved.

### **Dedications**

To my family for their unending support and encouragement.

### **Acknowledgements**

I would like to thank my advisor, Dr. Ani Hsieh, for her support and guidance over the past few years through this work and the rigors of graduate studies. I would like to thank Dr. Eric Forgoston and Dr. Phil Yecko as well for the collaborative efforts in this project. Thanks are also due to Alex Abad and Zachary Block for their help, and to my lab mates at the Scalable Autonomous Systems lab: Matt, Ken, James, Bill, Milligan, Dhanushka, Dan and Hadi.

This thesis is based upon the work supported by the National Science Foundation (NSF) grant IIS-1253917 and OISE-1131011.

## Table of Contents

List of Figures .....	iii
Abstract .....	vi
1. Introduction.....	1
1.1 Motivation .....	1
1.2 Related Work .....	3
1.3 Contributions of this thesis .....	3
2. From Models to Data .....	5
2.1 Flow Model .....	5
2.2 Tanks .....	6
2.3 Particle Image Velocimetry .....	6
3. Analysis .....	8
3.1 Lagrangian Coherent Structures.....	8
3.2 Finite Time Lyapunov Exponents.....	9
3.3 Koopman operator, eigenfunctions and modes .....	12
3.4 Dynamic Mode Decomposition .....	13
4. Design of the Experimental Infrastructure .....	17
4.1 Tanks .....	17
4.1.1 Low Reynolds Tank .....	18
4.1.2 High Reynolds Tank .....	19
4.1.3 Multi-Robot Tank .....	21
4.2 Flow Generation .....	21
4.2.1 Flow Generation System.....	22
4.2.2 Motor Control .....	25
4.3 Data Collection .....	27
5. Experimental Method .....	28
5.1 Experimental Procedure .....	28
5.1.1 PIV Data Collection Parameters .....	29
5.1.2 PIV Data Processing Parameters .....	30
6. Experimental Results .....	31
6.1 Overview .....	31

6.2	PIV Resultant Vector Fields.....	31
6.3	FTLE Analysis.....	33
6.3.1	Choice of Integration Time .....	33
6.3.2	Experimental FTLE vs Double Gyre Model .....	38
6.4	Invariance of Ridge Location .....	39
6.5	Dynamic Mode Decomposition .....	46
7.	Applications.....	50
7.1	Multi-Robot LCS Tracking Strategy .....	50
7.2	Autonomous Surface Vehicles.....	52
8.	Conclusion and Future Work .....	54
8.1	Conclusions .....	54
8.2	Future Work .....	54
A.	Supplementary Images .....	57
A.1	PIV Resolved Vector Fields.....	57
A.2	Resultant FTLE fields.....	60
	Bibliography .....	66

## List of Figures

2.1 Phase portrait for the model given by (2.1) with $A = 1$ , $\mu = 0$ , $\varepsilon = 0$ , $\psi = 0$ , and $s = 1$ ....	5
3.1 Diagram illustrating the initial and final locations of 4 advected particles in a flow.....	10
3.2 (a) Static double gyre vector field and (b) corresponding FTLE field for (2.2) with $A = 1$ and $s = 1$ .....	11
4.1 The LoRe tank, showing (a) independently controlled motors and (b) bank of spinning cylinders.....	18
4.2 The HiRe tank setup. Pictured is the tank with a 12 motor array set-up inside the control box, the overhead PIV camera system and the black lights used to illuminate the tracer particles.....	20
4.3 Experimental setup of flow tank with 12 driven cylinders in the control box.....	20
4.4 MR tank and overhead motion capture camera system.....	21
4.5 The HiRe tank Flow Driver. On the left is the encapsulated motor attached to its weighted base. On the right is driven cylinder that interacts with the fluid medium to generate flow.....	23
4.6 The flow driver for the Multi-Robot tanks .....	24
4.7 (a) The water tight unit housing the motor control boards. (b) The power supply unit and Arduino which provide power to the motor controller box and communication signals between the controllers and the PC. ....	26
6.1 A comparison of (a) the PIV resolved velocity field from HiRe tank data and (b) the velocity field from the Double Gyre model. ....	32
6.2 The vector field of the double-gyre model with particles initially placed straddling a known LCS boundary and advected forward in time by the flow.....	33
6.3 Distance separated vs. FTLE values for the double-gyre model with different integration times. ....	35
6.4 Distance separated vs. FTLE values for experimental data with different integration times.	37
6.5 Snapshot of the first run at 45 rpm with an integration time of $T = 4.0$ s.....	38
6.6 A comparison of (a) the FTLE for a time invariant double-gyre (b) against the time series average FTLE value of experimental data and (c) an instantaneous snapshot of the FTLE field. ....	40
6.7 Time series average FTLE field with ridge locations denoted by black marks found using the weighted average method discussed in 6.4.....	42
6.8 Snapshot FTLE field with ridge locations denoted by black marks found using the weighted average method discussed in 6.4. ....	43

6.9 A surface plot showing the complex topology of the snapshot FTLE field .....	43
6.10 A view of the FTLE ridge in Fig. 6.8 with the $1-\sigma$ deviation plotted in blue '+' signs and the time series average value plotted in black '*' symbols.....	44
6.11 The combined, forward and backward, FTLE for the full reconstruction of the flow field from a 14 mode DMD computation.....	47
6.12 The combined, forward and backward, FTLE for the primary mode reconstruction of the flow field from a 14 mode DMD computation.....	47
6.13 The combined, forward and backward, FTLE for the full reconstruction of the flow field from a 59 mode DMD computation.....	48
6.14 The combined, forward and backward, FTLE for the primary mode reconstruction of the flow field from a 59 mode DMD computation.....	48
6.15 A plot of the imaginary vs. real parts of the Ritz values computed for a 59 mode DMD computation. The color of each point displays the magnitude. The first 4 dominate modes are annotated. .....	49
7.1 .....	51
7.2 Examples of the two classes of autonomous surface vehicles used for this work, with and without 3D printed cap. Pictured on the left are the mASVs, and on the right are the mASVFs with flow sensors.....	53
A.1 Snapshot of PIV resolved vector field for the first run at 60 rpm .....	57
A.2 Snapshot of PIV resolved vector field for the first run at 75 rpm .....	58
A.3 Snapshot of PIV resolved vector field for the first run at 90 rpm .....	58
A.4 Snapshot of PIV resolved vector field for the first run at 105 rpm .....	59
A.5 Snapshot of PIV resolved vector field for the first run at 120 rpm .....	59
A.6 Snapshot FTLE for first run at 60 rpm .....	60
A.7 Time series average FTLE for first run at 60 rpm .....	60
A.8 Snapshot FTLE for first run at 75 rpm.....	61
A.9 Time series average FTLE for first run at 75 rpm .....	61
A.10 Snapshot FTLE for first run at 90 rpm.....	62
A.11 Time series average FTLE for first run at 90 rpm .....	62
A.12 Snapshot FTLE for first run at 105 rpm .....	63
A.13 Time series average FTLE for first run at 105 rpm.....	63

A.14 Snapshot FTLE for first run at 120 rpm .....	64
A.15 Time series average FTLE for first run at 120 rpm.....	64

**Abstract**

An Experimental Testbed for Creating Controlled Lagrangian Coherent Structures

Dennis J. Larkin

Advisor: M. Ani Hsieh, Ph.D.

In recent years there has been a surge in the development and deployment of Autonomous Surface and Underwater Vehicles for the continuous monitoring of dynamical processes occurring in the ocean such as biological and physical phenomenon. However, motion planning and control strategies for these systems are quite challenging to develop as they operate in a continuously time-varying environment that is both stochastic and aperiodic. Recently it has been shown that although these oceanic flows are stochastic and aperiodic they exhibit patterns which strongly govern transport properties. We are specifically interested in a class of these coherent structures known as Lagrangian Coherent Structures (LCS). Real time mapping of these structures is challenging due to their lagrangian nature. However, recent works have shown the feasibility of using a team of autonomous robots to track the location of these structures in real-time. Field validation of these and similar tracking strategies is often cost and time prohibitive. This paper describes the development of an experimental testbed that can reliably produce LCS at desired locations for sustained periods of time. The goal is to develop an inexpensive research instrument that can be employed to validate various autonomous vehicle deployment strategies subject to realistic flow conditions.



## 1. Introduction

### 1.1 Motivation

In recent years there has been a surge in the development and deployment of Autonomous Surface and Underwater Vehicles (ASV/AUV) for the continuous monitoring of dynamical processes occurring in the ocean such as biological and physical phenomenon. However, motion planning and control strategies for these systems are quite challenging to develop as they operate in a continuously time-varying environment that is both stochastic and aperiodic.

Despite various attempts by Government, Public and Private groups over the past 40 years to measure ocean flows through a combination of satellite, stationary, surface, and at depth sampling technologies existing data sets are still of finite-time and of low spatio-temporal resolution. The distance between sampling points versus the length of typical ASV's lends to a scaling issue with the incorporation of this data for motion planning and control strategies.

Recently it has been shown that although these oceanic flows are stochastic and aperiodic they exhibit patterns which strongly govern transport properties. We are specifically interested in a class of these coherent structures known as Lagrangian Coherent Structures (LCS) which are differentiated by the use of a physical measure from the flow field. These structures are important because they characterize transport properties and are quantifiable by local measures of Finite-Time Lyapunov Exponents (FTLE). The FTLE are similar to Lyapunov exponents in dynamical systems analysis in that they describe a local measure for the rate of separation of initially nearby trajectories, the difference being that FTLE uses a finite time horizon. LCS are representative of separatrices that divide the phase space of the system into dynamically distinct regions. The FTLE analysis for 2D-flow results in a field of scalar values where LCS present themselves as 1-dimensional boundaries that exhibit maximum FTLE measures. LCS boundaries have also been shown to coincide with fuel optimal paths in the ocean. These structures are also useful for defining boundaries of stochastic eddies, and have been shown to be useful for predicting contaminant transport in the ocean. In addition, LCS have been shown to correspond to regions of a flow where mobile sensors tend to escape their monitoring region of interest.

Mapping the locations of the LCS present in real ocean flows would thus be of great benefit. Unfortunately, these maps cannot be computed in real time due to the Lagrangian basis being laid

on the evolution of the flow. Instead the region of interest must be sampled, processed and then after some finite period of time we have a map. This map may now be useless though as the flows are typically time-varying and aperiodic, meaning that in the time to compute the FTLE and find the LCS the system may have changed and the structures are no longer present where they appear on our map.

Although, it may be impossible to construct a real-time map of the LCS by sampling the desired velocity field and computing the FTLE it is possible to track the structures in real-time. This can be done using the proper interior maximum (PIM) triple method (reference). A numerical technique used to locate trajectories in a chaotic dynamical system which remain in a region with no attractors for an arbitrarily long period of time. The PIM triple technique is capable of finding these trajectories using only local knowledge of the phase space of the system at three points along a line segment crossing the trajectory. Interpolating these measurements along the line segment yields a minimum velocity which shows the direction of the trajectory. Extending this to the real-time tracking of LCS is therefor, in theory, quite simple. A multi-robot control strategy for tracking the LCS using the PIM triple technique with 3 ASV's was devised.

This focus of this paper is the construction and validation of a testbed for the evaluation of the multi-robot LCS tracking strategy. The objective was to design a testbed capable of producing controllable flows that exhibit similar dynamical features as those found in the ocean. In addition, the testbed should enable the gathering of high resolution data of realistic quasi-2D flows for analysis and further development of distributed sensing and ocean monitoring strategies.

## 1.2 Related Work

At the current time there are many works utilizing the power of LCS for the analysis of fluid flows such as the original works by Haller [13–18, 37, 38] discussing the use and properties of LCS with a focus on the study of aperiodic flows and the identification of such structures. To Shadden [1, 22, 42] who utilized LCS analysis to show optimal glider paths and study aperiodic flows. These studies either look at the use of LCS as a tool to help understand complex dynamics in the systems they study or evaluate the inherent properties of LCS. Therefore, to our knowledge this is the first work attempting to produce LCS in a controlled laboratory setting for the purpose of creating consistent LCS for the evaluation of the multi-agent tracking strategy. The work presented here does not try to study the properties of the fluid flow itself but rather how to create these structures in desired locations and pattern for optimal testing of the tracking strategy in a consistent and repeatable fashion.

## 1.3 Contributions of this thesis

This work presents the design of an experimental testbed capable of creating controlled Lagrangian coherent structures in gyre like flows in laboratory setting. Specifically, the goal is to develop an experimental testbed that can be used to validate existing and future distributed autonomous sensing strategies in stochastic and time-varying environments. Towards this end, the main contribution of this thesis includes:

1. The development of two flow tanks equipped with gyre flow driving mechanisms capable of creating controlled and repeatable Lagrangian coherent structures; and
2. Validation of the flows created in the high Reynolds number (HiRe) tank against the analytical flow model given by the wind driven double-gyre model.

Based on our results, similar flows can be created in the Multi-Robot (MR) tank where multi-robot tracking and controls strategies can be validated [34].

This thesis presents a detailed exposition and discussion of these contributions and is organized as follows. Chapter 2 presents an overview of the model the system presented in this thesis was designed to replicate as well as how data will be collected. Chapter 3 summarizes our analysis techniques. Chapter 4 gives a detailed look at all of the components that make up the system with a brief summary of their function. Chapter 5 lays-out in detail the experimental procedures

followed for collecting data. Chapter 6 then presents the results of the experiment. Chapter 7 discusses applications of producing controlled LCS. Conclusions and future work are then presented in chapter 8.

## 2. From Models to Data

This chapter begins by describing the flow model that was used for the development of an experimental testbed for validating and improving the multi-robot Lagrangian Coherent Structure tracking strategy. It then proceeds with a brief explanation of gathering data on an experimental flow by using Particle Image Velocimetry (PIV).

### 2.1 Flow Model

The primary model used for the development and evaluation of the multi-robot coherent structure testbed has been the wind-driven double gyre model. The mathematical representation is given by

$$\dot{x} = -\pi A \sin\left(\pi \frac{f(x,t)}{s}\right) \cos\left(\pi \frac{y}{s}\right) - \mu x, \quad (2.1a)$$

$$\dot{y} = \pi A \cos\left(\pi \frac{f(x,t)}{s}\right) \sin\left(\pi \frac{y}{s}\right) \frac{df}{dx} - \mu y, \quad (2.1b)$$

$$f(x,t) = \varepsilon \sin(\omega t + \psi) x^2 + (1 - 2\varepsilon \sin(\omega t + \psi))x. \quad (2.1c)$$

Where in (2.1),  $A$  approximately determines the amplitude of the velocity vectors,  $\omega/2\pi$  gives the oscillation frequency,  $\varepsilon$  determines the amplitude of the left-right motion of the separatrix between the gyres,  $\psi$  is the phase,  $\mu$  determines the dissipation, and  $s$  scales the dimensions of the workspace. When  $\varepsilon = 0$ , the double-gyre flow is time-independent, while for  $\varepsilon \neq 0$ , the gyres undergo a periodic expansion and contraction in the  $x$  direction.

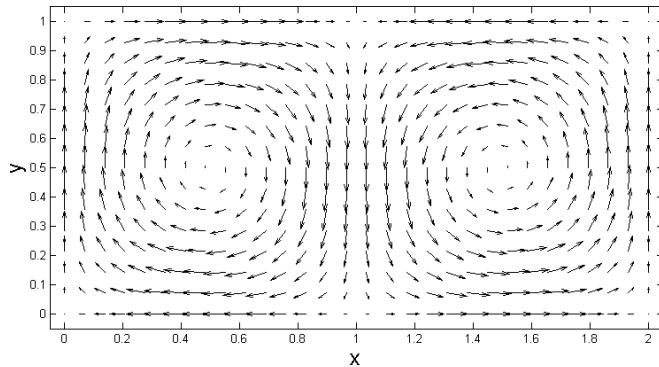


Figure 2.1: Phase portrait for the model given by (2.1) with  $A = 1$ ,  $\mu = 0$ ,  $\varepsilon = 0$ ,  $\psi = 0$ , and  $s = 1$ .

The wind driven double gyre model was chosen due to its long standing history in oceanography modeling real currents as well as the fact that it has been shown to exhibit Lagrangian Coherent Structures (LCS). The LCS in the double gyre model coincide with the separatrices between the gyres. Setting  $\varepsilon = 0$  and  $\mu = 0$  the equations are reduced to (2.2), which is the static wind driven double gyre model where there is no movement of the separatrix, thus the location of the LCS is constant.

$$\dot{x} = -\pi A \sin\left(\frac{\pi x}{s}\right) \cos\left(\frac{\pi y}{s}\right), \quad (2.2a)$$

$$\dot{y} = \pi A \cos\left(\frac{\pi x}{s}\right) \sin\left(\frac{\pi y}{s}\right). \quad (2.2b)$$

More information about LCS will be given in the next two sections.

## 2.2 Tanks

In order to produce these structures in real flows a set of tanks were built. The smallest is the Low Reynolds (LoRe) tank and is used for experiments dealing with creating LCS in the low Reynolds regime. The second tank is a bit larger and is titled the High Reynolds (HiRe) tank. This tank is used for studying the production and control of LCS in a high Reynolds regime similar to that of the ocean which is the regime where the autonomous surface vehicles will be implementing the multi-agent LCS tracking strategy. The final tank is dubbed the Multi-Robot (MR) tank which operates at similar Reynolds numbers as the HiRe tank. This tank is used for validation of the multi-agent LCS tracking strategy with actual vehicles. More detailed information can be found on these systems in chapter 4.

## 2.3 Particle Image Velocimetry

The work presented in this paper relies on the ability to resolve vector fields of real fluid flows. This is accomplished using Particle Image Velocimetry (PIV) which is an optical method of flow visualization that gives instantaneous velocities of the flow field. This is accomplished by seeding the flow with particles that accurately follow the dynamics of the fluid. Typically a pulsed laser sheet is used to illuminate the particles on a desired plane of interest. High speed cameras then record a series of images which go through a post-processing procedure to determine the velocity field.

In the post-processing the images are divided into many regions and an autocorrelation or cross-correlation technique is used between the same region at two consecutive time steps to determine the local instantaneous velocity. Repeating this process over the entire sequence of images resolves the velocity flow field over a given period of time. In order for this technique to work accurately several consideration must be taken such as the proper sizing of particles and image acquisition rate.

The determination of particle size is crucial to properly resolving the dynamics of the flow [28]. Ideally from the perspective of tracking the fluid the smaller and less dense the particle the more accurately it follows the streamlines of the flow. This is due to the drag forces on the particle overpowering any inertial forces. However, density is usually defined by the medium that the particles will be suspended in, making size the main parameter to be adjusted. When using smaller particles resolution and light scattering may become issues. A problem known as peak locking occurs when a single particle is represented by too few pixels. Thus with the need to capture these particles clearly on an image and have them large enough to avoid peak-locking a trade-off must be made. Sizing of the particles is typically determined using Stokes number (2.3). Typically it is desired to find the largest particle applicable to the flow being examined.

$$Stk = \frac{\rho d^2 U}{18\mu L} \quad (2.3)$$

For Stokes number  $\rho$  is the particle density,  $d$  is the particle diameter,  $U$  is the free stream velocity,  $\mu$  is the dynamic viscosity of the fluid and  $L$  is the characteristic length of the obstacle. Stokes number characterizes the ratio of drag forces on the particle from the flow to its own inertial forces. Low Stokes numbers indicate a particle that will be dominated by drag forces whilst a large Stokes number represents a particle whose inertial forces will dominate its dynamics. It has been shown that for accurate tracking of the flow a  $Stk \ll 0.1$  produces tracking errors less than 1% [7].

In order to capture the dynamical structures of interest it is also important to sample the field at the proper frequency. The acquisition rate  $f_{acq}$  should be selected such that  $\Delta t = 1/f_{acq}$ , where  $\Delta t$  is time between camera captures, is sufficient for the relevant velocities in the field of view. Increasing  $f_{acq}$  will allow the analysis to capture faster dynamics. There are many different techniques in PIV more information can be found in [27].

### 3. Analysis

In this chapter a description of Lagrangian Coherent Structures (LCS) is presented with an extension of how to compute Finite Time Lyapunov Exponents (FTLE). Following this the methods used to build a reduced order model of the flow field which still contains the dominant characteristics of the flow and more importantly for our purposes leaves the LCS invariant from the data set. Our tool of choice in building a reduced order model was the Dynamic Mode Decomposition (DMD) proposed by [40]. The DMD is based on the approximation of the Koopman operator, which is an infinite dimensional operator, with a finite dimensional operator that is similar which can be used to evaluate the flow field for reduced order modeling.

#### 3.1 Lagrangian Coherent Structures

In time-invariant dynamical systems with hyperbolic fixed points there exist stable and unstable manifolds which separate the phase space into dynamically distinct regions. An example of such manifolds are the separatrices, mentioned in the section on flow models, between the gyres of the wind-driven double gyre model. As we saw with the FTLE analysis of the time-invariant wind driven double gyre model there was a corresponding peak of FTLE values as depicted by the red shades on the FTLE map in Figure 3.2 at the location of these separatrices. These FTLE peaks present themselves as ridges within the FTLE field which we refer to as Lagrangian Coherent Structures. The ridges are characterized by the sharpness of their peaks or more mathematically the magnitude of the gradient of the FTLE field to either side of the ridge. Where the FTLE magnitude declines on either side of the LCS. The larger the magnitude of the gradient to either side of the peak the stronger the separation between initially nearby trajectories. Thus making them analogous to unstable manifolds.

Although in time-varying systems there is no clear notion of stable and unstable manifolds they still exhibit analogous LCS ridges in the FTLE field. Where these LCS still delineate dynamically distinct regions of the flow. These ridges however are also time dependent. One can draw a parallel here between studying time-varying systems using LCS and the study of time-invariant systems through use of stable and unstable manifolds.

In fluid dynamics LCS define transport barriers making them very useful to study. A great

example of this would be having knowledge of these structures during an oil spill. Since they act as no mixing boundaries it would allow for a more effective concentration of clean-up efforts. There are also many biological and physical processes such as algae blooms and salinity profiles that are more readily studied with knowledge of these structures. Aside from an environmental standpoint these structures are also of use to robotics. In the realm of path planning and motion control LCS have been shown to be time and fuel optimal glider paths. Knowledge of these structures and the dynamics of the regions they define could allow vehicles to further harness the power of the local dynamics rather than fighting them.

### 3.2 Finite Time Lyapunov Exponents

Lyapunov exponent analysis is a very common and powerful tool used in dynamical systems theory especially for non-linear time varying systems. Lyapunov exponents are used to determine the rate of separation between two initially close trajectories in the phase space. For a continuous system the maximal Lyapunov exponent can be found with (3.1). Where  $\lambda$  is known as the Lyapunov exponent,  $\delta_0$  is the initial separation between trajectories and  $\delta(t)$  is the separation between the trajectories at time  $t$ .

$$\lambda = \lim_{t \rightarrow \infty} \lim_{\delta_0 \rightarrow 0} \frac{1}{t} \ln \frac{|\delta(t)|}{|\delta_0|} \quad (3.1)$$

Evaluation of the equation shows that the maximal Lyapunov exponent is found by taking the limit as the initial separation of the trajectories goes to zero while time tends to infinity. The maximal Lyapunov exponent is most commonly used for system behavior since as  $t \rightarrow \infty$  the maximal value will overcome the smaller Lyapunov exponents. However, for a system with an  $n$ -dimensional phase space there will be  $n$  Lyapunov exponents.

Finite Time Lyapunov Exponents (FTLE) act as a suitable approximation to the standard Lyapunov exponents. A primary benefit of FTLE is that most observed data sets are of a bounded or finite period of time, making FTLE suitable. Unlike the Okubo-Weiss criterion which measures instantaneous separation rate the FTLE measures the average separation rate [1] of trajectories. This is a crucial distinction as instantaneous streamlines can quickly diverge from particle trajectories. Thus, FTLE more closely represent actual particle trajectories. Although, the FTLE analysis is done over a vector field and results in a field of values at a given time it is not thought of as an Eulerian view but rather as a Lagrangian one do to evaluating the trajectories of the system.

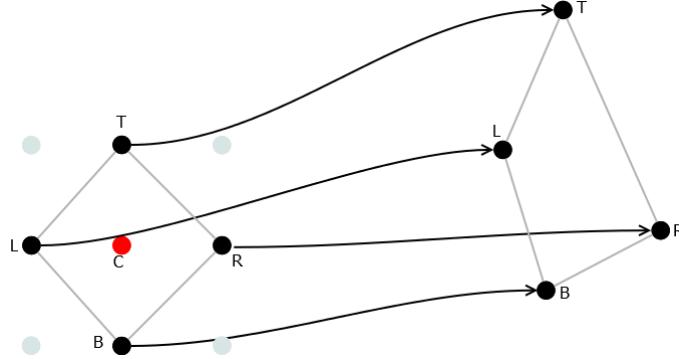


Figure 3.1: Diagram illustrating the initial and final locations of 4 advected particles in a flow.

Computation of the FTLE begins by initiating a grid of particles on the vector field to be analyzed. A basic example in 2-dimensional space of such an initial grid can be seen in the left half of Figure 3.1, where the FTLE evaluation is being done about point  $x_C$  and requires its four immediate neighboring points  $\{x_T, x_R, x_B, x_L\}$ . The four points  $\{x_T, x_R, x_B, x_L\}$  are then advected forward in time through the velocity field for period  $T$ , where  $T$  is the finite integration period time. Advecting the particles is typically done using numerical integration techniques. A central difference approximation is then used to construct the deformation tensor  $G$  about  $x_C$  in (3.2).

$$\mathbf{G} = \begin{bmatrix} \frac{x_R(t_0+T)-x_L(t_0+T)}{x_R(t_0)-x_L(t_0)} & \frac{x_T(t_0+T)-x_B(t_0+T)}{y_T(t_0)-y_B(t_0)} \\ \frac{y_R(t_0+T)-y_L(t_0+T)}{x_R(t_0)-x_L(t_0)} & \frac{y_T(t_0+T)-y_B(t_0+T)}{y_T(t_0)-y_B(t_0)} \end{bmatrix}. \quad (3.2)$$

In this 2-dimensional case the system will have two Lyapunov exponents. We are interested in the maximal value as this will indicate the greatest stretching. Using the  $L_2 - norm$  to find the magnitude of the deformation we come to right Cauchy-Greene deformation tensor  $G^T G$ . Now by letting,

$$\lambda = \max(\text{eig}(\mathbf{G}^T \mathbf{G})). \quad (3.3)$$

We find the FTLE value via the standard Lyapunov equation

$$\sigma_{x_C}(t_0) = \frac{1}{|T|} \ln \sqrt{\lambda}. \quad (3.4)$$

The result is a scalar field representing the average divergence rate of initially nearby trajectories at all points of the initial grid that was placed on the vector field. The resultant FTLE field for the static wind-driven double gyre can be seen in Figure 3.2. Where red shades denote higher values

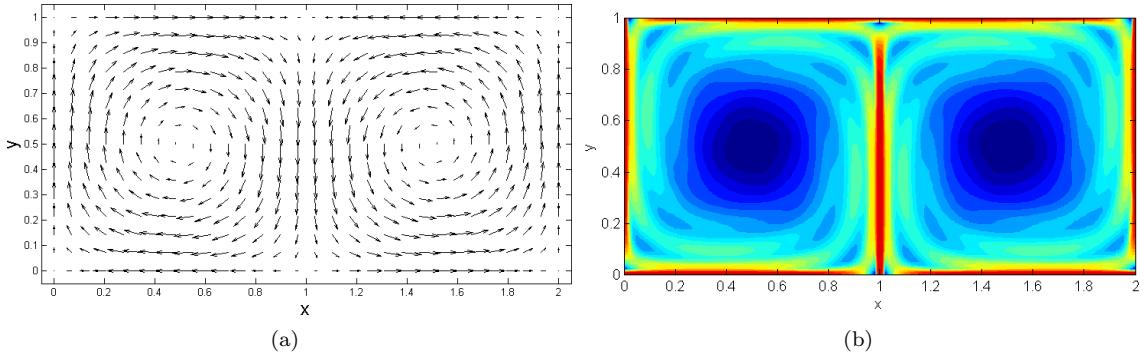


Figure 3.2: (a) Static double gyre vector field and (b) corresponding FTLE field for (2.2) with  $A = 1$  and  $s = 1$ .

of FTLE and the blue shades lower values. Notice that the regions of highest FTLE lie on the separatrixies of the gyres.

It should be noted that the resolution of the FTLE results is dependent on the density of the grid, or number of particles, that are used for the computation. The denser the grid the finer the resolution with the possibility of detecting smaller structures. This increased resolution however comes at the expense of computational time. For a static system one needs only compute the FTLE field once but for a time-varying system it becomes necessary to recompute the field periodically. This is because as the flow changes so may the given FTLE value at any point.

Typically experimental data sets are discrete and require some extra processing. In order to properly advect the particles accurately an interpolation scheme of the velocity vectors is typically implemented. Since vector field interpolation in general is non-unique [39] we arbitrarily chose to separate the components of the velocity into their own fields and interpolated over each of them. A trilinear interpolation scheme was used over both fields as it has been shown to be sufficient for the tracking of coherent structures. In general it has been shown that FTLE maxima are relatively insensitive to the interpolation scheme used for the computation [19].

Another interesting problem with real data is the handling of boundary conditions. Many times in real data sets, as in the tanks we present later, there are structures such as walls or, in more realistic ocean flows, coastlines. The problem here is that due to the integration length particles may leave the region or 'hit a wall' before their full FTLE integration period is completed. When this occurs the FTLE of the particle with its neighbors must be calculated at the time of exit. This clearly throws error into the analysis but is made worse if too many particles leave at once.

### 3.3 Koopman operator, eigenfunctions and modes

The Koopman operator is a composition operator which in this case evolves the system from one state to the next. Given the system

$$\dot{\mathbf{z}} = \mathbf{F}(\mathbf{z}) \quad (3.5)$$

Where  $A$  is the state space and  $\mathbf{z} \in A$ ,  $\mathbf{F}$  is a possibly non-linear vector-valued function such that the dimension of  $\mathbf{F}(\mathbf{z})$  is the same as  $\mathbf{z}$ . Define  $\mathbf{S}^t(\mathbf{z}_0)$  as the position in the state space at time  $t$ . We then define an arbitrary vector-valued observable  $\mathbf{g}$  defined on the Hilbert space such that  $\mathbf{g} : A \rightarrow \mathbb{R}^m$  where  $m$  is the dimension of the observable. Then from the initial condition  $\mathbf{z}_0$  parameterized by  $t$  the observable  $\mathbf{g}$  is given by

$$\mathbf{g}(t, \mathbf{z}_0) = \mathbf{g}(\mathbf{S}^t(\mathbf{z}_0)) \quad (3.6)$$

The family of operators  $U^t$  that work on the vector-space of  $\mathbf{g}$  and are parameterized by  $t$  are

$$U^t \mathbf{g}(\mathbf{z}_0) = \mathbf{g}(\mathbf{S}^t(\mathbf{z}_0)) \quad (3.7)$$

Where for a fixed  $t = \tau$  we see that  $U^\tau : \mathbf{g}(\mathbf{z}_0) \rightarrow \mathbf{g}(\tau, \mathbf{z}_0)$ . The operator  $U^t$  is known as the Koopman Operator. Although the underlying system dynamics may be non-linear and of finite dimension, the Koopman operator is linear and infinite dimensional. The linearity comes from the definition in (3.7). Performing an eigen-decomposition of  $U$  it is possible to express the vector-valued observables  $\mathbf{g}$  in terms of the Koopman eigenfunctions  $\phi_j$  by

$$\mathbf{g}(\mathbf{z}) = \sum_{j=1}^{\infty} \phi_j(\mathbf{z}) \mathbf{y}_j, \quad (3.8)$$

where  $\{\mathbf{y}_j\}$  are the set of vector coefficients which we call *Koopman modes*. Working under the assumption that  $\mathbf{g}$  lies within the span of the eigenfunctions then using the definition in (3.7) and the general eigen-decomposition

$$U \phi_j(\mathbf{z}) = \lambda_j \phi_j(\mathbf{z}), \quad (3.9)$$

we are able to explicitly state

$$\mathbf{g}_k(\mathbf{z}) = \sum_{j=1}^{\infty} \lambda_j^k \phi_j(\mathbf{z}_0) \mathbf{y}_j. \quad (3.10)$$

The growth rate and frequency of each mode is thus dictated by  $\lambda_j$  the set of Koopman eigenvalues. The benefit of this Koopman analysis being that the only information needed about the system is the observables. This is very practical for work on experimental systems where either there is little known about the system dynamics or they are very difficult to model.

### 3.4 Dynamic Mode Decomposition

DMD is an Arnoldi type method for finding eigenvalues and eigenfunctions of systems. Arnoldi techniques are based on the use of Gram-Schmidt orthogonalization process to create a set of  $n$  vectors which span the Krylov subspace  $\mathcal{K}_n$ . These vectors should give a good approximation to the eigenvectors of the system. DMD is quite similar to many other iterative techniques but differentiates itself by harnessing the power of the Koopman operator. This makes the analysis very applicable to experimental data where little is known about the system dynamics. The reason for this being that Koopman analysis doesn't require knowledge of the underlying system matrix. All that is required for the calculation is a time series of snapshots of the observable vector field.

Referring to section 3.1 to show the connection to the Koopman operator. Assume that we have a vector-valued observable  $\mathbf{v}_k = \mathbf{g}(\mathbf{z}_k)$ , where  $\mathbf{g}$  and  $\mathbf{z}$  are the same as before. The DMD technique utilizes the observables  $\{\mathbf{v}_k\}_{k=1}^N$  to find the complex Ritz vectors  $\{\mathbf{y}_j\}_{j=1}^{N-1}$  and values  $\{\lambda\}_{j=1}^{N-1}$  of the system such that

$$\mathbf{v}_k = \sum_{j=1}^{N-1} \lambda_j^k \mathbf{y}_j, \quad k = 1, 2, \dots, N-1 \quad (3.11a)$$

$$\mathbf{v}_N = \sum_{j=1}^{N-1} \lambda_j^N \mathbf{y}_j + \mathbf{r}, \quad \mathbf{r} \perp \text{span}(\mathbf{v}_1, \dots, \mathbf{v}_{N-1}) \quad (3.11b)$$

when  $\{\lambda\}_{j=1}^N$  are distinct. Where  $\mathbf{r}$  is the residual vector of the projection to the  $N$ th observable. The Ritz values and vectors are approximations of the true eigenvalues and eigenvectors and receive their name from the mathematician who pioneered the numerical technique of finding them.

The computation of the DMD algorithm begins by arranging the observable vectors, which we shall now refer to each observable as a snapshots, in sequential order to form the matrix

$$\mathbf{V}_1^N = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N\}, \quad (3.12)$$

where  $\mathbf{v}_i$  is a column vector representing the  $i$ th flow field. In this assignment  $\Delta t$  should be constant between each consecutive snapshot. Assuming the existence of a linear operator  $\mathbf{A}$  that maps  $\mathbf{v}_i$  to  $\mathbf{v}_{i+1}$  such that,

$$\mathbf{v}_{i+1} = \mathbf{A}\mathbf{v}_i \quad (3.13)$$

and that such a mapping is nearly the same for all  $i = \{1, 2, \dots, N-1\}$ . For a non-linear system, such as is our case, this results in a linear tangent approximation to the solution. The structure we have placed on the data sequence with assumption of the mapping  $\mathbf{A}$  allows us to evaluate the sequence as a Krylov sequence (3.14). This will allow us to extract the dynamic characteristics of the system described by  $\mathbf{A}$ .

$$\mathbf{V}_1^N = \{\mathbf{v}_1, \mathbf{A}\mathbf{v}_1, \mathbf{A}^2\mathbf{v}_1, \dots, \mathbf{A}^{N-1}\mathbf{v}_1\} \quad (3.14)$$

Now we must assume that for a system being driven at a steady state that after a certain number of snapshots the data becomes linearly dependent. Leveraging this idea and saying that the vectors  $i = \{1, 2, \dots, N-1\}$  are linearly independent and that the  $N$ th snapshot makes  $\mathbf{V}_1^N$  linearly dependent. Then we can express  $\mathbf{v}_N$  as

$$\mathbf{v}_N = a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \dots + a_{N-1}\mathbf{v}_{N-1} + \mathbf{r} \quad (3.15a)$$

$$\mathbf{v}_N = \mathbf{V}_1^{N-1}\mathbf{a} + \mathbf{r} \quad (3.15b)$$

Where  $\mathbf{v}_N$  is the final snapshot,  $\mathbf{a} = [a_1, a_2, \dots, a_{N-1}]^T$  is a vector of cofactors for the linearly independent set that make  $\mathbf{v}_N$  and  $\mathbf{r}$  is the residual due to being fitted data. Since we don't know what is in the  $A$  matrix we must find a similar matrix whose spectra we can evaluate. We proceed by restating the equations above as,

$$A\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{N-1}\} = \{\mathbf{v}_2, \mathbf{v}_3, \dots, \mathbf{v}_N\} = \{\mathbf{v}_2, \mathbf{v}_3, \dots, \mathbf{V}_1^{N-1}\mathbf{a}\} + \mathbf{r}\mathbf{e}_{N-1}^T, \quad (3.16)$$

which leads to,

$$\mathbf{A}\mathbf{V}_1^{N-1} = \mathbf{V}_2^N = \mathbf{V}_1^{N-1}\mathbf{S} + \mathbf{r}\mathbf{e}_{N-1}^T \quad (3.17)$$

where  $\mathbf{e}_{N-1} \in \mathbb{R}^{N-1}$  as the  $(N-1)$ th unit vector. Where  $\mathbf{S}$  is the companion matrix of  $\mathbf{A}$ , its structure can be seen below.

$$\mathbf{S} = \begin{bmatrix} 0 & 0 & \cdots & 0 & a_1 \\ 1 & 0 & & 0 & a_2 \\ 0 & 1 & & 0 & a_3 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & a_{N-1} \end{bmatrix} \quad (3.18)$$

Performing a QR decomposition of  $\mathbf{V}_1^{N-1}$  one can find the vector  $\mathbf{a} = \mathbf{R}^{-1}\mathbf{Q}^*\mathbf{v}_N$  which gives the last column of  $\mathbf{S}$ . It has been shown by [40] that although this decomposition is analytically correct it yields an ill-conditioned algorithm. It is instead suggested to use the singular value decomposition  $\mathbf{V}_1^{N-1} = \mathbf{U}\Sigma\mathbf{W}^*$  which when substituted into (3.17) yields

$$\mathbf{A}\mathbf{U}\Sigma\mathbf{W}^* = \mathbf{V}_2^N = \mathbf{U}\Sigma\mathbf{W}^*\mathbf{S} + \mathbf{r}\mathbf{e}_{N-1}^T \quad (3.19)$$

Then pre-multiplying by  $\mathbf{U}^*$ , rearranging and dropping the right hand side we find,

$$\mathbf{U}^*\mathbf{A}\mathbf{U} = \mathbf{U}^*\mathbf{V}_2^N\mathbf{W}\Sigma^* = \tilde{\mathbf{S}} \quad (3.20)$$

Where  $\tilde{\mathbf{S}}$  is well-conditioned for calculating the eigenvalues and eigenvectors that approximate those of  $\mathbf{A}$ . Authors of [40] also notes that a further advantage of this operation, besides a more robust calculation of the low dimensional representation of  $A$ , is the opportunity to account for a rank-deficiency in the data sequence  $V_1^{N-1}$  via a restriction to a limited projection basis  $U$  given by the non-zero singular values of  $\Sigma$ . This representation uses only information gathered from observation and requires no previous knowledge of the system dynamics. With the well conditioned  $\tilde{\mathbf{S}}$  a standard Arnoldi algorithm is used to determine the eigenvalues and eigenvector of  $\tilde{\mathbf{S}}$ . The dynamic modes are given by  $\phi_i = \mathbf{U}\mathbf{y}_i$ , where  $\mathbf{y}_i$  is the  $i$ th eigenvector of  $\tilde{\mathbf{S}}$  satisfying  $\tilde{\mathbf{S}}\mathbf{y}_i = \mu_i\mathbf{y}_i$ .

Commonly due to the number of observable points taken in each snapshot the row space may be several orders of magnitude larger than that of the column space. With the SVD algorithm used above this leads to increased computational costs. In the attempt to reduce the computational cost we apply a common technique when using the snapshot method with highly rectangular matrices. The technique continues with the SVD of  $\mathbf{V}_1^{N-1} = \mathbf{U}\Sigma\mathbf{W}^*$  we right multiply  $\mathbf{V}_1^{N-1}$  by it's complex conjugate resulting in a square matrix and a simpler decomposition. Dropping the  $(N-1)$  superscript

for notational convenience below we have,

$$\begin{aligned}
 V_1^* V_1 &= (\mathbf{U} \boldsymbol{\Sigma} \mathbf{W}^*)^* \mathbf{U} \boldsymbol{\Sigma} \mathbf{W}^* \\
 &= \mathbf{W} \boldsymbol{\Sigma}^* \mathbf{U}^* \mathbf{U} \boldsymbol{\Sigma} \mathbf{W}^* \\
 &= \mathbf{W} \boldsymbol{\Sigma}^2 \mathbf{W}^*
 \end{aligned} \tag{3.21}$$

Knowing that  $\boldsymbol{\Sigma}$  is a diagonal matrix and  $\mathbf{W}$  is a unitary operator this operation can be performed much quicker. We can now calculate  $\mathbf{U}$  by rearranging the SVD of  $\mathbf{V}_1^{N-1}$  so that we have  $\mathbf{U} = \mathbf{K} \mathbf{W} \boldsymbol{\Sigma}^*$ , this gives us all the quantities we need to find  $\tilde{\mathbf{S}}$  at a reduced computational cost.

It has been shown by [9] that for a set of real or complex data  $\{\mathbf{v}\}_{k=1}^N$  there exists complex  $\{\lambda\}_{j=1}^N$  and  $\{\mathbf{y}\}_{j=1}^N$  such that (3.11) is satisfied, provided that  $\{\lambda\}_{j=1}^N$  are distinct. It has been shown in [23] that the decomposition itself is unique. The DMD method is very successful at finding dynamic modes of the system and requires a lesser amount of data than the General Laplace Analysis. The DMD also seems better at capturing characteristic modes over shorter sampling periods that have larger spatial extents [30].

## 4. Design of the Experimental Infrastructure

Although simulation results are quite nice it is much more informative to utilize data in the environment within which the strategy was designed, the ocean. Real oceanic flows which are both aperiodic and highly stochastic would allow us to gain insight on the performance of the tracking strategy. Unfortunately, from the stance of an investigator the ocean is a very difficult place to conduct such studies. Current data sets from the ocean are of both low spatial and temporal resolution. This low spatio-temporal resolution lends to sampling scales that are vastly different than the scales of the ASV's dynamics. This increases the difficulty of incorporating the interaction of ocean dynamics and vehicle dynamics into control models. Let us also note cost restrictions and a lack of repeatability from uncontrollable ocean flows.

Thus, an experimental testbed was built in house in order to validate and further improve the multi-robot manifold tracking strategy. A system which is capable of producing repeatable and controllable quasi-2D flows which exhibit LCS similarly to real ocean currents. The system consists of several tanks; the Low Reynolds (LoRe) tank, the High Reynolds (HiRe) tank and the Multi-Robot (MR) tank as well as two types of ASV's. We refer to this system as the Multi-Robot Coherent Structure (MCoSte) testbed [24]. A description of the different tanks is given in 4.1 followed by a description of the flow generation and control system in 4.2 and ending with the details of our PIV system.

### 4.1 Tanks

The objective in the construction of a laboratory testbed was to generate and measure repeatable and controllable complex flows which possess similar structures to that of ocean currents. To do this the Low Reynolds (LoRe) tank, the High Reynolds (HiRe) tank and the Multi-Robot (MR) tank were built. As mentioned previously the wind-driven double gyre model is the basis of much of our simulations as it has been shown to have very useful properties. Thus, the tanks are based on recreating the double gyre structure. A basic description of each tank can be found in Table 4.1 which gives a quick comparison of the set.

Table 4.1: Comparison of LoRe, HiRe, and MR flow tanks.

	LoRe Tank	HiRe Tank	MR Tank
Re regime	$10^1 \sim 10^3$	$> 10^4$	$> 10^4$
Size	$0.1 \times 0.1 \times 0.02 \text{ m}^3$	$0.6 \times 0.6 \times 0.3 \text{ m}^3$	$3.0 \times 3.0 \times 1.0 \text{ m}^3$
Medium	water/glycerol-water mix	water	water
Typical liquid depth	2 cm	14 cm	30 cm
Flow driving devices	Two banks of 8 cylinders	12+ indep. cylinders	12+ indep. cylinders
Flow measurement	Laser PIV	Visible PIV/PTV	none

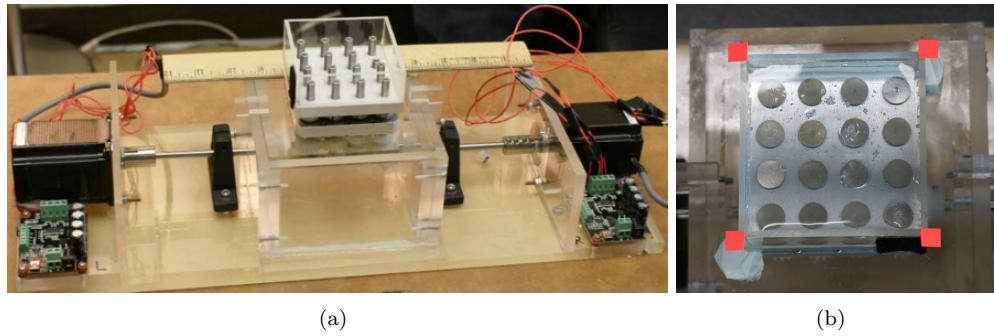


Figure 4.1: The LoRe tank, showing (a) independently controlled motors and (b) bank of spinning cylinders.

#### 4.1.1 Low Reynolds Tank

The smallest of the tanks is the LoRe Tank, pictured in Fig 4.1. The purpose of which is to produce controlled and realistic flows in the low Reynolds number regime onto which precise perturbation are made to the flow to study their effect on the coherent structures. The tank walls are transparent and make an 8 cm square which is typically filled to a depth of 2 cm with a mixture of glycol and water. Two independent stepper motors each drive a  $2 \times 4$  bank of counter-rotating disks which in their entirety make a  $4 \times 4$  array. The rotating disks reside at the bottom of the tank, submerged under the fluid. The design of the stepper motors and gearing to the disks allow for precise control of their rotational velocity. The motors are controlled through Matlab allowing the motors to drive at time-dependent velocities to create complex flows. A high quality laser PIV system is used to compute the flow field.

#### 4.1.2 High Reynolds Tank

The HiRe tank, a counterpart to the LoRe tank, operates in the Reynolds number regime  $Re > 10^4$  which is similar to that of which the ASV's will be working in and allows for more unsteady and realistic flows. Reproducing the double gyre structure in this higher Reynolds regime we expect the system to exhibit strongly nonlinear time-dependent behavior. Although we anticipate such behavior the general global structure of the flow should still remain quite similar. The tank must be able to produce these flows repeatedly and controllably at least in the global sense while also controlling perturbations and evaluating their effect on the presence of coherent structures in the flow.

To accomplish this, a  $60 \times 60 \times 30$  cm<sup>3</sup> high clarity glass flow tank, pictured in Fig. 4.2, was constructed. A smaller box which is called the control box rests inside the main tank and is used to ensure proper geometry of the system, this can be seen in Fig. 4.3. The tank is filled with water to a depth of 14 cm. Flows in the tank are generated via reconfigurable arrays of DC geared motors which drive cylindrical bodies that are submerged in the fluid, more on these in 4.2.1. These flow driving units are mounted within the tank such that their axis of rotation is perpendicular to the bottom surface of the tank in order to generate the quasi-2D flow fields. It would be possible for future work within this same tanks to rotate the axis of rotation such that it is no longer perpendicular with the bottom surface of the tank in order to create more complex 3D flow fields. Each motor is equipped with magnetic encoders enabling closed-loop speed control. The dynamics of the flow field are changed by setting the speed and direction of the various motorized cylinders. To create a nearly time-independent flow, such as that of the static double gyre model, one would place the motors in the desired pattern and run the motors at a constant speed with the proper directions of rotation. The motors are independently controlled as well which allows for the creation of more complex time-varying flows.

Flows generated in the HiRe tanks are measured with our PIV system from LaVision which is discussed in detail in section 4.3. The PIV results provide us with vectors fields at specified time intervals which allows us to compute the FTLE and also perform the DMD to create a reduced order model of the system.

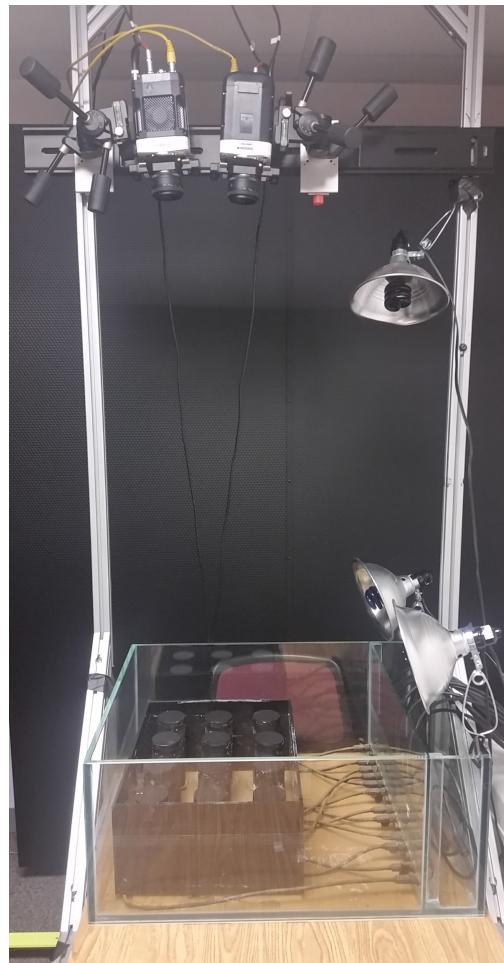


Figure 4.2: The HiRe tank setup. Pictured is the tank with a 12 motor array set-up inside the control box, the overhead PIV camera system and the black lights used to illuminate the tracer particles.



Figure 4.3: Experimental setup of flow tank with 12 driven cylinders in the control box.

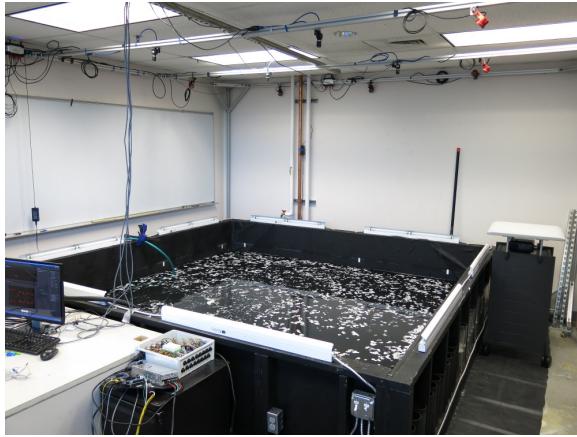


Figure 4.4: MR tank and overhead motion capture camera system.

#### 4.1.3 Multi-Robot Tank

The Multi-Robot (MR) tank, seen in Fig. 4.4, is the largest tank and is designed for the implementation and development of the multi-robot coherent structure tracking strategy. The tank needed to be large enough to accommodate a small networks of robots within the flow field while also producing LCS on a scale comparable to that of the robots physical tracking capabilities. Thus a tank that is  $3 \times 3 \text{ m}^2$  was built that is typically filled to a depth of 30 cm. The MR tank is very similar to the HiRe tanks as both operate if flow regimes of  $Re > O(10^4)$ . Like the HiRe tank the MR tank uses a set of reconfigurable DC gear motors which drive cylindrical bodies of varying size to generate flow, more on these in 4.2.1. The MR tank is also equipped with an overhead OptiTrack motion capture system which provides real-time rigid body tracking of the ASV's at 120 Hz. From this the position and orientation of each vehicle in the tank can be identified.

#### 4.2 Flow Generation

Flow generation in the HiRe and MR tanks are both performed using sets of geared DC motors which are capable of driving cylindrical bodies of several sizes. The work presented in this thesis however is limited to cylindrical bodies with radii  $\{6.4, 10.2\}$  cm for the HiRe and MR tanks respectively. Each motor is equipped with a magnetic encoder which enables closed-loop speed control allowing for accurate input control to the fluid system. Driving each motor is a RoboClaw motor controller which has built in PID control as well as many other functionalities. The motor controllers use serial communication to communicate with an Arduino which is coded with the desired motor

speeds and direction. This section discusses the construction of the flow driver assemblies as well as the system that controls all of the motors and the accuracy at which the motors operate for the experiments conducted in this thesis.

#### **4.2.1 Flow Generation System**

Both assemblies currently use the same DC motor and motor control system. The motors are basic DC gear motors purchased from Pololu, their properties can be seen in Table 4.2. The motors are intended to be submerged and thus a waterproof housing had to be built to protect the motors. Originally an assembly made of clear polycarbonate was used, making it easy to detect leak should they occur. Using rubber O-rings at joints and screw connections as well as a seal between the motors drive-shaft and where it protrudes through the top plate. However, after much use and testing, it became evident that in higher performance operating regimes the thermal energy created by the motor could not be dissipated fast enough through the polycarbonate. As a solution the end caps, specifically the top plate which the motor housing has direct contact with, were changed to aluminum and a thermal conductive paste was used where the motor housing and top plate contact one another to increase efficiency in transferring the thermal energy to the surrounding fluid. This solved the initial over-heating problem. An IP6 rated panel connector was used in the bottom plate to connect the motor to its power and signal wiring. The final motor housing assemblies can be seen in Fig. 4.5.

All metallic parts in the system are made with multipurpose 6061 aluminum while all screws are 304 stainless steel due to the corrosive environment of operation. It should be noted that further research found several companies who make water-proofed motors on the scale we required with better performance specs. The cost of these motors however was very high compared to our in house solution.

#### **HiRe tank Flow Driver**

In the HiRe tank flow driver assembly the motor casing is connected to a weighted base with the motor shaft pointing away from the base. The weighted base controls the cable feed to the motor and has a layer of neoprene rubber adhered to its bottom to prevent the motor from sliding across the glass tanks surface during operation. The weighted base is made with a bearing seat to accommodate the roller bearing which stabilizes the rotational axis of the driven cylindrical body. The driven cylindrical body, a section of pipe that has been cut and turned to uniformity, is attached

Table 4.2: DC Motor Specs

	Specs
Dimensions	(25 × 25 × 52) mm
Gear Ratio	34.014 : 1
Operating Voltage	6 ± 3 V
Free Run @ 6V	285 rpm
Free Run @ 6V	450 mA
Stall torque	4.3 kg-cm
Stall current	6 A
Encoder	48 CPR quadrature encoding
Encoder Resolution	1633 counts per rev



Figure 4.5: The HiRe tank Flow Driver. On the left is the encapsulated motor attached to its weighted base. On the right is driven cylinder that interacts with the fluid medium to generate flow.



Figure 4.6: The flow driver for the Multi-Robot tanks

to a hub that fixes it to the motor shaft. The motor shaft itself helps to stabilize the driven body as well as power its rotation. The assembly is 6.4 cm in diameter and 27 cm in height. The assembly can be seen in Fig. 4.5.

### **MR tank Flow Driver**

The MR tank flow driver is quite similar in overall design to that of the HiRe tank. The difference being that due to the larger diameter, extended length and weight of the driven cylindrical body it instead rides on two stainless steel bearings mounted on a central support shaft which extends from the weighted base. The motor in this system does not act as a method of stabilizing the rotational axis of the driven cylinder it only provides the necessary torque. The MR tank drivers also utilize 3D printed hubs to reduce machining time and costs. At the time of writing the driven cylinders are made of mild steel. This was due to the pricing and sourcing of the proper size tubing in either aluminum or stainless steel while validating their function. The mild steel parts will in the future be changed to either aluminum or stainless steel. The assembly is 10.2 cm in diameter and 68.5 cm in height. The assembly can be seen in Fig. 4.6.

#### 4.2.2 Motor Control

##### Devices and Configuration

Control of the system during the time of this writing is achieved through the use of an Arduino micro-controller. In development is interfacing the motor controllers with MATLAB so that processes may be graphically shown to the user on a PC while also allowing for recording and processing of the data. The Arduino is currently coded with the designated operation of each motor for the duration of a run. The Arduino communicates with a set of 8 RoboClaw  $2 \times 5\text{ A}$  motor controllers through packet serial data. The  $2 \times 5\text{ A}$  designation signifies the controller is capable of independently controlling two motors with a maximum current on each line of 5 A. The motor controllers are housed in a water proof container with a common external power supply, as seen in Fig. ???. The use of IP6 panel connectors allows for the power and signal cables to pass through the box without affecting its water tightness.

The motor controller is capable of reading quadrature encoding and comes with a built in PID routine which allows for closed-loop speed control for each motor. The encoder values and motor speeds can also be read from the RoboClaw allowing for the use of an external control system. We chose to use the internal PID routine for the experiments in this paper after its effectiveness was evaluated with our motors, further discussed below.

##### Consistency Evaluation

With the need to run many motors simultaneously and accurately over multiple motor controller devices it was important to test the consistency of the PID speed control routine across several devices. To perform this task 12 motors were used with 6 RoboClaw devices. Each device was set with the same PID parameters and speed setting. Rotational speeds of the drive cylinders were measured using a non-contact tachometer from ExTech, model 461920, which has an accuracy of  $\pm 0.05\%$  with a resolution of 0.1 rpm. The results of test can be seen in Table 4.3 where the test was performed at 5 different operating speeds.

As can be seen from the table the speed for different motors and motor controllers is quite consistent from 45 – 90 rpm. The consistency starts to deteriorate around 120 rpm, however, this is the highest operation speed we reach during the experiments in this paper. Thus, for our purposes we can say that under these operating parameters the speed control routine is consistent and accurate enough across all devices.

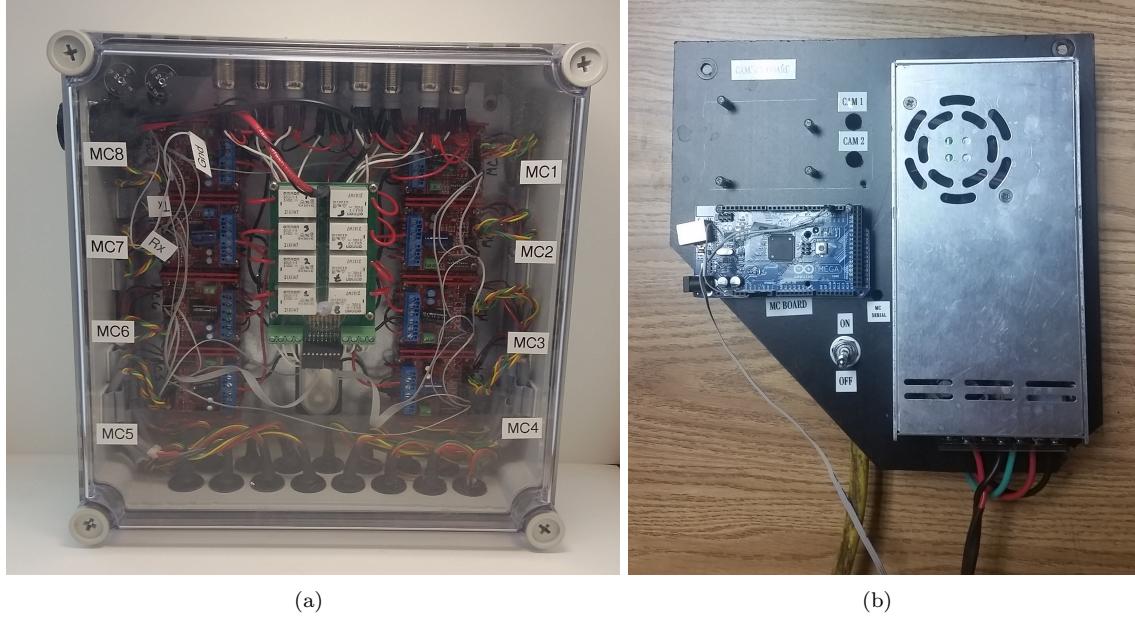


Figure 4.7: (a) The water tight unit housing the motor control boards. (b) The power supply unit and Arduino which provide power to the motor controller box and communication signals between the controllers and the PC.

Table 4.3: DC motor consistency evaluation results

	45 rpm	60 rpm	75 rpm	90 rpm	120 rpm
Motor 1	45.1	59.9	74.7	89.6	119.6
Motor 2	44.3	59.1	74.9	89.2	118.1
Motor 3	45.2	59.8	75.1	90.1	119.5
Motor 4	45.4	59.7	75.1	90.1	118.8
Motor 5	45.0	59.6	74.9	89.8	119.4
Motor 6	45.0	59.2	75.2	89.5	118.8
Motor 7	45.1	59.5	74.8	89.4	119.5
Motor 8	44.9	59.5	74.8	89.7	119.1
Motor 9	45.2	59.7	74.8	89.9	120.3
Motor 10	45.1	59.4	75.0	89.6	119.4
Motor 11	45.0	59.9	74.7	89.9	119.7
Motor 12	44.9	59.8	74.9	89.6	119.7
Standard Deviation	0.26	0.26	0.16	0.27	0.56

### 4.3 Data Collection

The method of extracting velocity measurements from the flow field is performed through the use of Particle Image Velocimetry (PIV), previously mentioned in 2.3. The system operated for the experiments performed in this thesis was produced by LaVision and operates on their DaVis 8 software package. To perform the high-speed imaging two Phantom Miro M110 cameras are used in stereo and utilize Scheimpflug adapters with Nikon AF Nikkor 50 mm lenses. Due to geometric constraints imposed by our experimental set-up the use of a laser sheet was made very difficult, instead particle illumination came from an overhead arrangement of black lights which caused the tracer particles to fluoresce. The particles used were purchased from Cospheric, product i.d. UVYGPMS-0.97 300–355  $\mu\text{m}$ , and fluoresce a yellow-green color when excited by black light. The particles were pretreated in a solution of Tween 80, another product from Cospheric, a surfactant that minimizes particle clumping.

## 5. Experimental Method

### 5.1 Experimental Procedure

The goal of this experiment is to show that the HiRe tank can consistently reproduce LCS in the same locations and to see how motor speed effects the LCS location. Originally the experiment was designed to operate with 16 flow drivers within the whole domain of the HiRe tank. Unfortunately, in the first attempts to run the experiment there were issues with motor positioning and movement. Most of these issues were caused by the stiffness and memory effect of the cabling used to control the Flow Drivers. This was even more evident when the cabling was submerged in the cold water the tank was filled with to help cool the motors. Flow drivers that required shorter cable lengths for positioning were lifting and vibrating out of place as the cable wasn't flexible enough to stay in place under the small weight of the motor. Some longer cables as well would begin to produce a force as the cable stiffened in the lower temperatures and wanted to return to it's original shape causing Flow Drivers to slowly vibrate out of position. Even with heating the cables and trying to get the material to relax in the desired position issues persisted. Attempts to let the water rise back to room temperature still did not completely resolve the issue and also began to lead to motors burning out after long operation periods.

The solution was to use a sub-domain of the tank and create a set of walls with a bottom plate that had holes equal in diameter to that of the motor base which were spaced appropriately in order to hold the flow drivers in place. This smaller tank within the HiRe Tank will be called the control tank. Unfortunately this control tank solution required the experiment to go from 16 to 12 flow drivers.

The experiment was performed in the HiRe tank and used 12 Flow-Drivers arranged in a  $3 \times 4$  grid. Data was collected at 6 different motor speeds, found in Table 5.1, in order to see the effect of motor speed on LCS location. Then each motor speed was sampled 3 different times to show repeatability of the system. Each Flow Driver was positioned 11.3 cm apart, measuring from the centers of rotation from it's neighbors in the same column or same row. Each Flow Driver at a boundary was positioned 5.65 cm away from the center of rotation to the nearest point on the wall. This was done to make the boundary conditions of each gyre the same geometrically. Spacing was kept constant by the use of the control tank. The use of 12 flow drivers ensures a system of two

Table 5.1: Motor Speeds

	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5	Speed 6
Motor Speeds (rpm)	45.0	60.0	75.0	90.0	105.0	120.0

gyres that do not see improper boundary conditions due to the presence of a wall. The flow drivers are spun such that the direction of the rotation is opposite that of any adjacent flow driver along the same row or column, the same as in the wind-driven double gyre model.

In order to truly show re-producibility the experiment is performed in a methodical way which is outlined here.

1. Start the flow drivers at the slowest speed and allow them to run for 1 minute before data acquisition.
2. Perform data acquisition then advance flow driver speed to the next greater value and again wait one minute before data acquisition.
3. Repeating steps 1 & 2 until the data has been collected for the highest speed.
4. At this point shut down the flow driving system and allow the fluid to come to rest.
5. Once the fluid has come to rest repeat steps 1 through 4 twice more.

This process should result in 3 different runs at each speed for a total of 18 runs.

### 5.1.1 PIV Data Collection Parameters

Each image set was collected under the same recording parameters. An image set is the series of images of the flow that will be fed into the PIV processing routine to determine the flow velocities. Each image set consists of a 30 s recording at 30 Hz giving a total of 900 images per camera per test. Imaging at a rate above 30 Hz showed no significant change in the resultant dynamics post PIV processing but a large increase in processing time. Sampling at a slower frequency than 30 Hz began to show poor resolution and aliasing effects post PIV processing, thus the experiments were performed at 30 Hz. The images taken from the two cameras at a single time point have approximately a 50% overlap. A common coordinate system is given to the images during a camera calibration phase internal to the DaVis 8 software before running a new experiment. The cameras utilize a single-frame single-exposure method in capturing each image.

### 5.1.2 PIV Data Processing Parameters

The image sets are fed to the PIV routine for processing. There are many processing methods available within the DaVis 8 software, here the settings used for the experiments in this thesis are described. Within DaVis 8 the processing operation 'PIV time-series' was chosen to process the image sets. A stereo cross-correlation method was performed with three passes. The first pass was performed with  $64 \times 64$  pixel windows that have a 50 % overlap. The second and third passes were both performed with  $32 \times 32$  pixel windows that have a 50 % overlap. A median filter was used such that any data that was not within 2 standard deviations of its neighboring particles was replaced with the median value.

## 6. Experimental Results

### 6.1 Overview

In this section the results of the experiments are presented first looking at the resolved velocity vector fields of the HiRe tank from the PIV system. Following will be an evaluation of choosing a suitable integration time for the Finite Time Lyapunov Exponent (FTLE) analysis on each of the data sets. This will transition into the resultant FTLE fields from the experimental data sets and a look at their similarity to the double-gyre model the experimental set-up was designed to reproduce. From there an analysis of the time-invariance of the ridge location will be evaluated to show that the flow is indeed producing Lagrangian Coherent Structures (LCS) at a constant position as we saw with the time-invariant double-gyre model. This precludes the evaluation of ridge location invariance between runs with the same parameters performed at different times after letting the fluid come to a rest and restarting the system, thus showing repeatability. A look at using Dynamic Mode Decomposition (DMD) to create a reduced order model and smooth out higher order dynamics that are not of interest to our investigation is then presented with similar FTLE evaluation as described in the initial portion of this section.

### 6.2 PIV Resultant Vector Fields

Utilizing the collection and processing parameters discussed earlier for the PIV system it was found that the flow produced in the experiments suitably matches that of the double-gyre model. Graphs of the two vector fields can be found in Fig. 6.1 where the experimental data is shown in (a) and the double-gyre model is shown in (b). The data has been cropped to remove the wall boundary conditions present in the experimental flow which do not exist in the analytical double-gyre model. The cropping focuses on the inner two flow drivers of the system which sit on  $x = 0$  at approximately  $y = \pm 55$ . The vector fields qualitatively look very similar. Notice the 6 saddle points that can be found presenting themselves in a  $2 \times 3$  array that sits over  $x = 0$  and the straights between the two adjacent gyres within the same column or row which strongly matches that of the model.

It should be noted that During the PIV processing masking of the flow drivers before processing created large holes in the data around the motors. This came from a lack of particles being present in the fluid directly surrounding the flow driver. The appearance of a turbulent region around the

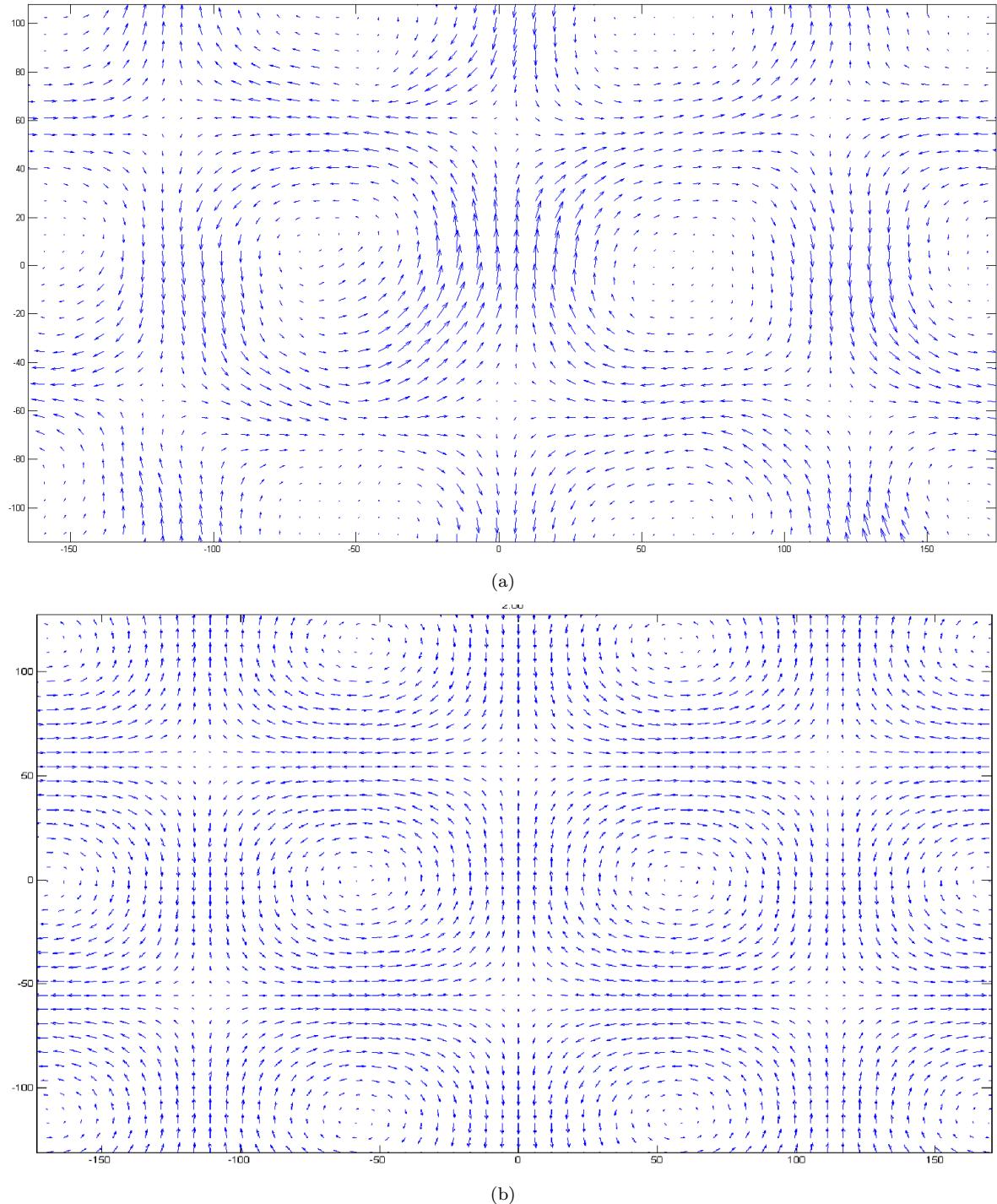


Figure 6.1: A comparison of (a) the PIV resolved velocity field from HiRe tank data and (b) the velocity field from the Double Gyre model.

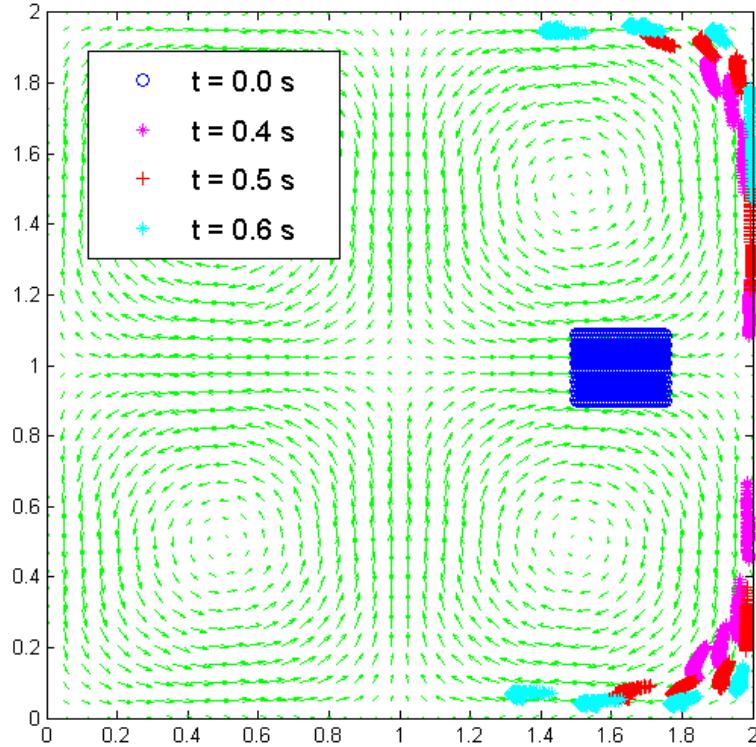


Figure 6.2: The vector field of the double-gyre model with particles initially placed straddling a known LCS boundary and advected forward in time by the flow.

flow driver that repels particles from its surface seems to be the culprit. The solution was to leave the flow drivers unmasked as a sufficient number of particles cling to the wall of the flow driver and allow an accurate measurement of the local flow. The area which the flow drivers occupied was then interpolated over to create the continuous looking vector field in Fig. 6.1. This interpolation should not affect the FTLE calculation at all as it provides the system there with a continuously monotonic dynamic change presenting no 'FTLE ridge'. Graphs of the velocity vector fields of each run speed can be found in Appendix A.1.

### 6.3 FTLE Analysis

#### 6.3.1 Choice of Integration Time

Choosing a proper integration time is necessary to providing meaningful results. Longer integration times will typically reveal more prominent structures, however, too long of a time and the particles may return to their initial separation and yield a low FTLE value where a LCS truly does

exist. Thus a sufficiently long time must be sought after that maximizes the definitiveness of the LCS. At the time there is no concrete method of choosing a proper integration time. To resolve an integration time that would yield useful results the double-gyre model was first evaluated to see the relation between FTLE values and particle separation. This was done in order to try and determine the range of integration times which best capture the sought after LCS while minimizing the chance of particles returning to their initial separation. This entailed evaluating the static double-gyre model over the range  $[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]$  s of integration periods. Following that a set of particles was placed straddling an LCS ridge and propagated forward in time by the flow field for the same time span as the integration period. The FTLE values and the distances separated were then compared to look for changes in behavior that would show better or lower quality integration periods. The graphs are presented in Fig. 6.3. Integration times between 0.1 s and 0.5 s display high FTLE values and the data looks similar to that of a exponential growth curve which is to be expected due to the calculation of the FTLE. Above an integration period of 0.5 s the trend begins to disappear as FTLE values become smaller for equivalent distance separations that were seen in the shorter integration period times. Notice also that the density of particles within a small range of FTLE values that occurs for the integration periods of 0.5 s to 0.7 s. This compacting of many particles to a small FTLE range is caused by them no longer drifting farther apart as time goes on and since the FTLE value is inversely proportionally to the integration period the FTLE value of these particles drops. Thus it should be said that for this model 0.4 s is a sufficiently long integration time as the relevant structures have been captured and to integrate longer is a waste of time and processing power and provides reduced insight on the dynamics of the system.

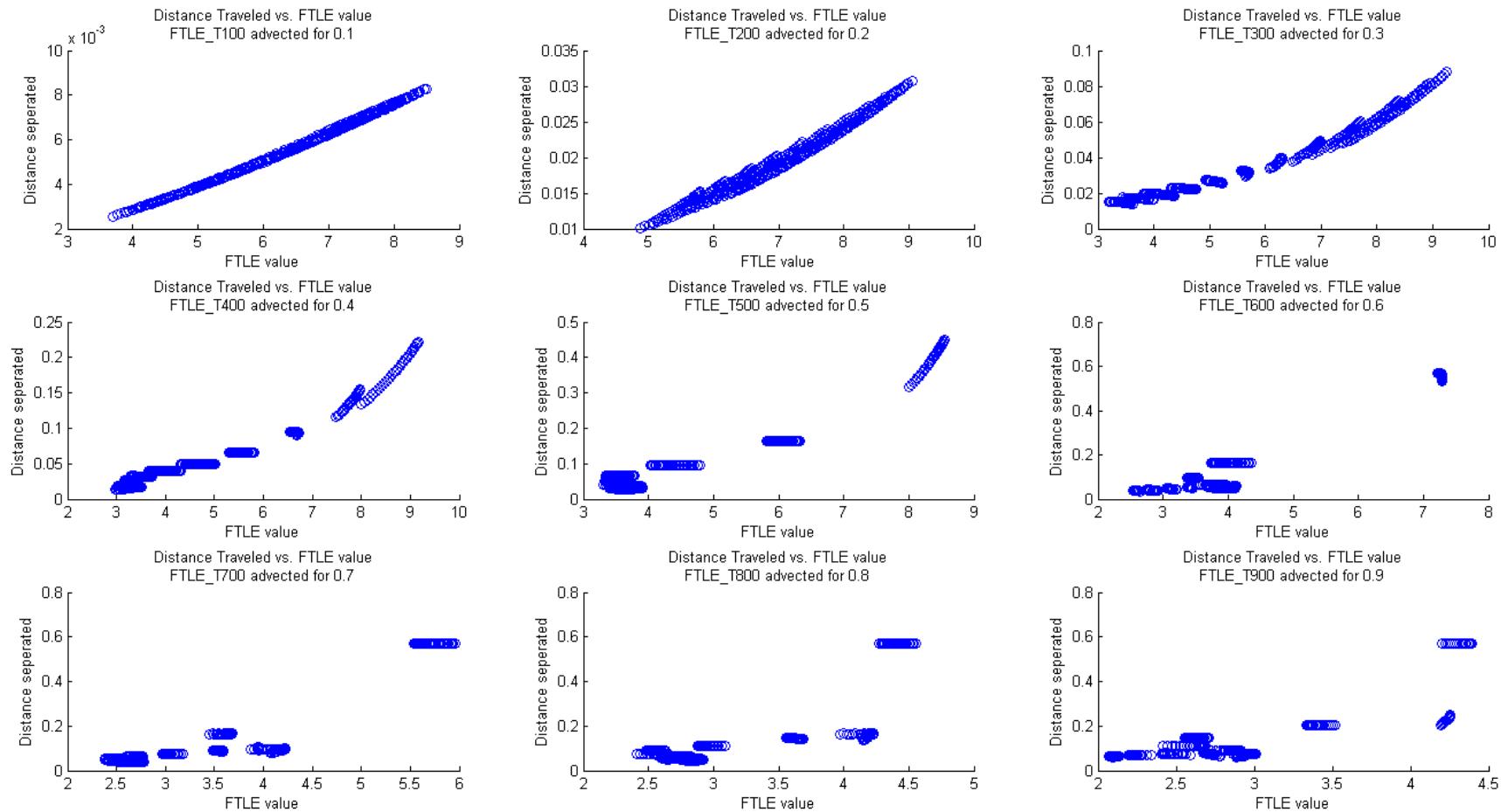


Figure 6.3: Distance separated vs. FTLE values for the double-gyre model with different integration times.

A suitable relation between the given model and the integration time must now be found that is scalable to the experimental system. Taking a look at the model flow field again and the final location of the propagated particles around the designated  $0.4\text{ s}$  integration period, which can be seen in Fig. 6.2, there are visual clues to answer the question of how does this knowledge scale. Looking at the distance the particles travel around a gyre for the given integration periods shows that for chosen period of  $0.4\text{ s}$  the particles make it approximately one-third of the way around the gyre. Therefore, a suitable estimate at a proper integration time may be based on the time it takes a particle, initialized proximal or preferably straddling a LCS, to traverse one-third of the gyre adjacent to the LCS. The experimental system is without a doubt much more complex than the simple analytical model that this principle is based on but testing of the experimental model showed this method to work quite well at determining a sufficiently long integration period. Each experiment used this method for finding a suitable integration period and with minimal change provided clean results. A view of the distance separated versus FTLE values can be seen for several integration periods of experimental data can be seen in Fig. 6.4. For this particular run an integration period of  $T = 4.0\text{ s}$  was used whose results can be seen in Fig. 6.5. Although a time of  $5.0\text{ s}$  appears to work as well from the graphs in Fig. 6.4 it produced very similar results and took a significantly greater amount of time to compute as that of the  $4.0\text{ s}$  integration time that was chosen. Evaluating the time it took for a particle in the experimental flow to travel one-third of the way around a gyre was done with two methods. The first was by visual inspection of the raw camera footage fed into the PIV processor. The second was to take the resultant vector fields from the PIV processing and initiate a set of particles between two gyres and propagate them forward in time until they reached the one-third revolution mark. The two methods showed good correspondence so visually tracking a particle, which could be quite challenging, should be used in future evaluation just for confirmation when the propagation of particles is not producing a good initial guess at the integration time.

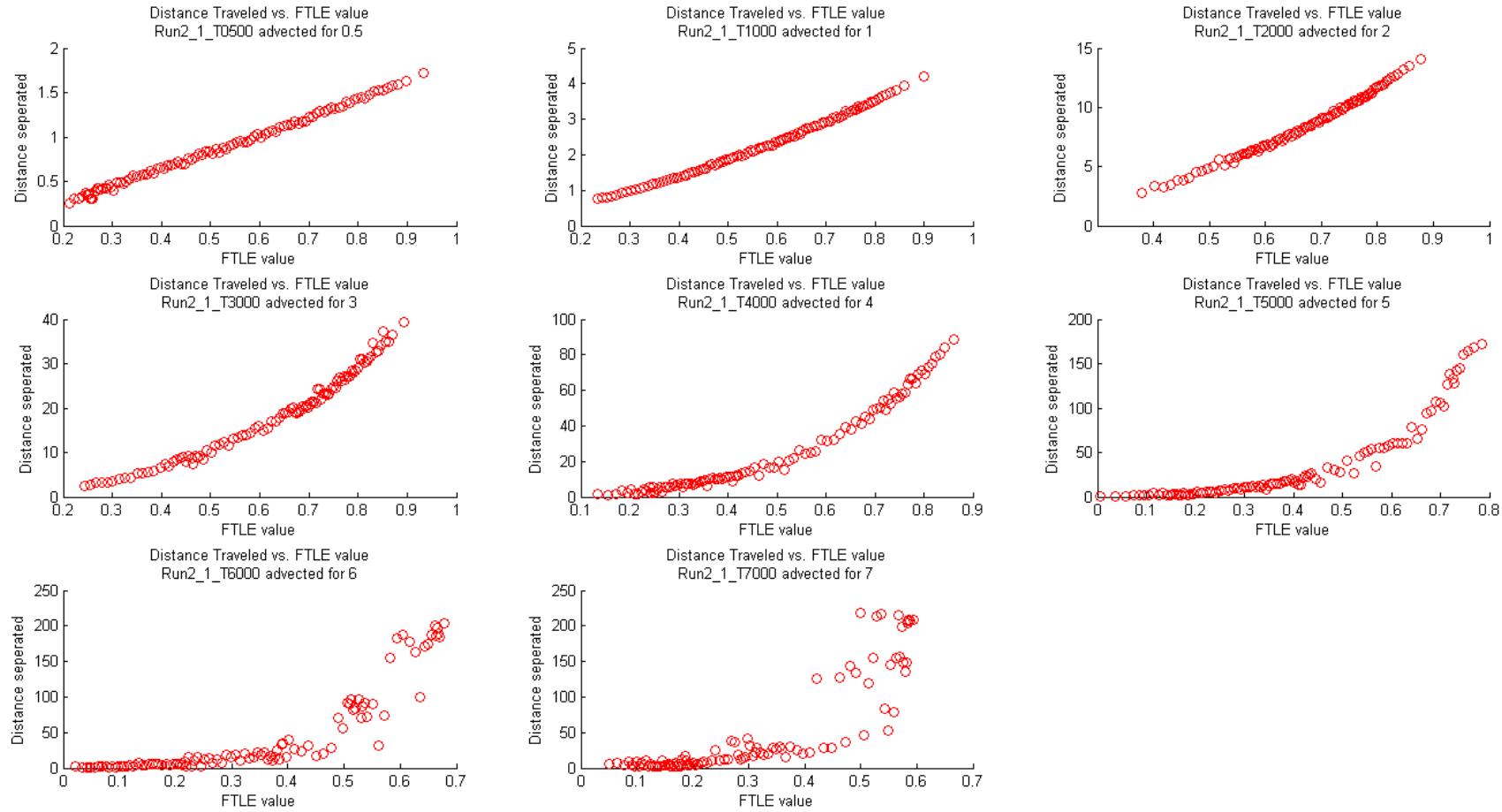


Figure 6.4: Distance separated vs. FTLE values for experimental data with different integration times.

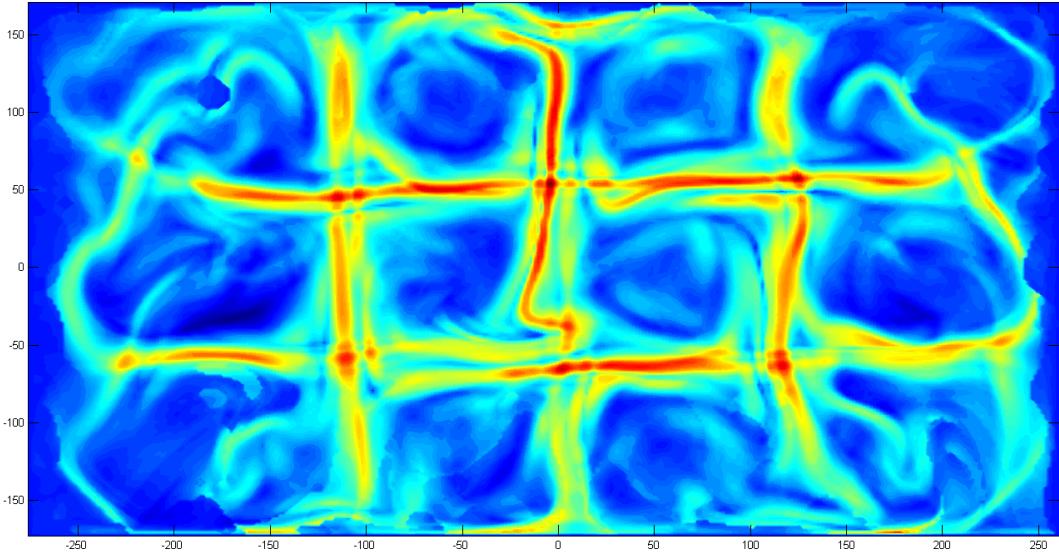


Figure 6.5: Snapshot of the first run at 45 rpm with an integration time of  $T = 4.0$  s.

### 6.3.2 Experimental FTLE vs Double Gyre Model

The time-invariant double-gyre model shows very clean cut FTLE ridges as can be seen in Fig. 6.6 (a) which was the basis that was trying to be reproduced in a real physical system. Such clearly defined and well organized ridges are unlikely to present themselves in real data, however, it is only desired to consistently reproduce a structure very similar to that of the double-gyre model that allows for testing of the multi-robot coherent structure tracking strategy discussed more in 7.1. In Fig 6.6 (b) is the time series average of an FTLE field produced by the experimental data. The time series average is calculated as,

$$\text{TimeSeriesAverage} = \frac{1}{N} \sum_1^N \sigma(x, y) \quad (6.1)$$

where  $\sigma(x, y)$  is the FTLE value at a given position and  $N$  is the number of sample points through time. This gives an understanding of where the LCS ridge typically lies within our noisy experimental data. This concept will be used more in the next section in regards to evaluating how 'steady' the location of a LCS ridge is with respect to the mean location. Comparing 6.6 (a) and (b) it is shown that the experiment was successful at recreating the same LCS pattern as the model. The FTLE instantaneous and time series average images for the other run speeds can be found in Appendix A.2. Looking at 6.6 (c), which represents an instantaneous view in time from the time series, it

should be noticed that occasionally several parallel LCS ridges occur in the same location. This is one reason for using the time series average to see the general pattern of where the LCS exist. The cause for the existence of these multiple ridges is unknown they may be a true part of the dynamics of the system or possibly created due to noisy measurements within the experimental procedure and data processing. More work must be done to determine the causality of their existence. Although here only seen at an instance, if one were to watch these FTLE fields through time one would find them not to be static but to continuously vary while maintaining the same general pattern to an overall degree. Which leads one to ask the question, 'to what spatial extent do these ridges shift?'

#### 6.4 Invariance of Ridge Location

To determine the invariance of a ridges location becomes a bit challenging when dealing with data where several parallel LCS ridges occur within a particularly close region. To evaluate this property and overcome the issue of several very close LCS ridges the task was broken into several parts. First was to begin by looking at the time series average and define a mean LCS ridge location. This action was performed because the averaging in effect smooths out multiple nearby parallel ridges into a single more well-defined ridge that closely represents the basis LCS pattern of the double-gyre model. In order to evaluate to what spatial extent the LCS ridge moves a method of defining a single ridge in each time instance of the FTLE field was needed in order to compare it to the mean location.

Defining a single ridge location in the presence of several locally relevant ridges was done by finding all peaks that satisfy  $|\vec{\mu} - \vec{p}| < \delta$ . Where  $\vec{\mu}$  is the time series average peak location,  $\vec{p}$  is any other peak in the same raster. An area weighting method was utilized in order to define a single ridge location from the set of points. Finding the ridges within the field was done by using MATLAB's *findpeaks()* function with a defined threshold such that no peak below the threshold value is returned. The function returns the peak values, locations, and half-height width. The only drawback to this function is that it only works on a function of one dependent variable. Thus a raster scanning processed was used to perform this function across  $y$  for all  $x$  and separately across  $x$  for all  $y$ . To carry the process out successfully regions where the scan ran along a ridge were excluded so that each scan would only cross LCS ridges and not travel along them. Doing this along both the  $x$ -axis and  $y$ -axis and looking at the union of the two sets would find LCS rides that ran

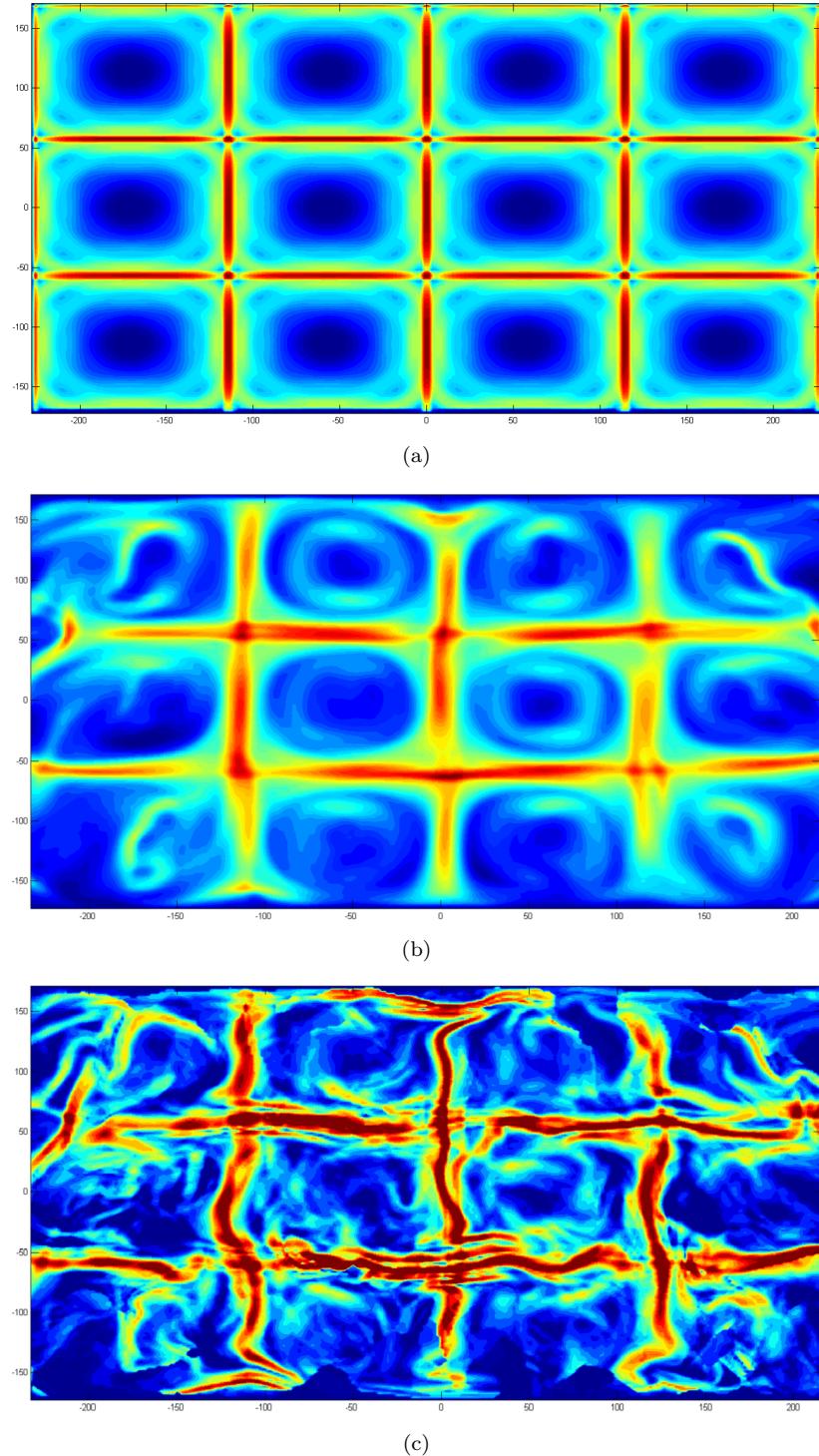


Figure 6.6: A comparison of (a) the FTLE for a time invariant double-gyre (b) against the time series average FTLE value of experimental data and (c) an instantaneous snapshot of the FTLE field.

both along the  $x$ -axis and  $y$ -axis. The area weighting method utilized was,

$$W_i = \frac{h_i w_i}{\sum_{m=1}^N h_m w_m} \quad (6.2)$$

where  $N$  is the number of points in the set and  $i$  enumerates the points in the set. The variable  $W_i$  is the weight,  $h_i$  is the height of the peak or FTLE value and  $w_i$  is the half-height width. The calculation of the new mean peak is straight forward from this point and is performed as

$$p_j = \sum_1^N p_i W_i \quad \forall |p_i - \mu_j| < \delta \quad (6.3)$$

where  $j$  enumerates the number of LCS ridges in a raster of the time series average,  $p_j$  denotes the weighted average point that will be defined as the ridge in further analysis,  $p_i$  is a LCS ridge point in the point set of a common ridge and  $\delta$  defines some bound on points that are considered to be sharing a common LCS ridge from the time series average.

The results of this algorithm can be viewed for the time series average FTLE in Fig. 6.7. The focus of the analysis is concentrated on the two interior gyres which are displayed in the figure. This was done because LCS ridges that approach the boundary wall of the experimental tank are not consistent enough in their location to make the evaluation of them worth while. The saddle points are not evaluated either. This comes as a by-product of only raster scanning where the scan traverses the ridge. However, the argument is made that if the LCS ridges meeting at the saddle point are constant then so should be the saddle point.

The results of the LCS ridge finding algorithm can be seen in Fig. 6.8. The weighted averaging can be seen to shift the ridge points off of what appears to be a dominant ridge occasionally such as the horizontal strip of points in the lower left of Fig. 6.8. This occurs due to the presence of another LCS ridge which runs parallel to the ridge that appears to be more dominant. It would be incorrect to exclude this secondary ridge which has FTLE values quite close to that of the ridge that appears to dominate. As the system evolves the weaker looking ridge occasionally grows stronger and the two merge together to become a single ridge again as well giving reason to take it into account. It is also unknown at the time of writing whether or not the presence of these parallel ridges in such close proximity are actually sets of true LCS ridges or by-products of noise or spatial sampling rate. The complex topography of the FTLE field can be seen in Fig. 6.9, where the left corner of the surface is the lower left corner as seen in Fig. 6.8. In the absence of many ridges in close proximity

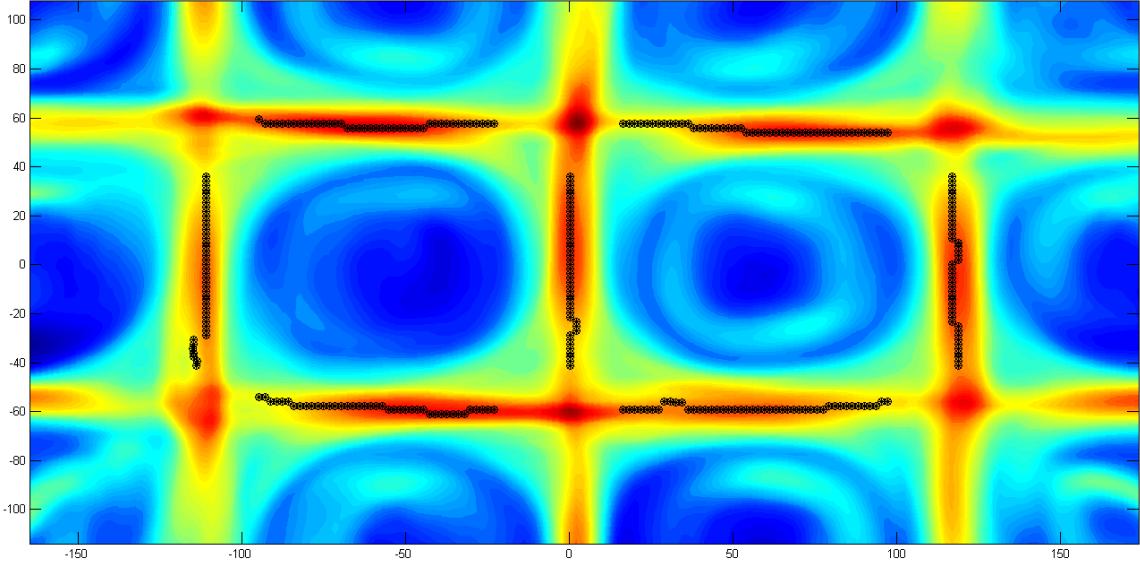


Figure 6.7: Time series average FTLE field with ridge locations denoted by black marks found using the weighted average method discussed in 6.4.

the ridge finding algorithm does very well at locating the center of a ridge and in their presence still performs quite well.

Applying the ridge finding algorithm to each snapshot of the FTLE field and comparing all of the points in the same raster and associated with the same LCS ridge a standard deviation can be calculated. The standard deviation is found using the time series average ridge locations as the mean of the set. To compute the standard deviation for raster scanning along the x-direction let  $X_k^n(i, j)$  be the sets of points found using the ridge finding algorithm where  $n = \{1, 2, 3\}$  denotes the run number,  $k = \{1, 2, 3\}$  denotes the LCS ridge number,  $i$  denotes the raster coordinate and  $j = \{1, 2, \dots, m\}$  enumerates the data points found at any triplet of  $\{n, k, i\}$  through time. Then the standard deviation for a particular raster and ridge location can be calculated as

$$\sigma_k(i) = \sqrt{\frac{1}{m} \sum_{j=1}^m (X_k^n(i, j) - \mu_k(i))^2} \quad (6.4)$$

where  $\sigma$  represents 1-standard deviation and  $\mu$  is the mean. The same technique can be applied to raster scanning across the y-direction.

The standard deviation of the ridge location using this method turns out to be quite small for all of the runs performed in the experiment and can be seen in Table 6.1. Figure 6.10 is the same view as Fig. 6.7 with the mean location marked in a black asterisk with a  $1-\sigma$  deviation marked

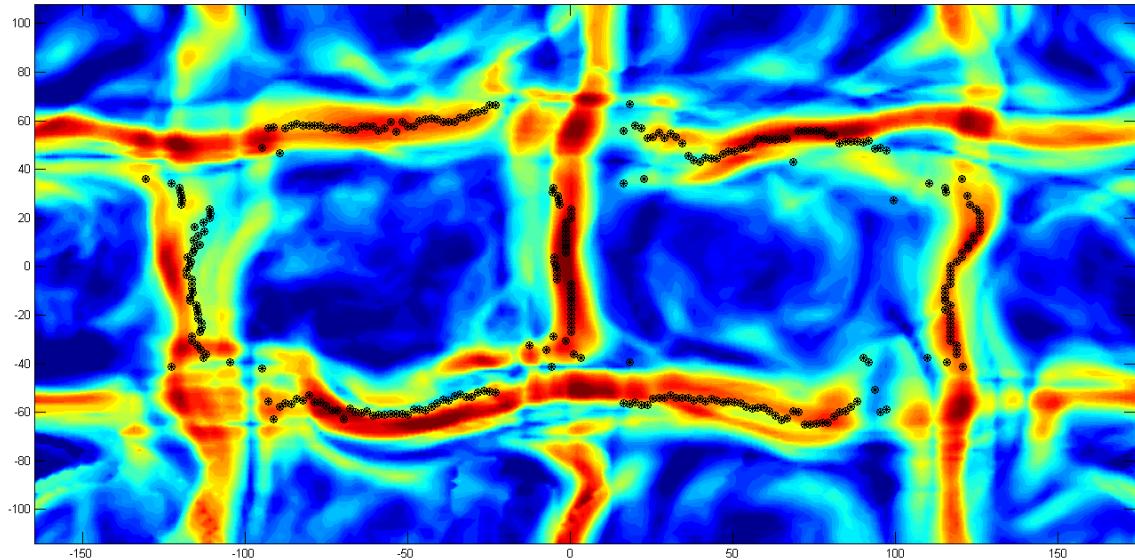


Figure 6.8: Snapshot FTLE field with ridge locations denoted by black marks found using the weighted average method discussed in 6.4.

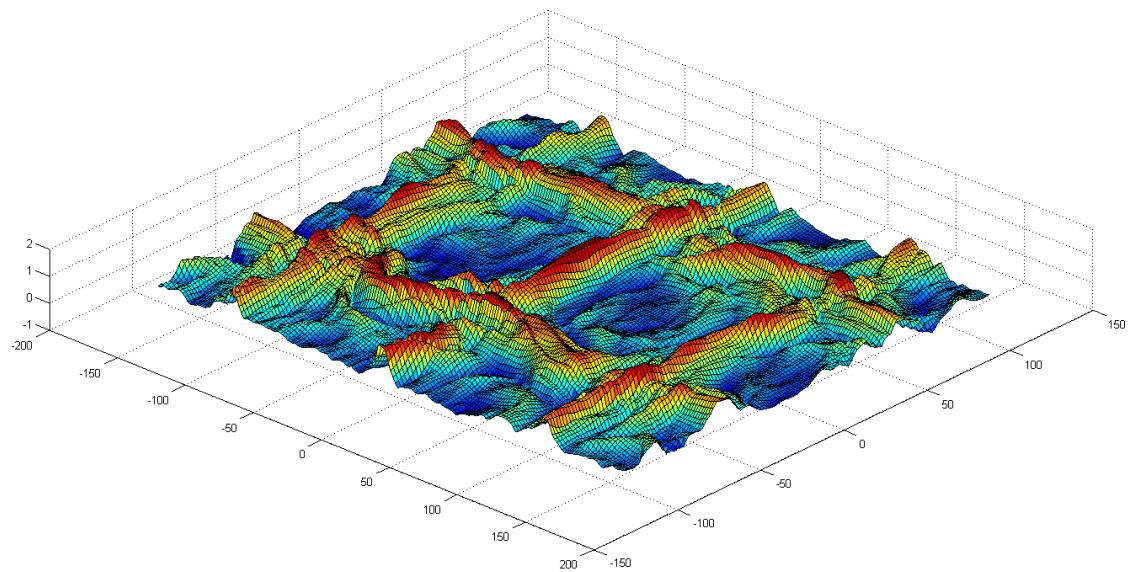


Figure 6.9: A surface plot showing the complex topology of the snapshot FTLE field

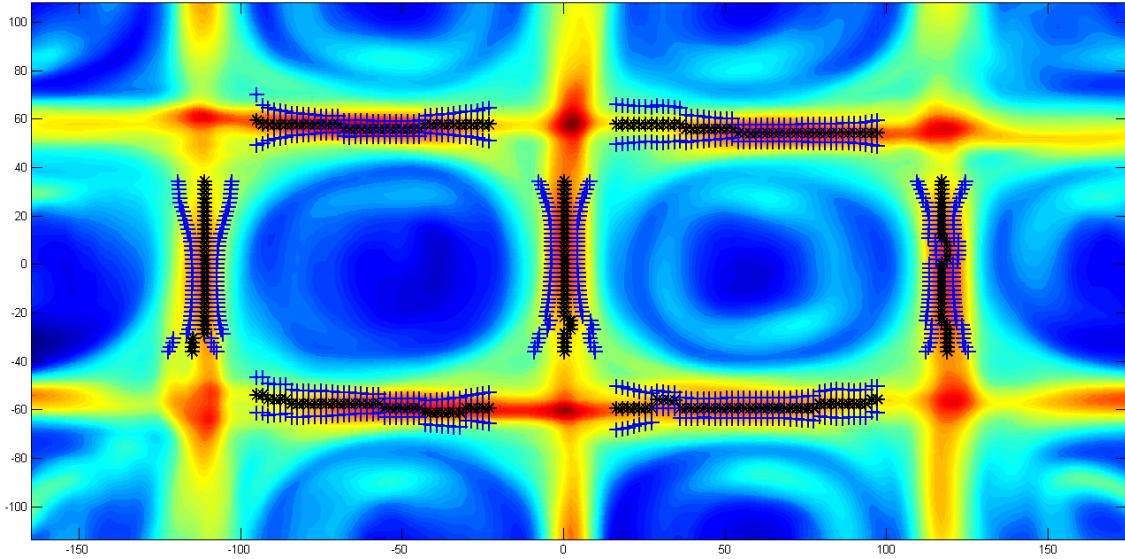


Figure 6.10: A view of the FTLE ridge in Fig. 6.8 with the  $1-\sigma$  deviation plotted in blue '+' signs and the time series average value plotted in black '\*' symbols.

Table 6.1: Mean and maximum standard deviations of LCS ridge location.

Flow Driver rpm	Run 1		Run 2		Run 3	
	mean- $\sigma$	max- $\sigma$	mean- $\sigma$	max- $\sigma$	mean- $\sigma$	max- $\sigma$
45	6.558	13.347	6.554	17.817	7.355	18.565
60	5.045	15.178	5.841	14.393	6.585	14.946
75	5.111	10.564	5.513	12.023	6.048	10.127
90	5.503	12.060	6.628	24.229	6.307	19.970
105	5.315	14.250	5.566	16.605	5.816	11.446
120	5.672	16.960	5.971	10.175	5.774	10.0195

by blue plus signs for each ridge point. Notice in the figure the deviation decreases to its smallest magnitude near  $y = 0$  which is the region farthest away from the effects of the saddle points and increases as a saddle point comes closer. When playing the instantaneous FTLE fields through time as a video it is observed that there tends to be many occurrences where multiple strong LCS ridges appear close together near saddle points and are biased to one side or the other of the time series average LCS ridge location. These events along with the weighted averaging method of determining LCS ridge location is what increases the magnitude of the standard deviation as a saddle point is approached.

To show that the system can consistently reproduce LCS in the same location three runs at each speed were performed as outlined in 5.1. To determine the consistency of the LCS location between

Table 6.2: Mean and maximum standard deviations of LCS ridge location over multiple runs.

Flow Driver rpm	All Runs	
	mean- $\sigma$	max- $\sigma$
45	7.605	18.957
60	6.707	17.470
75	5.935	13.438
90	6.645	12.625
105	6.006	9.757
120	6.674	12.407

different runs the ridge data computed for each run as discussed previously was used to determine the standard deviation for all of the runs performed at the same speed. This was done by taking the union of the sets and defining

$$\bar{X}_k = \bigcup_{n=1}^3 X_k^n(i, j) \quad (6.5)$$

where  $n$ ,  $k$ ,  $i$  and  $j$  are the same as Eq 6.4. Then using the time series average mean locations from run 1 as the reference mean for the computation. The standard deviation is given by

$$\sigma_k(i) = \sqrt{\frac{1}{m} \sum_{j=1}^m (\bar{X}_k^n(i, j) - \mu_k(i))^2} \quad (6.6)$$

which is the same as Eq 6.4 but with  $\bar{X}$ . The results from this are presented in Tab 6.2. The standard deviations increases by approx 15% which is about a 1mm physical shift. Thus with stopping and restarting the system the LCS present themselves consistently in the same location allowing for repeatable experimentation.

## 6.5 Dynamic Mode Decomposition

The dynamic mode decomposition is not heavily used in the work presented in this thesis but is needed to accomplish the goals of future work on this project. As such, only a brief discussion and demonstration of its capability to evaluate this system are given. It was previously mentioned that the DMD resolves spatial and temporal modes of the system which can be used to reconstruct the system with a reduced order model. Presented in Fig 6.11 is the FTLE analysis results of the DMD reconstructed flow field from a 14 mode decomposition of the experimental data and in Fig 6.13 is the same graph except it comes from a 59 mode decomposition. It is easily seen that with an increased number of modes used in the DMD the cleaner the results appear. The number of modes resolved using DMD is one less than that of the sample set. Therefore, the number of modes is limited by the number of samples and even further by the need for the samples to create a linearly independent set.

Modal dominance is calculated by the magnitude of the norm of each mode after scaling them to satisfy Eq 3.11. The larger the magnitude the more dominant the mode. The plot of the resultant modal Ritz values can be seen in Fig 6.15 with their corresponding magnitude displayed by the color of the point on the scatter plot. The most dominant mode in the decomposition tends to have a real value very close to 1 and an imaginary value of 0 which is seen in the plot.

In the experiments presented for this thesis the analysis was to show that a flow which produced LCS similar to that of the double gyre model could be consistently achieved. In the last section it was demonstrated that the experimental system does just and that the flow field is quasi-static. Thus, a single mode reconstruction of the flow from the DMD results works quite well for these experiments as can be seen in Figs 6.12 & 6.14 which are for the 14 mode and 59 mode decompositions respectively. Notice that with the reduced order model much of the higher frequency dynamics are gone removing much of the minor FTLE ridges that are present in the full reconstruction which give poorer resolution as to the true location of the major LCS ridges. The DMD technique produces a good reduced order model of the system from experimental data. The DMD technique will be instrumental in future work for constructing a control strategy that controls the locations of LCS in the flow.

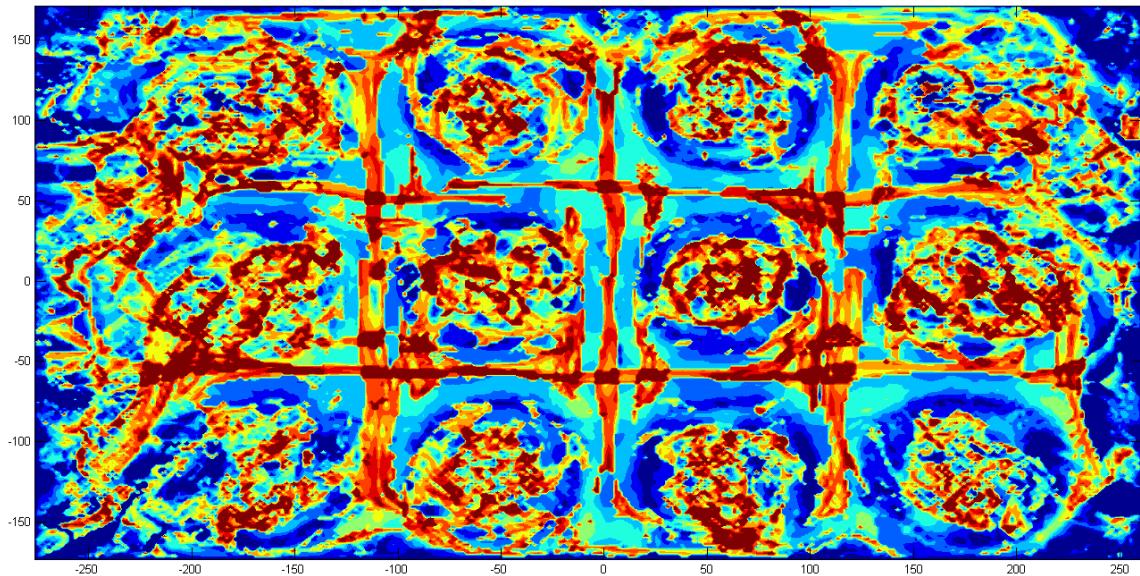


Figure 6.11: The combined, forward and backward, FTLE for the full reconstruction of the flow field from a 14 mode DMD computation

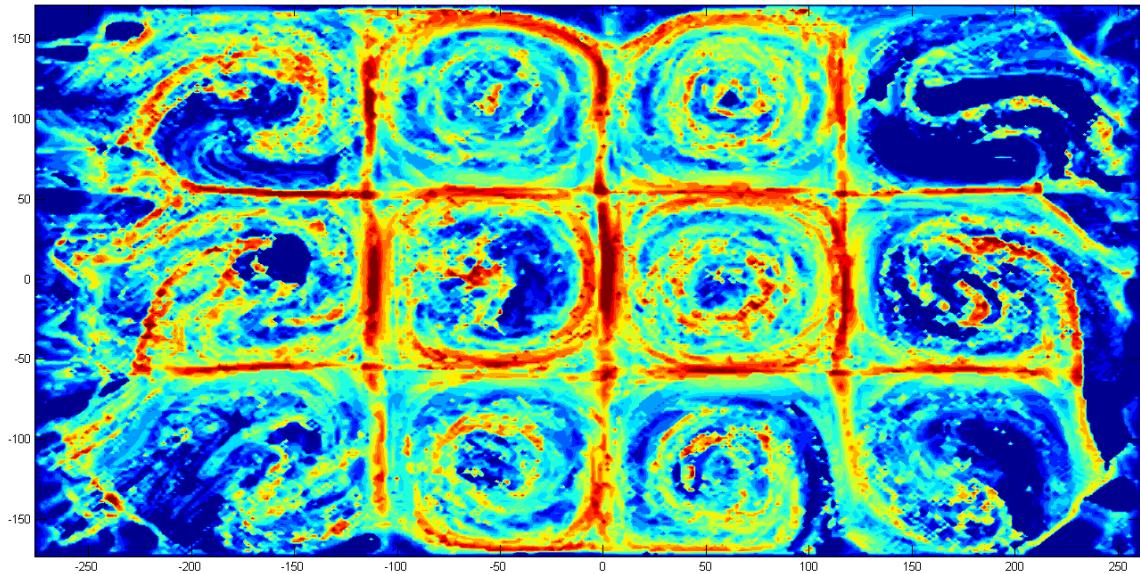


Figure 6.12: The combined, forward and backward, FTLE for the primary mode reconstruction of the flow field from a 14 mode DMD computation

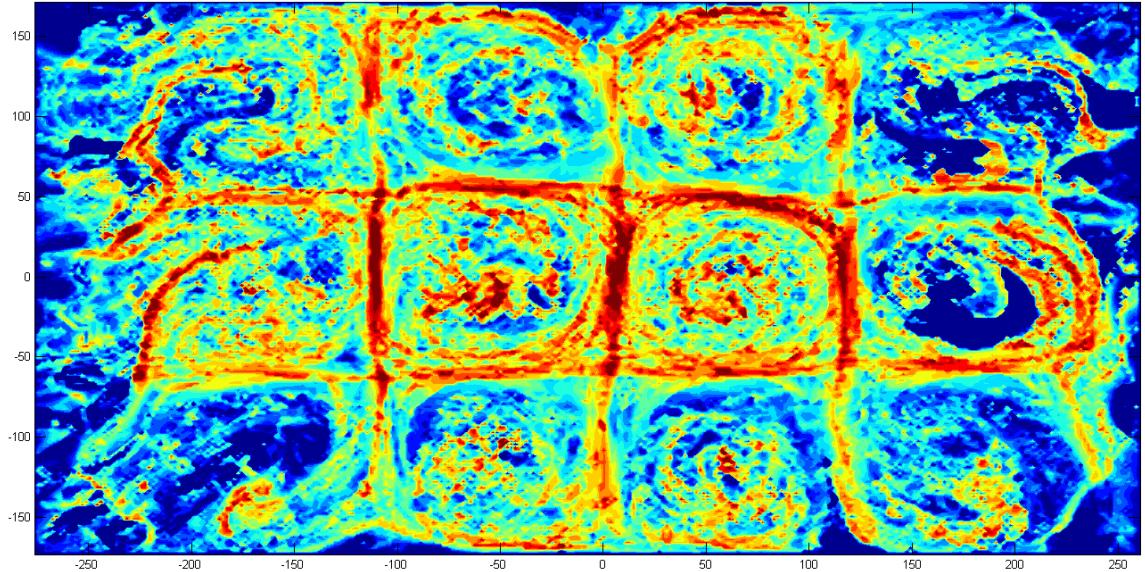


Figure 6.13: The combined, forward and backward, FTLE for the full reconstruction of the flow field from a 59 mode DMD computation

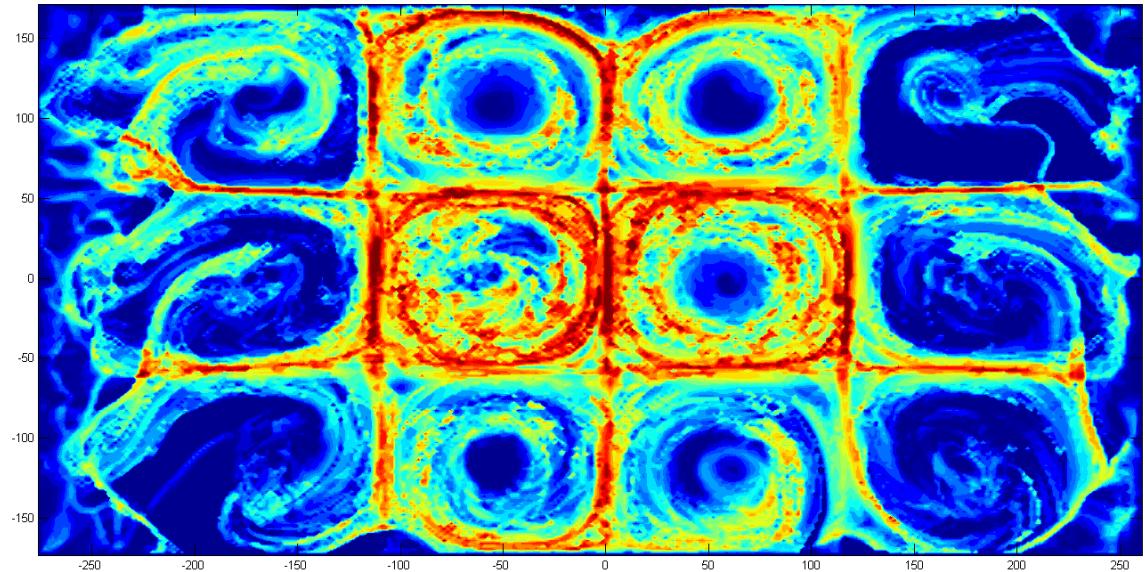


Figure 6.14: The combined, forward and backward, FTLE for the primary mode reconstruction of the flow field from a 59 mode DMD computation

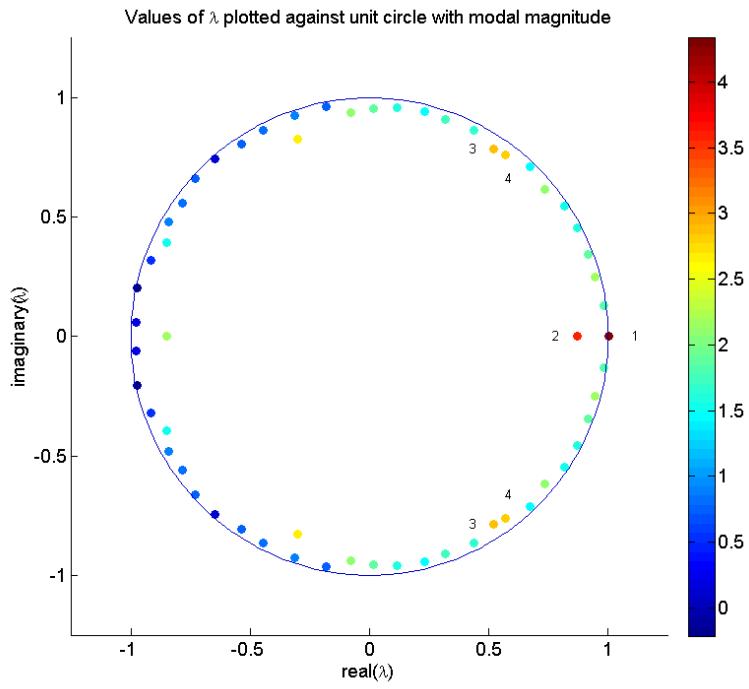


Figure 6.15: A plot of the imaginary vs. real parts of the Ritz values computed for a 59 mode DMD computation. The color of each point displays the magnitude. The first 4 dominate modes are annotated.

## 7. Applications

This section presents the basic principles behind the multi-agent LCS tracking strategy which is the main motivation behind creating this experimental testbed. It is followed by a brief discussion about the Autonomous Surface Vehicles that are used as well. This section is purely offered as insight to the project in its entirety and is not a necessary read for the work presented in this thesis.

### 7.1 Multi-Robot LCS Tracking Strategy

Inspired by the Proper Interior Maximum (PIM) triple technique a control methodology was developed for the tracking of LCS utilizing 3 ASV's [21]. In this section I will give a brief overview of the PIM triple technique and the control methodology as its validation and improvement in implementation is the greater reason for the work contributed by this thesis.

The PIM triple is a numerical technique used to locate stationary trajectories in a chaotic dynamical system which remain in a region with no attractors for an arbitrarily long period of time. For the dynamical system  $\dot{\mathbf{x}} = F(\mathbf{x})$ , let  $\mathcal{D} \subset \mathbb{R}^2$  be a closed and bounded set which doesn't contain any attractors of  $\dot{\mathbf{x}} = F(\mathbf{x})$ . The escape time of a point  $\mathbf{x} \in \mathcal{D}$  is given by  $T_E(\mathbf{x})$ , which is the time it takes  $\mathbf{x}$  to leave  $\mathcal{D}$  under  $\dot{\mathbf{x}} = F(\mathbf{x})$ . Then by locally sampling from three points along a line segment  $J$  that runs perpendicular to, or straddles, the manifold being tracked and defining the three points as  $\{\mathbf{x}_L, \mathbf{x}_C, \mathbf{x}_R\}$  which respectively stand for left, center and right. Where  $\{\mathbf{x}_L, \mathbf{x}_C\}$  are taken to be the end points of the line segment  $J$  with  $\mathbf{x}_C$  located between them. These points make a Proper Interior Maximum triple if  $T_E(\mathbf{x}_C) > \max\{T_E(\mathbf{x}_L), T_E(\mathbf{x}_R)\}$  and  $[\mathbf{x}_R, \mathbf{x}_L]$  is a proper subset of  $J$ .

The problem is set up as a numerical technique with the first iterates defined by  $J_0$  and  $\epsilon_0 > 0$  where  $\epsilon_0$  discretizes  $J_0$  such that  $\mathbf{x}_{R_0} = \mathbf{q}_0 < \mathbf{q}_1 < \dots < \mathbf{q}_m = \mathbf{x}_{L_0}$ . Then by interpolating we evaluate  $T_E(\mathbf{q}_j)$  for all  $\mathbf{q}_j$  such that a new triplet  $\{\mathbf{q}_{i-1}, \mathbf{q}_i, \mathbf{q}_{i+1}\}$  can be found which satisfies the requirements of the PIM triple, where  $\mathbf{q}_i = \arg \max_{j=1, \dots, M} T_E(\mathbf{q}_j)$ . The process is then restarted on the next iteration setting  $J_1 = [\mathbf{q}_{i-1}, \mathbf{q}_{i+1}]$  and discretizing the new segment by  $0 < \epsilon_1 < \epsilon_0$ . It can also be shown that if the initial conditions of the system are set to the PIM triple  $\mathbf{x}(t) = \{\mathbf{x}_L, \mathbf{x}_C, \mathbf{x}_r\}$  and the system is propagated forward in time that the line segment containing  $\mathbf{x}(t + \Delta T)$  will also be a PIM triple. Therefore, the system can be propagated forward and the PIM triple technique can be used on all  $J_{t+\Delta T}$  to reveal the entire stable manifold.

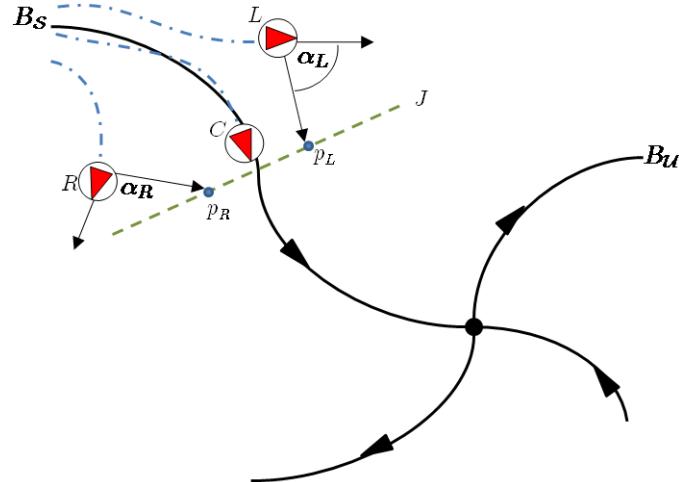


Figure 7.1

Transferring this technique to a control methodology for a fleet of 3 ASV's follows. Imagine each vehicle as one of the points in a PIM triple, each capable of localization with respect to one another as well as local flow sensing capabilities. The vehicles are each modeled with differential drive kinematics shown by (7.1) with their global position defined by  $\mathbf{x}_i = [x_i, y_i]^T$ . Where  $V_i$  is the forward velocity of the vehicle with respect to the flow and  $\theta_i$  is the heading angle with respect to the global frame, both of these are designated as control inputs. The local 2D flow field around the vehicle is described by  $\mathbf{u}_i = [u_i, v_i]^T$ .

$$\dot{x}_i = V_i \cos \theta_i + u_i, \quad (7.1a)$$

$$\dot{y}_i = V_i \sin \theta_i + v_i, \quad (7.1b)$$

We will talk about the vehicles as  $L$ ,  $C$  and  $R$  respectively as we did with the points in the PIM explanation and as in Figure 7.1. The vehicles  $L$  and  $R$  act as the extremum points straddling the stable manifold  $B_s$  also seen in Figure 7.1. The goal of these vehicles is to provide knowledge of their local flow fields while staying in proximity of vehicle  $C$ . Vehicle  $C$  is tasked, using the PIM triple technique, to follow  $B_s$ .

Vehicles  $L$  and  $R$  are independently prescribed with two control states, a passive state which we will refer to as  $U_p$  where  $V_i = 0$  and an active state  $U_a$ . The active state is characterized by  $V_i = \|(\mathbf{p}_i - \mathbf{x}_i) - \mathbf{u}_i\|$ , where  $\mathbf{p}_i$  is the desired location to reach on the projected line segment  $J_{t+\Delta T}$ ,

and  $\theta = \alpha_i(t)$ , where  $\alpha$  is the angle between the current and desired heading as can be seen in Figure 7.1. When bounded within  $d_{min} < |\mathbf{x}_C - \mathbf{x}_i| < d_{max}$  from vehicle  $C$  the vehicle will remain in a passive state. Should it leave these bounds it will enter the active state to move itself back within the bounds.

The objective of vehicle  $C$  is to follow the manifold  $B_s$  using its local data as well as the data collected by vehicles  $L$  and  $R$ . Letting  $\mathbf{x}(t_0) = \{\mathbf{x}_L(t_0), \mathbf{x}_C(t_0), \mathbf{x}_R(t_0)\}$  be a PIM triple on  $J_0$ , straddling  $B_s$ , where the Euclidean distance  $d(\mathbf{x}_c, B_s) < \epsilon_0$  and once again discretizing the system by  $\epsilon_0 > 0$  such that  $\mathbf{x}_{R_0} = \mathbf{q}_0 < \mathbf{q}_1 < \dots < \mathbf{q}_m = \mathbf{x}_{L_0}$  we resort to the PIM triple technique of earlier. Interpolating over the vector field using the Inverse Distance Weighting method, shown below where  $w_{ik} = \|\mathbf{x}_i(k) - \mathbf{q}_i\|^{-2}$ , to find all  $\mathbf{q}_j$ . We then search for the manifold  $B_s$  which is approximately located at  $\mathbf{q}_i = \arg \max_{j=1,\dots,M} \mathbf{u}(\mathbf{q}_j)$ .

$$\mathbf{u}(\mathbf{q}_j) = \sum_k \sum_{i=1}^N \frac{w_{ik} \mathbf{u}_i(k)}{\sum_k \sum_{i=1}^N w_{ik}}$$

Projecting the new PIM triple forward in time gives the desired destination of each vehicle. Where vehicle  $C$  should go to  $\mathbf{q}_i(t_0 + \Delta T)$  and vehicles  $L$  and  $R$  should move to  $\mathbf{q}_{i-1}(t_0 + \Delta T)$  and  $\mathbf{q}_{i+1}(t_0 + \Delta T)$  respectively. Making the desired heading of each vehicle,

$$\mathbf{V}_{d_j} = a(\mathbf{q}_j(t_0 + \Delta T) - \mathbf{x}_j(t_0)) \quad (7.2)$$

Where  $j = \{L, C, R\}$ ,  $a$  is a gain and  $\mathbf{q}_j(t_0 + \Delta T)$  is the new iteration of the PIM triple propagated forward in time. Continuing with this strategy allows the team to track a LCS in real-time.

An important notion is the selection of the parameter  $d_{max}$  such that curvature of the manifold being tracked  $\rho_{min}(B_*) > d_{max}$ . If  $\rho_{min}(B_*) < d_{max}$  an erroneous case would occur causing the method to fail. Typically there is no knowledge of  $\rho_{min}(B_*)$  in real systems at the time of operation so a suitable  $d_{max}$  must be selected.

## 7.2 Autonomous Surface Vehicles

Tracking experiments are performed with two different types of ASV's. The first type is titled micro Autonomous Surface Vehicle (mASV) and is a smaller vehicle seen in Figure 7.2 with a differential thrust drive, an on-board suite of inertial sensors and wireless communication capabilities.

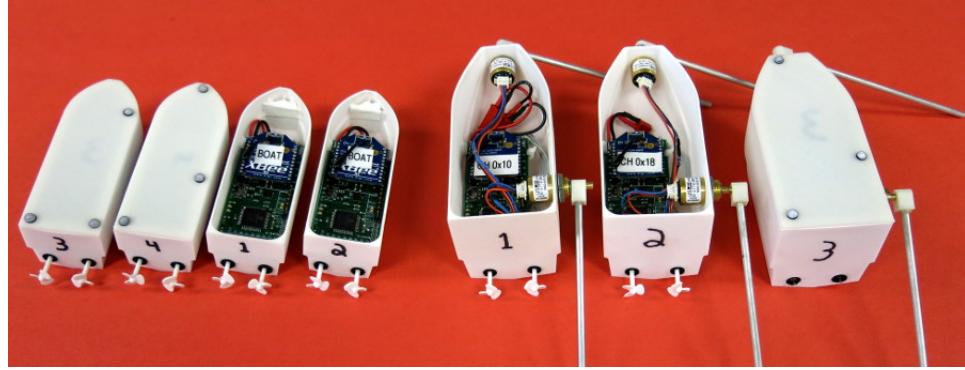


Figure 7.2: Examples of the two classes of autonomous surface vehicles used for this work, with and without 3D printed cap. Pictured on the left are the mASVs, and on the right are the mASVF's with flow sensors.

Table 7.1: Comparison of mASV and mASVF vehicle classes from [?].

	mASV	mASVF
Mass	46.8 g	109.5 g
Moment of Inertia (yaw)	$3 \cdot 10^{-5} \text{ kg m}^2$	$1.2 \cdot 10^{-4} \text{ kg m}^2$
Overall length	10.2 cm	12.8 cm
Width	3.3 cm	6.3 cm
Draft	2.3 cm	3.2 cm
Max. forward speed	$\sim 0.2 \text{ m/s}$	$\sim 0.2 \text{ m/s}$
Max. turning rate	$\sim 2.0 \text{ rad/s}$	$\sim 1.0 \text{ rad/s}$
Forward drag coefficient	$8.0 \text{ m}^{-1}$	unknown
Rotational drag coefficient	$1.0 \text{ rad}^{-1}$	unknown
Battery life	$> 2 \text{ hours}$	$\sim 1 \text{ hour}$
Local flow measurement	none	2 flow sensing rod/encoder assemblies

The second type is titled micro Autonomous Surface Vehicle - Flow Sensing (mASVF) and is a slightly larger vehicle than the first but is essentially the same except for the added capability of local flow sensing, the vehicle can be seen in Figure 7.2. A more comprehensive explanation on the construction and capabilities of these ASV's can be found in [?]. The mASV's are primarily used for simulated flow tracking experiments while the mASVF's are used in experiments with real flow fields. A comparison of the vehicles is give in Table 7.1 with basic performance characteristics. The scaling of the flow from the HiRe tank to the MR tank becomes very important in relation to these vehicles. The structures must be large enough such that a team of 3 of these robots are capable of tracking the structure.

## 8. Conclusion and Future Work

### 8.1 Conclusions

This work has presented the development and validation of an experimental testbed capable of consistently reproducing Lagrangian Coherent Structures in a quasi-2D flow field. The experimental system itself was designed in such a way that the flow drivers can be arranged and independently controlled in many different patterns allowing greater flexibility for future experiments that desire different flow fields than the double gyre model. The experiments performed during this testing demonstrated that a real flow producing LCS similar to those of the double-gyre model is feasible, validating the use of the model for much of the previous simulation work that was done for the multi-agent LCS tracking strategy. It was also shown that the LCS can be evaluated as being quasi-static in their location to within some finite boundary in this system. Thus, it has been demonstrated that LCS can be produced in a laboratory setting with consistent repeatability aiding to the future development of the multi-agent LCS tracking strategy. The ability of the Dynamic Mode Decomposition to resolve relevant spatial and temporal modes of the system was shown as well. This ability to use the DMD on such a system is important because it allows us to extract relevant dynamics of the flow from a purely observable standpoint without any understanding of the underlying system dynamics.

Although this work focused on showing that the LCS ridges are stationary, in space and time within some finite spatial boundary, it should be noted that there does appear to be a periodic time varying component to the LCS location near saddle points in several instances when observing the FTLE field as a video that look similar to the Time-Varying Double Gyre. The rotational speed of the flow drivers in this case did not appear to greatly affect the location of the LCS or the size of the bounded region which they occupy. A more rigorous analysis and comparison to the Time-Varying Double Gyre may show the system to follow that model more closely than that of the Static Double Gyre.

### 8.2 Future Work

A significant amount of work on this project still remains in order to understand the relation between how LCS evolve with respect to the flow generation system. The experiments carried

out in this thesis were performed with a single distance between cylinders. As mentioned in the results there were not many differences in LCS structure between different motor speeds. This may potentially be due to the distance between cylinders being fixed. Continuation of this work should start with an evaluation of the relationship between LCS location and the distance between flow drivers. Examining the effect of separation distance at several different speeds should reveal more about the interaction between flow generation and the dynamics of LCS in the flow. Once a better understanding of that relationship has been reached the effect of flow driver cylinder diameter could be evaluated. With an understanding of the relationship of LCS dynamics with flow driver separation distance, speed and distance it will be possible to talk about scalability. Once there is evidence of scalability it can be resolved whether or not the MR tank will behave as hypothesized.

As future experiments become more complex, such as individual time-varying motor speeds, a graphical interface for the control and monitoring of the motor system would be very useful. Currently in place is an Arduino which works as a middle man from the PC to the Motor Controller boards. For the complexity level of the experiments in this thesis it worked well but would quickly become tedious when dealing with more complex experiments.

Moving on towards true model synthesis experiments should be performed where a single motor has its speed increased or decreased relative to the rest of the set. Performing this for each motor and computing the DMD or another similar modal decomposition will allow for the production of a set of basis functions that can be used to model the system for given motor speed inputs. In the  $4 \times 3$  flow driver layout the evaluation of symmetry should be examined. Should there exist a symmetry in the basis modes it may prove that fewer experiments are required in order to yield future basis functions from more advanced operating schemes. From there the evaluation of more than one motor spinning at a different speed should be performed. Utilizing the DMD again the set of basis vectors can begin to grow until the set becomes linearly dependent, if it ever does. This type of examination should also be performed while evaluating the relationship of LCS location with flow driver speed, separation distance and diameter in-order to build a full picture of the system. More advanced schemes can then be performed such as the evaluation of the transients caused by step inputs or ramp inputs to the system. The assessment of all these things will lead to a model of the system based on real data that will allow the user to 'design' LCS patterns in the system based on motor control.

The experimental stand should be improved as well especially the PIV camera mounting system. A stiffer camera mounting system which reduces vibration amplitudes caused by the motors should

be created. Currently the camera stand has the potential to wobble which may be introducing more noise into the system. The resultant vector fields from the PIV appear to be clean but the ability to rule out such vibration as a source of noise would strengthen the experimental results. As discussed in the results section it was also unclear if the multiple closely packed parallel LCS ridges were a cause of low spatial sampling. This should be tested by moving the cameras closer to increase spatial sampling frequency. This should resolve the question of whether multiple LCS ridges exist.

## Appendix A. Supplementary Images

### A.1 PIV Resolved Vector Fields

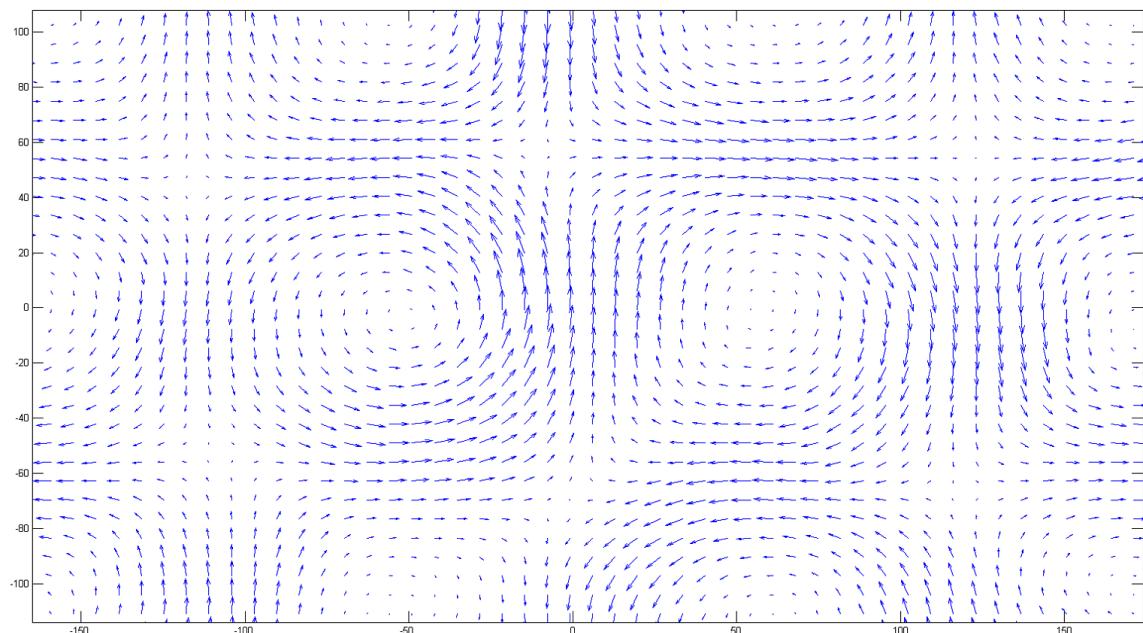


Figure A.1: Snapshot of PIV resolved vector field for the first run at 60 rpm

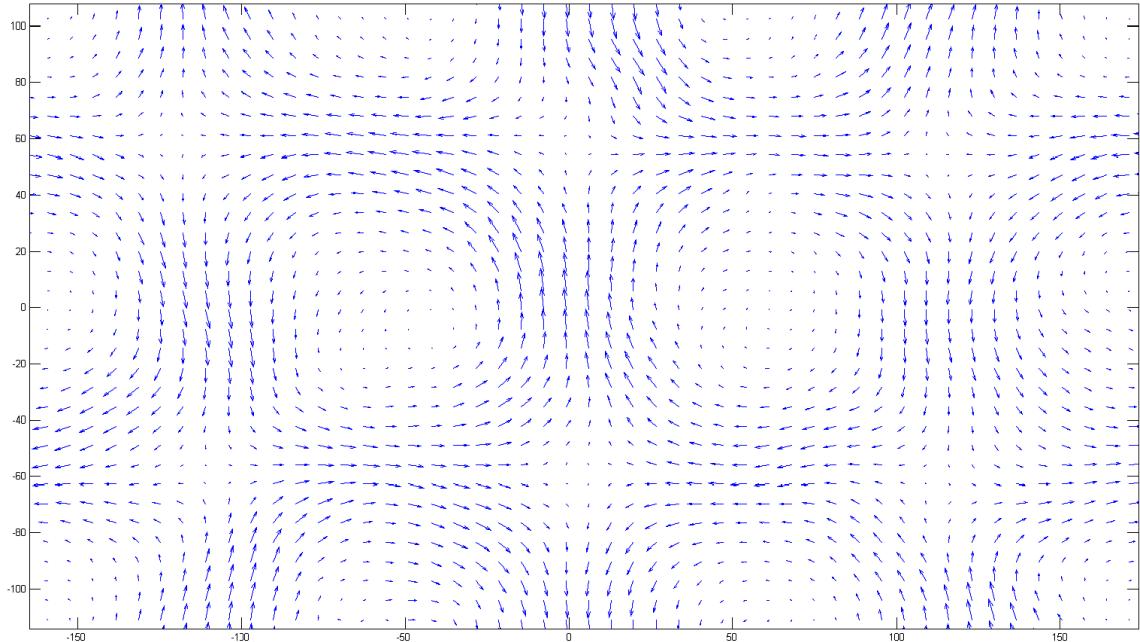


Figure A.2: Snapshot of PIV resolved vector field for the first run at 75 rpm

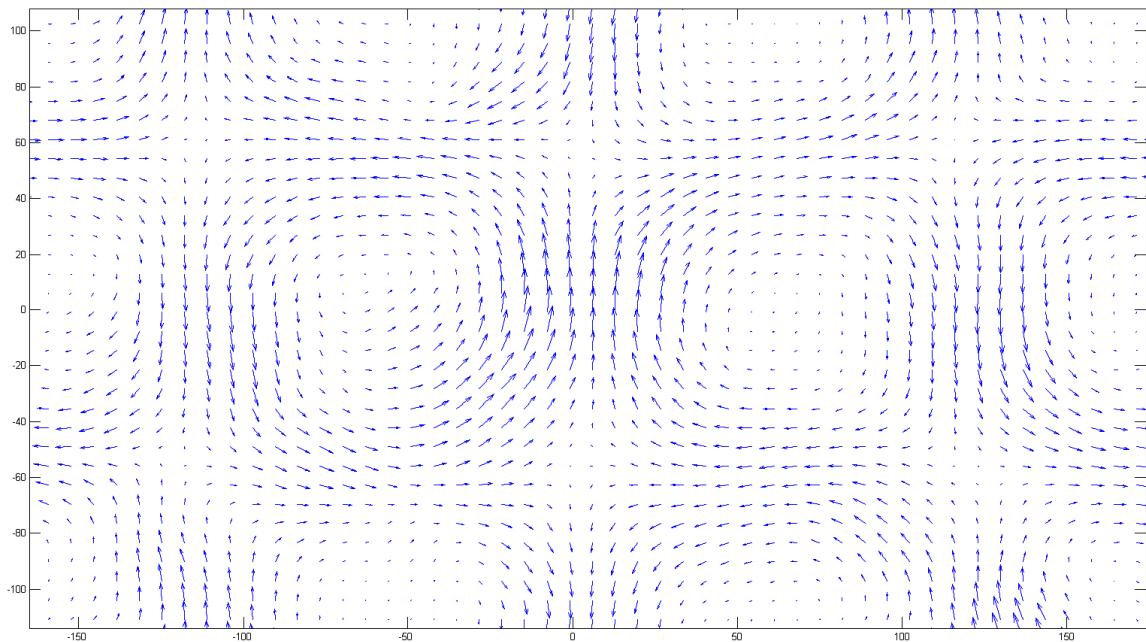


Figure A.3: Snapshot of PIV resolved vector field for the first run at 90 rpm

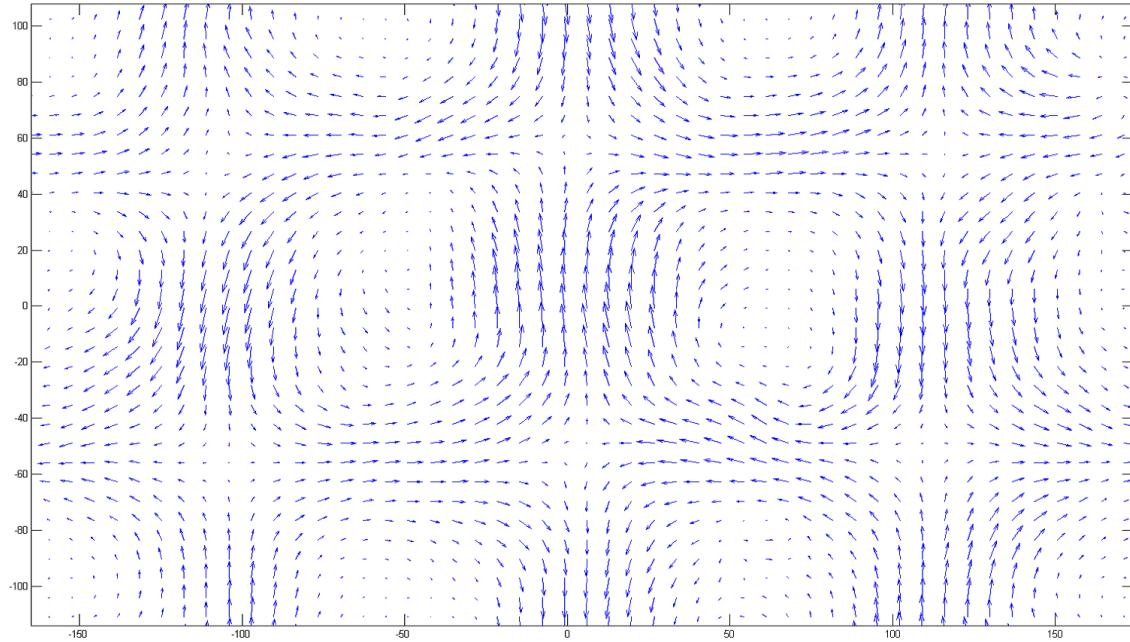


Figure A.4: Snapshot of PIV resolved vector field for the first run at 105 rpm

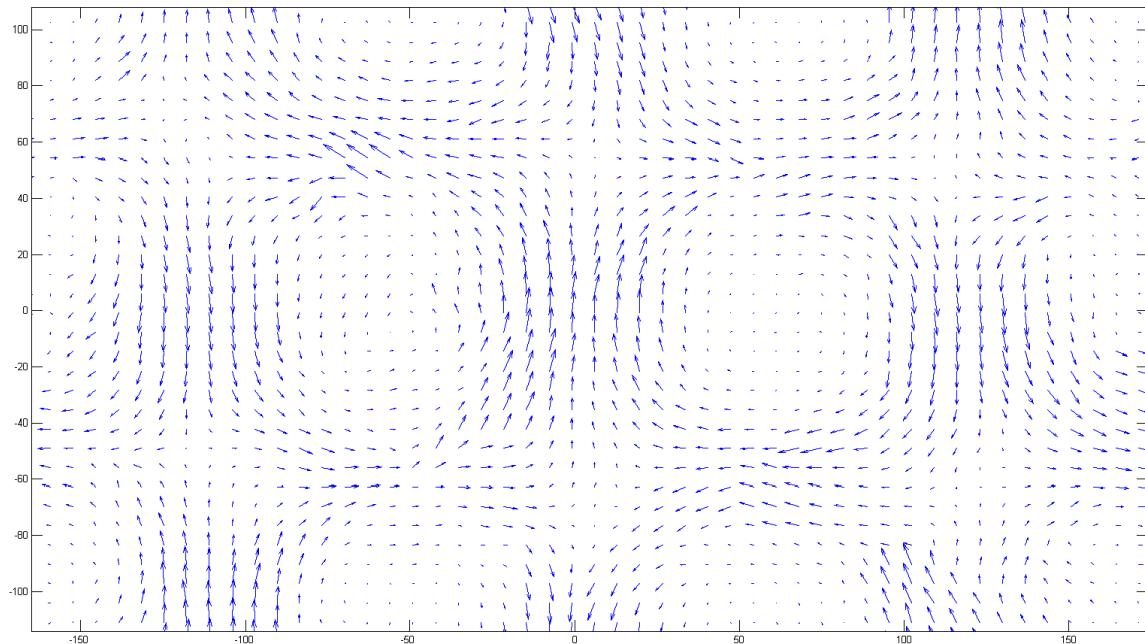


Figure A.5: Snapshot of PIV resolved vector field for the first run at 120 rpm

### A.2 Resultant FTLE fields

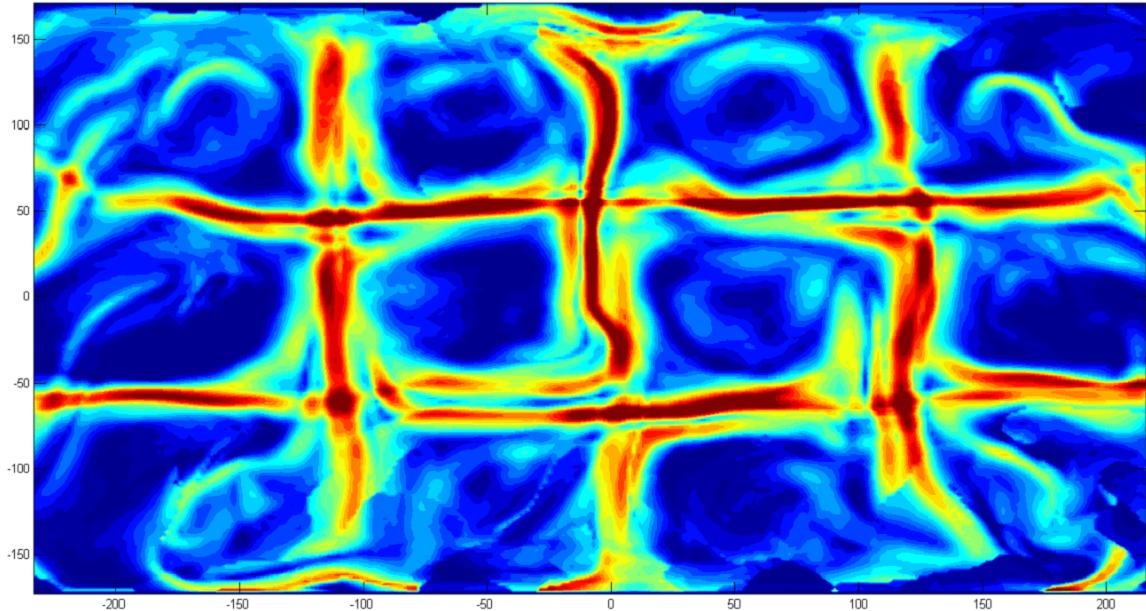


Figure A.6: Snapshot FTLE for first run at 60 rpm

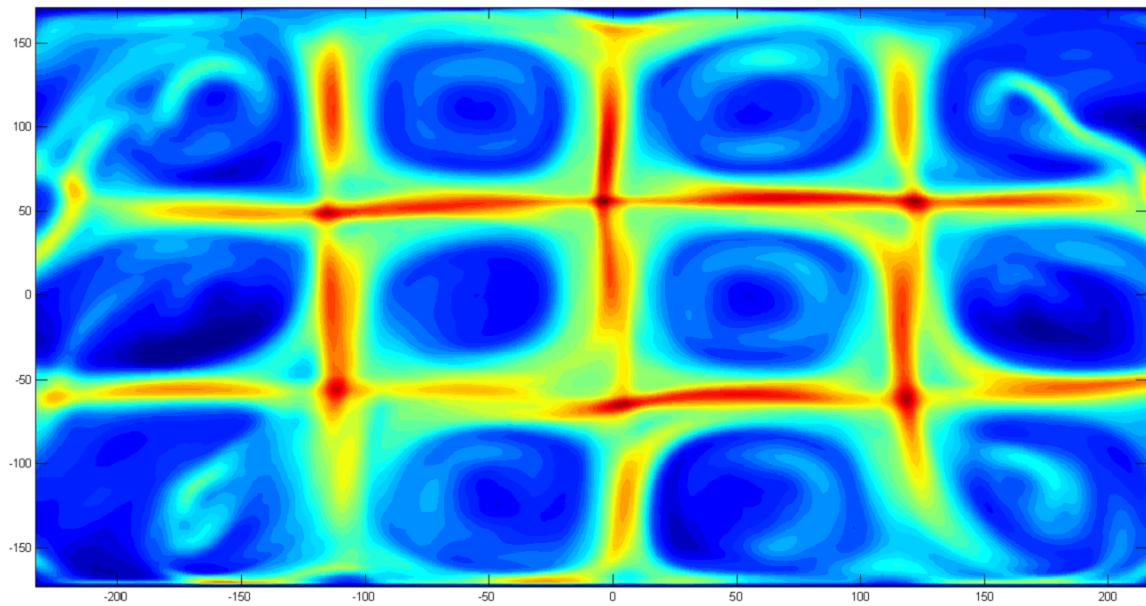


Figure A.7: Time series average FTLE for first run at 60 rpm

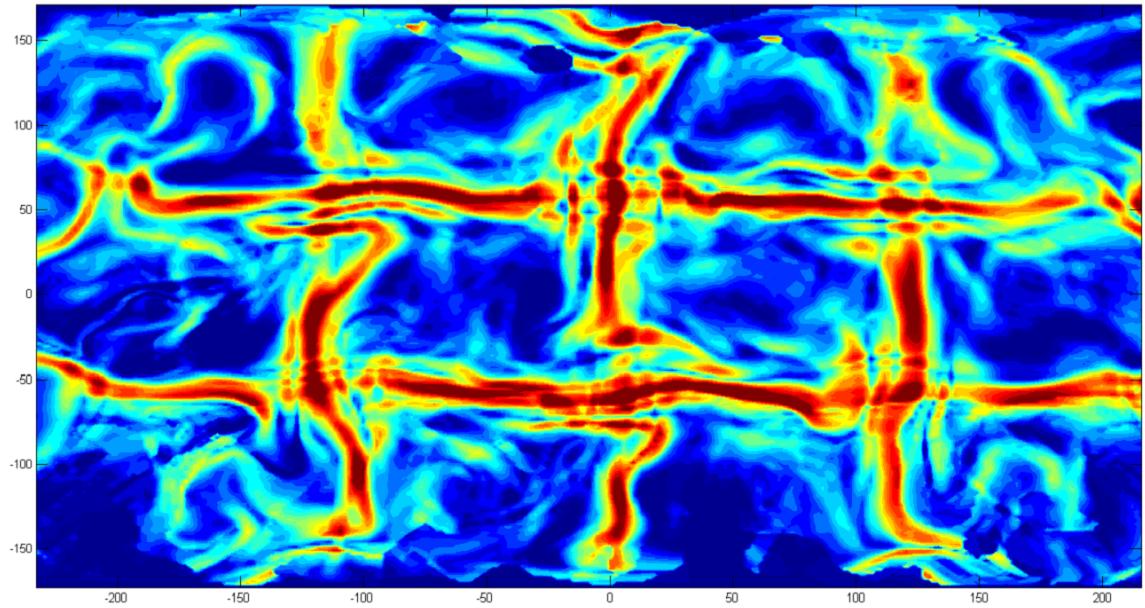


Figure A.8: Snapshot FTLE for first run at 75 rpm

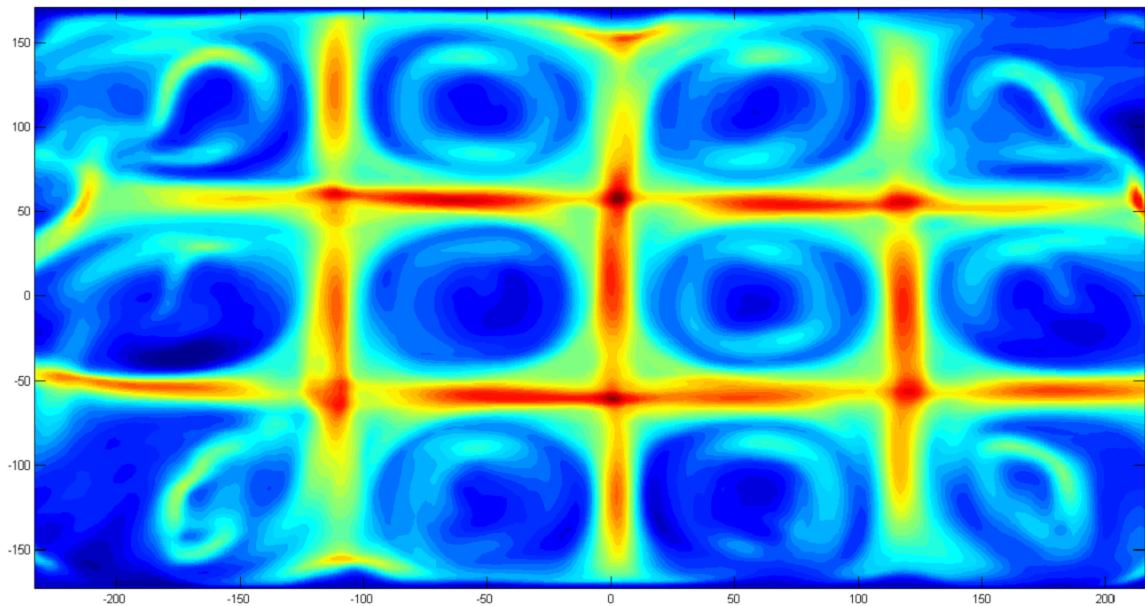


Figure A.9: Time series average FTLE for first run at 75 rpm

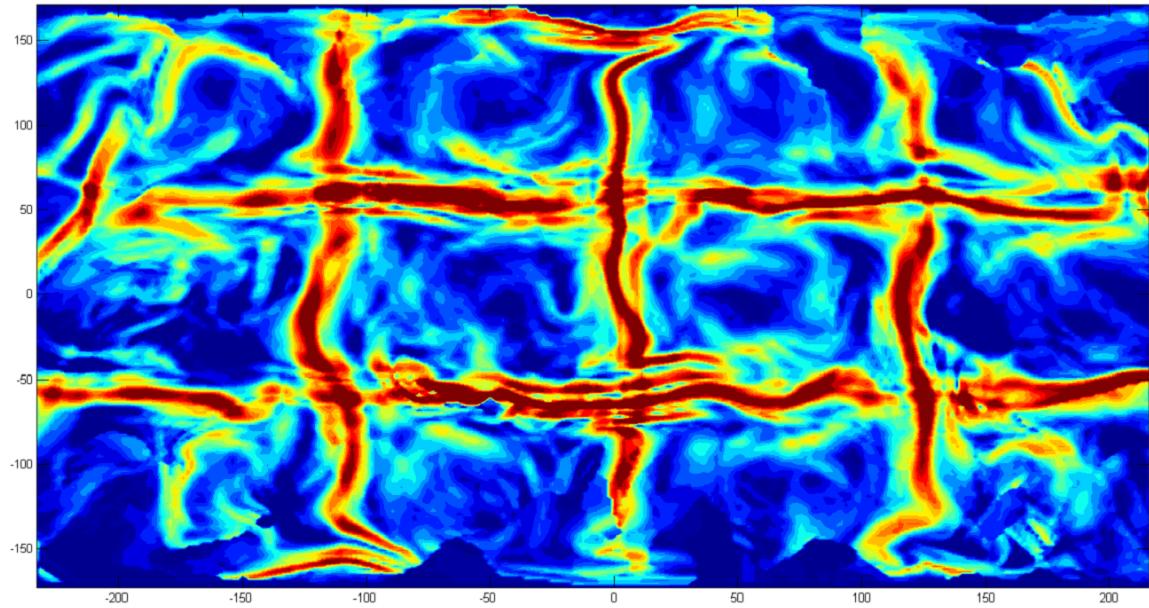


Figure A.10: Snapshot FTLE for first run at 90 rpm

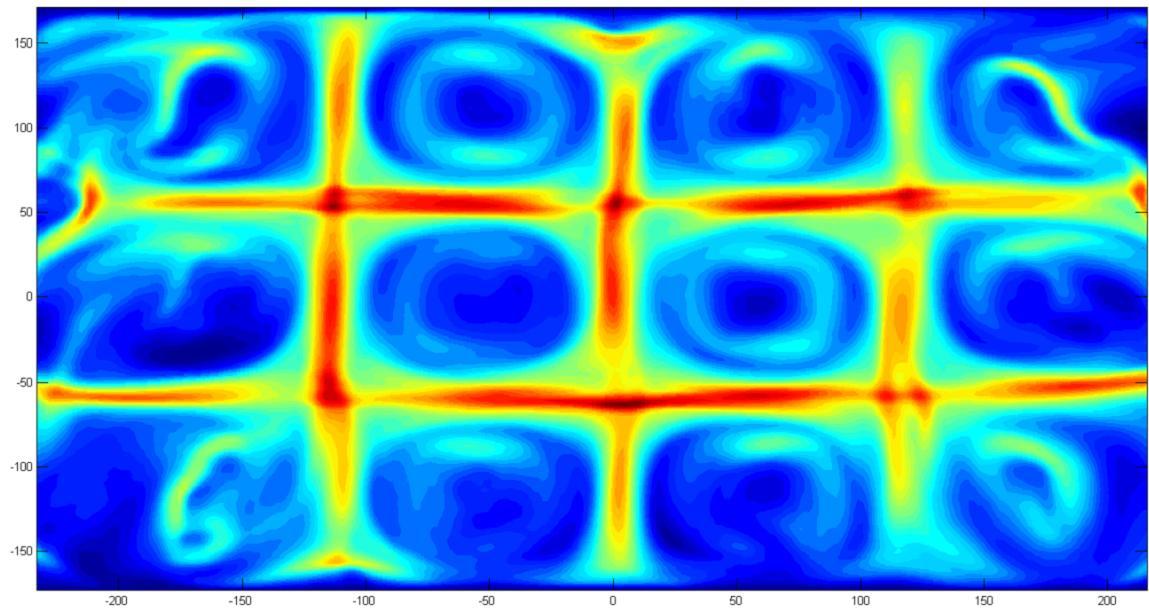


Figure A.11: Time series average FTLE for first run at 90 rpm

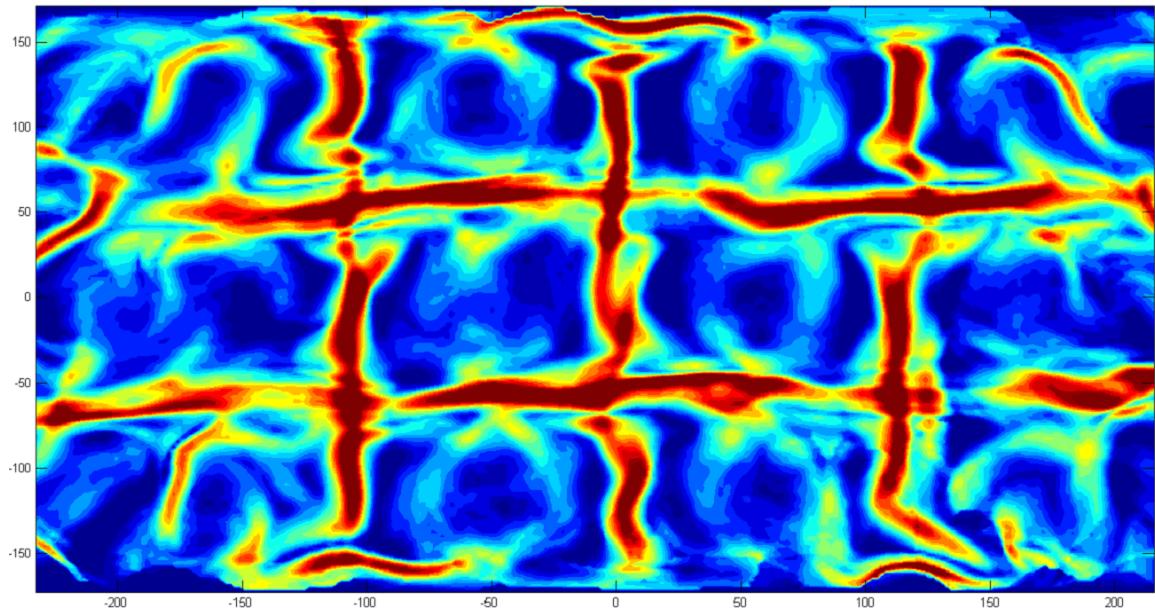


Figure A.12: Snapshot FTLE for first run at 105 rpm

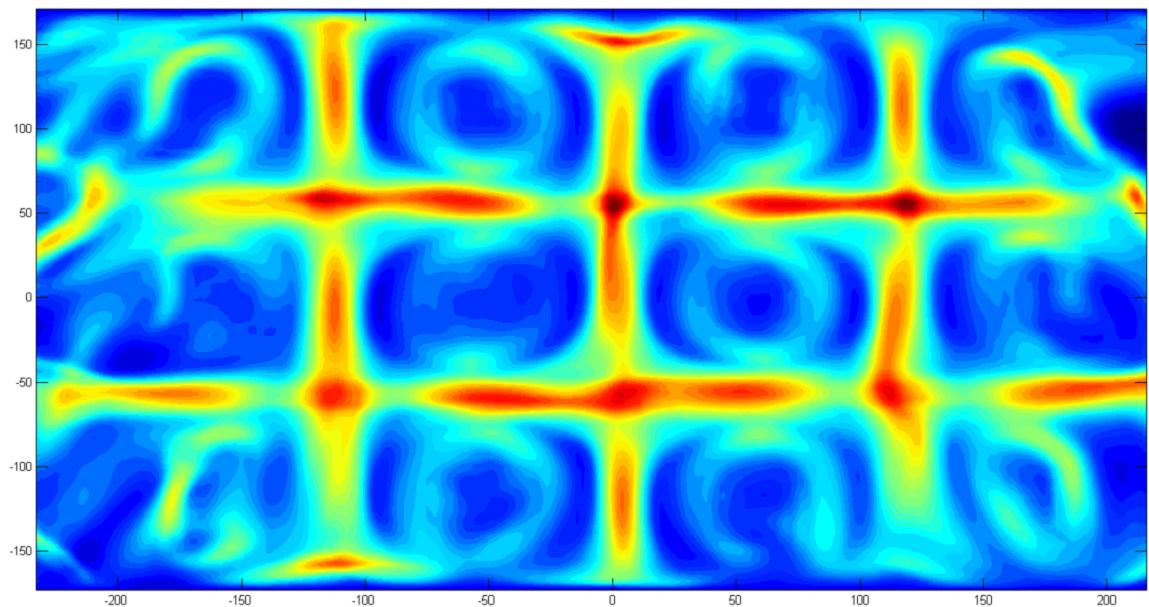


Figure A.13: Time series average FTLE for first run at 105 rpm

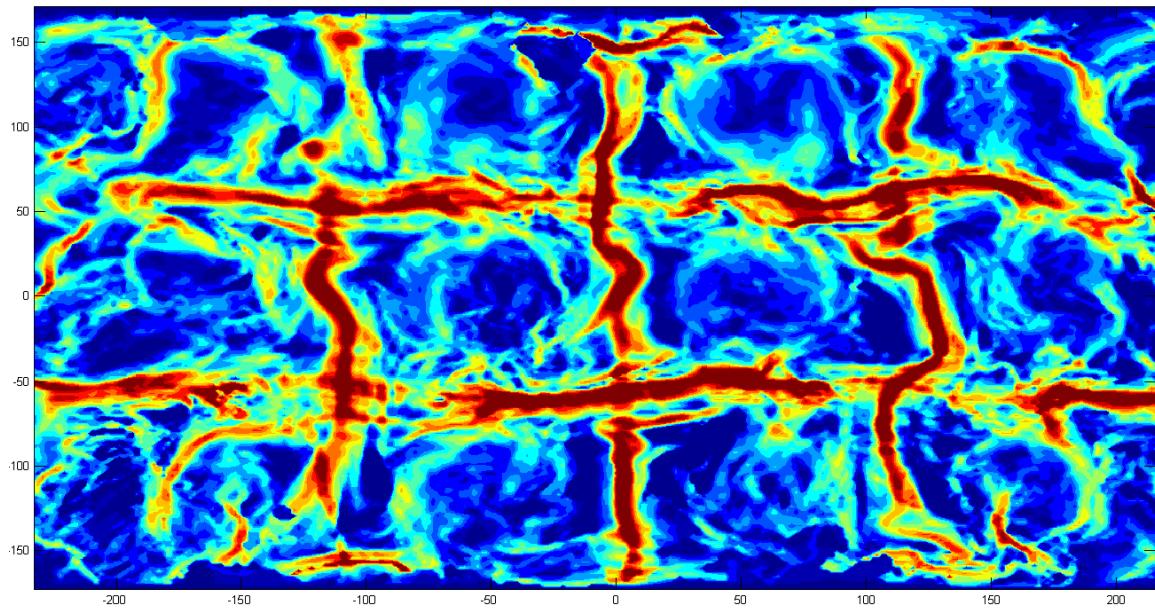


Figure A.14: Snapshot FTLE for first run at 120 rpm

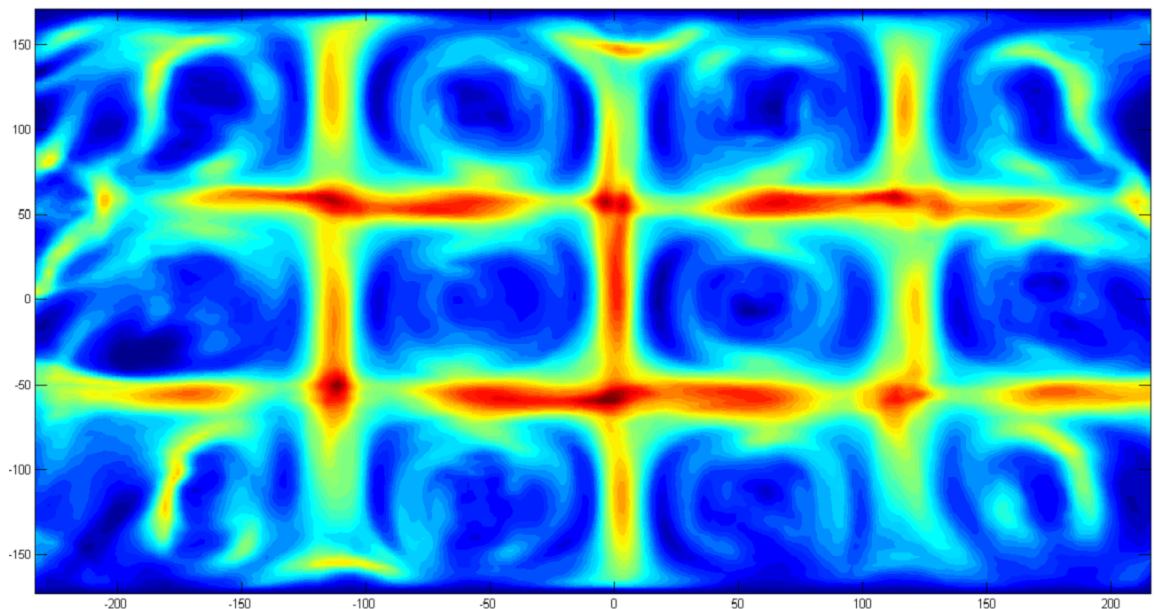


Figure A.15: Time series average FTLE for first run at 120 rpm



## Bibliography

- [1] Lagrangian coherent structures: Analysis of time-dependent dynamical systems using finite-time lyapunov exponents.
- [2] MATLAB piv toolbox. <http://www.oceanwave.jp/softwares/mpiv/>.
- [3] Naitonal HF RADAR network - surface currents.
- [4] C. C. Landreth P. W. Offutt A. K. Prasad, R. J. Adrian. Effect of resolution on the speed and accuracy of particle image velocimetry interrogation. *Experiments in Fluids*, 13:105–116, June 1992.
- [5] Ronald J. Adrian, Richard P. Wildes, Michael J. Amabile, Ann marie Lanzillotto, and Tzong shyng Leu. Particle imaging techniques for experimental fluid mechanics. In *In Proc. Conf. Comp. Vision Pattern Rec*, pages 969–975, 1991.
- [6] Juan C. Agui and J. Jimenez. On the performance of particle tracking. *Journal of Fluid Mechanics*, 185:447–468, 1987.
- [7] Alexander Yarin Cameron Tropea and John Foss. Springer handbook of experimental fluid mechanics. 2007.
- [8] Shervin Bagheri Philipp Schlatter Clarence W. Rowley, Igor Mezic and Dan S. Henningson. Reduced-order models for flow control: balanced models and koopman modes. In *Seventh IUTAM Symposium on Laminar-Turbulent Transition Bookseries*, pages 43–50, 2009.
- [9] Shervin Bagheri Philipp Schlatter Clarence W. Rowley, Igor Mezic and Dan S. Henningson. Spectral analysis of nonlinear flows. *Journal of Fluid Mechanics*, 641:115–127, December 2009.
- [10] R. Doerner, B. Hübinger, W. Martienssen, S. Grossmann, and S. Thomae. Stable manifolds and predictability of dynamical systems. *Chaos, Solitons, & Fractals*, 10(11):1759–1782, November 1999.
- [11] Eric Forgoston, Lora Billings, Phillip Yecko, and Ira B. Schwartz. Set-based corral control in stochastic dynamical systems: Making almost invariant sets more invariant. *Chaos*, 21(013116), 2011.
- [12] V. M. Goncalves, L. C. A. Pimenta, C. A. Maia, B. Dutra, and G. A. S. Pereira. Vector fields for robot navigation along time-varying curves in n-dimensions. *IEEE Trans. on Robotics*, 26(4):647–659, 2010.
- [13] G. Haller. Lagrangian coherent structures from approximate velocity data. *Physics of Fluids (1994-present)*, 14(6), 2002.
- [14] G. Haller and A.C. Poje. Finite time transport in aperiodic flows. *Physica D: Nonlinear Phenomena*, 119(34):352 – 380, 1998.
- [15] G. Haller and G. Yuan. Lagrangian coherent structures and mixing in two-dimensional turbulence. *Phys. D*, 147:352–370, Dec 2000.
- [16] George Haller. Distinguished material surfaces and coherent structures in three-dimensional fluid flows. *Physica D: Nonlinear Phenomena*, 149(4):248–277, 2001.
- [17] George Haller. A variational theory of hyperbolic lagrangian coherent structures. *Physica D: Nonlinear Phenomena*, 240(7):574–598, 2011.

- [18] George Haller and Themistoklis Sapsis. Lagrangian coherent structures and the smallest finite-time lyapunov exponent. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 21(2):023115, 2011.
- [19] C. S. Harrison and G. A. Glatzmaier. Lagrangian coherent structures in the California Current System - sensitivities and limitations. *Geophysical and Astrophysical Fluid Dynamics*, 106:22–44, October 2010.
- [20] CH Hsieh, ZP Jin, D Marthaler, BQ Nguyen, DJ Tung, AL Bertozzi, and RM Murray. Experimental validation of an algorithm for cooperative boundary tracking. In *Proc. of the 2005 American Control Conference.*, pages 1078–1083, 2005.
- [21] M.A. Hsieh, E. Forgoston, T.W. Mather, and I.B. Schwartz. Robotic manifold tracking of coherent structures in flows. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 4242–4247, May.
- [22] T. Inanc, S.C. Shadden, and J.E. Marsden. Optimal trajectory generation in ocean flows. In *American Control Conference, 2005. Proceedings of the 2005*, pages 674 – 679, 8-10, 2005.
- [23] Jonathan H. Tu Kevin K. Chen and Clarence W. Rowley. Variants fo dynamic mode decomposition: Boundary condition, koopman and fourier analyses. *Journal of Nonlinear Science*, 22:887–917, April 2012.
- [24] Dennis Larkin, Matthew Michini, Alexandra Abad, Stephanie Teleski, and M Ani Hsieh. Design of the multi-robot coherent structure testbed (mcoste) for distributed tracking of geophysical fluid dynamics. In *ASME 2014 International Design Engineering and Technical Conference*. American Society of Mechanical Engineers, August 2014.
- [25] Francois Lekien, Chad Couillette, Arthur J. Mariano, Edward H. Ryan, Lynn K. Shay, George Haller, and Jerry Marsden. Pollution release tied to invariant manifolds: A case study for the coast of florida. *Physica D: Nonlinear Phenomena*, 210(12):1 – 20, 2005.
- [26] K. Mallory, M. A. Hsieh, E. Forgoston, and I. B. Schwartz. Distributed allocation of mobile sensing swarms in gyre flows. *Nonlinear Processes Geophysics*, 20(5):657–668, 2013.
- [27] Steven T. Wereley Markus Raffel, Christian E. Willert and Jurgen Kompenhans. *Particle Image Velocimetry*. Experimental Fluid Mechanics. Springer, 2007.
- [28] R. Mei. Velocity fidelity of flow tracer particles. *Experiments in Fluids*, 22:1–13, November 1996.
- [29] Igor Mezic. Spectral properties of dynamical systems, model reduction and decompositions. *Journal of Nonlinear Dynamics*, 41:309–325, 2005.
- [30] Igor Mezic. Analysis of fluid flows via spectral properties of the koopman operator. *Annual Review of Fluid Mechanics*, 45:357–78, 2013.
- [31] M. Michini, M. A. Hsieh, E. Forgoston, and I. B. Schwartz. Experimental validation of robotic manifold tracking in gyre-like flows. In *IEEE International Conference on Intelligent Robots and Systems (IROS 2014)*, June 2014.
- [32] M. Michini, M.A. Hsieh, E. Forgoston, and I.B. Schwartz. Robotic tracking of coherent structures in flows. *Robotics, IEEE Transactions on*, PP(99):1–11, 2014.
- [33] M. Michini, H. Rastgoftar, M. A. Hsieh, and S. Jayasuriya. Distributed formation control for collaborative tracking of manifolds in flows. In *IEEE American Control Conference (ACC 2014)*, June 2014.

- [34] Matthew Michini. Tracking of Manifolds and Coherent Structures in Flows: Simulations and Experiments. 2014.
- [35] Matthew Michini, Kenneth Mallory, Dennis Larkin, M Ani Hsieh, Eric Forgoston, and Philip A Yecko. An experimental testbed for multi-robot tracking of manifolds and coherent structures in flows. In *ASME 2013 Dynamic Systems and Control Conference*, pages V002T32A002–V002T32A002. American Society of Mechanical Engineers, 2013.
- [36] H. E. Nusse and J. A. Yorke. A procedure for finding numerical trajectories on chaotic saddles. *Physica D Nonlinear Phenomena*, 36:137–156, jun 1989.
- [37] T. Peacock and G. Haller. Lagrangian coherent structures: The hidden skeleton of fluid flows. *Physics Today*, 66(2):41–47, 2013.
- [38] Hayder Salman, Jan S Hesthaven, Tim Warburton, and George Haller. Predicting transport by lagrangian coherent structures with a high-order method. *Theoretical and Computational Fluid Dynamics*, 21(1):39–58, 2007.
- [39] Joseph T Schaefer and Charles A Doswell III. On the interpolation of a vector field. *Monthly Weather Review*, 107(4):458–476, 1979.
- [40] Peter J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 676:5–28, 2010.
- [41] C. Senatore and S.D. Ross. Fuel-efficient navigation in complex flows. In *American Control Conference, 2008*, pages 1244 –1248, june 2008.
- [42] Shawn C. Shadden, Francois Lekien, and Jerrold E. Marsden. Definition and properties of lagrangian coherent structures from finite-time lyapunov exponents in two-dimensional aperiodic flows. *Physica D: Nonlinear Phenomena*, 212(34):271 – 304, 2005.
- [43] S.H. Strogatz. *Nonlinear Dynamics And Chaos*. Studies in nonlinearity. Sarat Book House, 2007.
- [44] S. Susca, S. Martinez, and F. Bullo. Monitoring environmental boundaries with a robotic sensor network. *IEEE Trans. on Control Systems Technology*, 16(2):288–296, 2008.
- [45] Ioana Triandaf and Ira B. Schwartz. A collective motion algorithm for tracking time-dependent boundaries. *Mathematics and Computers in Simulation*, 70:187–202, 2005.
- [46] J. Goree Y. Feng and Bin Liu. Accurate particle postiion measurement from images. *Review of Scientific Instruments*, 78, 2007.
- [47] Weizhong Zhang, T. Inane, S. Ober-Blobaum, and J.E. Marsden. Optimal trajectory generation for a glider in time-varying 2d ocean flows b-spline model. In *Robotics and Automation, 2008. ICRA 2008. IEEE Int. Conf. on*, pages 1083 –1088, may 2008.
- [48] Sunil Ahuja Zanhua Ma and Clarence W. Rowley. Reduced-order models for control of fluids using the eigensystem realization algorithm. *Theoretica and Computational Fluid Dynamics*, 25:233–247, June 2011.