Reduced-order modeling and adjoint-based optimization of flows with fluid-structure interactions

by

Bolun Xu

B.S., University of Science and Technology of China, China, 2014

M.Eng., University of Science and Technology of China, China, 2018

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Alan Levin Department of Mechanical and Nuclear Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2023

Abstract

Fluid-structure interaction (FSI) is ubiquitous in nature, and has drawn great attention in fluid mechanics community for about a century. However, the complexity of FSI has perplexed the physical analysis and understanding on this problem. In addition, FSI often brings a huge parametric space when moving solid boundaries exist, which makes the parametric space too computationally or experimentally costly to explore thoroughly.

In the present thesis, first a global proper orthogonal decomposition (POD) and Galerkin projection based reduced-order model (ROM) has been developed to represent the essential physics of a FSI system with moving solid boundaries. The ROM is able to work on both numerical and experimental two-dimensional (2D) and three-dimensional (3D) dataset, and has shown adequate accuracy in the reconstruction of flow dynamics and the prediction of key aerodynamic properties, while keeping the computational cost very low. Then an adjoint approach has been derived, based upon the ROM and the conventional adjoin approach, to achieve fast flow control and optimization of a FSI system with moving solid boundaries. Different strategies have been designed to guarantee the accuracy of ROMs during the optimization process. The adjoint-ROM has been applied to the stabilization of the flow field as well as the aerodynamic force optimization of 2D flows past oscillating cylinders and NACA0012 airfoils. The adjoint-ROM approach has been proved effective and fast, of which the computational cost saving may make a near-real-time flow control possible.

Additionally, the adjoint-based approach has been leveraged to study two important yet sophisticated FSI problems: the gust mitigation problem, and the hydrofoil schooling problem. The gust mitigation has been investigated for 2D and 3D heaving-pitching wing models under low Reynolds numbers. The streamwise gust and the transverse gust have been mitigated for both models. The non-cylindrical enabled full-order model (FOM) based adjoint approach has been used for all cases, while the adjoint-ROM developed in the present

work has been used for 2D weak gust mitigation only. The mitigation of gust has been realized by minimizing a well-designed objective function to recover the mean lift of the base flow, while keeping the lift profile as steady as possible. The FOM-based adjoint approach was applied to both 2D and 3D strong gusts, and was highly effective in recovering the target lift, while keeping a reasonable computational cost. The adjoint-ROM approach was used for 2D weak gusts. It was able to mitigate the gust impact with relatively lower controllability and partially recovered lift. Moreover, its online computation was fast enough for real-time control.

The hydrofoil schooling problem has been studied by FOM-based adjoint approach for the first time to optimize the hydrodynamic force on trailing hydrofoils in a school. Both 2D oscillating rigid hydrofoils and undulating flexible hydrofoils have been studied at modest Reynolds numbers. It was found the adjoint approach effectively optimized the motion and the formation of the hydrofoil school to achieve huge drag reduction as well as drag-to-thrust conversion on trailing hydrofoils. The vortex-foil interaction was investigated and was found to be crucial for the optimization of hydrodynamic force.

Reduced-order modeling and adjoint-based optimization of flows with fluid-structure interactions

by

Bolun Xu

B.S., University of Science and Technology of China, China, 2014

M.Eng., University of Science and Technology of China, China, 2018

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTER OF PHILOSOPHY

Alan Levin Department of Mechanical and Nuclear Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2023

Approved by:

Major Professor
Dr. Mingjun Wei

Copyright

Abstract

Fluid-structure interaction (FSI) is ubiquitous in nature, and has drawn great attention in fluid mechanics community for about a century. However, the complexity of FSI has perplexed the physical analysis and understanding on this problem. In addition, FSI often brings a huge parametric space when moving solid boundaries exist, which makes the parametric space too computationally or experimentally costly to explore thoroughly.

In the present thesis, first a global proper orthogonal decomposition (POD) and Galerkin projection based reduced-order model (ROM) has been developed to represent the essential physics of a FSI system with moving solid boundaries. The ROM is able to work on both numerical and experimental two-dimensional (2D) and three-dimensional (3D) dataset, and has shown adequate accuracy in the reconstruction of flow dynamics and the prediction of key aerodynamic properties, while keeping the computational cost very low. Then an adjoint approach has been derived, based upon the ROM and the conventional adjoin approach, to achieve fast flow control and optimization of a FSI system with moving solid boundaries. Different strategies have been designed to guarantee the accuracy of ROMs during the optimization process. The adjoint-ROM has been applied to the stabilization of the flow field as well as the aerodynamic force optimization of 2D flows past oscillating cylinders and NACA0012 airfoils. The adjoint-ROM approach has been proved effective and fast, of which the computational cost saving may make a near-real-time flow control possible.

Additionally, the adjoint-based approach has been leveraged to study two important yet sophisticated FSI problems: the gust mitigation problem, and the hydrofoil schooling problem. The gust mitigation has been investigated for 2D and 3D heaving-pitching wing models under low Reynolds numbers. The streamwise gust and the transverse gust have been mitigated for both models. The non-cylindrical enabled full-order model (FOM) based adjoint approach has been used for all cases, while the adjoint-ROM developed in the present

work has been used for 2D weak gust mitigation only. The mitigation of gust has been realized by minimizing a well-designed objective function to recover the mean lift of the base flow, while keeping the lift profile as steady as possible. The FOM-based adjoint approach was applied to both 2D and 3D strong gusts, and was highly effective in recovering the target lift, while keeping a reasonable computational cost. The adjoint-ROM approach was used for 2D weak gusts. It was able to mitigate the gust impact with relatively lower controllability and partially recovered lift. Moreover, its online computation was fast enough for real-time control.

The hydrofoil schooling problem has been studied by FOM-based adjoint approach for the first time to optimize the hydrodynamic force on trailing hydrofoils in a school. Both 2D oscillating rigid hydrofoils and undulating flexible hydrofoils have been studied at modest Reynolds numbers. It was found the adjoint approach effectively optimized the motion and the formation of the hydrofoil school to achieve huge drag reduction as well as drag-to-thrust conversion on trailing hydrofoils. The vortex-foil interaction was investigated and was found to be crucial for the optimization of hydrodynamic force.

Table of Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Dr. Mingjun Wei, for his invaluable guidance, support, and encouragement during the past five years in my PhD journey. Dr. Wei not only gave me helpful suggestions when I ran into problems in my research, but also fostered a relaxed and open lab culture, always encouraging us to explore our own research interests and pursue our own ideas freely. Under Dr. Wei's unique way of mentoring without much constrain or micromanagement, I was able to develop my research skills to independently come up with new ideas and solve problems, which are valuable for my future career. I cannot think of a better advisor to work with than Dr. Wei.

I would like to thank Dr. John T. Hrynuk from U.S. Army Research Lab for his constant support. Dr. Hrynuk's advice from an experimentalist's perspective always inspired me to think deeper on the physics. More importantly, this PhD work cannot be accomplished without the fellowship provided by ARL in the past two years, and the financial support was also important to allow me to attend different academic conferences. I would like to thank my supervisory committee members, Dr. Shih-Kang Fan, Dr. Hitesh Bindra, and Dr. Chih-Hang Wu, for their willingness to serve in the committee and advice on my research. I would also like to thank all professors who I have worked with as teaching assistant: Dr. B. Terry Beck, Dr. Jeremy Roberts, Dr. Ronald Brockhoff, Dr. Jack Xin, and Dr. Raj Kumar Pal.

I would like to thank my previous and current colleagues in Dr. Wei's group: Dr. Wei Zhang, Dr. Elnaz Rezaian, Dr. Kun Jia, Dr. Sherif Elsayed, Daniel Colgan, for their thoughtful help and insightful discussions. On the lab, they were reliable research buddies; off the lab, we were also good friends to share some hobbies and have fun together. I also want to thank group alumni Dr. Haotian Gao and Dr. Min Xu, for their timely suggestion when I first started the research, and guidance in job hunting.

The PhD journey could be stressful and frustrating sometimes, and it felt even more so for an international student setting foot on an unfamiliar continent for the first time, and then experiencing an unprecedented pandemic. However, so many friends I have come across literally lit up my life and made this journey never feel lonely. I would like to thank Dr. Hongliang Wang, Fan Liu, Dr. Yijie Gui, Hao Deng, Armin Ezzati, Jon-Michael Grote, Dr. Khalid Abdelaziz, for all the great memories and happy time with them. I also want to thank Dr. Richard "Dick" Akins and Eva Sue Akins deeply, for their constant kind help to make me more and more involved in the community. The accommodation from their family always made me feel home.

This journey definitely would not be possible without the great support from my family at the other side of the ocean: my parents, uncle, aunt, and grandmother who passed away few years ago. Despite we never managed to reunite in person, their consistent love and encouragement always inspired me to move forward adamantly. Last but not least, I would like to thank my lovely wife Jia Tan. You are magic to me. To be able to spend the rest of my life with you makes me even more accomplished than to be a PhD. I can't wait to embark on a new journey in the next phase of my life with you, and wish you all the best in your own pursuit of being Dr. Tan one day.

Dedication

To my parents, my wife Jia, and our cat Piggy.

Chapter 1

Introduction

1.1 Fluid-structure interactions: challenges and inspirations

Fluid-structure interaction (FSI) is ubiquitous in nature[2]. It occurs at a variety of biological scales, from the cells moving in the human body, the blood flowing through the circulation system[3], to the swimming fish in the ocean, and flying insects and birds in the sky. In human society, fluid-structure interaction is still omnipresent, as can be found from the huge turbines of wind farms, jet aircraft, and small-scale unmanned-air vehicles (UAVs). In all those biological phenomena and engineering applications lies rich fluid physics, which has drawn the attention of fluid mechanics community for almost a century. To analyze and understand the physics numerically, a model that can resolve both fluid and structure is needed[4]. However, there are still many existing difficulties in building computational FSI models, including the formulation, boundary conditions, numerical discretization, and the treatment of fluid-structure coupling, as summarized in Bazilevs et al.[5]. The complexity of the FSI problem has raised many questions to be answered as follows. (1) Considering the high computational cost to resolve complicated FSI problem, is it possible to find a simplified model that can capture the essence of the physics in FSI while keep the computational cost relatively low at the same time? (2) FSI often results in a massive parametric space that

may be too huge to thoroughly explore. Therefore, flow optimization with FSI is usually sophisticated, yet highly desirable for the design of aircrafts and marine vehicles. Is there any approach that is able to find the optimal solution in an FSI system while also avoiding the heavily costly computation? If the simplified model mentioned in question (1) exists, can it be applied to the optimization as well? As a matter of fact, these questions have been the major inspiration for the author to propose the current work. In the rest of chapter 1, the numerical model of FSI is introduced in section 1.2, which lays the very foundation of all reduced-order modeling and optimization approaches developed and applied in this thesis. Then, modal decomposition enabled reduced-order modeling methods are introduced in section 1.3. The fundamental formulations and algorithms of conventional adjoint-based flow optimization are introduced in section 1.4. Highlights of this thesis are summarized in section 1.5. An outline of the remaining chapters is presented in section 1.6.

1.2 Direct numerical simulation of FSI

The flow studied in the present work is unsteady, incompressible and viscous, thus is governed by the continuity equation and the incompressible Navier-Stokes equation:

$$\begin{aligned} \nabla \cdot \boldsymbol{u} &= 0 \\ \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} &= -\nabla p + \frac{1}{Re}\nabla^2 \boldsymbol{u}, \quad \text{with} \quad \boldsymbol{u} = \boldsymbol{V} \quad \text{at} \quad \partial\Omega_s. \end{aligned} \tag{1.1}$$

The above equations are non-dimensionalized by a characteristic length L, a reference velocity U^* (the incoming flow velocity in this thesis), and the density ρ and viscosity μ of the fluid. The normalization produces a dimensionless parameter Reynolds number Re:

$$Re = \frac{\rho U^* L}{\mu}. \tag{1.2}$$

With appropriate boundary conditions (BCs), and proper temporal and spatial discretization, the equation 1.1 can be solved numerically.

However, when there is a solid structure immersed in the fluid as shown in figure 1.1, extra efforts might be needed for the spatial discretization and the setup of boundary conditions. A conventional approach is to deploy structured or unstructured meshes conforming to the solid body. The task of mesh generation for a complex solid body can be significantly costly. Depending on the numerical methods used sequentially based on the conformal mesh, a transformation of governing equations from the physical domain to the computational domain might be required, which will increase the computational cost at each grid point. Additionally, an eminent drawback of the conformal mesh appears when the solid structure is moving and deforming in the fluid domain, which means a new mesh needs to be generated at every time step, with the old solution being projected onto the new mesh. These steps will clearly have negative impact on the simplicity, accuracy, and computational cost of the solving process[6]. The aforementioned difficulties have prompted researchers to develop new approaches, among which is the immersed boundary method (IBM) used in the present thesis.

IBM was first developed in a series of marvelous pioneering works of Charles S. Peskin and his colleagues on the simulation of blood flow through heart valves[7–11]. The most distinguishing feature of IBM is that the numerical simulation with FSI can be performed on a Cartesian mesh, regardless of the number or shape of the solid structure, as shown in figure 1.1. When using a Cartesian mesh the solid boundary ($\partial\Omega_s$) may cut through the grid. Then the key step is to impose appropriate boundary conditions on the solid boundary (the black dots in figure 1.1) that is not conforming to the grid, which is usually realized by modifying the governing equations in the vicinity of the solid boundary (the grey area around the solid body in figure 1.1). The major advantage of IBM over the conventional approach is that the mesh generation process is more straightforward and efficient, since only one stationary, non-deforming mesh needs to be built before the simulation, regardless of the type and motion of solid bodies. Therefore, IBM has become an extremely appealing alternative method to simulate flows with FSI, despite that it usually requires more grid points than the conformal mesh for better resolution of the solid boundary[6].

In order to satisfy the solid boundary condition on a Cartesian mesh, IBM introduces a

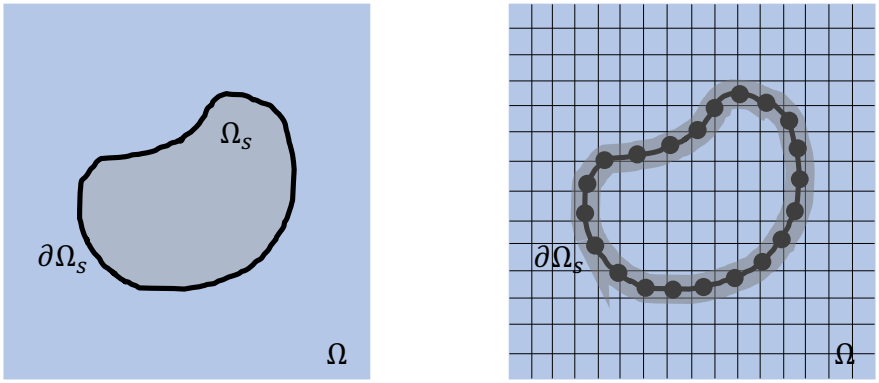

Figure 1.1: *The sketch of a solid structure immersed in the fluid, and the discretization of the computational domain with Cartesian mesh. Ω represents the fluid domain, Ω_s represents the solid structure, and $\partial\Omega_s$ is the solid boundary.*

body-force field $\boldsymbol{f}$ such that the desired velocity distribution $\boldsymbol{V}$ on the solid boundary can be assigned properly, accordingly the Navier-Stokes equation is modified as:

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} = -\nabla p + \frac{1}{Re}\nabla^2\boldsymbol{u} + \boldsymbol{f}. \tag{1.3}$$

Generally there are two specific categories of forcing-type IBM: the continuous forcing approach, where the forcing is incorporated in to continuous equations, and the discrete forcing approach, where the forcing is added after the equations are discretized[6]. The discrete forcing approach is highly dependent on the spatial discretization, but it has the capability to avoid the "diffuse boundary" problem caused by the continuous forcing approach, which can ultimately lead to a sharp fluid-structure interface with high resolution. Therefore, this approach has gained higher popularity recently, with different methods in this category being developed in Mohd-Yusof[12]; Fadlun et al.[13]; Mittal et al.[14]; Taira and Colonius[15]; Mittal et al.[16]; Zhao et al.[17]. The discrete IBM was also applied in the previous work of the author's group[18–21], as well as in this thesis.

Without loss of the generality, a time-discretized modified equation 1.3 with a simple first-order time-forward scheme is:

$$\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^n}{\Delta t} = [-\nabla p - (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \frac{1}{Re}\nabla^2\boldsymbol{u}]^n + \boldsymbol{f}, \tag{1.4}$$

where the superscript n denotes the values at time step t_n. Implementing the discrete forcing approach with sharp interface as developed in Mittal et al. [16], the forcing term f can be expressed nominally as:

$$\boldsymbol{f} = \begin{cases} [(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} - \frac{1}{Re}\nabla^2\boldsymbol{u}]^n + \frac{1}{\Delta t}(\boldsymbol{V} - \boldsymbol{u}^n), & \text{in} \quad \Omega_s \\ 0, & \text{otherwise}, \end{cases} \tag{1.5}$$

where f only takes effect in the solid occupied domain Ω_s. All following work on the reduced-order modeling and the flow optimization will be developed based on modified governing equations 1.4 and 1.5.

1.3 Reduced-order modeling

Even a seemingly simple flow configuration under modest conditions, such as the supercritical flow past a fixed cylinder [22], may produce rich and complex fluid dynamics with a wide range of spatial and temporal features, not to mention the perplexing fluid physics of flows with more sophisticated FSI. Fortunately for many flows, there are often physically more important features that stand out and can represent the core characteristics. Therefore, extracting these physically important features would be a natural first step in the analysis of complex flows. Typically these steps will start with a modal decomposition of either experimental or numerical data of the fluid flows. Then with the modes obtained through the modal decomposition, a reduced-order model (ROM) can be constructed usually by projection method to obtain the lower-order dynamics which can represent the essential physics of the original flow fields. An analogy of the modal decomposition may be made by image compression. Figure 1.2 shows different images of the author's cat. The original photo was taken with the highest resolution which consists of 1960×1878 pixels. Each of other three pictures was compressed, and can be regarded as the reconstruction of "lower-order modes" with smaller size. It can be observed clearly that excessive compression may cause some of them to lose most of the information from the original picture (more blurry).

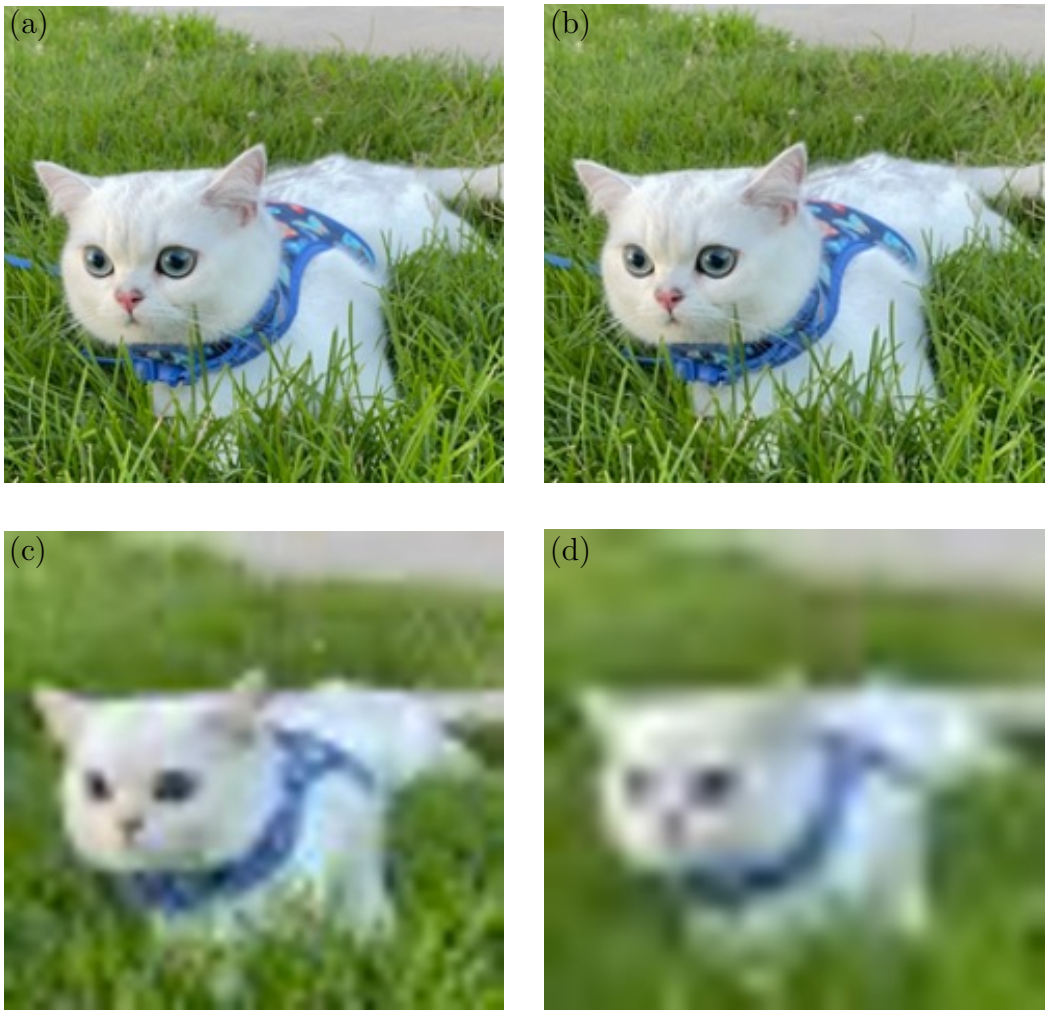

Figure 1.2: *The pictures of cat "Piggy" before and after the image compression. (a)* 1960×1878 *pixels (the original); (b)* 11.67% *size, 100 modes; (c)* 2.33% *size, 20 modes; (d)* 1.16% *size, 10 modes.*

Therefore, some criterion is needed to determine how many lower-order modes are necessary to keep the reconstruction of the original data accurate enough, which will be addressed later.

1.3.1 Modal decomposition: basics of proper orthogonal decomposition

Modal decomposition based ROM methods have been studied for decades, including proper orthogonal decomposition (POD)[23;24], balanced proper orthogonal decomposition (balanced

POD)[25;26], spectral proper orthogonal decomposition (spectral POD)[27;28], dynamic mode decomposition (DMD)[29], and so on. These approaches are all data-driven, which means the construction of modes required the data fed by experimental measurements or numerical simulations[30]. Among all modal decomposition approaches, POD is arguably the most popular one and the bedrock of modal decomposition techniques to extract coherent structures from the flow data. Since it was invented back in 1960s[31], POD has been extensively used in fluid mechanics community. The major success of POD is that it is able to provide physically interpretable spatial-temporal decompositions of flow data[32]. The details of POD method will be reviewed as follow.

Considering the governing partial differential equations (PDEs) 1.1, the velocity vector field $\boldsymbol{u}(\boldsymbol{x}, t)$ would be the solution, though the highly nonlinear PDEs themselves can hardly be solved analytically. But some ideas can still be borrowed from one common technique to analytically solve linear PDEs, which is the separation of variables. When using this method, a specific form of solution is presumed where space and time are independent:

$$\boldsymbol{u}(\boldsymbol{x}, t) = a(t)\boldsymbol{\phi}(\boldsymbol{x}). \tag{1.6}$$

Although the strong nonlinearity of equations 1.1 makes it impossible to obtain a trivial solution as equation 1.6, a series of basis functions, which hierarchically represent different dimensions of the complex fluid flow system, may still be able to construct the solution and split the space and time:

$$\boldsymbol{u}(\boldsymbol{x}, t) = \sum_{i=0}^{\infty} a_i(t)\boldsymbol{\phi}_i(\boldsymbol{x}), \tag{1.7}$$

where $a_i(t)$ is the time coefficients representing the temporal dimension, and $\boldsymbol{\phi}_i(\boldsymbol{x})$ are the modes representing the spatial dimension. Note that the upper limit of the superposition 1.7 is infinity, corresponding to the infinite dimension of a fluid flow system. Now, the primary task is to determine the coefficient $a_i(t)$ and the mode $\boldsymbol{\phi}_i(\boldsymbol{x})$ respectively. Figure 1.3 briefly summarizes the procedure of modal decomposition with POD. When applying POD on $\boldsymbol{u}(\boldsymbol{x}, t)$, $\boldsymbol{\phi}_i(\boldsymbol{x})$ are POD modes accordingly. The modes can be regarded as a set

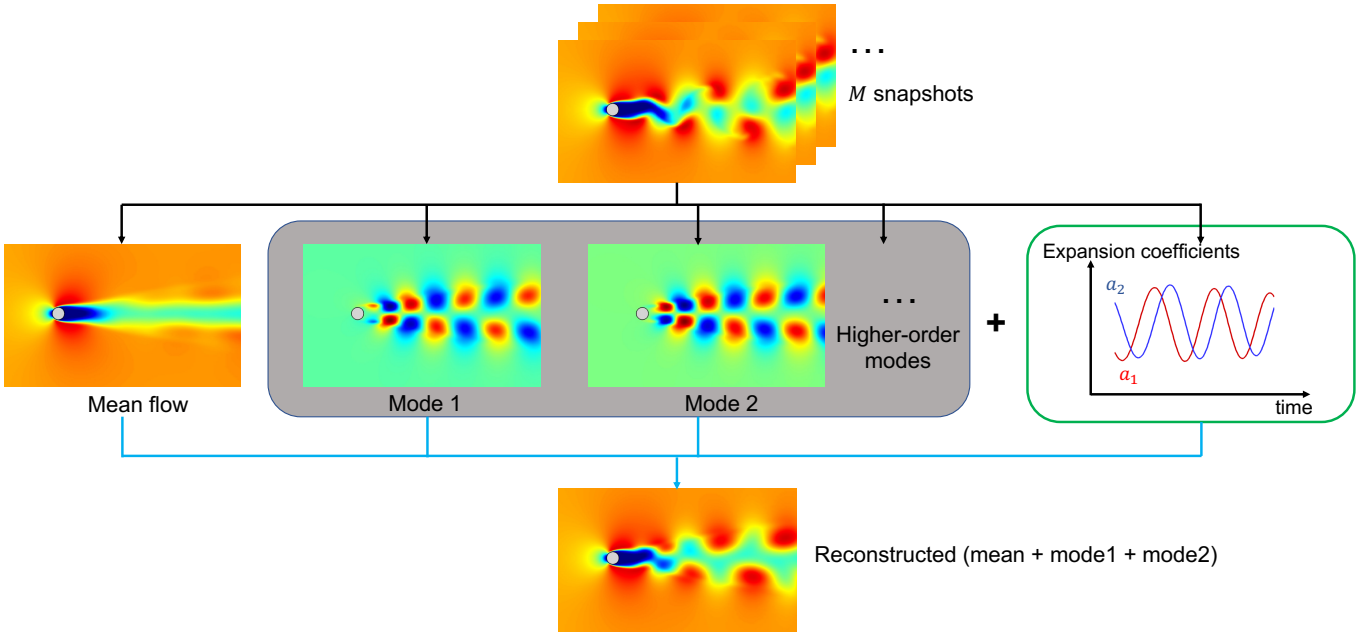

Figure 1.3: *POD of a 2D flow past a fixed cylinder under* $Re = 200$. *The illustrations shows the nonlinear flow being captured by the mean flow and the first two POD modes.*

of basis functions that represent flow field $\boldsymbol{u}(\boldsymbol{x}, t)$ optimally. First, for flow fields at all time moments, the temporally averaged flow field $\bar{\boldsymbol{u}}(\boldsymbol{x})$ should be subtracted from each $\boldsymbol{u}(\boldsymbol{x}, t)$, which generates a collection of finite-dimensional data vectors $\boldsymbol{u}'(t)$ as the fluctuation of unsteady flow fields:

$$\boldsymbol{u}'(t) = \boldsymbol{u}(\boldsymbol{x}, t) - \bar{\boldsymbol{u}}(\boldsymbol{x}) \in \mathbb{R}^N, \quad t = t_1, t_2, ..., t_M, \tag{1.8}$$

where N and M are the dimension of space and time respectively. It was proved that seeking optimal basis functions $\boldsymbol{\phi}_i(\boldsymbol{x})$ can be converted to solving an eigenvalue problem of the covariance matrix $\boldsymbol{R}$[23;24], which is constructed as:

$$\boldsymbol{R} = \sum_{m=1}^{M} \boldsymbol{u}'(t_m)\boldsymbol{u}'^T(t_m) = \boldsymbol{X}\boldsymbol{X}^T \in \mathbb{R}^{N\times N}, \tag{1.9}$$

where the matrix $\boldsymbol{X}$ is the assembly of M snapshots of fluctuation data $\boldsymbol{u}'_i$:

$$\boldsymbol{X} = [\boldsymbol{u}'_1 \quad \boldsymbol{u}'_2 \quad ... \quad \boldsymbol{u}'_M] \in \mathbb{R}^{N\times M}. \tag{1.10}$$

Note that N is usually a massive number for fluid flows, which is equal to the spatial degrees of freedom of flow data (number of grid points) times the number of flow variables (2 for 2D flows, for example). The eigenvalue problem is formulated as:

$$\boldsymbol{R}\boldsymbol{\phi}_i = \lambda_i \boldsymbol{\phi}_i, \quad \boldsymbol{\phi}_i \in \mathbb{R}^N, \quad \lambda_1 \geq ... \geq \lambda_N \geq 0. \tag{1.11}$$

By solving the above equation, the eigenvectors $\boldsymbol{\phi}_i(\boldsymbol{x})$ can be found which are also POD modes. One important feature of POD modes is that they are orthogonal, which means the inner product (denoted by bracket operator $\langle \cdot \rangle$) between the modes satisfies:

$$\langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_k \rangle = \int_\Omega \boldsymbol{\phi}_i \cdot \boldsymbol{\phi}_k d\Omega = \delta_{ik}, \quad i, k = 1, ..., N, \tag{1.12}$$

where Ω is the domain of inner product and δ is the Kronecker symbol. The eigenvalues λ_i show how well the original flow field is captured by each POD mode in terms of the kinetic energy. In other words, the larger the λ_i is, the more important role the corresponding $\boldsymbol{\phi}_i$ plays in capturing the kinetic energy of the flow field. As a result, λ_i can be leveraged to determine the number of POD modes needed to represent the original flow field. An example of the eigenvalues for the flow past a fixed cylinder under $Re = 200$ is shown in figure 1.4. It can be found that the first two λ_i are much larger than the rest, and the first 10 POD modes have already captured nearly 100% of the total kinetic energy. Therefore, practically the leading r out of N modes (in decreasing order with respect to the value of λ_i) are retained to produce a lower-order representation of the flow such that:

$$\sum_{i=1}^{r} / \sum_{i=1}^{N} \approx 1. \tag{1.13}$$

The modes with order higher than r are truncated, considering they are negligible from the energy perspective. With these modes being selected, the flow field can be represented with

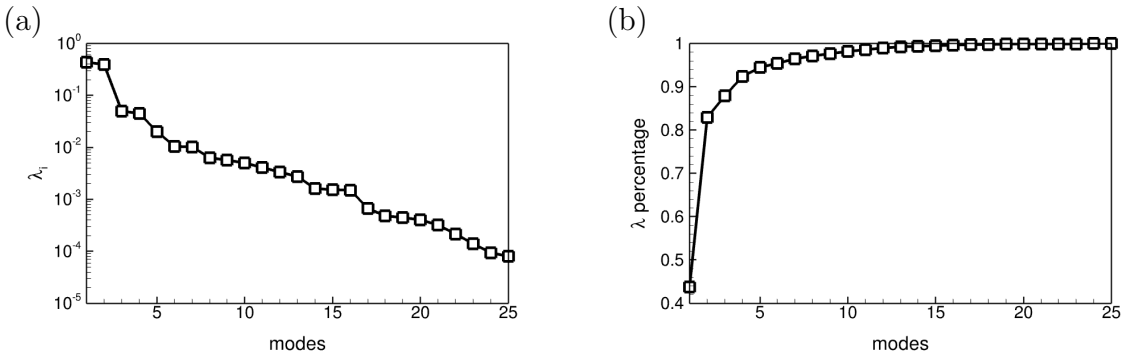

Figure 1.4: *An example of eigenvalues λ_i obtained by solving POD of 2D flow past a fixed cylinder under $Re = 200$. (a): The first 25 eigenvalues; (b): the cumulative percentage of kinetic energy captured by the first 25 POD modes.*

finite series of r modes in an optimal manner:

$$\boldsymbol{u}(\boldsymbol{x},t) - \bar{\boldsymbol{u}}(\boldsymbol{x}) = \boldsymbol{u}'(t) \approx \sum_{i=1}^{r} a_i(t)\boldsymbol{\phi}_i(\boldsymbol{x}), \tag{1.14}$$

where the temporal expansion coefficients $a_i(t)$ can be obtained by projecting the fluctuation field $\boldsymbol{u}'(t)$ onto POD modes:

$$a_i(t) = \langle \boldsymbol{u}'(t), \boldsymbol{\phi}_i(\boldsymbol{x}) \rangle. \tag{1.15}$$

The POD method introduced above is called classical POD. A major drawback of it is that the dimension of the covariance matrix $\boldsymbol{R}$ is usually so huge ($N \times N$) that the eigenvalue problem (equation 1.11) is impossible to solve practically. Instead of tackling $\boldsymbol{R}$ directly, the method of snapshots has been developed to solve the eigenvalue problem of an alternative matrix with a much smaller size[23]:

$$(\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{\psi}_i = \lambda_i\boldsymbol{\psi}_i, \quad \boldsymbol{\psi}_i \in \mathbb{R}^M, \tag{1.16}$$

where the matrix $\boldsymbol{X}^T\boldsymbol{X}$ has a dimension of $M \times M$. Note that M is the number of discrete time levels, thus is much smaller than N. Equation 1.16 is solvable and it still shares the same nonzero eigenvalues λ_i with the original equation 1.11. Moreover, the POD modes can

be recovered by the new eigenvectors $\boldsymbol{\psi}_i$ as well:

$$\boldsymbol{\phi}_i = \boldsymbol{X}\boldsymbol{\psi}_i \frac{1}{\sqrt{\lambda_i}}, \quad j = 1, 2, ..., M. \tag{1.17}$$

The method of snapshots is arguably the most widely used POD method in fluid mechanics due to the drastic reduction in computational time and memory resources, so it was also used in the present work. Note that the above derivation is based on the snapshots generated on a domain discretized by uniform mesh. Practically, when POD is performed on a domain with non-uniform mesh, the matrix $\boldsymbol{X}^T\boldsymbol{X}$ in the eigenvalue problem 1.16 should be modified as $\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}$, where $\boldsymbol{W}_{n\times n}$ is the weight matrix that holds the spatial weight (i.e. the size of each mesh cell).

It is worth to know that POD modes can also be computed in an equally swift manner by singular value decomposition (SVD). Using the same matrices $\boldsymbol{X}$ and $\boldsymbol{W}$ as aforementioned, a new matrix $\boldsymbol{Y}$ can be generated:

$$\boldsymbol{Y} = \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{X}. \tag{1.18}$$

Then a thin SVD can be performed as:

$$\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T. \tag{1.19}$$

Accordingly, a correlation matrix can be obtained by:

$$\boldsymbol{Y}^T\boldsymbol{Y} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \boldsymbol{V}\boldsymbol{\Sigma}^2\boldsymbol{V}^T. \tag{1.20}$$

It is easy to find that:

$$\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{V} = \boldsymbol{V}\boldsymbol{\Sigma}^2, \tag{1.21}$$

which means the columns of $\boldsymbol{V}$ are the eigenvectors of matrix $\boldsymbol{Y}^T\boldsymbol{Y}$, which is equivalent to

the matrix $\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}$ used in the method of snapshots. So the POD modes $\boldsymbol{\phi}$ are solved by:

$$\begin{aligned}\boldsymbol{\phi} &= (\boldsymbol{W}^{-\frac{1}{2}}\boldsymbol{Y})\boldsymbol{V}\boldsymbol{\Lambda}^{-\frac{1}{2}} \\ &= \boldsymbol{W}^{-\frac{1}{2}}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\Lambda}^{-\frac{1}{2}} \\ &= \boldsymbol{W}^{-\frac{1}{2}}\boldsymbol{U}.\end{aligned} \tag{1.22}$$

More details of SVD-based technique can be found in the book by Brunton and Kutz[32].

At the end of this section the pros and cons of POD are briefly summarized. As demonstrated earlier, the most attractive property of POD is the orthogonality of its modes, which allows for the construction of reduced-order model with minimal number of modes. In addition, POD modes are simple to compute, of which the method of snapshots is able to deal with high-dimensional fluid flow datasets. However, POD ignores the correlations that are higher than the second order. The importance of the modes is assessed by the order of kinetic energy they capture, instead of by the order of dynamical importance, but this drawback can be addressed and improved by balanced POD and DMD method. Furthermore, for some highly turbulent flows, the number of POD modes required to satisfy equation 1.13 is very large, which overshadows the goal to significantly reduce the order of the original flow system. Despite the above limitations, the flows studied in the present work are mostly in the laminar flow regime with modest Reynolds numbers, thus the POD method still remains robust here.

1.3.2 Galerkin projection based ROM

Through the modal decomposition by POD, a subspace consisting of r leading POD modes is constructed, and the lower-order dynamics of the original snapshots $\boldsymbol{u}(\boldsymbol{x}, t)$ is also obtained as $a_i(t)$. However, so far $a_i(t)$ contains no dynamics beyond the range of $\boldsymbol{u}(\boldsymbol{x}, t)$, thus is still infeasible to predict the flow dynamics of future time moments. Fortunately, $a_i(t)$ at arbitrary time moments can be achieved by solving another equation built upon the original Navier-Stokes equation, the velocity decomposition (equation 1.14), and the Galerkin-projection.

First, the velocity expanded with the first r POD modes

$$\boldsymbol{u}(\boldsymbol{x},t) \approx \bar{\boldsymbol{u}}(\boldsymbol{x}) + \sum_{i=1}^{r} a_i(t)\boldsymbol{\phi}_i(\boldsymbol{x}) \tag{1.23}$$

is substituted in equation 1.1. Then the Galerkin projection is performed with the same definition of inner product as equation 1.12, where each term in equation 1.1 is projected onto the subspace, i.e. POD modes $\boldsymbol{\phi}_i$. Due to the orthogonality of POD modes, a group of ordinary differential equations (ODEs) about $a_i(t)$ can be derived:

$$\frac{da_i}{dt} = \sum_{j=0}^{r} L_{ij}a_j + \sum_{j=0}^{r}\sum_{k=0}^{r} Q_{ijk}a_j a_k, \quad i = 1, 2, ..., r, \tag{1.24}$$

where

$$\begin{aligned} L_{ij} &= \frac{1}{Re}\langle \nabla^2 \boldsymbol{\phi}_j, \boldsymbol{\phi}_i \rangle, \\ Q_{ijk} &= -\langle \nabla \cdot (\boldsymbol{\phi}_j \boldsymbol{\phi}_k), \boldsymbol{\phi}_i \rangle. \end{aligned} \tag{1.25}$$

For the purpose of brevity, the mean flow $\bar{\boldsymbol{u}}(\boldsymbol{x})$ is absorbed in the velocity expansion (equation 1.24) and $\boldsymbol{\phi}_0 = \bar{\boldsymbol{u}}(\boldsymbol{x})$ hereafter, with $a_0(t) = 1$. The ODEs in equation 1.24 are the general POD-Galerkin projection based reduced-order model (ROM) of the original flow system. Note that the pressure term in Navier-Stokes equation is not accounted for ROM. As discussed by Noack et al.[33], pressure term can be neglected when appropriate boundary conditions are proposed, which is the situation of the present work.

The POD-Galerkin ROM has been proven successful over decades. However, there are still limitations even for a configuration as simple as the flow past a fixed 2D cylinder. Some major limitations are well summarized in Loiseau et al.[34], including stability issue, difficulties in transient flows, and so on. Another big issue of the existing ROM methods is the challenge brought by moving or deforming solid boundaries immersed in the flow domain, which intrigues the present work on the global POD-Galerkin ROM approach. The details of this new ROM approach will be presented in chapter 2.

1.4 Flow optimization and adjoint approach

As demonstrated by Gunzburger[35], flow control and optimization basically has been a long-lasting,s ancient practice of mankind. For instance, the construction and application of any dam, canal, irrigation ditch, etc., all are efforts attempting to control the fluid flow to benefit the human life. In the fluid mechanics community, flow optimization has also been exercised and studied for more than a century. Back in early 1900s, Prandtl, a pioneer of modern fluid mechanics, experimentally studied boundary layer control on the suction side of a cylinder in incoming laminar flow, to reduce the drag by delaying flow separation[36]. Generally speaking, a well-posed flow optimization problem has three main ingredients:

1. the objective to achieve for the problem, which can be drag reduction, lift/thrust enhancement, delay of turbulent transition, etc.;

2. the control parameters, which can be tuned to reach the desirable objectives;

3. the constraints, which here are the physical laws and governing equations to describe the fluid dynamics properly, and ultimately set up the range for control parameters.

Based on the physical understanding of a given flow, the control parameters that might possibly have impact on the objective can be determined. Then these control parameters can form a parametric space. In principle, the optimal objective can be found as long as the parametric space is completely explored through either experiments or numerical simulations. However, for a complicated flow such as flow with FSI studied in the present work, the parametric space is often too massive to thoroughly explore, thus a brute-force parametric study will be impossible. The dilemma of huge parametric space has stimulated the development of flow optimization methods based on mathematics. Among all these methods, gradient-based methods have been widely used, due to their high efficiency and accuracy to find the local minimum on a convex manifold in even a huge parametric space, thus it is also an ideal choice for optimizing complicated FSI problems. Tuncer and Kaya[37] used gradient-based method to optimize the thrust and propulsive efficiency of the 2D flow over

a plunging and pitching wing. The gradient was computed by perturbing the control parameters and computing the unsteady flow field numerically, which resulted in the excessive computational cost of solving the Navier-Stokes equations as many times as the number of control parameters. As a different route, the adjoint-based approach allows for an inexpensive computation on the gradient, regardless of the number of control parameters. Figure 1.5 illustrates the relation between the computational cost and the number of control parameters for different methods. The computational cost of parametric study usually grows exponentially as the number of control parameters increases. On the contrary, when using adjoint-based approach, the computational cost barely grows even with a huge parametric space. This is one of the biggest advantages over other methods, specifically for cases where the inputs (control parameters) outnumber the outputs (objectives). In fact, for the most of adjoint-based applications, only one objective is controlled. The formulations of the adjoint-based approach will be reviewed in the following subsection to demonstrate how it is able to obtain the gradient quickly during the optimization.

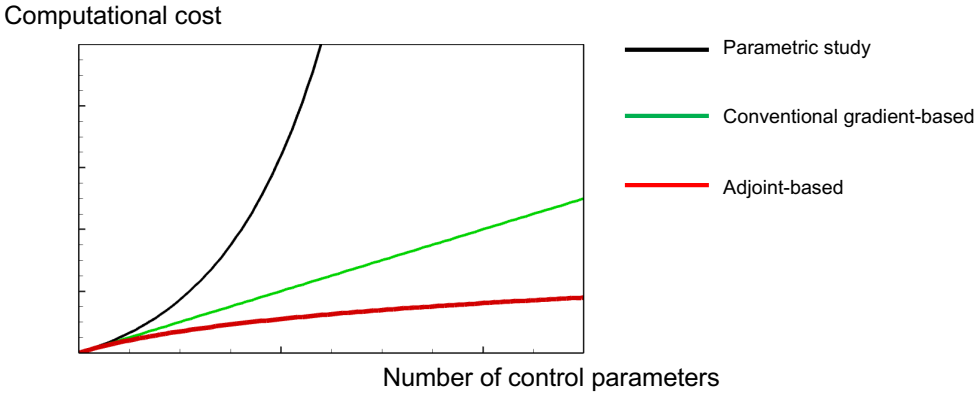

Figure 1.5: *An illustration of computational cost versus the number of control parameters for different methods. The cost is an estimation.*

1.4.1 Generic adjoint formulations

Typically, the adjoint-based approach is classified into two categories: continuous adjoint[38;39] and discrete adjoint[40;41], where the former formulates all adjoint equations for the optimization and then discretize them numerically, while the latter spatially discretizes the governing

equations before determining the adjoint counterpart[42]. The continuous adjoint approach was chosen in the present work, due to the fact that it has better clarity in the adjoint equation with terms actually showing physical meaning. Therefore, the term "adjoint-based approach" appearing hereafter only refers to the continuous adjoint approach.

The adjoint-based approach was first derived for the optimization of incompressible turbulent channel flows where the control was introduced by adding an unsteady wall-normal velocity γ at the boundary Γ[38;42]. Three vectors containing flow variables are required to build the adjoint formulations: the flow state variables $\boldsymbol{q} = [p \quad \boldsymbol{u}]^T$, the flow perturbation variables $\boldsymbol{q}' = [p' \quad \boldsymbol{u}']^T$, and the adjoint variables $\boldsymbol{q}^* = [p^* \quad \boldsymbol{u}^*]^T$. Considering the boundary condition and for the brevity of the expression, the Navier-Stokes equation 1.1 is rewritten as:

$$\begin{aligned} \mathcal{N}(\boldsymbol{q}) &= 0, \\ \boldsymbol{u} &= -\gamma \boldsymbol{n} \quad \text{on} \quad \Gamma, \end{aligned} \tag{1.26}$$

where $\mathcal{N}(\boldsymbol{q})$ is the nonlinear Navier-Stokes operator:

$$\mathcal{N}(\boldsymbol{q}) = \begin{bmatrix} \frac{\partial u_j}{\partial x_j} \\ \\ \frac{\partial u_i}{\partial t} + u_j(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) - \frac{1}{Re}\frac{\partial u_i}{\partial x_j^2} + \frac{\partial p}{\partial x_i} \end{bmatrix}, \tag{1.27}$$

An analytical objective function $\mathcal{J}$ of physical interest is needed for the optimization. For the purpose of demonstration, the turbulent kinetic energy in the flow domain Ω is optimized here:

$$\mathcal{J} = \frac{1}{2}\int_0^T \int_\Omega |\boldsymbol{u}|^2 d\Omega dt. \tag{1.28}$$

With a well-defined objective $\mathcal{J}$ and control γ, the essential procedure of optimization is to determine the sensitivity of $\mathcal{J}$ to small perturbations of γ. This sensitivity can be

quantified by using a Fréchet derivative[43] of $\mathcal{J}$ such that:

$$\begin{aligned}\mathcal{J}' &= \lim_{\epsilon \to 0} \frac{\mathcal{J}(\gamma + \epsilon\gamma') - \mathcal{J}(\gamma)}{\epsilon}, \\ &= \int_0^T \int_\Omega g(\gamma)\gamma' d\Omega dt.\end{aligned} \tag{1.29}$$

By computing the nominal first-order derivation of $\mathcal{J}$ with respect to an arbitrary control γ as above, the sensitivity is now evaluated by the gradient $g(\gamma)$. With the same differentiation rule and equation 1.28, the perturbation of the objective is:

$$\mathcal{J}' = \int_0^T \int_\Omega \boldsymbol{u} \cdot \boldsymbol{u}' d\Omega dt. \tag{1.30}$$

As the control γ is perturbed by a small fraction, it is physically intuitive that the flow state variables $\boldsymbol{q}$ should be perturbed by an amount of $\boldsymbol{q}'$ as well. The perturbed Navier-Stokes equation can be derived based on $\boldsymbol{q}'$ with the same fashion of Fréchet derivative:

$$\begin{aligned}\mathcal{N}'(\boldsymbol{q}') &= 0, \\ \boldsymbol{u}' &= -\gamma' \boldsymbol{n} \quad \text{on} \quad \Gamma,\end{aligned} \tag{1.31}$$

where $\mathcal{N}'(\boldsymbol{q}')$ is the linearized Navier-Stokes operator:

$$\mathcal{N}'(\boldsymbol{q}') = \begin{bmatrix} \frac{\partial u_j'}{\partial x_j} \\ \\ \frac{\partial u_i'}{\partial t} + u_j'(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i}) - \frac{1}{Re}\frac{\partial u_i'}{\partial x_j^2} + \frac{\partial p'}{\partial x_i} \end{bmatrix}, \tag{1.32}$$

Using the perturbed Navier-Stokes equations, the adjoint variables $\boldsymbol{q}^*$ are introduced as Lagrangian multipliers to impose the flow equations 1.26, as the whole problem is regarded as a constrained optimization problem subject to the governing equation. Then the perturbation of enhanced objective function is rewritten as:

$$\mathcal{J}' = \int_0^T \int_\Omega \boldsymbol{u} \cdot \boldsymbol{u}' d\Omega dt + \int_0^T \int_\Omega \boldsymbol{q}^* \cdot \mathcal{N}'(\boldsymbol{q}') d\Omega dt. \tag{1.33}$$

Using the adjoint identity

$$\langle \mathcal{N}'(\boldsymbol{q}'), \boldsymbol{q}^* \rangle = \langle \boldsymbol{q}', \mathcal{N}^*(\boldsymbol{q}^*)\boldsymbol{q}^* \rangle, \tag{1.34}$$

the adjoint equation can be derived by performing integration by parts on the second term of equation 1.33:

$$\begin{aligned} \mathcal{N}^*(\boldsymbol{q}^*) &= \mathcal{F}^*, \\ \boldsymbol{u}^* &= 0 \quad \text{on} \quad \Gamma, \end{aligned} \tag{1.35}$$

where

$$\mathcal{N}^*(\boldsymbol{q}^*) = \begin{bmatrix} & -\frac{\partial u_j^*}{\partial x_j} & \\ & & \\ -\frac{\partial u_i^*}{\partial t} - u_j^*(\frac{\partial u_i^*}{\partial x_j} + \frac{\partial u_j^*}{\partial x_i}) - \frac{1}{Re}\frac{\partial u_i^*}{\partial x_j^2} - \frac{\partial p^*}{\partial x_i} \end{bmatrix}, \tag{1.36}$$

and $\mathcal{F}^* = [0 \quad \boldsymbol{u}]^T$. Note that the adjoint equation needs to be solved backward in time, due to the sign of the time derivative and viscous term. Consequently equation 1.33 is converted as:

$$\mathcal{J}' = B + \int_0^T \int_\Omega \boldsymbol{q}' \cdot [\mathcal{N}^*(\boldsymbol{q}^*) - \mathcal{F}^*] d\Omega dt, \tag{1.37}$$

where B comprises all boundary-related terms:

$$B = \int_\Omega (u_j^* u_j')|_0^T d\Omega + \int_0^T \int_\Gamma n_j [u_i^*(u_j u_i' + u_j' u_i) + p^* u_j' - \frac{1}{Re}(u_i^* \frac{\partial u_i'}{\partial x_j} - u_i' \frac{\partial u_i^*}{\partial x_j}) + u_j^* p'] d\Gamma dt. \tag{1.38}$$

With carefully imposed boundary conditions for $\boldsymbol{q}^*$ and by comparing B against equation 1.29, the gradient $g(\gamma)$ is obtained:

$$g(\gamma) = p^*. \tag{1.39}$$

This simple expression of $g(\gamma)$ indicates that for this particular example, the adjoint pressure field p^* is the measure of the sensitivity of the turbulent kinetic energy which is controlled by the wall-normal blowing or suction. In theory, regardless of the number of control parameters in $\boldsymbol{\gamma}$, similar steps can be taken to derive the adjoint equation. Once adjoint equation is solved, the gradient can be computed momentarily with adjoint variables.

1.4.2 Control updates: conjugate gradient method

Algorithm 1: The generic algorithm of adjoint-based flow optimization in a main iteration k.

Iteration k begins

1. Solve governing equations 1.26 with $\gamma^k \rightarrow \boldsymbol{q}, \mathcal{J}^k$
2. Derive adjoint equations 1.35, solved backward in time $\rightarrow \boldsymbol{q}^*$
3. Compute gradient g^k
4. Solve equation 1.42 $\rightarrow \beta^k$
5. Solve equation 1.41 $\rightarrow d^k$

Line-search sub-iteration begins

6. Calculate temporary $\gamma_t = \gamma^k + \alpha d^k$, then solve equation 1.26 with this γ_t

if α *is equal to* $argmin_\alpha \mathcal{J}(\gamma^t)$ **then**
  $\alpha^k = \alpha$;
else
  Repeat step 6.
end

Line-search sub-iteration ends

7. Update control $\gamma^{k+1} = \gamma^k + \alpha^k d^k$

if $|\mathcal{J}^{k+1} - \mathcal{J}^k| < \epsilon$ *(ϵ is a very small number)* **then**
  Iteration k ends, optimization is converged
else
  Repeat step 1 to 7.
end

With the gradient g achieved in the prior section, the *Polak-Ribiere* variant of the conjugate gradient method[44] was used in the present work to update the control parameters iteratively. The control parameter γ can be updated by:

$$\gamma^{k+1} = \gamma^k + \alpha^k d^k, \tag{1.40}$$

where the superscript indicates the times of main iteration, d is the fastest descent direction, α is the optimal step size obtained from the line-search computation such that $\alpha = \text{argmin}_\alpha \mathcal{J}(\gamma)$. Initially d^1 is set to be $-g^1$, and computed thereafter by

$$d^k = -g^k + \beta^k d^{k-1}. \tag{1.41}$$

β^k is determined by

$$\beta^k = \frac{(g^k - g^{k-1}) \cdot g^k}{g^{k-1} \cdot g^{k-1}}, \tag{1.42}$$

where gradient g^k is already obtained by solving corresponding adjoint equations. The generic algorithm of adjoint-based flow optimization is summarized in algorithm 1.

Note that the adjoint-based approach reviewed in this section is derived under the framework of fixed flow domain. However, when extending it to a domain with moving/deforming solid boundaries, some problems will emerge due to ill-defined integration over a morphing domain. Therefore new techniques are needed to address these difficulties. The improved adjoint-based approach will be introduced in chapter 3.

1.5 Highlights of this work

In order to answer the questions asked in the end of section 1.1, the present work is focused on:

- developing a ROM that is able to represent the essential physics of a FSI system with moving solid boundaries;
- developing a fast optimization method based on adjoint-ROM equations for a FSI system with moving solid boundaries;
- applying adjoint-based approach to the optimization of complicated FSI problems: including gust mitigation with heaving-pitching airfoil, and the hydrofoil schooling problem

Highlights and contributions of the present work can be summarized as:

1. A global POD-Galerkin ROM was developed for flows with moving solid boundaries, which can work for both numerical and experimental dataset. The ROM showed adequate accuracy in the reconstruction of flow fields and the prediction of key aerodynamic features, while the computational cost remained very low.

2. Based on the new global POD-Galerkin ROM, an adjoint enabled fast flow optimization method was developed for the first time for FSI with moving solid boundaries. Different strategies to choose proper ROM during the optimization were studied. The adjoint-ROM approach was then applied to the optimization on 2D flows over moving cylinder and heaving-pitching airfoil. The results demonstrated the effectiveness of the approach, and the computational cost was reduced drastically for all cases.

3. Gust mitigation with a 2D/3D heaving-pitching wing was studied using an adjoint-based approach for the first time. Effective controls on the wing motion were achieved by optimization to not only recover the original mean aerodynamic force on the wing, but also alleviate the unsteadiness of the force to some extent. The full-order model (FOM) based adjoint approach was able to handle relatively strong gusts with very good recovery of the lift force, while the adjoint-ROM approach was able to mitigate weak gusts, with lower controllability but within extremely short computational time.

4. Adjoint-based approach was extended to optimize the FSI system with multiple moving solid bodies for the first time. The drag force of 2D rigid and flexible hydrofoils in different arrangements of formation was optimized by controlling the motion and formation of trailing hydrofoils. Significant drag reduction and drag-to-thrust conversion were achieved by adjoint-based optimization. The analysis on vortex-structure interactions of optimal schooling shed some light upon the hydrodynamic mechanisms of fish schooling behavior.

1.6 The outline of the remainder

The rest of this thesis is arranged in the following manner: Chapter 2 introduces the development of a global POD-Galerkin projection based ROM, and its application on numerical as well as experimental datasets; chapter 3 reviews non-cylindrical calculus enabled adjoint approach based on FOM equations, and then introduces the development of an adjoint-ROM based fast flow optimization approach, and its application on 2D flows over moving cylinder

and heaving-pitching airfoil; chapter 4 implements both FOM-based adjoint approach and adjoint-ROM approach to solve the gust mitigation problem for 2D and 3D heaving-pitching wings; chapter 5 leverages FOM-based approach on the hydrodynamic performance optimization of multiple rigid and flexible hydrofoils; the conclusions are drawn in chapter 6, with the insights into future work being discussed.

Chapter 2

Global POD-Galerkin ROM approach for flows with moving solid boundaries

2.1 Introduction

High-fidelity large-scale numerical simulation plays a critical role in research to understand and analyze fluid dynamics. This is particularly true in fluid-solid systems with complex motion and interactions, such as the classical dynamic stall problem which has attracted the attention of the fluid dynamics community for decades[45–47]. However, the direct application of high-fidelity computation is hindered by its computational cost in situations where computational speed is critical. Thus, a reduced-order model (ROM) is often needed to provide a feasible low-fidelity solution with much lower computational cost and lower-but-sufficient computational accuracy[48;49].

Just as reviewed in chapter 1, proper orthogonal decomposition (POD)-Galerkin projection[24;50]has been one of the most popular approach in the category, and has shown success in many research areas including incompressible free shear layers[51;52], the flow past fixed cylinder[53–56], compressible flows[57;58], aeroacoustics in cavity flows[59;60], etc. A common factor enabling the success of these ROMs is that the flows considered are defined in a fixed fluid domain with either fixed solid boundaries or no-solid boundary (i.e. infinite domain),

which is required by the original derivation of the approach. Recently, more efforts have been made to extend the application of POD-Galerkin projection to a domain with moving solid boundaries or structures[61–64]. To model the flow past a heaving airfoil, Lewin and Haj-Hariri[65] defined the POD modes in the coordinate fixed on the airfoil to avoid the domain problems caused by the motion and allow for an almost direct application of traditional POD-Galerkin projection. The only modification required in their approach was to change far-field boundary conditions and add extra non-inertial terms to the governing equations before projection, which was based on the new body-fixed coordinate in a periodic heaving motion. Aside from the inconvenience caused by non-inertial coordinates, the approach was only applicable to non-deforming bodies with prescribed simple motions. For flow past an oscillatory cylinder, Noack et al.[66] and Tadmor and Noack[67] introduced an additional actuation mode for moving-boundary-imposed unsteadiness, as well as a cylinder-fixed coordinate, to effectively represent the local motion while still using the POD modes constructed on a fixed domain. The separate actuation mode allowed for an easy formulation for direct flow control using the same actuation. However, the approach was limited to simple motions which can be sufficiently described by a single actuation mode.

Similar to its application in numerical simulation, a Lagrangian-Eulerian framework may also be applied in model reduction to map a physical domain with moving boundaries to a computational domain with fixed boundaries, thus the application of POD-Galerkin projection in the fixed computational domain became straightforward. The mapping idea was successfully applied in the modeling of an inviscid and irrotational flow passing a cylinder with streamwise oscillation[68;69], incompressible viscous flow passing a pitching and plunging foil[70], and experimental data of the flow passing a pitching and plunging foil[71;72]. Though promising and mathematically rigorous, the Lagrangian-Eulerian approaches were often complex in formulation due to the complexity of mapping functions and associated Jacobian in governing equations. More importantly, the applications were mostly limited to small amplitudes for the solid body motion or deformation to avoid ill-conditioned matrices in computation. Wang and Shoele[73] proposed to use conformal mapping to largely reduce the complexity in the transformation between the moving physical domain and the fixed computational

domain. However, their approach was restricted to two-dimensional configurations.

To avoid complex mapping functions, a completely different route was taken to define an arbitrary stationary domain including both fluid and solid while allowing a direct application of POD-Galerkin projection on a modified equation in the combined fluid-solid domain[62–64;74]. A recent work by Liberge and Hamdouni[74], extended from their earlier work on one-dimensional Burgers equation[75], defined global POD modes from a global fluid-solid velocity field and successfully built a ROM for the flow passing a spring-attached cylinder oscillating at small amplitude. The same idea, to define global POD modes in a combined fluid-solid domain, was leveraged in the current work as well as our earlier work on ROMs for flows around flapping wings and moving cylinders[62–64]. However, there was a notable difference in Galerkin projection on the modified equation in the combined fluid-solid domain: the derivation by Liberge and Hamdouni[74] was based on so-called fictitious domain method[76;77] while the current work was based on immersed-boundary method (IBM)[6;12;13]. When solid mechanics and material properties are not considered, such as in applications with solid boundaries moving with prescribed motion or deformation, the IBM-based formulation is more convenient in derivation and more easily extended to applications with large motions and deformation.

In the persent work global POD modes are defined in a combined fluid-solid domain to avoid the problem caused by a morphing fluid domain with moving solid boundaries. The governing equation is defined in the same combined domain as a modified Navier-Stokes equation with an IBM-based formulation. The Galerkin projection of the modified equation in the combined domain on the global POD modes leads to a global ROM capable of capturing fluid dynamics with moving solid boundaries, which are represented by additional terms coming from the modification to the original Navier-Stokes equation. Those additional terms are used to describe solid-motion effects come from an integration of terms embedded only in the solid domain with motion/deformation. So, the additional terms for a global ROM need to be updated constantly to keep up with their constant motion.

Based on two different methods to describe solid motion, there are two different numerical implementations for the update of these additional ROM terms: (1) Continuous Solid

Domain Method (CSDM) where the solid motion is described as a continuous motion of a continuous solid domain; (2) Decomposed Solid Domain Method (DSDM) where the solid motion is further decomposed to a superposition of solid modes and needs to be computed on-the-fly by a separate "solid ROM". The CSDM implementation is straightforward and accurate, but is computationally more expensive; the DSDM implementation is much faster by modeling the solid motion, at the cost of slightly reduced accuracy. Both CSDM and DSDM are implemented in the global POD-Galerkin projection framework and compared in this work. It is worth noting that the focus of current study is to consider an arbitrary solid motion with one-way coupling from solid to fluid only, and fully-coupled fluid-structure interaction is not considered here but such extension is possible.

In the remainder of the chapter, section 2.2 briefly introduces the formulations for POD modes, then describes the new methodology to build ROM for moving solid boundaries. In section 2.3, the new ROM approach is applied on different cases for 2D and 3D numerical simulation or experiments, and related discussion is provided. At the end, section 2.4 summarizes and concludes the chapter.

2.2 Methodology

2.2.1 Global POD-Galerkin projection in a time-varying fluid domain

As reviewed in section 1.3, the traditional approach of POD-Galerkin projection requires fixed fluid domain, which is not the case for flows with moving solid boundaries and structures. In recent studies[63;74;78], a globally stationary domain including both fluid and solid areas is considered in both POD and Galerkin projection to solve the theoretical challenges posed by the time-varying fluid domain with moving solid boundaries or structures. Though there is distinct difference in application areas, the current usage of a combined fluid-solid domain for simplicity resembles the same idea behind a popular numerical simulation approach for moving boundaries, the immersed boundary method[6–8;10]. This method leverages

the combined domain to allow simple fixed meshes for discretization, where the effect of moving boundary/structure is represented as extra body-force terms added to the original Navier-Stokes equation in specific "solid" area. More details about IBM can be found in chapter 1.

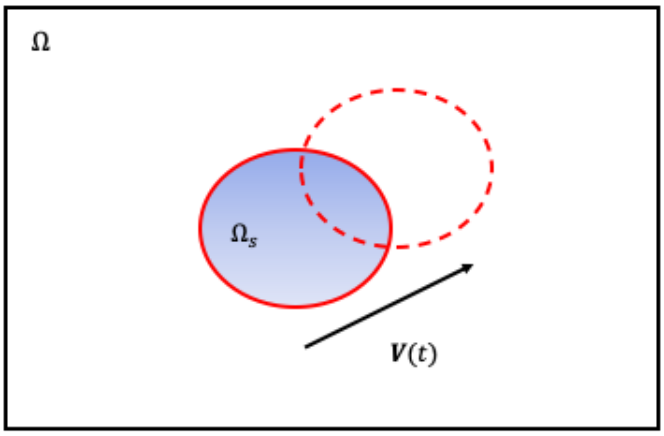

Figure 2.1: *Fluid and solid domains in a global POD-Galerkin projection, with Ω_s being the solid domain moving at velocity $\boldsymbol{V}(t)$ and Ω being the combined fluid-solid domain.*

As shown in figure 2.1, the combined fluid-solid domain is Ω and a moving solid domain Ω_s is defined with $\boldsymbol{V}(t)$ as its velocity. A global POD is defined in the combined domain Ω with a mathematically rigorous inner product (defined by Hilbert space) in the same Ω. For clarity, in the rest of this chapter, $\langle\ ,\rangle$ is used for an inner production defined over the whole combined domain Ω and $\langle\ ,\rangle_s$ is used for an inner product over the solid domain Ω_s only. To be consistent, a global Galerkin projection is applied on the same domain Ω, which would require a modified Navier-Stokes equation 2.1 as introduced in chapter 1 [13;16–18]:

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} = -\nabla p + \frac{1}{Re}\nabla^2\boldsymbol{u} + \boldsymbol{f}, \tag{2.1}$$

where $\boldsymbol{f}$ is the body forcing term numerically added in solid domain Ω_s to define the trajectory of solid motion and to satisfy boundary conditions for fluids [12;13]. Similar to the forcing term 1.5 reviewed in section 1.3, the body force $\boldsymbol{f}$ may be defined to advance the time from

t_n to t_{n+1}:

$$\boldsymbol{f} = \begin{cases} [(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} - \frac{1}{Re}\nabla^2\boldsymbol{u}]^n + \frac{1}{\tau}(\boldsymbol{V} - \boldsymbol{u}^n), & \text{in} \quad \Omega_s \\ 0. & \text{otherwise} \end{cases} \tag{2.2}$$

Note that the notation of the above forcing term is slightly different than equation 1.5 introduced in section 1.3, for the time step inherited from DNS is denoted as τ. The time step of DNS is denoted as Δ_0 only for this chapter, to distinguish it from the time step of the discretized ROM equations. The velocity of solid motion $\boldsymbol{V}(t)$ would be satisfied at the end of each time advancement τ, which represents the time scale to match the dynamic changes in this process. However, selecting a value of τ the same as the time step used in ROM computation may not lead to an optimal solution due to the time relaxation and delay in complex dynamic systems. The value of τ used in the ROM computation in the current work has been adjusted by trial and error for better accuracy, though choosing the same value as Δt used in current ROM computation provided computational results with comparable accuracy. A theoretical estimation to give a range for reasonable τ will be discussed later.

2.2.2 Continuous Solid Domain Method (CSDM)

A standard Galerkin projection of equation 1.3 on to the first N global POD modes $\boldsymbol{\phi}_i$ leads to a global ROM for fluid-solid domain:

$$\frac{da_i}{dt} = \sum_{j=0}^{N}(L_{ij} - L'_{ij})a_j + \sum_{j=0}^{N}\sum_{k=0}^{N}(Q_{ijk} - Q'_{ijk})a_j a_k + C'_i, \quad i = 1, 2, ..., N, \tag{2.3}$$

which is similar to the traditional ROM in equation 1.24, but with extra parameters representing the solid motion with velocity $\boldsymbol{V}$ embedded in solid domain Ω_s:

$$\begin{aligned} L'_{ij} &= \langle(\frac{1}{Re}\nabla^2\boldsymbol{\phi}_j + \frac{1}{\tau}\boldsymbol{\phi}_j), \boldsymbol{\phi}_i\rangle_s \\ Q'_{ijk} &= -\langle\nabla\cdot(\boldsymbol{\phi}_j\boldsymbol{\phi}_k), \boldsymbol{\phi}_i\rangle_s \\ C'_i &= \langle\frac{1}{\tau}\boldsymbol{V}, \boldsymbol{\phi}_i\rangle_s, \end{aligned} \tag{2.4}$$

while the old parameters L_{ij} and Q_{ijk} keep the same definitions but over the extended domain Ω.

The above method is named Continuous Solid Domain Method (CSDM) for an overall global POD-Galerkin approach, since an integration over a continuous solid domain is necessary to provide the correction parameters representing solid motion. CSDM method is simple and promising in applications[63]. However, the integrations of coefficients in equation 2.4 at each time step to accommodate the time-variant solid domain is very computationally expensive and overshadows the computational merits of ROMs.

2.2.3 Decomposed Solid Domain Method (DSDM)

To reduce computational cost on equation 2.4, another variation of the global POD-Galerkin approach, Decomposed Solid Domain Method (DSDM) is developed. DSDM decomposes the time-variant solid domain into Lagrangian POD modes, just as Eulerian POD modes are developed for fluid dynamics. Thus the solid motion may be described by a reduced-order model of the solid modes. The impact of solid motion to fluid dynamics may be represented by the solid modes and their dynamic ROM to avoid expensive computation in equation 2.4. In fact, a separate ROM for solid motion also brings the convenience to further development of two-way fluid-solid coupling as shown below.

In DSDM, a characteristic scalar function $\chi_s(\boldsymbol{x}, t)$ (a.k.a. shape function) is introduced:

$$\chi_s(\boldsymbol{x}, t) = \begin{cases} 1, & \text{in} \quad \Omega_s \\ 0, & \text{otherwise} \end{cases}, \tag{2.5}$$

which allows for rewriting the coefficients in equation 2.4 as:

$$\begin{aligned} L'_{ij} &= \langle (\frac{1}{Re}\nabla^2\boldsymbol{\phi}_j + \frac{1}{\tau}\boldsymbol{\phi}_j), \chi_s\boldsymbol{\phi}_i \rangle \\ Q'_{ijk} &= -\langle \nabla \cdot (\boldsymbol{\phi}_j\boldsymbol{\phi}_k), \chi_s\boldsymbol{\phi}_i \rangle \\ C'_i &= \langle \frac{1}{\tau}\boldsymbol{V}, \chi_s\boldsymbol{\phi}_i \rangle, \end{aligned} \tag{2.6}$$

defined by an inner product of a fixed combined domain Ω. Similar to the fluid field, the solid domain and its motion, defined now by the characteristic function χ_s, can also be decomposed to individual modes $\psi_m(\boldsymbol{x})$ (i.e. Lagrangian POD modes) to reduce a continuous solid domain to its low-dimensional representation with a truncation in the number of modes:

$$\chi_s(\boldsymbol{x},t) = \sum_{m=0}^{\infty} b_m(t)\psi_m(\boldsymbol{x}) \approx \sum_{m=0}^{M} b_m(t)\psi_m(\boldsymbol{x}). \tag{2.7}$$

The decomposition of the solid domain and its motion provide an approximation of the coefficients in equation 2.4:

$$\begin{aligned}
L'_{ij} &\approx \sum_{m=0}^{M} \langle (\frac{1}{Re}\nabla^2\boldsymbol{\phi}_j + \frac{1}{\tau}\boldsymbol{\phi}_j), \psi_m\boldsymbol{\phi}_i\rangle b_m(t) = \sum_{m=0}^{M} L^*_{ijm} b_m(t) \\
Q'_{ijk} &\approx -\sum_{m=0}^{M} \langle \nabla\cdot(\boldsymbol{\phi}_j\boldsymbol{\phi}_k), \psi_m\boldsymbol{\phi}_i\rangle b_m(t) = \sum_{m=0}^{M} Q^*_{ijkm} b_m(t) \\
C'_i &\approx \sum_{m=0}^{M} \langle \frac{1}{\tau}\boldsymbol{V}, \psi_m\boldsymbol{\phi}_i\rangle b_m(t) = \sum_{m=0}^{M} C^*_{im} b_m(t),
\end{aligned} \tag{2.8}$$

where new coefficients L^*_{ijm}, Q^*_{ijkm} and C^*_{im} are all time-independent and need only to be computed once at the beginning like other coefficients in equation 2.3. The new model, which removes expensive inner product integration to update coefficients, is:

$$\frac{da_i}{dt} = \sum_{j=0}^{N}(L_{ij} - \sum_{m=0}^{M} L^*_{ijm}b_m)a_j + \sum_{j=0}^{N}\sum_{k=0}^{N}(Q_{ijk} - \sum_{m=0}^{M} Q^*_{ijk}b_m)a_j a_k + \sum_{m=0}^{M} C^*_{im}b_m, \quad i = 1, 2, ..., N, \tag{2.9}$$

To close this system, a level-set-type equation[79] depicting the evolution of the solid boundary is introduced:

$$\frac{\partial\chi_s(\boldsymbol{x},t)}{\partial t} + \boldsymbol{V}\cdot\nabla\chi_s(\boldsymbol{x},t) = 0, \tag{2.10}$$

and it is projected onto solid modes ψ_m for the time evolution of b_i:

$$\frac{db_i}{dt} = -\sum_{m=0}^{M}\langle \boldsymbol{V}\cdot\nabla\psi_m(\boldsymbol{x}), \psi_i(\boldsymbol{x})\rangle b_m, \quad i = 1, 2, ..., M, \tag{2.11}$$

where $\boldsymbol{V}$ is the velocity at the fluid-solid interface and it is assumed known in cases with prescribed solid motion. In a general three-dimensional rigid body motion, the Eulerian velocity $\boldsymbol{V}(t)$ has 6 degrees of freedom:

$$\boldsymbol{V}(t) = \sum_{p=1}^{6} T_p(t)\boldsymbol{X}_p(\boldsymbol{x}), \tag{2.12}$$

where $\{T_p\} = \{V_x(t), V_y(t), V_z(t), \omega_x(t), \omega_y(t), \omega_z(t)\}$ represents the time-varying prescribed translations and rotations respectively on the extended coordinate $\boldsymbol{X} = (\boldsymbol{n}_x, \boldsymbol{n}_y, \boldsymbol{n}_z, \boldsymbol{R}_x, \boldsymbol{R}_y, \boldsymbol{R}_z)$. Thus, the temporal variation components of the motion may be separated from the computation in equation 2.9 and equation 2.11 for terms:

$$\begin{aligned} C^*_{im} = \langle \frac{1}{\tau}\boldsymbol{V}, \psi_m \boldsymbol{\phi}_i \rangle = \frac{1}{\tau}\sum_{p=1}^{6} T_p(t)\langle \boldsymbol{X}_p, \psi_m\boldsymbol{\phi}_i\rangle \qquad \text{and} \\ \langle \boldsymbol{V}\cdot\nabla\psi_m(\boldsymbol{x}), \psi_i(\boldsymbol{x})\rangle = \sum_{p=1}^{6} T_p(t)\langle \boldsymbol{X}_p\cdot\nabla\psi_m(\boldsymbol{x}), \psi_i(\boldsymbol{x})\rangle. \end{aligned} \tag{2.13}$$

The separation of time-varying components reduces the computationally-intensive spatial integration of ROM coefficients from needing to be computed at each time step to only once at the beginning, and it drastically reduced the online computational time. The dramatic speed-up of the above DSDM ROM formulation offers the capability of real-time computation with lower but adequate resolution to study fluid problems with moving solid boundaries.

Both CSDM ROM and DSDM ROM equations were solved numerically with semi-implicit scheme which guaranteed 2-order temporal precision. The details of this numerical scheme are presented in appendix A.

2.3 Results and discussion

The global POD-Galerkin projection approach, both CSDM and DSDM, were applied in this section on different types of high-fidelity high-resolution 2D/3D computational and experimental databases. The ROM results were benchmarked against the original datasets

for accuracy and efficiency.

2.3.1 Flow past 2D oscillatory cylinder

The global ROM approach was first applied to a canonical case of the two-dimensional flow past an oscillatory cylinder under a prescribed sinusoidal motion along the vertical (transverse) direction y:

$$y(t) = y_0 \sin(2\pi f t), \tag{2.14}$$

where $y_0 = 0.65$ and $f = 0.1568$, non-dimensionalized by the cylinder diameter D and the far-field incoming flow velocity U^*. At Reynolds number $Re = 200$, the chosen parameters led to a "P+S" type of vortex street[22], as shown in figure 2.2. The numerical simulation, in this case and other cases of the paper, was performed by a well-validated incompressible flow solver[63;64;80] with the moving boundaries handled by the immersed-boundary method[16;81]. The computational domain was 30×30, non-dimensionalized by the cylinder diameter D, and it had a non-uniform mesh at 601×601 which was fine and nearly uniform in the near-field of the oscillatory cylinder and gradually stretched coarser moving towards the far-field.

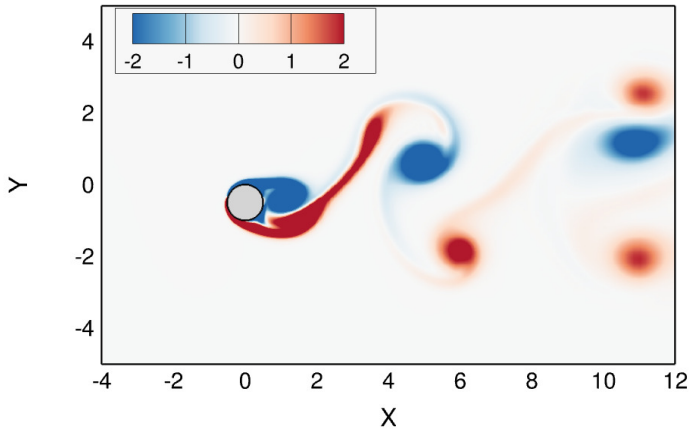

Figure 2.2: *A typical snapshot of the flow past 2D oscillatory cylinder contoured by vorticity.*

After the initial numerical and physical transition, data snapshots were taken every 50 time steps which were uniformly chosen at $\Delta t_0 = 0.0016$ in computation, and a total of

326 snapshots were taken between the dimensionless time $t = 640.0$ and $t = 666.0$ which covered four periods of vortex shedding from the cylinder. These snapshots were used to provide POD modes for model order reduction. In the current case, $N = 20$ global POD flow modes were used to build both CSDM ROM (equation 2.3) and DSDM ROM (equation 2.9). $M = 20$ POD solid modes were used in the DSDM ROM (equation 2.11) to model the solid motion. It is worth noting that larger spatial scales represented by low-order modes in ROM equations allowed for a coarser temporal discretization with larger time steps of $\Delta t = 0.1$ in the ROM computation (compared to $\Delta t_0 = 0.0016$ in DNS). The time scale $\tau = 0.05$ was used for this calculation. Figure 2.3 shows the mean flow (zeroth mode) and the first 5 POD modes of the flow field. For DSDM ROM in particular, the solid motion was also decoupled to POD solid modes, and figure 2.4 shows the first 6 POD solid modes.

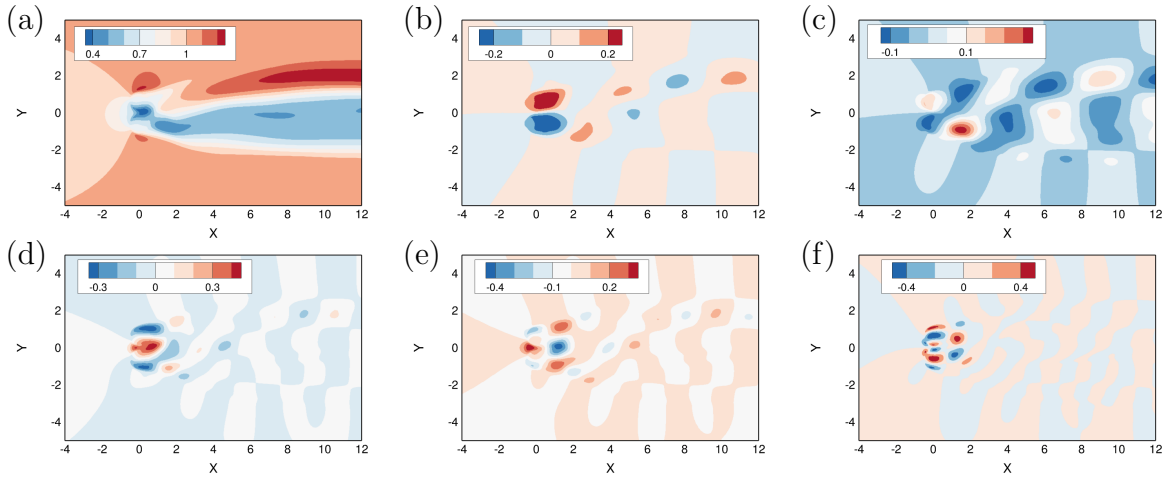

Figure 2.3: *The leading POD modes of the flow past 2D oscillatory cylinder: (a) the mean flow (zeroth mode) and (b – f) the first 5 global POD modes, contoured by horizontal velocity* v_x.

As shown by the phase portraits of the leading fluid modes in figure 2.5 (a) and (b), both CSDM ROM and DSDM ROM successfully captured the dynamics compared to the direct low-order projection from the original DNS data (notated by "L-DNS" in the figure). For DSDM ROM, the phase portraits of leading solid modes are shown in Fig. 2.5 (c) and (d) and they also match well with the exact solid motion trajectory used in DNS.

For a more intuitive comparison in physical domain, figure 2.6 compares the flow fields

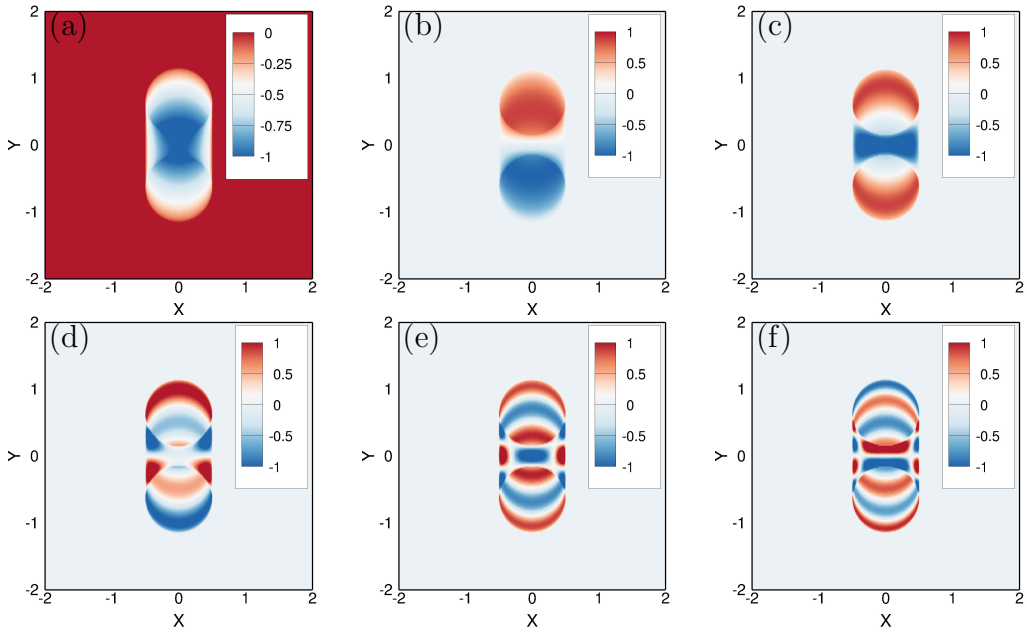

Figure 2.4: *The first 6 solid POD modes ψ_i of the shape function: (a – f) for mode 1 – 6 respectively.*

with the same contours of vorticity, from the original DNS, L-DNS, CSDM and DSDM methods. The L-DNS representation was calculated from the reconstruction of the first 20 POD modes and their time coefficients computed by a direct projection of DNS data (i.e. an exact presentation of DNS data in a 20-mode space for a fair comparison), while the CSDM and DSDM were reconstructed based on ROMs which also used the first 20 POD modes. Flow snapshots were taken at three consecutive moments within one vortex-shedding period: $t = 662.4$, $t = 664.0$ and $t = 666.0$. Overall, the CSDM and DSDM methods compared quite well with the DNS, and L-DNS results. In the region near the oscillating cylinder, noise and "shadows" appear in all low-dimensional representations including the low-order projection of DNS and CSDM/DSDM ROMs. If the low-order projection of DNS in the same space is considered the "exact solution", both CSDM and DSDM ROMs were able to accurately capture flow dynamics.

For a more quantitative comparison, the horizontal velocity component v_x was extracted along the centerline at $y = 0$ from the left ($x = -2$) to the right ($x = 10$) of the domain from the DNS, low-order projection of DNS, CSDM and DSDM data respectively, in figure

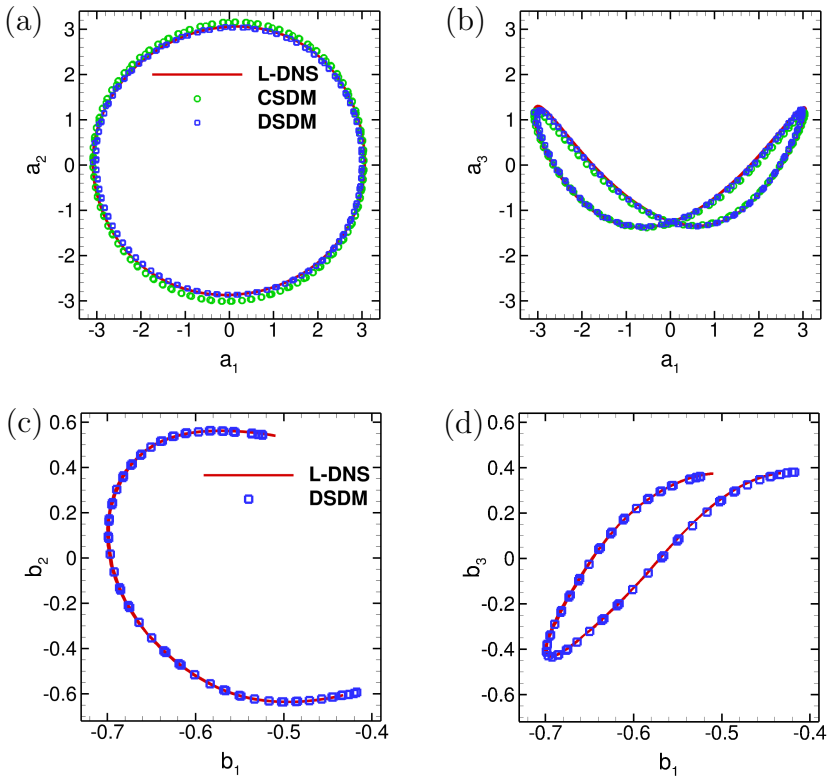

Figure 2.5: *The phase portraits of the time coefficients of the leading global POD modes and solid modes: (a) a_1 versus a_2; (b) a_1 versus a_3; (c) b_1 versus b_2; (d) b_1 versus b_3.*

2.7. For all three time moments, there was a good agreement across most of the center line except for the region near $x = 0$, where is the area with large solid motion which makes it sensitive to low-order approximations.

The drag and lift forces exerted on the cylinder surface were also computed and shown in figure 2.8. Note that, since pressure was not involved in the POD or ROM calculations, the drag and lift coefficients C_D and C_L for low-order projection of DNS and ROM reconstruction were computed by a method proposed by Noca et al.[82] and Noca et al.[83], which uses the velocity field only for force calculation. The detailed derivation can be referred to appendix B. The forces calculated from the DNS results were smooth but exhibited significant "noise" caused by small-scale, high frequency structures in the flow. These smaller scale features were truncated by the POD mode downselect (20 modes), resulting in smoother curves. Essentially, the force history plotted from POD reconstruction is similar to a smoothed DNS

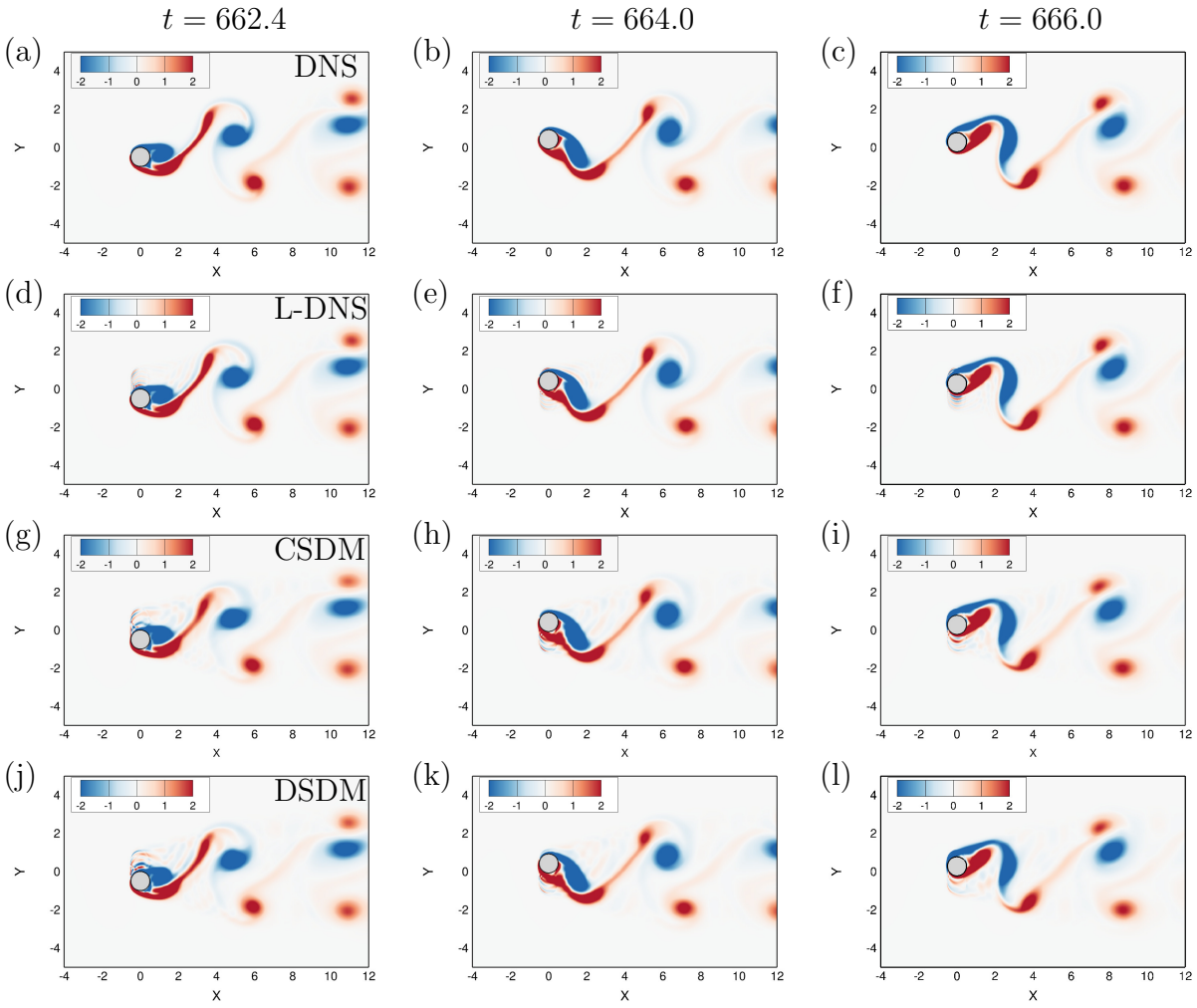

Figure 2.6: *The flow field contoured by vorticity at different time moments (t = 662.4, 664.0, and 666.0): (a – c) the original DNS data; (d – f) the low-order projection of DNS data with 20 modes; (g – i) CSDM ROM with 20 global POD modes; (j – l) DSDM ROM with 20 global POD modes and 20 solid modes.*

plot. The force history plotted from the CSDM and DSDM ROMs covering two periods predicted the lift force C_L well and matches with the plot of low-order projection of DNS in its amplitude and time periodicity. The ROM results are less accurate in their prediction of the drag force C_D in terms of the amplitude.

2.3.2 Flow past 3D oscillatory sphere

The global ROM method was then applied to a three-dimensional case to study the flow past an oscillatory sphere. Similar to its two-dimensional counter-part, the sphere oscillated

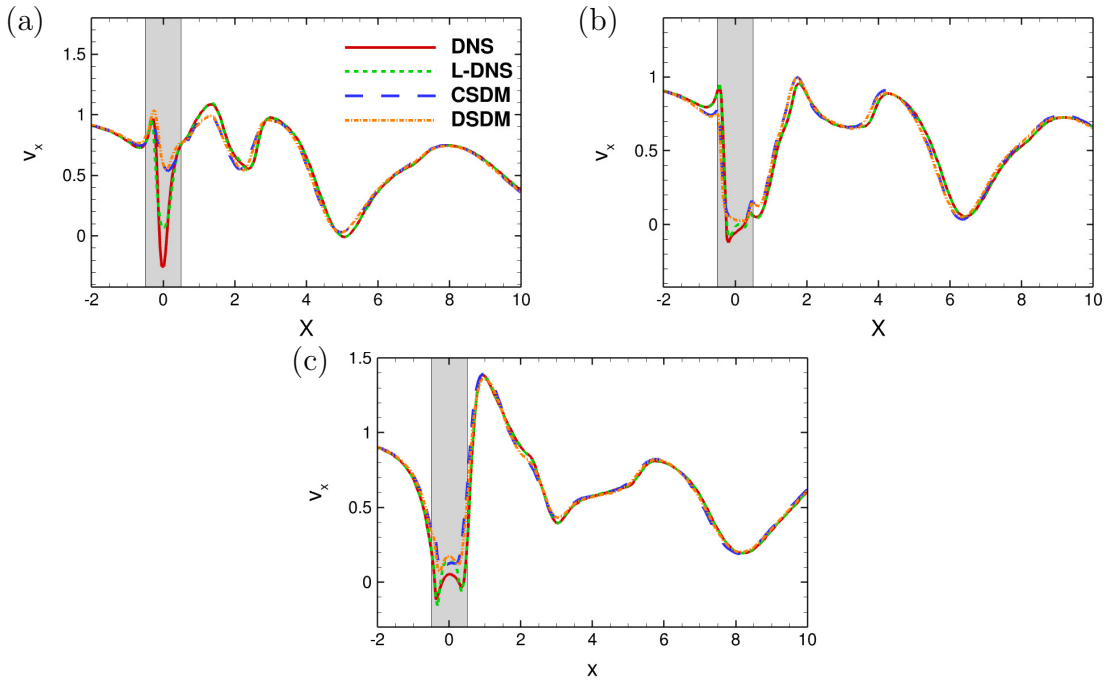

Figure 2.7: *The horizontal velocity v_x along the center line ($y = 0$) from $x = -2$ to $x = 10$ obtained from DNS, the low-order projection of DNS, CSDM ROM computation and DSDM ROM computation respectively: (a) $t = 662.4$, (b) $t = 664.0$, and (c) $t = 666.0$. The shade indicates the area that may be inside the cylinder.*

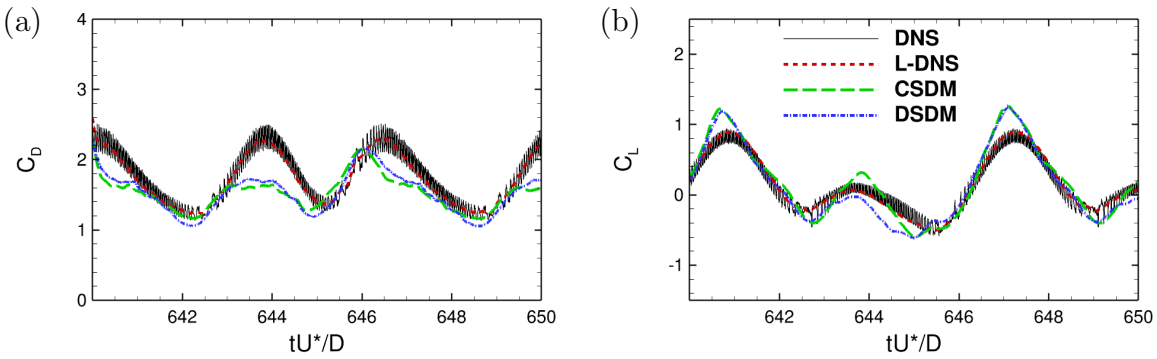

Figure 2.8: *The drag (C_D) and lift (C_L) coefficients obtained from DNS, low-order projection of DNS, CSDM ROM computation, and DSDM ROM computation.*

vertically with a sinusoidal motion $z(t) = z_0 \sin(2\pi f t)$, where $z_0 = 1.0$ and $f = 0.16$ were chosen here and non-dimensionalized by the diameter of the sphere D and the far-field incoming velocity U^* (figure 2.9). The Reynolds number $Re = 200$ was based on the same characteristic parameters. The computational domain was $15 \times 8 \times 8$ non-dimensionalized by D. Symmetric boundary conditions were enforced along y and z, both directions normal to

the incoming flow. A non-uniform mesh at $301 \times 161 \times 161$ was used to increase the near-field resolution of the flow surrounding the moving sphere. After the flow was fully developed and reached a stable vortex shedding state, data snapshots were taken every 20 time steps which were uniformly chosen at $\Delta t_0 = 1.54 \times 10^{-3}$, and a total of 401 data snapshots were taken between the dimensionless time $t = 30.8$ and $t = 43.1$ to cover two periods of vortex shedding from the sphere.

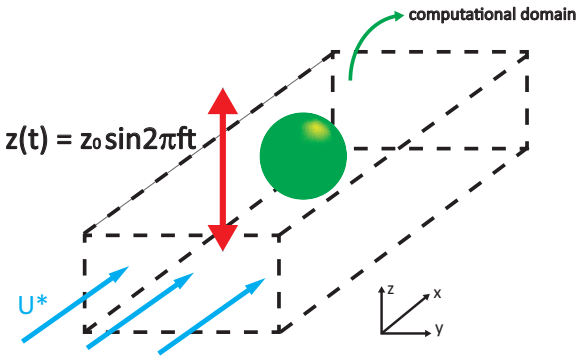

Figure 2.9: *The sketch of a 3D oscillatory sphere with incoming flow in a computational domain.*

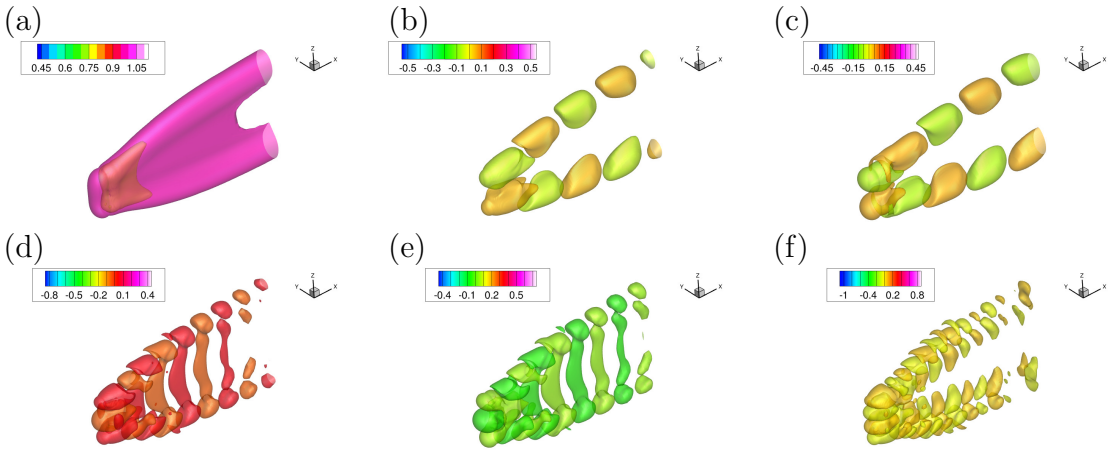

Figure 2.10: *The leading POD modes of the flow past 3D oscillatory sphere: (a) the mean flow (zeroth mode) and (b – f) the first 5 global POD modes, denoted by the iso-surfaces of streamwise velocity v_x: (a) iso-surfaces $v_x = 0.75$ and $v_x = 0.95$ are shown; (b – f) iso-surfaces $v_x = 0.05$ and $v_x = -0.05$ are shown.*

The mean flow (zeroth mode) and the first 5 POD modes are shown in figure 2.10. A time step of $\Delta t = 0.05$ and time scale $\tau = 0.1$ were used for the ROM calculations. Considering the

significantly increased cost to compute a three-dimensional inner product, which is required by the CSDM approach for every single time step, DSDM became the only feasible solution to develop an ROM in this case.

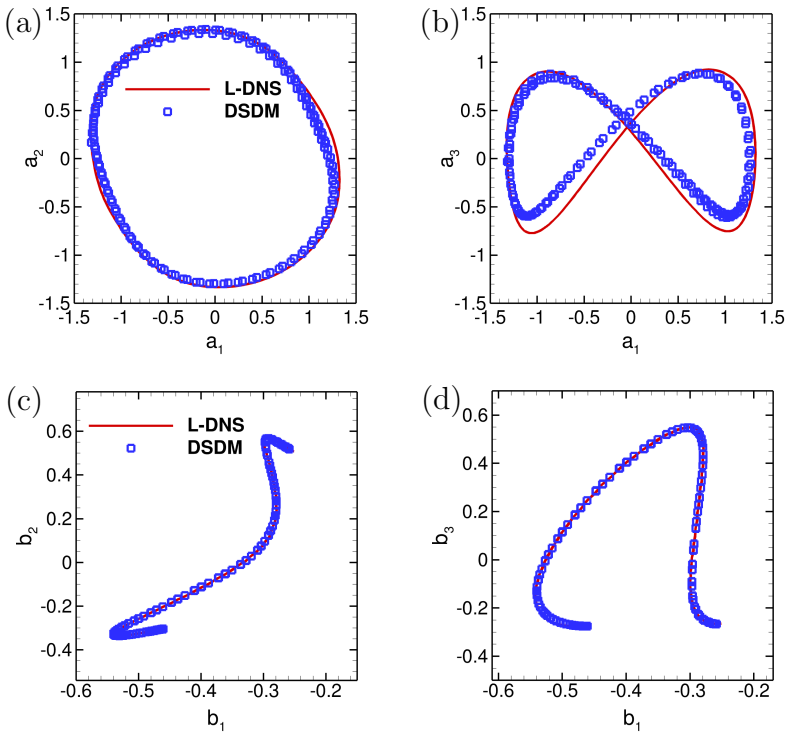

Figure 2.11: *The phase portraits of the time coefficients of the leading global POD and solid modes: (a) a_1 versus a_2; (b) a_1 versus a_3; (c) b_1 versus b_2; (d) b_1 versus b_3.*

To compare the dynamics computed by the DSDM ROM to the original dynamics of the flow, figure 2.11 plots the phase portraits of leading fluid and solid modes. The red lines represent the original dynamics by a direct projection of the original DNS data to the same low-order space of the POD modes used in ROM computation, while the blue circles are the DSDM ROM results. The comparison shows a good agreement between the dynamics reproduced by the ROM and the original dynamics of DNS data in both fluid flow and solid motion.

Figure 2.12 shows the time coefficients of the first two POD modes, obtained by the direct projection of the original flow onto POD modes, and solving DSDM equation (equation 2.9). Twenty POD modes as well as twenty solid modes were used for DSDM. It is found that

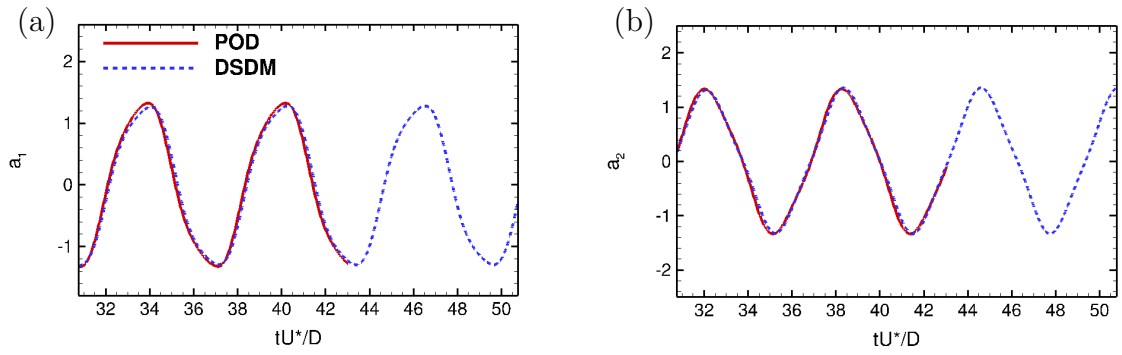

Figure 2.12: *The time coefficients of the first two POD modes as functions of dimensionless time.*

DSDM is capable of capturing the dynamics in 3D case in terms of amplitude and periodicity. The total energy of the system is strictly conserved which guarantees the stability of long-term simulation as the amplitude of $a_i(t)$ is well reserved. As shown in figure 2.12, the time coefficients by DSDM can keep the same amplitudes and periods even beyond the time range of the snapshots which POD modes are based on.

It can be seen that DSDM ROM captured the flow dynamics successfully compared to the direct projection from the original DNS data, with only small discrepancy between the limit cycles in figure 2.11 (b). The solid motion in DNS was also well captured by solid modes as shown in figure 2.11 (c) and (d).

The flow field was then reconstructed in a low-order space from the ROM computation. figure 2.13 compares snapshots of flow field at two different time moments $t = 40.0$ and $t = 43.1$, which are plotted respectively from the original full-order DNS data, the direct projection of the DNS data to the low-order space of the first 20 POD modes, and the flow reconstruction from the DSDM ROM in the same low-order space. The flow reconstruction by the DSDM ROM accurately represented the flow features (e.g. vortex shedding and wake structures) of the DNS flow, especially its low-order representation. In fact, it is arguably more fair to compare the DNS results and ROM results on the same low-order space (L-DNS). In the near-solid region, spurious flow structures appear in a way similar to the earlier two-dimensional case, as a result of lower resolution and higher sensitivity in the region.

For a quantitative comparison of the ROM's prediction of aerodynamic performance,

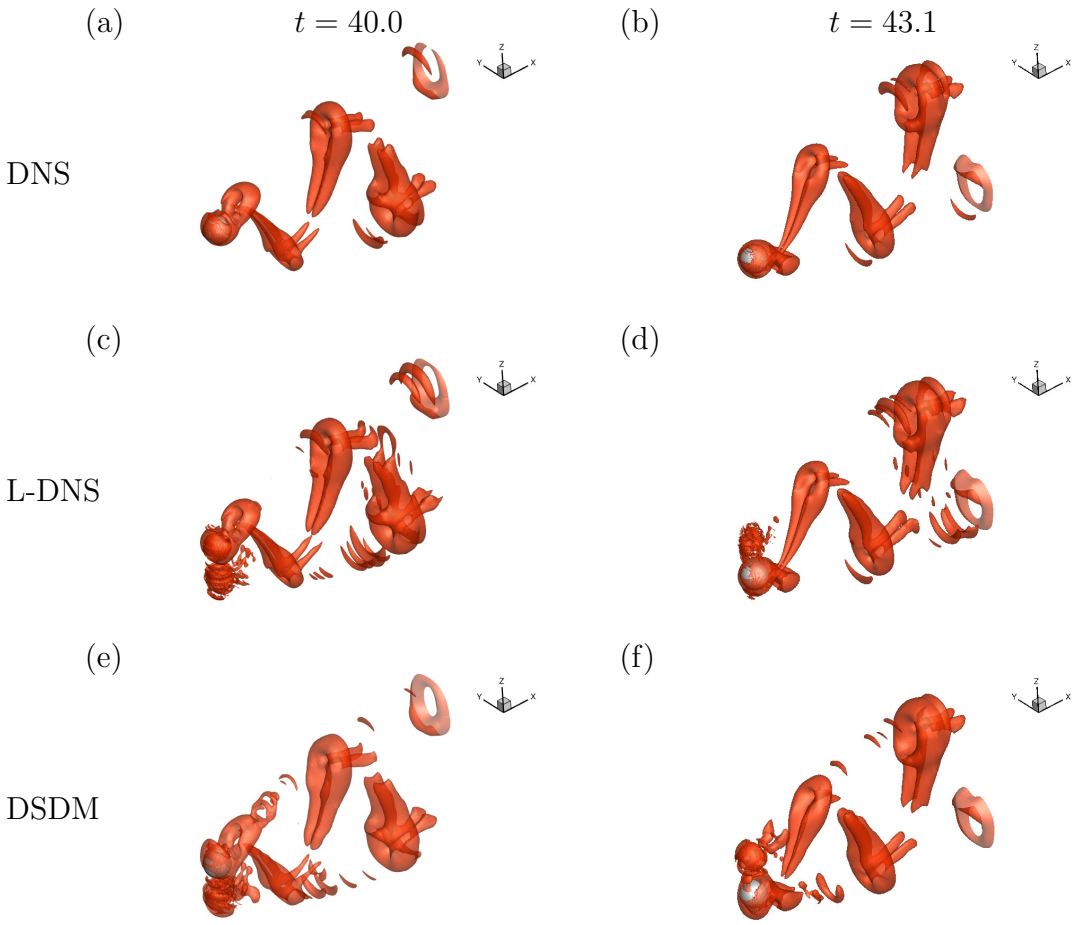

Figure 2.13: *The flow field at $t = 40.0$ and $t = 43.1$: (a – b) the original DNS data; (c – d) the low-order projection of DNS data with 20 modes; (e – f): DSDM ROM with 20 global POD modes and 20 solid modes. The iso-surface of $Q = 0.01$? is chosen here to compare three-dimensional vortex structures.*

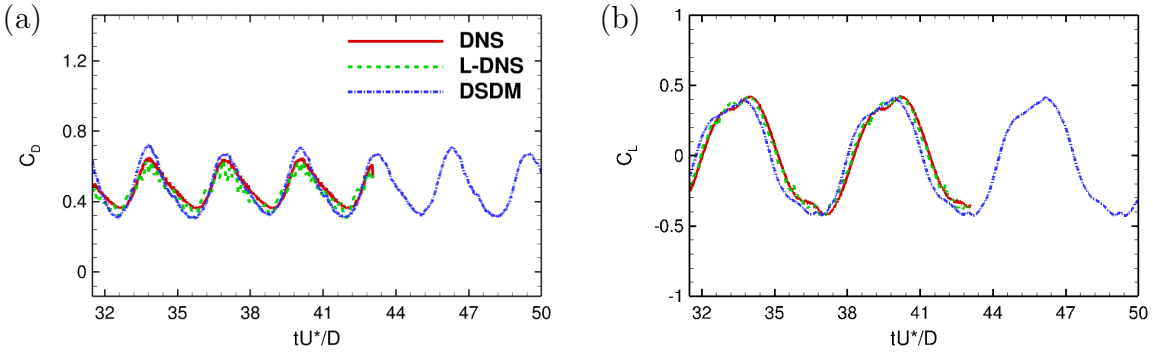

Figure 2.14: *The drag (C_D) and lift (C_L) coefficients obtained from DNS, the low-order projection of DNS, and DSDM ROM.*

drag and lift coefficients were computed from different flow reconstructions and plotted in figure 2.14. The original DNS data used in model reduction covers the time period between $t = 30.8$ and $t = 43.1$, which were indicated by the start and the end of drag and lift coefficients computed from the DNS and the low-order projection of DNS data. There was close matching of drag/lift force provided by the DSDM-ROM in the same time period. Notably, it is even more exciting to observe the ROM's capability to extend the force prediction beyond the original data as shown in the time period between $t = 43.1$ and $t = 50$. Though it is not a surprise in terms of the dynamics in a periodic system, the capability and robustness of the current ROM shown in its extended prediction (without forcing any periodicity) is remarkable and implies opportunities in its practical application.

2.3.3 Flow past 2D pitching-up NACA0012 airfoil: experimental data

The global ROM approach may also be applied to an experimental database in the same way as a numerical simulation database. With the turbulence from higher Reynolds number and noise from experimental measurement, the application on experimental data creates new challenges. In this section, the same methodology was applied to high-quality wind-tunnel PIV data of the flow past a pitching-up NACA0012 airfoil. The PIV data was taken in the Microsystem Aeromechanics Wind Tunnel (MAWT) facility at the Army Research Lab (ARL). For the clarity of the current paper, the experimental facility and its setup[78;84] were not detailed here. The experiment was conducted at $Re = 12,000$ defined by the incoming flow speed and airfoil chord length. The airfoil began with angle of attack $\alpha = 0$, and started to pitch up at $t = 16.5$ with angular velocity $\omega = 0.2$ around its quarter-chord point until reaching the maximum angle of attack $\alpha = 55°$. It is noted that the angular velocity is nondimensionalized by the same incoming flow and airfoil chord length for consistency, and it makes twice the value of the reduced pitching rate commonly used in other literatures with slightly different normalization[84].

A total of 394 PIV snapshots were taken for the entire process including the time period

before and after the pitching-up. To focus on the rapid evolution of dynamics during the pitching-up, the snapshots before and after the pitching-up were removed and only the 59 snapshots between $t = 16.5$ to $t = 21.6$ were used for POD analysis and ROM construction. Since the mesh and resolution of the raw experimental data are different from the computational framework for model reduction, a 2nd-order linear interpolation was used to fit the data of all frames to the same mesh of a computational domain 1.8×1.25 (chord length based dimensionless size). The first 20 global POD modes computed from the 59 PIV snapshots were used to build the ROM, which also used 20 solid modes computed directly from the pitching-up motion of the NACA0012 airfoil. A time scale of $\tau = 0.0075$ was used, while the time step used in ROM computation was $\Delta t = 0.01$, which is smaller than previous cases due the higher Reynolds number in experiment.

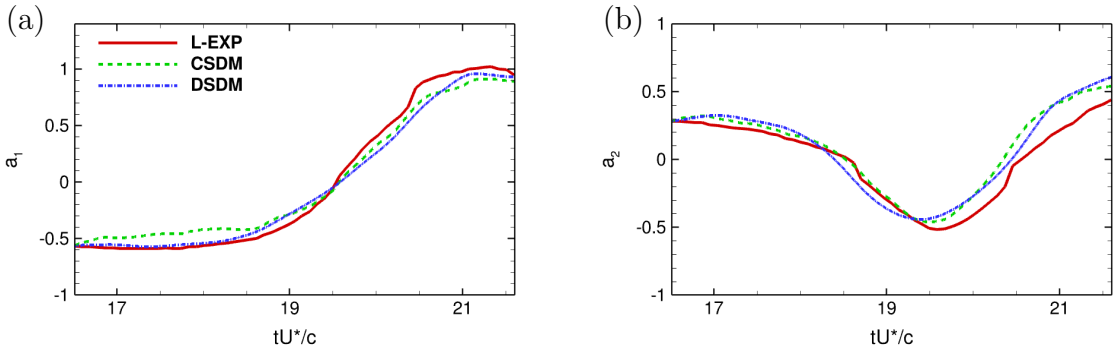

Figure 2.15: *The time coefficients of the first 2 global POD modes during the pitching-up of airfoil: a comparison between the direct low-order projection of DNS, CSDM ROM computation, and DSDM ROM computation.*

Figure 2.15 compares the time coefficients of the first 2 POD modes computed by CSDM and DSDM ROMs with the ones directly projected from the original PIV data in the same low-order space (i.e. "L-EXP"). Although the Reynolds number in the experiment was much higher than previous simulation cases, the dynamic behavior of the dominant energetic modes were accurately represented by both CSDM and DSDM ROMs. The ROMs also preformed well with the transition dynamics, which was considered more challenging to capture than previous limit-cycle dynamics (i.e. periodic fluid flow).

To have a more complete picture of flow dynamics, the original flow fields obtained

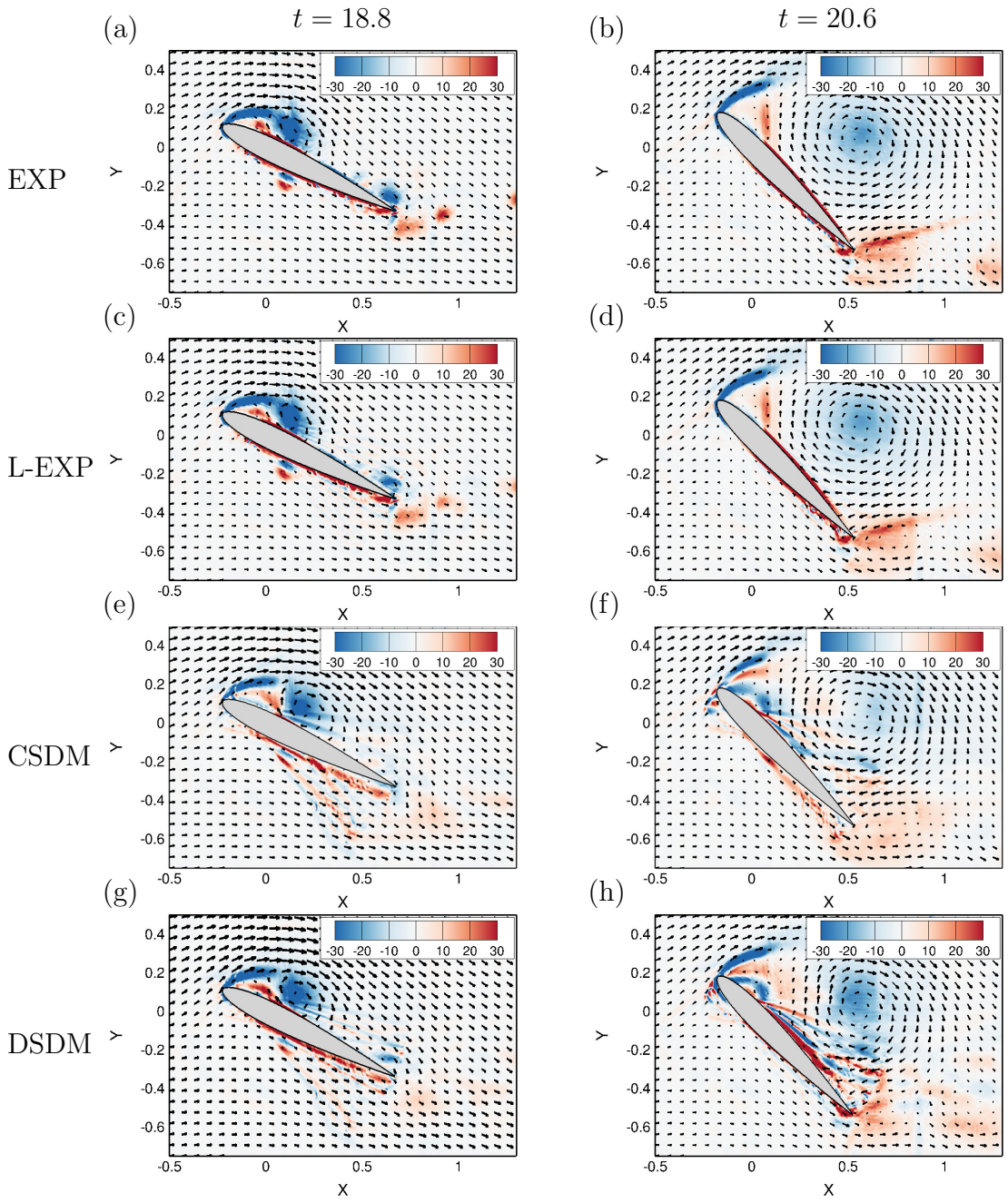

Figure 2.16: *The comparison of flow fields from (a – b) original experimental data, (c – d) the low-order projection of experimental data with 20 POD modes, (e – f) CSDM ROM with 20 fluid modes, and (g – h) DSDM ROM with 20 fluid and 20 solid modes, at* $t = 18.8$ *when* $\alpha = 26°$ *(left) and* $t = 20.6$ *when* $\alpha = 45°$ *(right). All flow fields are contoured by vorticity and overlaid with velocity vectors.*

from the raw PIV data (i.e. "EXP"), the low-order projection with the first 20 global POD modes (for fluid and solid) of experimental data (i.e. "L-EXP"), and the reconstructions from 20-mode CSDM/DSDM ROM computations are compared in figure 2.16. The two time

moments are chosen to focus on two distinct dynamic events during the pitching-up process: the growing and separation of leading-edge vortex (LEV) and the LEV shedding from the airfoil to pair with the trailing-edge vortex (TEV) towards the end of the pitching motion. The 20-mode low-order projection of experimental data, shown in figure 2.16 (c) and (d), represents the original flow data in a nearly identical fashion, demonstrating the possibility of maintaining the original dynamics in a low-order space (at least by an "ideal" ROM). More importantly, the flow fields computed directly from CSDM/DSDM ROMs, shown in figure 2.16 (e – h), resemble well the main dynamic features including the location and strength of LEVs. On the other hand, it is apparent that ROM reconstructions have some spurious flow structures near the airfoil surfaces and struggle in capturing TEV structures accurately. Furthermore, since CSDM uses the solid motion directly and DSDM uses a discrete representation of the actual motion, the flow reconstructions by CSDM ROM show less spurious structures.

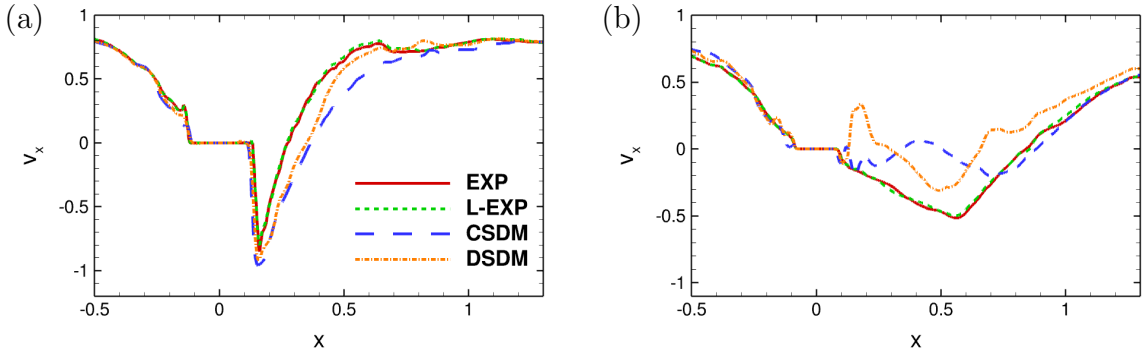

Figure 2.17: *The horizontal velocity component v_x as a function of x along $y = 0$ line at: (a) $t = 18.8$ and (b) $t = 20.6$.*

A more quantitative comparison is shown in figure 2.17 by retrieving streamwise velocity v_x along the centerline $y = 0$ from the same snapshots in figure 2.17. The streamwise velocity plotted from the direct projection of data is almost identical to the one from the original experimental data. The velocity computed by both CSDM and DSDM ROMs shows overall good agreement with the original flow at the beginning of the pitching-up when the angle of attack is small or medium and the LEV is still attached. However, when the the angle of attack increases and triggers vortex shedding and pairing, the ROM calculation becomes

inaccurate in the dynamically intense region near the pitching-up airfoil, though the ROM computation remains accurate in areas before and after the near-field region.

2.3.4 Flow past 2D rotating elliptic airfoil: experimental data

The experiment was conducted with the same equipments as used for pitching-up NACA0012 airfoil case at $Re = 12,000$. A full-span wing model with an elliptical shaped airfoil with a chord of 12 cm and thickness of 1.8 cm was pitched about mid-chord at a constant pitch rate. The pitch rate was held at a constant value of $\Omega^* = 0.12$. The schematic illustration of the experimental model is shown in figure 2.18. Again the turbulence at higher Reynolds number and the noise from experimental setup and measurement put the proposed ROM methodology in more challenges. POD was first applied on PIV snapshots for the bases, then the global Galerkin projection on the POD modes led to a global ROM. Different approaches for the integration of solid domain, CSDM and DSDM, are implemented and compared. A total of 360 PIV snapshots were used to compute POD modes for fluid flow. 20 POD fluid modes were used for both CSDM and DSDM, and another 20 POD solid modes were used for DSDM implementation. τ was set to be 0.0075 for the construction of ROM equations.

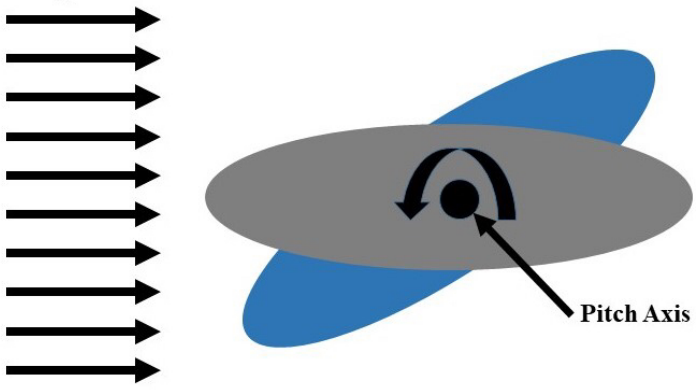

Figure 2.18: *Schematic drawing of the rotating elliptical airfoil.*

Figure 2.19 presents the time coefficients of first 2 POD modes. Although the Reynolds number was significantly high, the pitching motion is still dominant, thus the low-dimension

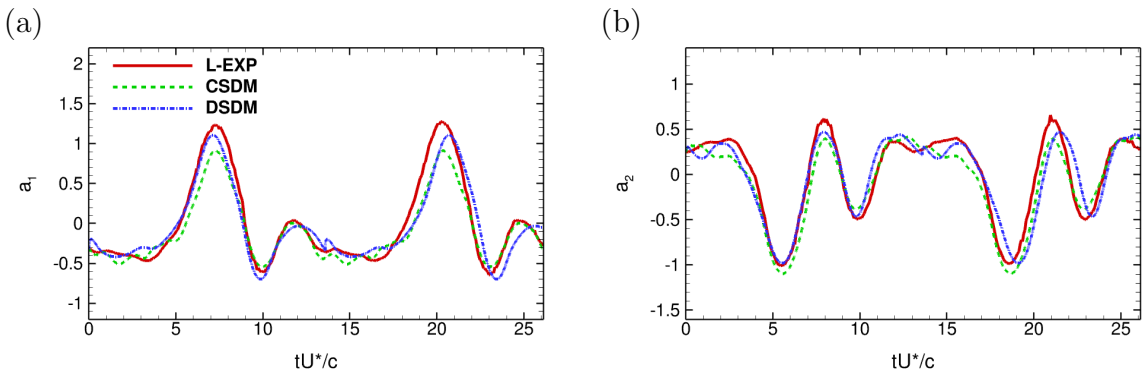

Figure 2.19: *The evolution of the time coefficients a_1 and a_2 for the experimental data at $Re = 12,000$ by L-EXP, the CSDM ROM, and the DSDM ROM.*

flow dynamics remained periodic. However, more fluctuations were observed in this experimental data. This might arise from experimental uncertainty, repeatability of the phase averaging, or even a transition to turbulence in some regions. It can be seen that both CSDM and DSDM accurately modeled the coefficients from the L-EXP of the experimental data. However, the DSDM results had slightly more fluctuation compared to the POD and CSDM, which may have been caused by the use of a smaller number of solid modes (20) in the DSDM computation. Such results show that the current global ROM methodology is able to accurately model the time coefficients with robustness for more complex flows even with uncertainties.

Flow fields with velocity vector and contours for vorticity at different phase angles are shown in figure 2.20 for both the experimental data (a – d) and CSDM rebuilt flow fields (e – h). Only near-field structures are shown due to the limited experimental resources and time. Since the same flow would repeat twice during one period it is enough to study the first half period. As expected, the flow is periodically dominated by DSV, TEV and the traditional bluff body shedding. When $\alpha = 90°$, DSV and TEV form simultaneously, and those two vortices are not symmetric, with the TEV being slightly more pre-dominant than the DSV. In figure 2.20 (e) it can be seen that ROM reconstruction can capture such asymmetry as well as the positions of the two vortices with a good level of accuracy. When the ellipse was pitched up to 135°, the flow was DSV dominated. Figure 2.20 (f) shows some non-physical fluctuations near the airfoil by ROM rebuilt, which may relate to the exclusion

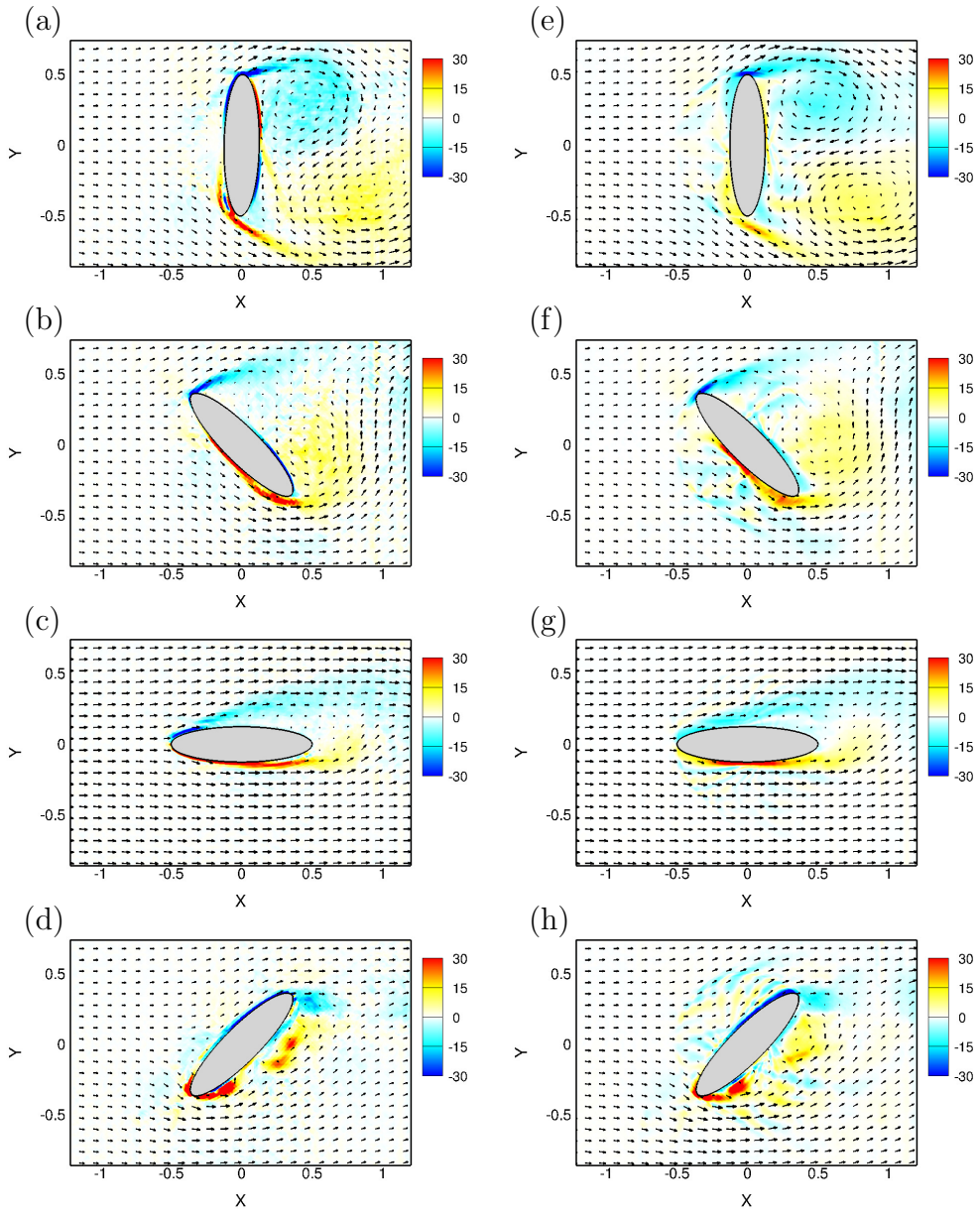

Figure 2.20: *The comparison of flow field with snapshots from PIV (a – d) and a reconstruction from ROM with CSDM (e – h) at different time and phase angles:* $\alpha = 90°$ *for (a) and (e),* $135°$ *for (b) and (f),* $180°$ *for (c) and (g), and* $225°$ *for (d) and (h). The vectors are for velocity field and the contours are for vorticity.*

of higher order information in the ROM. However TEV can still be captured clearly despite the non-physical results upstream of the airfoil. When $\alpha = 180°$ both DSV and TEV had convected downstream resulting in bluff body shedding to dominate the flow. The ROM reconstruction for this flow state accurately reconstructed the experimental flow field data.

At $\alpha = 225°$, figure 2.20 (h), which shows the early formation of the DSV, the ROM also accurately reconstructs the shear layer vortex (SLV) near the trailing edge.

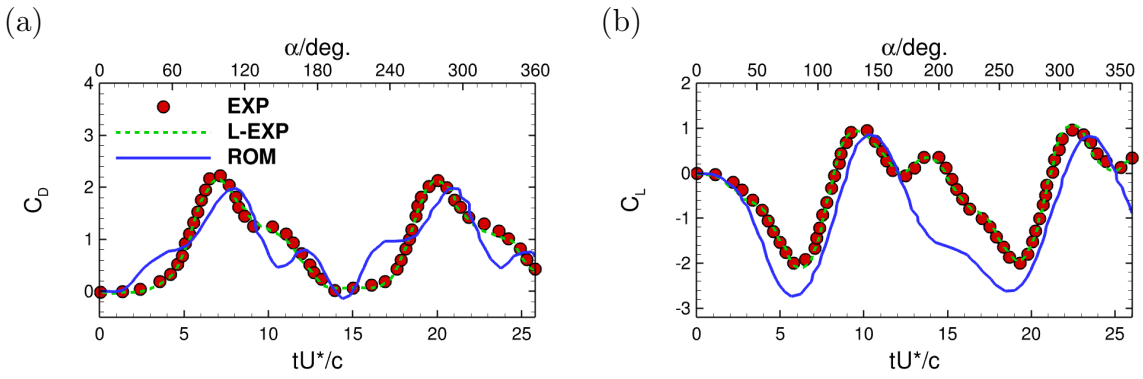

Figure 2.21: *The drag coefficient C_D and lift coefficient C_L obtained from for experimental measurements, L-EXP reconstruction and CSDM ROM reconstruction. The variation of the pitching angle is shown by the top x-axis.*

An analysis of the lift and drag coefficients was done and results are shown in figure 2.21. Forces were computed directly from velocity data of experiments, then the results from L-EXP reconstruction and ROM reconstruction with CSDM were calculated by the same method. While the L-DNS is able to recreate the lift and drag behaviors of the experimental data with high accuracy, the CSDM ROM deviated somewhat. In general it performed very well at predicting peak lift and drag values, although it over-predicted the negative lift peak as seen in figure 2.21 (b). Despite this, the timing of peak lift aligned well, while drag peaks had a slight lag. The accuracy of these results could potentially be improved by the inclusion of more modes in the ROM or by investigating the non-physical flow features that appear near the airfoil at some times (figure 2.20).

2.3.5 Computational cost analysis

Computational costs by DNS and CSDM/DSDM ROMs of all cases studied in prior sections are listed in this section to highlight the computational saving of the proposed method. Though two cases were based on experimental data, the computational cost of their ROMs remains meaningful in their comparison to the tremendous experimental cost in preparing the PIV data. The computational cost includes two parts with distinct nature, the offline

cost and the online cost. The offline cost refers to the time consumed in computing POD modes and the integration of all time-independent coefficients in ROM construction; the online cost refers to the time consumed in the all computation to advance the actual ROMs in time, including the time integrator of ROMs, and the potential update of time-dependent terms or coefficients in some approaches such as the CSDM coefficients in equation 2.4, which changes along with the solid motion. For a fair comparison, all computational costs (i.e. wall time) listed in table. 2.1 are measured by running on the same computer. The 2D computation for the flow past oscillatory cylinder and all ROM computation ran as a serial code, and the 3D computation for the flow past oscillatory sphere ran as a parallel code with 5× CPU power.

Table 2.1: *Comparison of the computational time of ROMs to the original data.*

Original data	CSDM	DSDM
2D oscillatory cylinder (DNS): 20h15m	offline: 5m online: 11h37m	offline: 1h22m online: 4s
3D oscillatory sphere (DNS): 6h38m†	N/A	offline: 26h37m online: 44s
Pitching-up airfoil (EXP)	offline: 6m online: 22h33m	offline: 47m online: 13s
Rotating elliptic airfoil (EXP)	offline: 7m online: 27h24m	offline: 1h8m online: 14s

† run in parallel with 5× CPU power.

The data in table 2.1 indicate a much improved computational efficiency by ROMs especially in the online portion. For the 2D oscillatory cylinder case, CSDM ROM reduced the computational cost to 57% and DSDM ROM reduced to only 6.7%. More important, DSDM ROM moves most of the computational cost to offline, which is computed only once in the initial development of ROMs, and reduces the mission-critical online cost to only 4 seconds (0.005% of the original computation) for the entire event which allows to run ROMs in real-time for computation, optimization, and control. For the 3D oscillatory sphere case, the original DNS cost is scaled by a factor of 5 to 33h10m to count for parallel computation,

and the overall DSDM ROM cost reduced to 81% of the scaled time. However, the mission-critical online cost of the DSDM ROM was reduced to a total of 44 seconds and 0.037% of the original cost, which allows again a real-time computation. For the experimental data of NACA0012 airfoil, the online cost of DSDM ROM was reduced to 13 seconds. Similar huge reduction of online cost is also observed for the experimental data of elliptic airfoil (only 14 seconds). The high computational effiiciency opens up the possibility of future work on real-time flow-control in scenarios such as real-time response to the dynamics stall of a pitching-up airfoil or a rotating airfoil. It is worth noting that the computational saving of ROMs comes from two factors: one is the drastic order reduction in degree of freedom of ROM equations, and the other is that much larger scales resolved by ROMs also allow integration within the ROM equations with much larger time steps in computation.

2.3.6 Error analysis

Lastly, the accuracy of ROMs is investigated for its sensitivity to a few key control parameters: the number of global POD modes N, the number of solid modes M in DSDM, and the nominal time scale τ for solid motion. The study focuses only on DSDM ROM as it is the only feasible choice for real-time online computation as shown in the prior section.

The amplitude of the first two modes, $A = \sqrt{a_1^2 + a_2^2}$, is often used to measure the behavior of attracters in their low-order phase portraits[54]. An averaged error E for ROM accuracy may therefore be defined by

$$E = \frac{1}{T} \int_0^T |A_{\mathrm{ROM}} - A_{\mathrm{L-DATA}}| dt, \tag{2.15}$$

where T is the total time of ROM computation, A_{ROM} is the amplitude computed by ROMs, and $A_{\mathrm{L-DATA}}$ is a direct lower-order projection of the original data (i.e. L-DNS or L-EXP).

Figure 2.22 shows the impact to model accuracy from choosing different number of modes, N for the global POD modes and M for the solid POD modes, while the time scale τ is fixed respectively at 0.05 and 0.1 for the two numerical simulations. For the two experimental

cases, $\tau = 0.0075$ and $\tau = 7.5 \times 10^{-4}$ were used for the pitching-up airfoil and rotating elliptic airfoil case respectively. In the study, when N is compared, all M is fixed at 20; when M is compared, all N is fixed at 20. In the first two numerical simulations, the average error is not sensitive to the choice of N between 10 and 30, and remains below 4%. In the experimental case of NACA0012 airfoil, however, the error first decreases from 9.5% to below 6% as N increases from 10 to 20, then the error increases again with additional modes. Larger error is found for the rotating elliptic airfoil case, with the lowest error around 10% achieved when $N = 10$ and $M = 20$. The increase of error with the number of global fluid modes N is from the inclusion of spurious high-frequency components in higher POD modes which are noisier in experimental data and also harder to resolve at high Reynolds number. On the other hand, the increase of solid modes M shows more consistent contribution to the model accuracy by improving the resolution of moving solid boundaries. For most of the computation in this paper, $N = 20$ and $M = 20$ are used as a balanced choice for ROMs of all three cases.

Contrary to the relatively low sensitivity to N, the error for all four cases is found more sensitive to M. As seen in figure 2.22, when increasing M for 10 to 20, the error decreased by about 2% for two-dimensional DNS case, about 4% for three dimensional DNS case and about 8% for the experimental case of pitching-up NACA0012, respectively. In addition, the error for the experimental case of rotating elliptic airfoil even decreased by one order of magnitude when M was increased from 5 to 20. Since solid modes are the low-order representations of the solid motion, it is crucial to keep enough number of solid modes when building ROM to maintain a relatively low error.

The dependence of τ on the error was also investigated with N and M both fixed at 20. As derived in equation 2.2, τ is originated from the discretization of Navier-Stokes equation with IBM, so its value should be closely related to the numerical treatment on the original Navier-Stokes equation. However, the choice of τ still remains open, for it does not necessarily have to be the same value as in DNS, furthermore there is no τ whatsoever in experiments. Therefore, it would be practical for a new ROM setup if an *a priori* τ can be estimated. A simple way to estimate the proper value of τ for ROM is provided, which leverages the

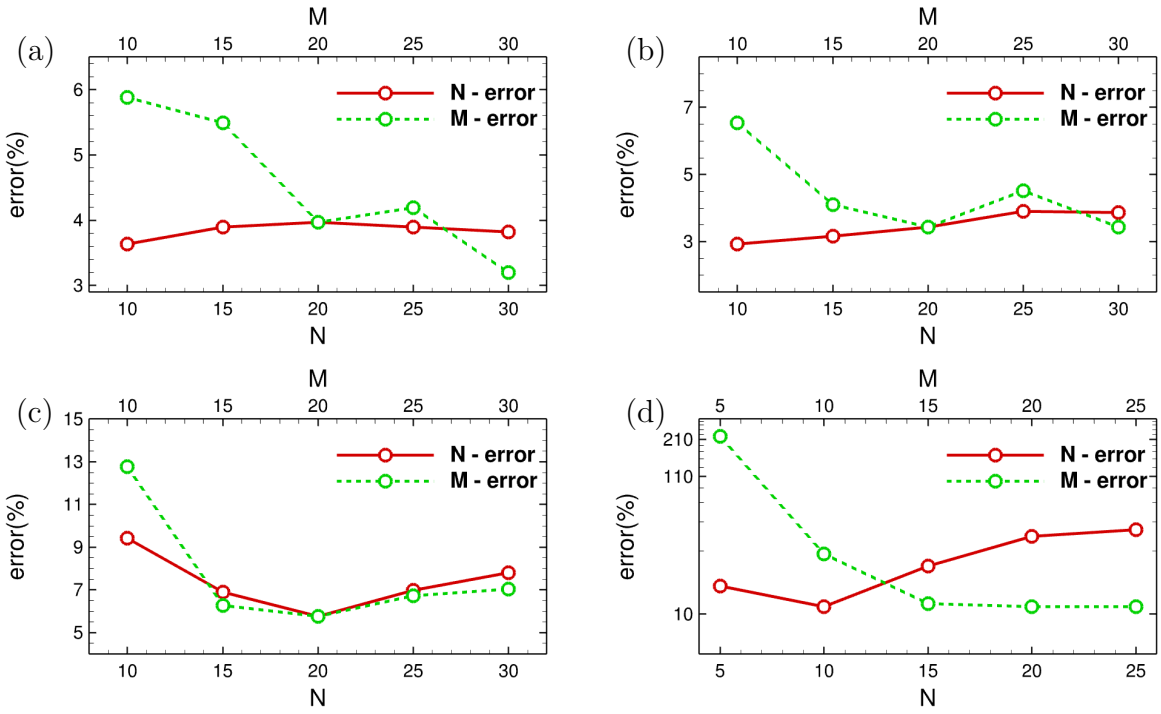

Figure 2.22: *ROM sensitivity to the number of POD modes for fluid N and for solid M in: (a) the flow past 2D oscillating cylinder; (b) the flow past 3D oscillating sphere; (c) the flow past pitching-up airfoil in the experiments; (d) the flow past rotating elliptic airfoil in the experiments, the y axis is in logarithmic scale.*

numerical stability criterion of the original DNS. A third-order Runge-Kutta scheme (RK3) was used in DNS, thus the stability limit the Courant–Friedrichs–Lewy number (CFL) can be given by $CFL = \sqrt{3}$[13]. With this CFL the criterion is designed as follow[20]:

$$\tau_0 = CFL \times \frac{\min\{\Delta x, \Delta y, \Delta z\}}{U_{\max}}, \tag{2.16}$$

where $U_{\max}$ is the maximum flow velocity in the associated FOMs, which can be approximated by the velocity of incoming flow, thus $U_{\max} = 1$ was used for all four cases in this work. Then with the mesh size used for the inner product in the computations of ROMs, an analytical estimation τ_0 for an optimal τ can be obtained, as shown in table 2.2.

On the other hand, the optimal τ can also be found numerically by parametric study, which is denoted as τ_{opt}. Figure 2.23 presents the variation of errors in terms of τ for DNS and experimental cases. It is found that for all cases, as τ increases, the error decreases first

Table 2.2: *τ_0 estimation and the optimal τ_{opt} for smallest error obtained from the numerical study for different cases.*

case	$\min\{\Delta x, \Delta y, \Delta z\}$	τ_0	τ_{opt}
2D oscillatory cylinder	0.008	0.0138	0.025
3D oscillatory sphere	0.0154	0.0266	0.075
pitching-up NACA0012 airfoil	0.00313	0.0054	0.0075
rotating elliptic airfoil	0.003	0.0087	0.00075

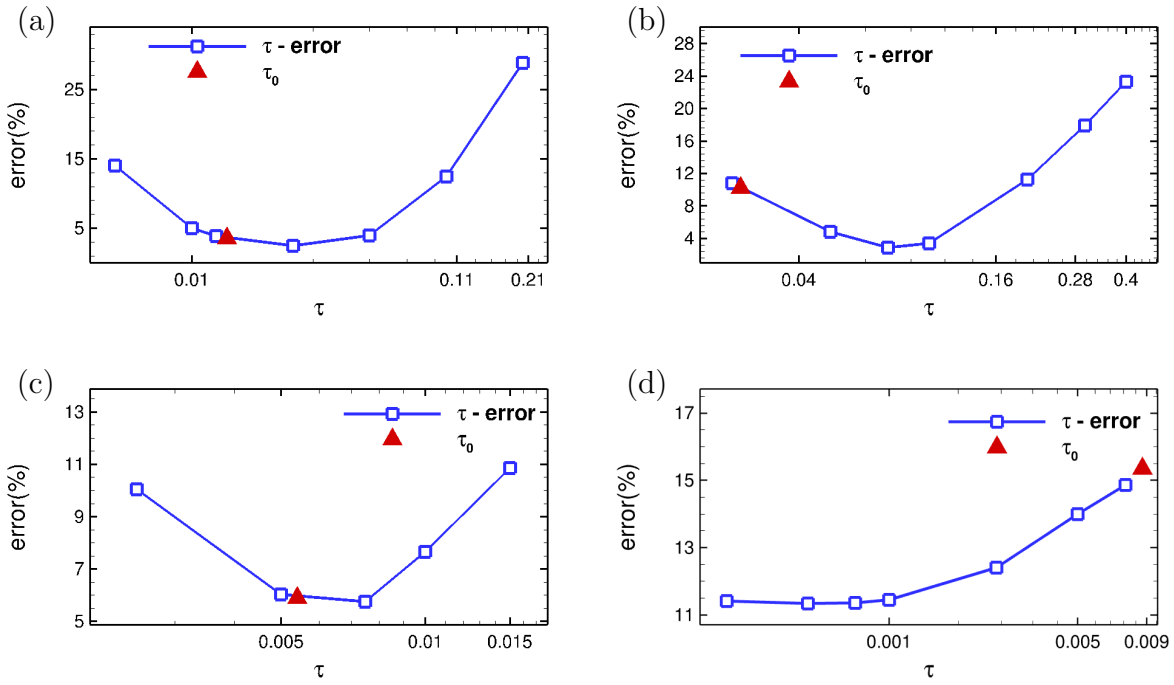

Figure 2.23: *ROM sensitivity to the time scale τ used in: (a) the flow past 2D oscillating cylinder; (b) the flow past 3D oscillating sphere; (c) the flow past pitching-up airfoil; (d) the flow past rotating elliptic airfoil. x axes are in logarithmic scale. τ_0 is given by Eq. 2.16.*

then grows, though specifically for the rotating elliptic airfoil case, only slight decrease of the error was observed. Therefore, τ_{opt} can be obtained to minimize the error, which is presented in table 2.2. It is found that for each case, τ_{opt} obtained from numerical experiments has the same order of magnitude as the associated τ_0, except for the rotating elliptic airfoil case, where τ_0 is an order higher of τ_{opt}. The comparison shows that though the criterion by

equation 2.16 is preliminary, it may still provide a reasonable and handy guidance to seek a proper τ for ROM, instead of going through a tedious trial-and-error process. However, more rigorous criteria are needed to determine a more accurate estimation of τ_0, considering the present criterion performs not so well for the rotating elliptic airfoil case.

2.4 Concluding remarks

In this chapter, a global POD-Galerkin projection for model order reduction was proposed to develop ROMs for complex fluid flows with moving solid structures. The method was based on globally defined POD modes in the combined fluid-solid domain and the projection of a modified Navier-Stokes equation simultaneously describing fluid and solid in the same combined domain. The global definition allowed for convenient application of POD-Galerkin projection in a fixed fluid-solid domain and avoided using complex and often expensive techniques, such as unsteady mapping functions, in its handling of a morphing fluid domain. To implement the impact from solid motion to the global fluid-solid ROM, two different descriptions for solid domains led to two types of global POD-Galerkin ROMs, namely CSDM and DSDM ROMs. CSDM ROM implemented the solid motion by defining its impact as an integration in the solid domain and therefore must be updated at each time step as the solid domain morphing continuously; DSDM ROM further decoupled the solid motion to the motion of discrete modes (i.e. shape functions) to replace the expensive integration with a few dynamic equations of these solid modes and speed up the computation to a real-time fashion at the cost of reduced accuracy in boundary resolution.

The global ROM methodology was applied to build ROMs from two numerical simulation datasets and one from experimental data. For all cases the use of 20 global POD modes and additional 20 solid modes (for DSDM only) allows ROMs to accurately capture basic dynamics. The accuracy of the prediction by both type of ROM equations was demonstrated by its close match with the original DNS and experimental data. This accuracy was demonstrated using the evolution of leading modes, reconstructed flow field, and key aerodynamic measures such as lift and drag. Though CSDM ROM provides a simplified dynamic equation for

analysis and also reduces the original cost significantly, its online computational cost remains too high for applications requiring real-time responses. On the other hand, DSDM ROM, by moving some computational cost to offline and at the expense of reduced boundary resolution, achieves real-time computational capability while retaining an accurate description of key dynamic features.

The sensitivity of the error of DSDM ROM to the number of POD modes N, the number of solid modes M, and the time scale τ, was studied. For both DNS cases, the error was found insensitive to N, which was lower than 4% for all N when M was fixed at 20. For the experimental cases, the lowest error was less than 6% when $N = 20$ for the pitching-up NACA0012 airfoil case, while a lowest error of 9% was achieved for the rotating elliptic airfoil case . M was found to have more impact on the error, for increasing M from 10 to 20 can significantly reduce the error all four cases. Additionally, it is found that an optimal τ exists to minimize the error for all cases, which can be preliminarily estimated priorly by the CFL criterion.

Chapter 3

Fast flow optimization with adjoint-ROM approach

3.1 Introduction

Moving body actuated flows are ubiquitous in the nature, which help natural flyers or swimmers to attain better performance of flying. Among these motions, the pitching-flapping-wing motion has drawn the interests of fluid mechanics community for a long time[85], since it effectively promotes the aerodynamic efficiency and maneuverability of many birds and insects. Although some understanding of its mechanism has been achieved by high-fidelity direct numerical simulations (DNS)[16;18;86–88] or well-designed experiments[89;90], the huge parameter space of this problem often prevents further physical understanding and optimization through a direct parametric study, let alone the real-time control. Many efforts have been made to reduce the complexity of the physical model or the dimension of the parameter space, such as the works by Berman and Wang[91], Trizila et al.[92], and Ghommem et al.[93], which limited the dimension of the control parameters. However, the computational cost for these optimal controls remained high, even when surrogate models were used. Gradient-based approaches have been applied to optimization[94;95], but the process was still computationally expensive due to the way the gradient was calculated.

Contrary to the aforementioned methods, adjoint-based methods pave a way to tackle a large number of control parameters simultaneously with computational cost not significantly scaling up[38;96]. The fundamentals of the conventional adjoint-based flow optimization was detailed in chapter 1. However, for the most problems in the present work, there exist moving solid boundaries in the flow domain, therefore the conventional adjoint approach which is derived on a fixed flow domain cannot be applied directly. Nadarajah and Jameson[97] used a domain transformation to map the physical domain with a moving solid boundary to a computational domain with only fixed boundary, to optimize the shape of a plunging airfoil for better aerodynamic performance with conventional adjoint-based approach. However, the mapping function would greatly increase the complexity as well as the computational cost of the optimization, which made it impossible for controls on moving solid trajectory or dynamic morphing solid boundaries. To avoid the difficulty of handling the domain mapping, non-cylindrical calculus has been leveraged as a powerful tool to deal with the difficulty of moving solid boundaries, and has been successfully applied to the optimization of heat transfer and shape control with FSI[98;99]. Recently, the author's research group has developed non-cylindrical calculus enabled adjoint approach to optimize the aerodynamic force and the power efficiency of flows over a flapping wing, which has shown its effectiveness on the optimization of flows with complicated FSI and morphing solid boundaries[80;88;100;101].

Despite the fact that the adjoint-based method is able to reduce the computational cost for large control spaces, numerical simulation is still required for most studies, which can still be computationally expensive, especially for 3D flows. Therefore, the real-time control for 3D flows is still not feasible with the current adjoint methods. A surrogate model might be developed to mitigate these challenges, which has lower dimension but can adequately capture the essential physics of the original flow system. Then the optimization can be conducted on this surrogate model instead of the original high-dimensional full-order model (FOM). An appropriate surrogate model would potentially enable real-time flow control due to the significant reduction in computational costs. Proper orthogonal decomposition (POD)-Galerkin projection based reduced-order model (ROM) is one of the most popular surrogate models, and has been successful when applied to numerous flows[52;54;58;102;103]. Recently, a

global POD-Galerkin projection based ROM has been developed to handle flows with moving solid boundaries[62–64;78;104–106]. This ROM was shown to be reasonably successful in capturing the dynamics of different 2D and 3D DNS and experimental databases, as comprehensively introduced in chapter 2. Hence it is a natural choice for the current study on the optimal control of flows with actuations from moving solid boundaries.

Some previous researchers also studied using ROMs for flow control. Noack et al.[66] and Tadmor and Noack[67] proposed an energy-based control scheme to control the vortex shedding behind a circular cylinder with a four-dimensional Galerkin ROM. The control approach yielded the optimal parametric amplitude for the smallest fluctuation energy, but had a very narrow region of validity in the control parameter space. Tallet et al.[107], Oulghelou and Allery[108], and Oulghelou and Allery[109] proposed a POD-ROM based flow control approach with adjoint method, to study the temperature control of the heated lid-driven cavity flow. Model adaptation methods, including Lagrangian interpolation on the mean fields (IMF) and modal interpolation based on properties of the tangent subspace of the Grassmann manifold (ITSGM), were embedded in the control process to enhance the robustness of the approach. ROM-based optimal control was also performed on driven cavity flows and backward-facing-step channel flows by Ito and Ravindran[110], Ravindran[111], and Ravindran[112]. A simple ROM based on the reduced-basis method with Hermite and Lagrange function was used in Ito and Ravindran[110], then later it was extended to POD-Galerkin ROM in Ravindran[111] and Ravindran[112]. The control approach was demonstrated to be feasible for the cases with boundary surface movement introduced as control. Despite the success in controlling specific flows in the above works, these methods were still limited to a fixed-domain framework. In the works by Bergmann et al.[113] and Bergmann and Cordier[114], the adjoint method was used for the optimal control of the 2D cylinder wake based on a POD-Galerkin ROM. The control was introduced by a azimuthal oscillation of the cylinder. By implementing the trust-region proper orthogonal decomposition (TRPOD) approach, the mean drag in those works was reduced by about 30% through the optimal control. However, this approach required extra kinds of modes, including the steady base flow, the rotary actuation mode, and shift modes, to construct the ROM. In addition, in spite of a rotating solid body involved in the study,

the whole computational domain still remained fixed. As a result, it is hard to extend the method to other control problems where the controls are imposed by arbitrary solid motions.

In the present work, a new adjoint based approach using the global POD-Galerkin ROM was developed for the optimal control of the flows actuated by moving solid boundaries. One advantage of the present approach is its flexibility of being able to tackle arbitrary actuations introduced by solid motions, thus greatly broadening the scope of applications, for instance, to study the flows of natural flyers and swimmers. Moreover, leveraging ROM as a surrogate model can drastically reduce computational time compared to DNS-base methods, which can possibly enable the real-time or near-real-time flow control. To guarantee the accuracy and robustness of ROMs used in the optimization where the flow condition may change, different strategies were designed to select the proper ROM for the control. Then the adjoint-ROM approach was applied to the optimal control of flows past moving 2D cylinders and oscillating 2D airfoils. The rest of this section is arranged in the following manner. Section 3.2 first reviews the derivation of non-cylindrical calculus enabled adjoint approach, then it details the comprehensive derivation of the adjoint-ROM approach for the fast optimal flow control, followed by the introduction of different strategies to select ROMs for the optimization. The applications of adjoint-ROM approach are introduced in section 3.3 on flows past moving cylinders and airfoils. In the end the concluding remarks are made in section 3.4.

3.2 Methodology

First the non-cylindrical calculus enabled adjoint approach is introduced for full-order model (FOM) based flow optimization. Then the adjoint-ROM approach developed in the present work is derived for fast flow optimization with moving solid boundaries. At last, different strategies to select ROMs for better accuracy of the surrogate model during the optimization is introduced.

3.2.1 Adjoint approach for moving solid boundaries

As depicted in figure 3.1 (a), when there is no motion nor deformation on the solid body in a domain, the flow control usually introduced by changing the flow velocity at the boundary, for example, the suction and blowing imposed in Bewley et al.[38], and the conventional adjoint-based approach can succeed in this scenario. When the control is introduced by moving or translating the solid body as shown in figure 3.1 (b), non-cylindrical calculus can avoid mapping the whole physical domain, but only transform the moving solid boundaries by using the transverse map velocity function $\boldsymbol{Z}$, which dramatically simplifies the adjoint formulation and the computational cost.

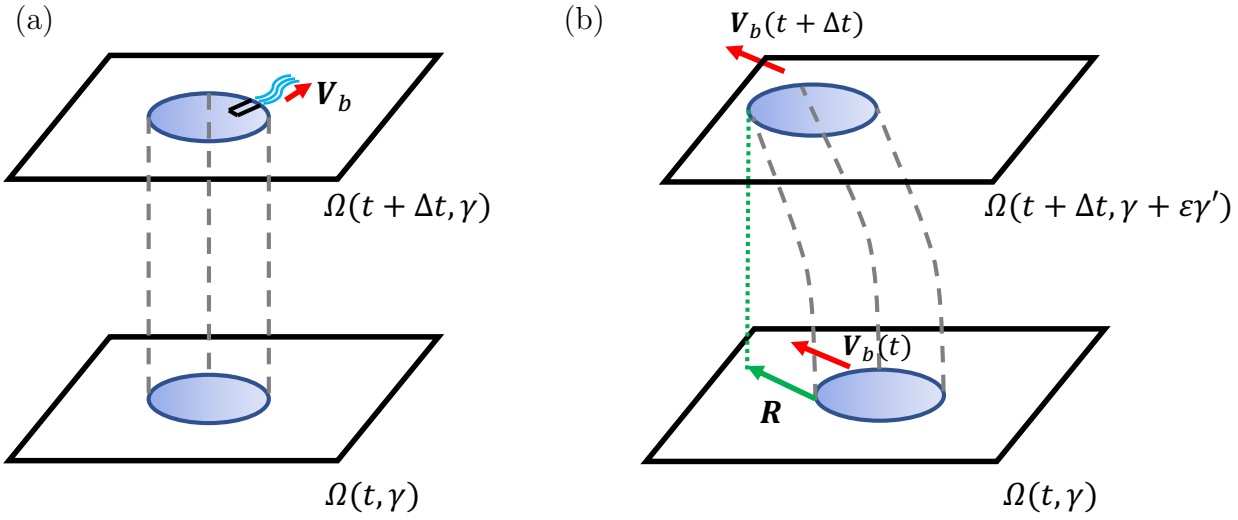

Figure 3.1: *The sketches of domain mapping for: (a) fixed solid boundary with control introduced by changing the velocity at boundaries; (b) moving solid boundary with control introduced by solid velocity and translation.*

A boundary-to-boundary mapping $\mathcal{T}$ is defined to describe the time evolution of the physical domain Ω with control γ:

$$\mathcal{T}(t,\gamma) : \Omega(t,\gamma) \rightarrow \Omega(t+\Delta t,\gamma). \tag{3.1}$$

As shown in figure 3.1 (b), the domain Ω evolves in time because of the motion of solid boundaries with moving velocity $\boldsymbol{V}(t,\gamma,\boldsymbol{x})$, therefore it is reasonable to define $\boldsymbol{V}$ as the

first-order derivative of the mapping function $\mathcal{T}$ with respect to the time variance Δt:

$$\boldsymbol{V}(t,\gamma,\boldsymbol{x}) = \frac{\partial \mathcal{T}(t,\gamma,\boldsymbol{x})}{\partial \Delta t}|_{\Delta t=0}. \tag{3.2}$$

Meanwhile, a transverse mapping $\tilde{\mathcal{T}}$ which maps the domain Ω with original control γ to the domain with perturbed control $(\gamma + \epsilon\gamma')$ at the same time moment:

$$\tilde{\mathcal{T}}(t,\gamma,\boldsymbol{x}) : \Omega(t,\gamma) \to \Omega(t,\gamma+\epsilon\gamma'). \tag{3.3}$$

Then the transverse mapping velocity $\boldsymbol{Z}$ can be defined similarly with respect to ϵ:

$$\boldsymbol{Z}(t,\gamma,\boldsymbol{x}) = \frac{\partial \tilde{\mathcal{T}}(t,\gamma,\boldsymbol{x})}{\partial \epsilon}|_{\epsilon=0} \tag{3.4}$$

Given a general continuous function f, the non-cylindrical material derivative in a Lagrangian framework $\dot{f}$ is defined by:

$$\dot{f}(t,\boldsymbol{x}) = \lim_{\epsilon\to 0} \frac{f(t,\gamma+\epsilon\gamma',\tilde{\mathcal{T}}) - f(t,\gamma,\boldsymbol{x})}{\epsilon}, \tag{3.5}$$

and the non-cylindrical shape derivative in an Eulerian framework f' is related to $\dot{f}$ by:

$$f' = \dot{f} - \boldsymbol{Z}\cdot\nabla f. \tag{3.6}$$

With the non-cylindrical material derivative, the relation between the transverse map velocity $\boldsymbol{Z}$ and the solid boundary velocity $\boldsymbol{V}$ can be determined by:

$$\dot{\boldsymbol{V}} = \frac{d\boldsymbol{Z}}{dt}. \tag{3.7}$$

Now that the fundamentals of non-cylindrical calculus are given, the perturbed governing equations with moving solid boundaries based on equation 1.31 can be modified with new

boundary conditions:

$$\begin{aligned} \mathcal{N}'(\boldsymbol{q}') &= 0 \quad \text{in} \quad \Omega, \\ \boldsymbol{u}' &= \dot{\boldsymbol{V}} - \boldsymbol{Z} \cdot \nabla \boldsymbol{u} \quad \text{on} \quad \partial\Omega_s, \\ \dot{\boldsymbol{V}} &= \frac{d\boldsymbol{Z}}{dt} \quad \text{on} \quad \partial\Omega_s, \\ \boldsymbol{Z}|_{t=0} &= 0 \quad \text{on} \quad \partial\Omega_s, \end{aligned} \tag{3.8}$$

where Ω_s is the solid domain. For the purpose of demonstration, the same objective function as equation 1.28 is used again here. By introducing the same adjoint variables $\boldsymbol{q}^*$, the perturbation of the enhanced objective function $\mathcal{J}'$ by using the non-cylindrical shape derivative is given as:

$$\mathcal{J}' = \int_0^T \int_\Omega \boldsymbol{u} \cdot \boldsymbol{u}' d\Omega dt + \int_0^T \int_\Omega \boldsymbol{q}^* \cdot \mathcal{N}'(\boldsymbol{q}') d\Omega dt + \int_0^T \int_{\partial\Omega_s} \boldsymbol{Z}^* \cdot (\frac{d\boldsymbol{Z}}{dt} - \dot{\boldsymbol{V}}) ds dt. \tag{3.9}$$

Similarly using integration by parts to group the perturbation terms, $\mathcal{J}'$ is simplified to have the same form as the equation 1.37 but different contents in each term, where B is updated as:

$$\begin{aligned} B = &\int_\Omega (u_j^* u_j') d\Omega|_0^T + B_\infty + \int_0^T \int_{\partial\Omega_s} u_i^* \sigma_{ij}' n_j ds dt \\ &- \int_0^T \int_{\partial\Omega_s} u_i' [\sigma_{ij}^* nj + u_j^* u_i n_i + u_i^* (u_j - V_j) n_j] ds dt \\ &+ \int_{\partial\Omega_s} Z_i Z^* ds|_0^T - \int_0^T \int_{\partial\Omega_s} (\frac{dZ_i^*}{dt} + Z_i^* \text{div}|_{\partial\Omega_s} \boldsymbol{V}) Z_i ds dt - \int_0^T \int_{\partial\Omega_s} \dot{V}_i Z_i^* ds dt, \end{aligned} \tag{3.10}$$

where B_∞ comprises all far-field terms, and tangential divergence of solid moving velocity $\text{div}|_{\partial\Omega_s} \boldsymbol{V}$ is defined by:

$$\text{div}|_{\partial\Omega_s} \boldsymbol{V} = \frac{\partial V_i}{\partial x_i}|_{\partial\Omega_s} - \frac{\partial V_i}{\partial x_i} n_j n_i. \tag{3.11}$$

$\mathcal{J}'$ can be further simplified if the following conditions are satisfied for the adjoint equations:

$$\begin{aligned} \mathcal{N}^*(\boldsymbol{q}^*) &= \mathcal{F}^*, \\ \boldsymbol{u}^* &= 0 \quad \text{on} \quad \partial\Omega_s, \\ \frac{dZ_i^*}{dt} + Z_i^* \text{div}|_{\partial\Omega_s} \boldsymbol{V} &= (\nabla \boldsymbol{u})^T \cdot \boldsymbol{\sigma} \cdot \boldsymbol{n} \quad \text{on} \quad \partial\Omega_s, \\ \boldsymbol{Z}^*|_{t=T} &= 0 \quad \text{on} \quad \partial\Omega_s. \end{aligned} \tag{3.12}$$

By solving the above adjoint equation, $\mathcal{J}'$ is simplified as:

$$\mathcal{J}' = -\int_0^T \int_{\partial\Omega_s} \dot{V}_i(\sigma_{ij}^* n_j + Z_i^*) ds dt. \tag{3.13}$$

Comparing this $\mathcal{J}'$ against equation 1.29, the gradient g_i is readily to be found as:

$$g_i = -(\sigma_{ij}^* n_j + Z_i^*). \tag{3.14}$$

Indeed the first term of the gradient is equivalent to equation 1.39, which is derived on a fixed domain. The second term is unique due to the moving solid boundary.

3.2.2 Adjoint-ROM approach

The full-order model of the incompressible flows are governed by the Navier-Stokes, equation 1.1. In order to resolve the moving solid boundary numerically on a fixed Cartesian grid, the discrete immersed boundary method (IBM) is used to represent the solid boundary conditions by adding a body force term to the right-hand side of Navier-Stokes equation. The same modified Navier-Stokes equation 1.3 as reviewed in chapter 1 is rewritten here for the readers' convenience:

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\nabla p + \frac{1}{Re}\nabla^2 \boldsymbol{u} + \boldsymbol{f}, \tag{3.15}$$

with the forcing term $\boldsymbol{f}$:

$$\boldsymbol{f} = \begin{cases} [(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} - \frac{1}{Re}\nabla^2\boldsymbol{u}]^n + \frac{1}{\tau}(\boldsymbol{V} - \boldsymbol{u}^n), & \text{in} \quad \Omega_s \\ 0, & \text{otherwise}, \end{cases} \tag{3.16}$$

where Ω_s is the domain occupied by solid body, and the superscript n denotes the discrete variables at n-th time step. By solving Navier-Stokes equation, high-fidelity solutions of the flow field can be obtained.

Using the database generated by DNS, a global POD-Galerkin projection approach is applied to construct a ROM on the combined domain for both fluid and solid, which is later leveraged as the surrogate model for optimal control. As introduced in chapter 2, global POD modes ($\boldsymbol{\phi}$) defined on the combined domain for fluid and solid can be generated by the method of snapshots. Only DSDM ROM is used in this chapter due to its extremely low online computational cost may allow for real-time or near-real-time flow control. For brevity, the DSDM ROM equations are rewritten as:

$$\begin{aligned} \frac{da_i}{dt} &= \sum_{j=0}^{N}(L_{ij} - \sum_{m=0}^{M} L^*_{ijm}b_m)a_j + \sum_{j=0}^{N}\sum_{k=0}^{N}(Q_{ijk} - \sum_{m=0}^{M} Q^*_{ijk}b_m)a_j a_k + \sum_{m=0}^{M} C^*_{im}b_m, \quad i = 1, 2, ..., N, \\ \frac{db_i}{dt} &= -\sum_{m=0}^{M} P^*_{im}b_m, \quad i = 1, 2, ..., M, \end{aligned} \tag{3.17}$$

where $P^*_{im} = \langle \boldsymbol{V}\cdot\nabla\psi_m, \psi_i\rangle$ represents the coefficients shown in equation 2.11. Here τ is still the time scale as introduced in chapter 2, of which the value is chosen to stabilize the numerical integration of ROMs and to minimize the error. $\boldsymbol{V}$ is the prescribed velocity of the rigid solid body, and will be specified for different flow configurations. This ROM is then used as a surrogate model which represents the essence of the flow dynamics of the original system (FOM), therefore the optimal solutions of the ROM should be approximately the same as of the associated FOM. However, ROMs are usually sensitive to changes in parameters, which would definitely happen during the control process. Therefore, different strategies to

select an appropriate ROM for the flow control loop will be designed later in section 3.2.3 to guarantee capability of ROMs to represent the original flow system when control parameters change.

Following previous notations, the adjoint-ROM approach is derived as follow. First, a well-defined objective function $\mathcal{J}$ is still needed as the target of flow control. Different from the equation 1.28 introduced in chapter 1, here a $\mathcal{J}$ is chosen to reach a prescribed velocity field $\boldsymbol{u}_0$ in a observation zone Ω_0 in the downstream:

$$\mathcal{J} = \int_{T_0}^{T_1} \int_{\Omega_0} |\boldsymbol{u} - \boldsymbol{u_0}|^2 d\Omega dt. \tag{3.18}$$

This objective function is a proper mathematic description for the stabilization of perturbed flows. The sensitivity of the objective function to the control $\boldsymbol{\gamma}$ can be computed by the perturbation of $\mathcal{J}$ subjected to a perturbation of $\boldsymbol{\gamma}$ with Fréchet differential (the same as equation 1.29):

$$\mathcal{J}' = \lim_{\epsilon \to 0} \frac{\mathcal{J}(\boldsymbol{\gamma} + \epsilon \boldsymbol{\gamma}') - \mathcal{J}(\boldsymbol{\gamma})}{\epsilon}. \tag{3.19}$$

When considering optimization, the general gradient function $g(\boldsymbol{\gamma})$ can be derived from the above sensitivity analysis as:

$$\mathcal{J}' = g(\boldsymbol{\gamma}) \cdot \boldsymbol{\gamma}'. \tag{3.20}$$

As previously noted, the state equations used in the present paper are the ROM equations as a surrogate model, and the control is introduced by solid motion, thus the control parameter $\boldsymbol{\gamma}$ is restricted to the velocity of solid motion $\boldsymbol{V}$. For a more concise expression,

the coefficients in equation 3.17 are further simplified as:

$$
\begin{aligned}
L'_{ij} &= L_{ij} - \sum_{m=0}^{M} L^*_{ijm} b_m \\
Q'_{ijk} &= Q_{ijk} - \sum_{m=0}^{M} Q^*_{ijk} b_m \\
C'_{im} &= C^*_{im} \\
P'_{im} &= P^*_{im}.
\end{aligned}
\tag{3.21}
$$

It is worth noting that hereafter the limits of summations are from 0 to N when the subscript is j or k; the limits are from 0 to M when the subscript is m by default. Let the ROM equation be $\boldsymbol{\mathcal{F}}_i(\boldsymbol{q}_i) - \boldsymbol{\mathcal{R}}_i = 0$, where

$$
\boldsymbol{\mathcal{F}}_i(\boldsymbol{q}_i) = \begin{bmatrix} \dot{a}_i - \sum_j L'_{ij} a_j - \sum_j \sum_k Q'_{ijk} a_j a_k - \sum_m C'_{im} b_m \\ \\ \dot{b}_i + \sum_m P'_{mi} b_m \end{bmatrix},
\tag{3.22}
$$

and $\boldsymbol{\mathcal{R}}_i = 0$. Here $\boldsymbol{q} = [a_i \quad b_i]^T$ are the state variables of ROM. Then the perturbed ROM equations $\boldsymbol{\mathcal{F}}'_i(\boldsymbol{q}'_i) - \boldsymbol{\mathcal{R}}'_i = 0$ can be obtained:

$$
\begin{aligned}
\dot{a}'_i = &\sum_j L'_{ij} a'_j + \sum_j \sum_k (Q'_{ijk} a_j a'_k + Q'_{ijk} a'_j a_k) - \sum_j \sum_m L^*_{ijm} a_j b'_m \\
&- \sum_j \sum_k \sum_m Q^*_{ijkm} a_j a_k b'_m + \sum_m \langle \frac{1}{\tau} \boldsymbol{\gamma}, \psi_m \boldsymbol{\phi}_i \rangle b'_m + \sum_m \langle \frac{1}{\tau} \boldsymbol{\gamma}', \psi_m \boldsymbol{\phi}_i \rangle b_m \\
\dot{b}'_i = &- \sum_m (\langle \boldsymbol{\gamma}' \cdot \nabla \psi_m, \psi_i \rangle b_m - \langle \boldsymbol{\gamma} \cdot \nabla \psi_m, \psi_i \rangle b'_m)
\end{aligned}
\tag{3.23}
$$

The adjoint variables $\boldsymbol{q}^*_i = [a^*_i \quad b^*_i]^T$ can be introduced as Lagrange multipliers in the same fashion as equation 1.33 to impose the state equations, so that the shape derivative of

the objective function, which is subjected to the constraints of state equations, is:

$$\mathcal{J}' = \int_T \int_{\Omega_0} 2(\boldsymbol{u} - \boldsymbol{u}_0) \cdot \boldsymbol{u}' dt d\Omega + \sum_i \int_T \boldsymbol{q}_i^* \cdot [\boldsymbol{\mathcal{F}}_i'(\boldsymbol{q}_i') - \boldsymbol{\mathcal{R}}_i'] dt. \tag{3.24}$$

In the above equation, the target velocity $\boldsymbol{u}_0$ in the observation zone Ω_0 is obtained from the snapshots by which the POD modes are generated. $\boldsymbol{u}$ is the instantaneous velocity of the controlled flow, which is given by POD-ROM reconstruction. Note that the ranges of i in the summation of the second term of equation 3.24 are from 1 to N for a_i^* related terms and from from 1 to M for b_i^* related terms respectively.

Using integration by parts, then separating the perturbation terms, we can have:

$$\mathcal{J}' = B - \sum_i \int_T \boldsymbol{q}_i' \cdot [\boldsymbol{\mathcal{F}}_i^*(\boldsymbol{q}_i^*) - \boldsymbol{\mathcal{R}}_i^*] dt, \tag{3.25}$$

where

$$\begin{aligned}
\boldsymbol{\mathcal{F}}_i^*(\boldsymbol{q}_i^*) &= \begin{bmatrix} \frac{da_i^*}{dt} + \sum_j [L'_{ji} + \sum_k (Q'_{jik} + Q'_{jki}) a_k] a_j^* \\ \\ \frac{db_i^*}{dt} + \sum_j [C'_{ji} - \sum_k (L^*_{jki} + \sum_m Q^*_{jkmi} a_m) a_k] a_j^* - \sum_m P'_{mi} b_m^* \end{bmatrix}, \\
\boldsymbol{\mathcal{R}}_i^* &= \begin{bmatrix} 2(\sum_j K_{ji} a_j - \sum_j M_{ji} a_{0j}) \\ 0 \end{bmatrix},
\end{aligned} \tag{3.26}$$

New coefficients K_{ij} and M_{ij} can be obtained by area integration of POD modes over Ω_s:

$$\begin{aligned}
K_{ij} &= \int_{\Omega_0} \boldsymbol{\phi}_i \cdot \boldsymbol{\phi}_j d\Omega, \\
M_{ij} &= \int_{\Omega_0} \boldsymbol{\phi}_{0i} \cdot \boldsymbol{\phi}_j d\Omega,
\end{aligned} \tag{3.27}$$

where $\boldsymbol{\phi}_0$ are the POD modes of the target flow and $\boldsymbol{\phi}$ the POD modes of the controlled flow. For simplicity during optimization, the control parameters are chosen in a way such that the POD modes will not change drastically when control is imposed, thus $\boldsymbol{\phi} = \boldsymbol{\phi}_0$.

The equations

$$\boldsymbol{\mathcal{F}}_i^*(\boldsymbol{q}_i^*) - \boldsymbol{\mathcal{R}}_i^* = 0 \tag{3.28}$$

are the adjoint equations. With those equations being satisfied, the only term left in $\mathcal{J}'$ (equation 3.25) is B:

$$B = \sum_{i=1}^{N}[a_i' a_i^*|_{T_0}^{T_1} - \frac{1}{\tau}\int_{T_0}^{T_1}\sum_m b_m\langle\boldsymbol{\gamma}', (\psi_m\boldsymbol{\phi}_i)\rangle a_i^* dt] + \sum_{i=1}^{M}[b_i' b_i^*|_{T_0}^{T_1} + \int_{T_0}^{T_1}\sum_m\langle\boldsymbol{\gamma}'\cdot\nabla\psi_m, \psi_i\rangle b_m b_i^* dt]. \tag{3.29}$$

It can be noticed that equation 3.29 can be further simplified if the following conditions are imposed:

$$\begin{aligned}
\boldsymbol{a}^*(T_0) &= 0,\\
\boldsymbol{b}^*(T_0) &= 0,\\
\boldsymbol{a}'(T_1) &= 0,\\
\boldsymbol{b}'(T_1) &= 0.
\end{aligned} \tag{3.30}$$

Equation 3.28 with initial and final conditions defines the adjoint equation, where $t = T_1$ is regarded as the 'initial' time, since the adjoint system typically evolves backwards in time. With the definition of the adjoint formulations, equation 3.29 can be further reduced, and by comparing to equation 3.20, $g(\boldsymbol{\gamma})$ can be eventually derived. However, for different flow configuration, the form of $g(\boldsymbol{\gamma})$ varies accordingly, which will be presented later when the application to different flows are discussed.

The optimal control problem can be solved iteratively. Here $g(\boldsymbol{\gamma})$ is a function of both $\boldsymbol{q}$ and $\boldsymbol{q}^*$. Therefore in each iteration, $\boldsymbol{q}$ is obtained first by solving the state equations (equation 3.17), after that objective function is updated. Then the adjoint equation (3.28) is solved backward in time, which generates the adjoint information $\boldsymbol{q}^*$. By evaluating equation 3.29, the gradient function is obtained. Then the control parameters can be updated iteratively by equation 1.40. It is worth noting that the above derivation is not the only way to obtain adjoint-ROM equations, for example, an alternative derivation is presented in

appendix C.

Adjoint-ROM formulations for force control

Adjoint-ROM formulations need modifications when the objective function is the aerodynamic forces on the moving solid body. Similar to the application in chapter 2, the aerodynamic forces is obtained solely by the velocity fields:

$$\boldsymbol{F} = -\frac{d}{dt}\int_{\Omega} \boldsymbol{u} dV + \oint_{\partial\Omega} \hat{\boldsymbol{n}} \cdot \boldsymbol{\Gamma} dS - \oint_{\partial\Omega_s} \hat{\boldsymbol{n}} \cdot (\boldsymbol{u}\boldsymbol{u}) dS, \tag{3.31}$$

It is worth noting that, for the applications studied in this paper, the term $-\oint_{\partial\Omega_s} \hat{\boldsymbol{n}} \cdot (\boldsymbol{u}\boldsymbol{u}) dS$ in equation 3.31 is negligible, which will be shown later in the case study. So the following derivations will only use the first two terms in equation 3.31. After normalization, the forces can be rewritten by force coefficient $\boldsymbol{C}_F = [C_D, C_L]$, of which the two components are drag and lift coefficients.

Now the objective function can be defined to represent the mean forces in a time window $[0, T]$:

$$\mathcal{J} = \frac{1}{T}\int_0^T \boldsymbol{C}_F dt. \tag{3.32}$$

With velocity decomposition given by global POD, $\mathcal{J}$ can be rewritten as a function of POD modes $\boldsymbol{\phi}_i$ and t:

$$\mathcal{J} = \frac{1}{T}\int_0^T \sum_{i=0} \{\frac{da_i}{dt}[\hat{C}_{F1}(\boldsymbol{\phi}_i) + \hat{C}_{F2}(\boldsymbol{\phi}_i, t)] + a_i \hat{C}_{F3}(\boldsymbol{\phi}_i) + \frac{1}{2}\sum_{j=0} a_i a_j \boldsymbol{S}_{ij}\} dt, \tag{3.33}$$

where

$$\hat{C}_{F1}(\boldsymbol{\phi}_i) = \oint_{\partial\Omega} -[(\boldsymbol{x} \cdot \boldsymbol{\phi}_i)\boldsymbol{I} + \boldsymbol{x}\boldsymbol{\phi}_i] \cdot \hat{\boldsymbol{n}} dS, \tag{3.34}$$

$$\hat{C}_{F2}(\boldsymbol{\phi}_i, t) = -\int_{\Omega_s(t)} \boldsymbol{\phi}_i dV, \tag{3.35}$$

$$\hat{C}_{F3}(\boldsymbol{\phi}_i) = \oint_{\partial\Omega} \{\frac{1}{Re}[(\boldsymbol{x} \cdot \nabla^2\boldsymbol{\phi}_i)\boldsymbol{I} - \boldsymbol{x}\nabla^2\boldsymbol{\phi}_i] + \frac{1}{Re}(\nabla\boldsymbol{\phi}_i + \nabla\boldsymbol{\phi}_i^T)\} \cdot \hat{\boldsymbol{n}} dS \tag{3.36}$$

and

$$\boldsymbol{S}_{ij} = \oint_{\partial\Omega} [(\boldsymbol{\phi}_i \cdot \boldsymbol{\phi}_j)\boldsymbol{I} - 2\boldsymbol{\phi}_i\boldsymbol{\phi}_j - 2\boldsymbol{\phi}_i\boldsymbol{\psi}_j] \cdot \hat{\boldsymbol{n}} dS, \tag{3.37}$$

with $\boldsymbol{\psi}_i = \boldsymbol{x} \times (\nabla \times \boldsymbol{\phi}_i)$. The reconstruction of velocity field at each time step to solve equation 3.31, which is very time-consuming, can be completely avoided by using this objective function. Instead, all spatially dependent coefficients in equation 3.33 except $\hat{C}_{F2}$ are time-independent, which can be calculated offline. Therefore, the computational cost can be greatly reduced by applying above transformation.

With this $\mathcal{J}$, the adjoint-ROM equations can be derived as:

$$\boldsymbol{\mathcal{F}}_i^*(\boldsymbol{q}_i^*) = \boldsymbol{\mathcal{R}}_i^*, \tag{3.38}$$

where

$$\begin{aligned} \boldsymbol{\mathcal{F}}_i^*(\boldsymbol{q}_i^*) &= \begin{bmatrix} \frac{da_i^*}{dt} + \sum_j [L'_{ji} + \sum_k (Q'_{jik} + Q'_{jki}) a_k] a_j^* \\ \\ \frac{db_i^*}{dt} + \sum_j [C'_{ji} - \sum_k (L^*_{jki} + \sum_m Q^*_{jkmi} a_m) a_k] a_j^* - \sum_m P'_{mi} b_m^* \end{bmatrix}, \\ \boldsymbol{\mathcal{R}}_i^* &= \begin{bmatrix} \frac{1}{2T} \sum_{j=0} a_j (\boldsymbol{S}_{ij} + \boldsymbol{S}_{ji}) - \frac{1}{T} \frac{\partial \hat{C}_{F2}}{\partial t} + \frac{1}{T} \hat{C}_{F3} \\ 0 \end{bmatrix}. \end{aligned} \tag{3.39}$$

It can be seen that the only difference is in $\mathcal{R}_i^*$ compared to equation 3.26.

Benchmark on a simple ODE system

The newly developed adjoint-ROM approach is worth validating on simple ordinary differential equations (ODEs), since ROM is also a group of ODEs. Key criteria to assess the adjoint approach include the accuracy of the optimal solution, as well as the accuracy of gradient. Here the approach was tested on the heat conduction problem in an object immersed in constant ambient temperature, with its spatial dimension being neglected. The problem is

governed by Newton's cooling law about the temperature T:

$$\dot{T} = k(T_r - T), \quad \text{with} \quad \mathrm{T}|_{\tau=0} = \mathrm{T}_0, \tag{3.40}$$

where k is the heat conductivity and T_r the room temperature. τ represents the time. This equation has an analytical solution when initial condition is properly given:

$$T(t) = T_r + (T_0 - T_r)e^{-k\tau}. \tag{3.41}$$

With $k = 2$, $T_r = 25$ and $T_0 = 40$, the numerical result and analytical one are compared in figure 3.2 to show the accuracy of the numerical solution.

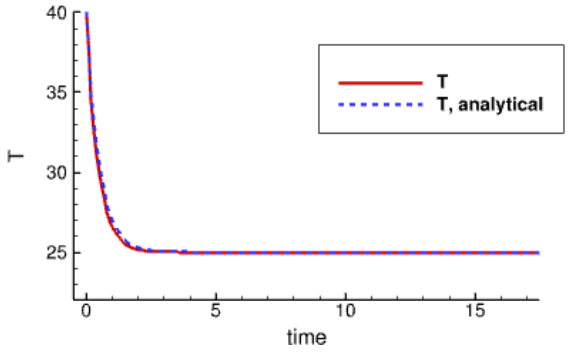

Figure 3.2: *The solution of Newton's cooling equation.*

Then optimal control of this equation is conducted by the adjoint approach. The room temperature T_r is the control parameter. The objective function $\mathcal{J}$ with a penalty term is designed as such that a local minimum can be achieved with $T_r \in [15, 70]$:

$$\begin{aligned} \mathcal{J} &= \frac{1}{\tau}\int_\tau (T^2 + p(T_r - T_{r0})^2)d\tau, \quad \text{when} \quad \mathrm{T_r} < \mathrm{T_{r0}}; \\ \mathcal{J} &= \frac{1}{\tau}\int_\tau T^2 d\tau, \quad \text{otherwise.} \end{aligned} \tag{3.42}$$

The variations of $\mathcal{J}$ with respect to T_r obtained by parametric study is shown in figure. 3.3. By introducing the Lagrangian multiplier T^*, a modified objective function $\mathcal{L}$ is generated

as:

$$\begin{aligned}\mathcal{L} &= \frac{1}{\tau}\int_{\tau}(T^2 + p(T_r - T_{r0})^2)d\tau + \int_{\tau} T^* \cdot [\dot{T} - k(T_r - T)]d\tau, \quad \text{when} \quad \mathrm{T_r} < \mathrm{T_{r0}};\\ \mathcal{L} &= \frac{1}{\tau}\int_{\tau} T^2 d\tau + \int_{\tau} T^* \cdot [\dot{T} - k(T_r - T)]d\tau, \quad \text{otherwise.}\end{aligned} \tag{3.43}$$

Let $\frac{\partial \mathcal{L}}{\partial T} = 0$ and by integrating by parts, the adjoint equation is given as:

$$\dot{T}^* - kT^* - \frac{2T}{\tau} = 0. \tag{3.44}$$

And the gradient g can be obtained by $g = \frac{\partial \mathcal{L}}{\partial T_r}$;

$$\begin{aligned}g &= -\int_{\tau} kT^* d\tau + \frac{2}{\tau}\int_{\tau} p(T_r - T_{r0})d\tau, \quad \text{when} \quad \mathrm{T_r} < \mathrm{T_{r0}};\\ g &= -\int_{\tau} kT^* d\tau, \quad \text{otherwise.}\end{aligned} \tag{3.45}$$

$T_{r0} = 30$ is used here. Then optimal solution is $T_{r,\mathrm{opt}} = 28.59$ which gives $\mathcal{J} = 873$, it matches the minimum found by parametric study as shown in figure 3.3. Lastly the gradients achieved by solving adjoint equation was compared against the benchmark given by finite difference method are shown in figure 3.4. Again, a good agreement is found which demonstrates the accuracy and effectiveness of the adjoint approach.

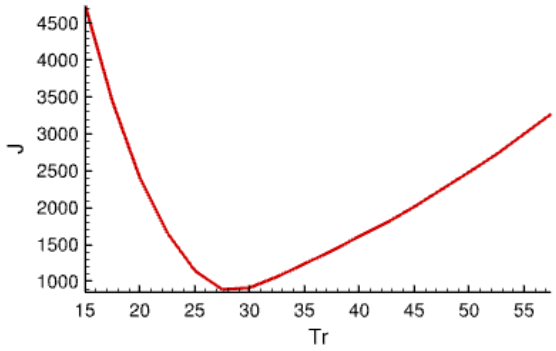

Figure 3.3: *The objective function versus T_r obtained by parametric study.*

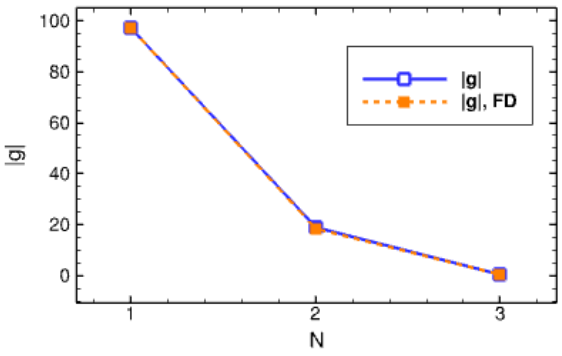

Figure 3.4: *The gradient at each iteration. Blue line is the gradient by solving adjoint equation, orange line is the benchmark of gradient obtained numerically by finite difference method.*

3.2.3 ROM selecting strategies

One of the biggest challenges for ROM-based optimization is keeping the accuracy of ROMs when control parameters alter. Unfortunately, there is arguably no rigorous mathematical assurance that the optimal solution given by ROM would always correspond to the local optimizer for the original high-fidelity system during the optimization process. Previously some efforts have been made to tackle this challenge[113;114]. In those studies, DNS had to be conducted to obtain new POD basis, once the old ones were not able to represent the physics of the original system as the flow was changed by control. By using this method, however, time-consuming DNS still has to be performed numerous times during the online process, which is not ideal for real-time flow control.

In the present work, two strategies have been designed to guarantee the accuracy of ROMs when control parameters change during the optimization, while keeping the online computational time low enough for possible real-time control. The first one is the "one-ROM" strategy, as illustrated in figure 3.5 (a). This strategy is used in cases where the control parameters do not change drastically so that there is no transition of flow regime taking place. A single ROM will be generated base upon the database of either the original flow or the perturbed flow, depending on the specific type of problems. This ROM is accurate to represent the flow dynamics of the whole parametric space studied (rendered by blue in figure 3.5 (a)), as long as the control parameters at each iteration are still located in this

domain. Nevertheless, *a posteriori* validation should be required after the optimization by comparing the FOM and ROM results of the optimal flow, to evaluate the robustness of the single ROM during the whole optimization process.

The second strategy is the "ROM-switching" strategy, as illustrated in figure 3.5 (b). It is designed for cases where the control parameters change too much to make single ROM valid for the whole parametric space. This strategy managed to move time-consuming DNS calculations from a cyclic calculation with updated ROMs to one database of pre-calculated ROMs. Practically, for any control $\boldsymbol{\gamma}$, the range of its possible values is often known before the optimization. Then in that range, M different $\boldsymbol{\gamma}^p \in \{\boldsymbol{\gamma}^1, \boldsymbol{\gamma}^2 ... \quad \boldsymbol{\gamma}^M\}$ can be chosen to fundamentally represent the parametric space. For each $\boldsymbol{\gamma}^p$, a DNS will be performed to generate snapshots for that specific control parameter. With POD-Galerkin projections, M ROMs in total will be built corresponding to these M controls (for example, in figure 3.5 (b), $M = 4$). Although the ROM is still locally sensitive to the change of control parameters, as long as $\boldsymbol{\gamma}$ changes in small increments within each iteration of the optimization, the adaptive ROM for this slightly altered $\boldsymbol{\gamma}$ can still be accurate. These M ROMs then form a library for the optimization. Consequently, during the online optimization process, no new DNS will be needed to guarantee ROMs for various control parameters. Instead, an existing ROM from the library will be selected whenever the control $\boldsymbol{\gamma}$ changes. The rule to select a proper ROM is given as such: for the current $\boldsymbol{\gamma}$, if the minimum of $|\boldsymbol{\gamma} - \boldsymbol{\gamma}^p|$ is achieved for a $\boldsymbol{\gamma}^p \in \{\boldsymbol{\gamma}^1, \boldsymbol{\gamma}^2 ... \quad \boldsymbol{\gamma}^M\}$, then the ROM based on $\boldsymbol{\gamma}^p$ will be used for the current iteration (for example, as shown in figure 3.5 (b), only the first and the forth ROM are used for the optimization). This rule guarantees that the ROM used at each iteration contains the physics that is closest to the original system based on the current $\boldsymbol{\gamma}$.

3.3 Results and discussion

The adjoint-ROM method was applied to the optimal control of 2D flows past an oscillating cylinder as well as a heaving-pitching NACA0012 airfoil. The control was imposed by changing the motion of solid body. For the oscillatory motion of the cylinder or the airfoil, a

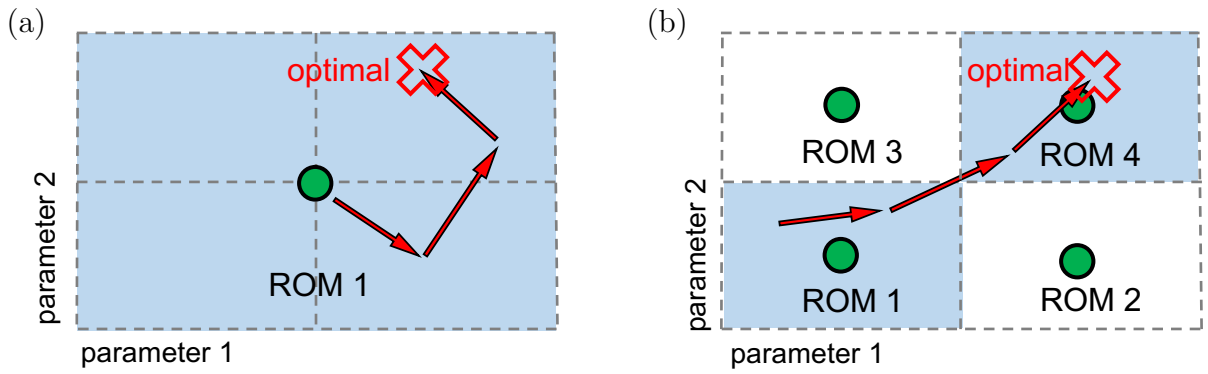

Figure 3.5: *The schematic illustration of two different strategies to select ROM during the optimization in a 2D parametric space: (a) one-ROM strategy, which uses a single ROM built upon the database of either the original flow or the perturbed flow; (b) ROM-switching strategy, which first generates several ROMs representing different control parameters in the parametric space offline before the optimization, then uses one ROM for the online iteration. Green dots show the control parameter at which the ROM is generated. Blue shade indicates that certain ROM is used for the optimization. Red arrows represent the change of control parameters at each iteration.*

group of universal harmonic functions are used to describe the possible oscillation in vertical or azimuthal direction:

$$\begin{aligned} Y(t) &= A\sin(2\pi ft), \\ \Theta(t) &= \theta\sin(2\pi ft+\phi), \end{aligned} \tag{3.46}$$

where A is the heaving amplitude, θ the pitching amplitude, and ϕ the phase delay angle between heaving and pitching motion. The number of control parameters was up to three for all cases studied in this section, with possible $\boldsymbol{\gamma} = [A, \theta, \phi]$. Two objectives were optimized. The first objective was to match a target flow field by minimizing the difference between velocity fields. The second objective was to optimize the aerodynamic force. Due to the different forms of the solid velocity for different configurations, the form of the gradient function was modified accordingly for different cases.

3.3.1 Optimal control of the flow past a 2D oscillatory cylinder

First the adjoint-ROM method was applied to the optimization of flow past a 2D oscillatory cylinder. Initially the cylinder was located at the origin of a 10×10 domain (normalized by the diameter of the cylinder D) discretized by a 601×401 non-uniform Cartesian mesh, which was denser in the area close to the cylinder. The Reynolds number based on D was $Re = 50$. The time t was normalized by the diameter of the cylinder D and incoming flow velocity U. Then the cylinder was oscillating in the vertical direction with a sinusoidal velocity based on equation 3.55:

$$\boldsymbol{V} = 2\pi f \gamma \sin(2\pi f t)\boldsymbol{e}_y. \tag{3.47}$$

The schematic of the flow configuration is briefly shown in figure 3.6 (a).

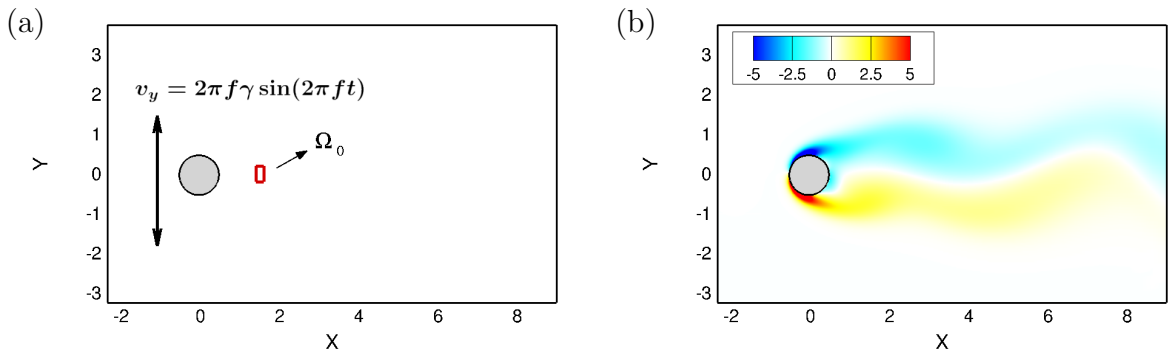

Figure 3.6: *(a): The schematic of the flow configurations, the red rectangular zone is Ω_0. (b): A snapshot of the target flow ($\gamma_t = 0.1$) contoured by the vorticity ω.*

The control parameter $\boldsymbol{\gamma}$ only had one degree of freedom which was the amplitude of the velocity A, thus for this case only γ was used interchangeable with A. The frequency $f = 0.2$ was fixed. The gradient with one degree of freedom $\boldsymbol{g}(\gamma)$ was thus given by:

$$\boldsymbol{g} = \int_T [\sum_i^N (-\frac{1}{\tau} \sum_m b_m \langle \boldsymbol{e}_y, \psi_m \boldsymbol{\Phi}_i \rangle a_i^*) + \sum_i^M (\sum_m \langle \boldsymbol{e}_y \cdot \nabla \psi_m, \psi_i \rangle b_m) b_i^*] \boldsymbol{V}_\gamma dt, \tag{3.48}$$

where $\boldsymbol{V}_\gamma$ is defined as:

$$\boldsymbol{V}_\gamma = 2\pi f \sin(2\pi f t), \tag{3.49}$$

by computing the shape derivative of $\boldsymbol{V}$.

The target flow was computed when $\gamma_t = 0.1$. As shown in figure 3.6 (b), there was a coherent wake flow in the downstream because of the very small amplitude of oscillation. Then POD was performed on 500 snapshots from dimensionless time $t = 500$ to $t = 520$. Then two variants of initial controls were chosen:

$$\gamma^{(0)} = 0.2, \tag{3.50}$$

and

$$\gamma^{(0)} = 0.5. \tag{3.51}$$

Twenty POD modes as well as solid modes were used to build the ROM. The time step used to solve ROM and adjoint equations was $\Delta t = 0.005$. Note that the adjoint equation usually has stricter stability requirement than the ROM equation, thus Δt is smaller than that used in the ROM solution.

The ROM results of the target flow and controlled flows are presented in figure 3.7 by the phase portraits of the time coefficients of the first two POD and solid modes. It can be seen that, with the initial control $\gamma^{(0)}$ the flow has quite different dynamics compared to the target flow. Through the control process by the oscillation of the cylinder, the flow dynamics should approach that of the target flow when the whole process was converged.

Then the optimal control was conducted to minimize the objective function given by equation 3.18 with "one-ROM" strategy. The observation zone Ω_0 was set in the downstream ($1.4 < x < 1.6$ and $-0.2 < y < 0.2$), which is sketched in figure 3.6 (a). In theory, since the objective function is designed to "match" the velocity of the target flow in Ω_0, once the objective function was converged, the value of control γ should be identical to the target one γ_i. Therefore the objective function chosen here can not only be used to stabilize a perturbed flow, but also provide a feasible way to validate the adjoint-ROM method.

As shown in figure 3.8 (a) and (b), when the initial control was set as $\gamma^{(0)} = 0.2$, $\mathcal{J}$ is about 10^{-1} at the beginning, then converges down to well below 10^{-10} after only three

iterations. As expected, the control parameter $\gamma^{(3)}$ after three iterations matches the target control parameter $\gamma_t = 0.1$. For $\gamma^{(0)} = 0.5$ case, the similar variations of $\mathcal{J}$ and γ can also be observed. As shown in figure 3.8 (c) and (d), after three iterations $\gamma^{(3)}$ converged to 0.1 with $\mathcal{J} < 10^{-10}$.

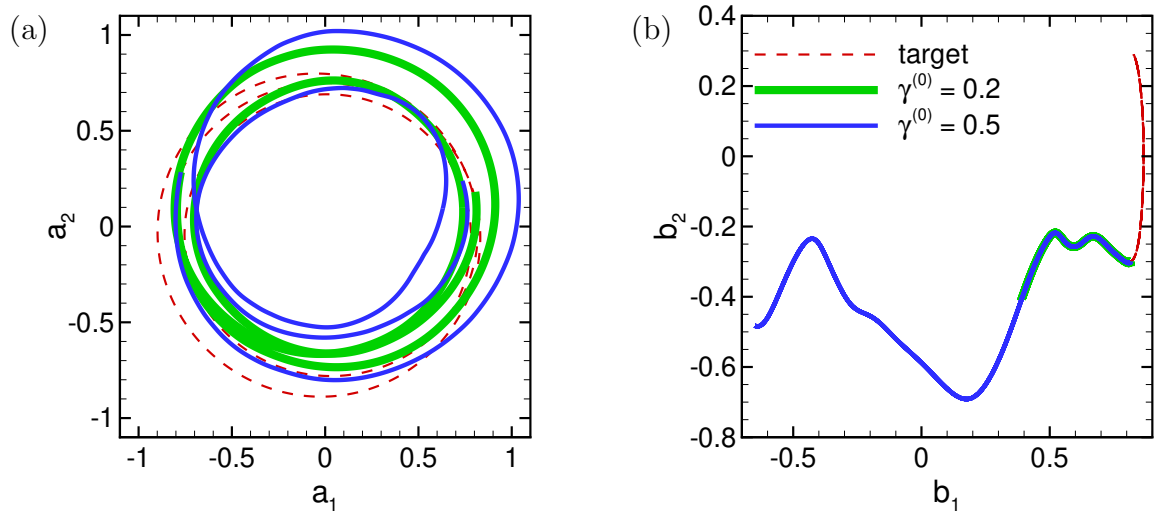

Figure 3.7: *The phase portraits of the time coefficients of the first two POD and solid modes for target flow with $\gamma_t = 0.1$ and controlled flow with initial controls $\gamma^{(0)} = 0.2$ and $\gamma^{(0)} = 0.5$, respectively. (a): a_1 versus a_2; (b): b_1 versus b_2.*

During the optimization process, the gradient of the objective function g is crucial to the accuracy of the approach. In order to validate the value of g given by solving the adjoint equations 3.28 and equation 3.48, the gradient computed by directly solving equation 3.19 with a finite difference method was also presented as a benchmark. The perturbation ϵ was set to 10^{-4}. The comparisons between the gradients are shown in figure 3.8 (e) for $\gamma^{(0)} = 0.2$ and in figure 3.8 (f) for $\gamma^{(0)} = 0.5$ respectively. It is seen that overall the two gradients are in very good agreement for both cases, which indicates the gradient g given by adjoint method was accurate enough for the optimization approach.

The advantage of adjoint-based methods is generally more apparent with a higher dimension parametric space. Here a second control parameter was added to the vertically oscillating cylinder case described above. Reynolds number was increased to $Re = 100$ for rich flow dynamics in the wake. The 2-DoF control was introduced to the oscillatory motion

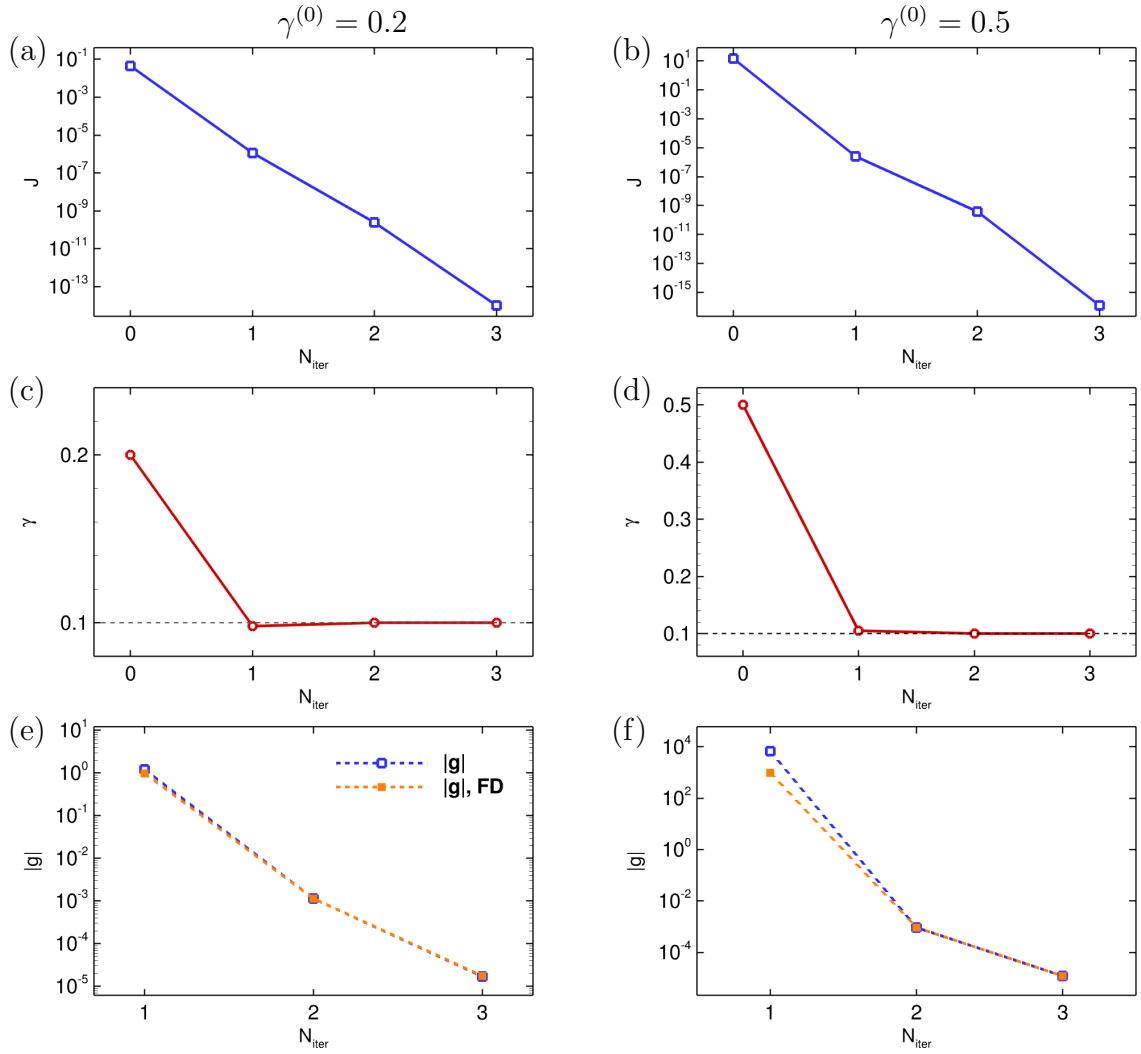

Figure 3.8: *Variation of the objective function $\mathcal{J}$ ((a) and (b)); variation of the control γ ((c) and (d)); and variation of the absolute value of the gradient g ((e) and (f)) with respect to the iteration number for $\gamma^{(0)} = 0.2$ (left) and $\gamma^{(0)} = 0.5$ (right). In (c) and (d), horizontal dashed lines show the target control parameter $\gamma_t = 0.1$. In (e) and (f), blue dashed lines with hollow squares are the gradients given by adjoint method, and orange dashed lines with solid squares are the gradients given by equation 3.19 numerically.*

with a velocity as

$$\boldsymbol{V} = \sum_{i=1}^{2} 2\pi f_i A_i \cos(2\pi f_i t)\boldsymbol{e}_y, \tag{3.52}$$

where the frequencies were fixed at $f_i = [0.1, \quad 0.2]$, and the two amplitudes A_i were optimized as a 2-degree-of-freedom (DoF) vector controller $\boldsymbol{\gamma} = [A_1, A_2]$. The corresponding

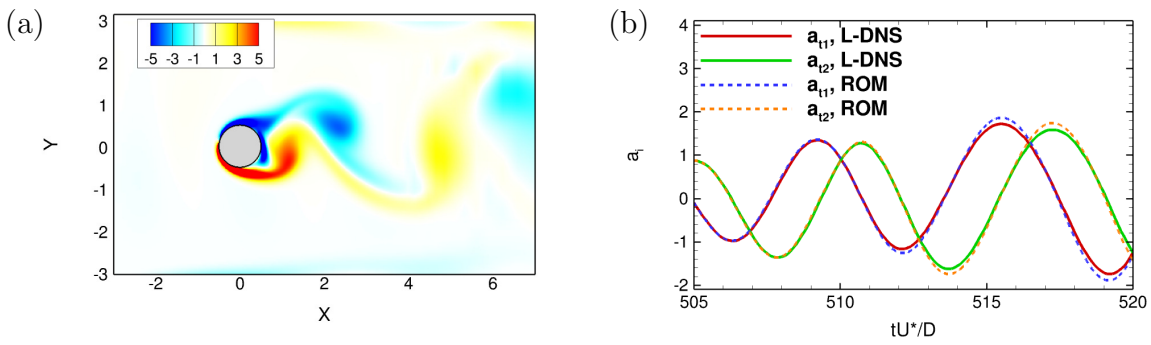

Figure 3.9: *(a): A snapshot of the two DoF control flow past a oscillatory cylinder contoured by the vorticity; (b) The time coefficients of the first two POD modes by the direct projection of the full-order model onto POD modes (L-DNS, solid lines) and by solving DSDM ROM equations (Eq. (3.17), dashed lines).*

gradient g_l for each control parameters γ_l is updated as:

$$\boldsymbol{g}_l = \int_T [\sum_i^N (-\frac{1}{\tau} \sum_m b_m \langle \boldsymbol{e}_y, \psi_m \boldsymbol{\Phi}_i \rangle a_i^*) + \sum_i^M (\sum_m \langle \boldsymbol{e}_y \cdot \nabla \psi_m, \psi_i \rangle b_m) b_i^*] \boldsymbol{V}_{\gamma l} dt, \quad l = 1, 2, \tag{3.53}$$

where

$$\boldsymbol{V}_{\gamma l} = -2\pi f_l \sin(2\pi f_l t), \quad l = 1, 2, \tag{3.54}$$

is the shape derivative corresponding to each control parameters.

The target flow was computed with $\boldsymbol{\gamma} = [0.1, \quad 0.2]$ which serves as the theoretical solution of the optimization. Figure 3.9 (a) shows a typical snapshot of the target flow. The target flow snapshots were used to compute POD modes and build a ROM for the optimization (from flows with off-target controls). POD modes were computed from a total of 151 snapshots between dimensionless time $t = 505.0$ to $t = 520.0$. Only 10 POD modes and another 10 solid motion modes were used in this case to build the ROM. As shown in figure 3.9 (b), the flow dynamics presented by the time coefficients of the first two POD modes was accurately captured by the 10-mode ROM in its comparison to the coefficients from direction projection of the full-order model (i.e. DNS data).

The same observation zone was used as in the earlier cases, and the objective function $\mathcal{J}$ was also the same as in equation 3.18. The initial control was chosen with $\boldsymbol{\gamma}^{(0)} = [0.25, 0.25]$.

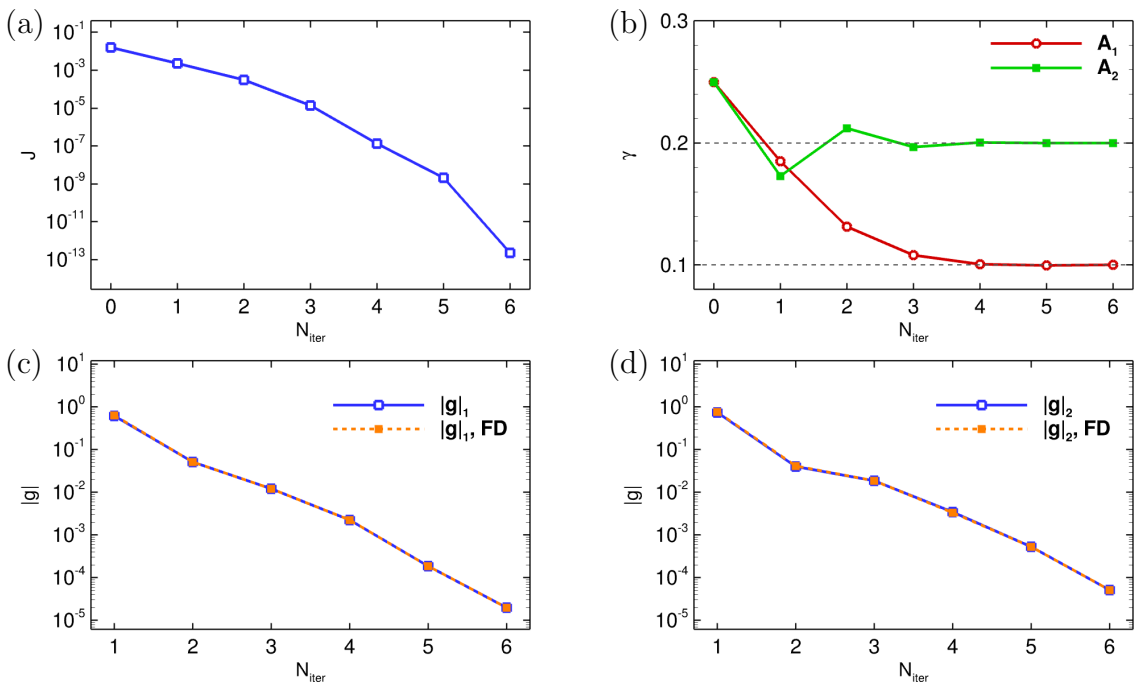

Figure 3.10: *Variation of the objective function $\mathcal{J}$ (a); variation of the control γ (b); and variation of the absolute value of the gradient g ((c) and (d)) with respect to the iteration number. In (b), horizontal dashed lines indicate the target control parameter $\boldsymbol{\gamma}_t = [0.1, 0.2]$. In (c) and (d), blue dashed lines with hollow squares are the gradients given by adjoint equation, and orange dashed lines with solid squares are the gradients given by equation 3.19 numerically.*

By implementing the same ROM-based adjoint optimal control method, $\mathcal{J}$ was converged to less than 10^{-13} after 6 iterations, as shown in figure 3.10 (a), and both control parameters reached the target values, $\boldsymbol{\gamma}^{(6)} = [0.1, 0.2]$, as shown in figure 3.10 (b). Figure 3.10 (c) and (d) compared both components of the gradient g from the adjoint computation, which shows very good agreement with the benchmarks.

3.3.2 Optimal control of the flow past a heaving-pitching airfoil

The adjoint-ROM was next applied to the optimization of flow past a 2D heaving-pitching NACA0012 airfoil with $Re = 100$. The schematic view of the flow configuration is presented in figure 3.11. Originally the quarter-chord point of the airfoil was located at the origin of a 15×10 domain (normalized by the chord length of the airfoil c) discretized by a 751×501 non-uniform Cartesian mesh, which was denser in the area close to the airfoil. The airfoil

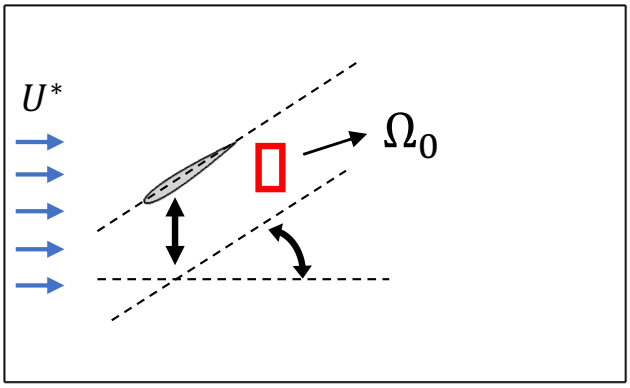

Figure 3.11: *The schematic illustration of the flow past a heaving-pitching NACA0012 airfoil. The red box indicated the observation zone Ω_0. The size is not to scale.*

can heave in the vertical direction (Y) and pitch in the azimuthal direction (Θ) freely as described by the harmonic function 3.55, which resulted in the velocity of the oscillation as:

$$\begin{aligned} \boldsymbol{V}_y(t) &= 2\pi f A \cos(2\pi f t)\boldsymbol{e}_y, \\ \boldsymbol{V}_\theta(t) &= 2\pi f \theta \cos(2\pi f t + \phi)\boldsymbol{e}_r, \end{aligned} \tag{3.55}$$

where $\boldsymbol{e}_y = [0 \quad 1]^T$ and $\boldsymbol{e}_r = [-y \quad x]^T$.

The control vector $\boldsymbol{\gamma} = [A, \theta, \phi]$, which made the problem a 3-DoF optimal control. The frequency f still remained 0.2. The objective was the same as previous cases of flow past an oscillating cylinder to match the flow field in the observation zone Ω_0, which was located in the downstream at $1.4 < x < 1.6$ and $-0.2 < y < 0.2$. Thus $\mathcal{J}$ was calculated by equation 3.18. Due to multiple control parameters, the gradient $\boldsymbol{g}$ is modified accordingly as:

$$\begin{aligned} \boldsymbol{g}_1 &= \int_T [\sum_i^N (-\sum_m b_m C_{im1}) a_i^* + \sum_i^M (\sum_m b_m K_{im1}) b_i^*] \boldsymbol{V}_{1\gamma} dt \\ \boldsymbol{g}_2 &= \int_T [\sum_i^N (-\sum_m b_m C_{im2}) a_i^* + \sum_i^M (\sum_m b_m K_{im2}) b_i^*] \boldsymbol{V}_{2\gamma} dt, \end{aligned} \tag{3.56}$$

where subscript "1" represents vertical (Y) direction and subscript "2" is azimuthal (Θ)

direction, and

$$
\begin{aligned}
C_{im1} &= \frac{1}{\tau}\langle \boldsymbol{e}_y, (\psi_m \boldsymbol{\phi}_i)\rangle \\
C_{im2} &= \frac{1}{\tau}\langle \boldsymbol{e}_r, (\psi_m \boldsymbol{\phi}_i)\rangle
\end{aligned}
\tag{3.57}
$$

and

$$
\begin{aligned}
K_{im1} &= \langle \boldsymbol{e}_y \cdot \nabla \psi_m, \psi_i)\rangle \\
K_{im2} &= \langle \boldsymbol{e}_r \cdot \nabla \psi_m, \psi_i)\rangle
\end{aligned}
\tag{3.58}
$$

Additionally, using equation 3.55 the first-order differentiations of V respective to control parameters are:

$$
\begin{aligned}
\boldsymbol{V}_{1\gamma} &= [2\pi f \cos(2\pi f t) \quad 0]^T, \\
\boldsymbol{V}_{2\gamma} &= [2\pi f \cos(2\pi f t + \phi) \quad -2\pi f \theta \sin(2\pi f t + \phi)]^T.
\end{aligned}
\tag{3.59}
$$

The target flow was generated when $\boldsymbol{\gamma}_t = [0.2, 30°, -30°]$, and this $\boldsymbol{\gamma}_t$ was also regarded as the theoretical benchmark of the optimization. Figure 3.12 (a) shows a typical flow field of the target flow. It can be seen that a single vortex street was formed in the wake without strong vortex pairing. The POD was performed based on 281 snapshots of the target flow covering dimensionless time $t = 149.8$ to $t = 159.9$. 20 global POD modes plus 20 solid modes were used to build DSDM ROM. τ was set to 0.03 for the stability and accuracy of ROM. The flow was perturbed by altering the control parameters abruptly to $\boldsymbol{\gamma}^{(0)} = [0.4, 45°, 30°]$. After the perturbation, the flow structures changed significantly as shown in figure 3.12. Due to high instantaneous angle of attack (AoA), apparent flow separation happened on both sides of the airfoil, which generated strong leading edge vortex (LEV) and trailing edge vortex (TEV). These vortices started pairing in the wake, altering the velocity field in the observation zone Ω_0 significantly.

"One-ROM" strategy was utilized for the optimization. The results of flow optimization are presented in figure 3.13. When the flow was highly perturbed, $\mathcal{J}$ was at about $\mathcal{O}(10)$, then

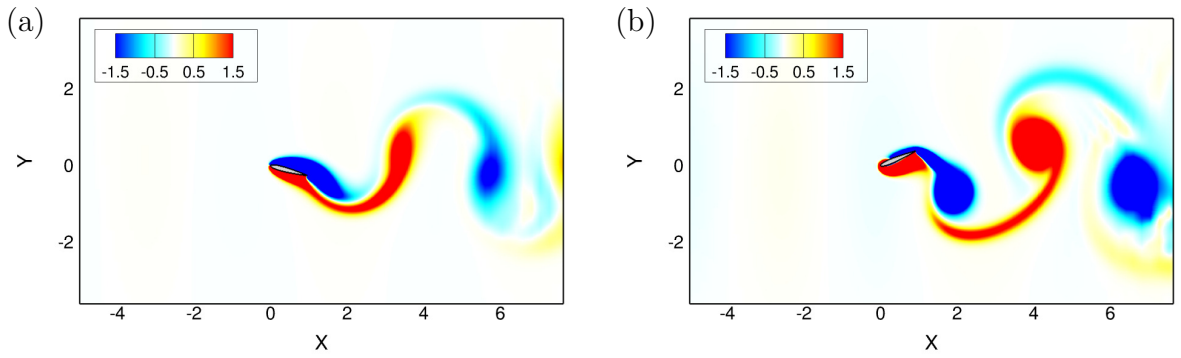

Figure 3.12: *A typical flow field at* $t = 159.9$ *contoured by vorticity. (a): The target (base) flow with* $\boldsymbol{\gamma}_t = [0.2, 30°, -30°]$*; (b): the perturbed flow with* $\boldsymbol{\gamma}_t = [0.4, 45°, 30°]$.

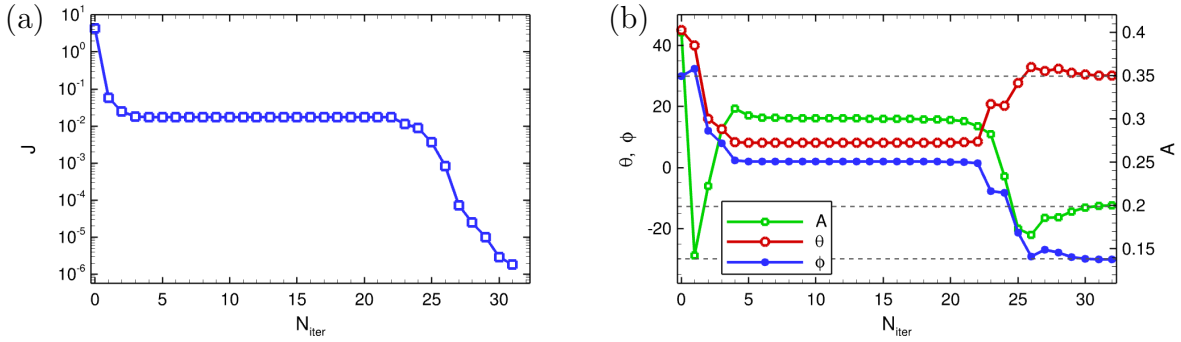

Figure 3.13: *Variation of the objective function* $\mathcal{J}$ *(a) and the variation of the control* $\boldsymbol{\gamma} = [A, \theta, \phi]$ *(b). The horizontal dashed lines indicate the target control parameters* $\boldsymbol{\gamma}_t = [0.2, 30°, -30°]$.

the flow control successfully decreased $\mathcal{J}$ by over 6 orders of magnitude after 32 iterations. It is noticed that due to the big perturbation and the increase of DoF of control, it took many more iterations to reach the convergence, compared to the oscillatory cylinder case in prior section. However, the control parameters $\boldsymbol{\gamma}$ were all converged to the theoretical benchmark $\boldsymbol{\gamma}_t$ in the end.

Each component of the gradient vector $\boldsymbol{g}$ was validated by comparing the value given by solving adjoint equations 3.28 and equation 3.56, against the gradient computed by directly solving equation 3.19 with finite difference method. The perturbation ϵ in finite difference computation was set to 10^{-4}. The comparisons between the gradients are shown in figure 3.14. It is seen that small mismatch existed for g_A and g_θ at some intermediate iterations, but eventually can have good agreement with the benchmarks when the optimization was

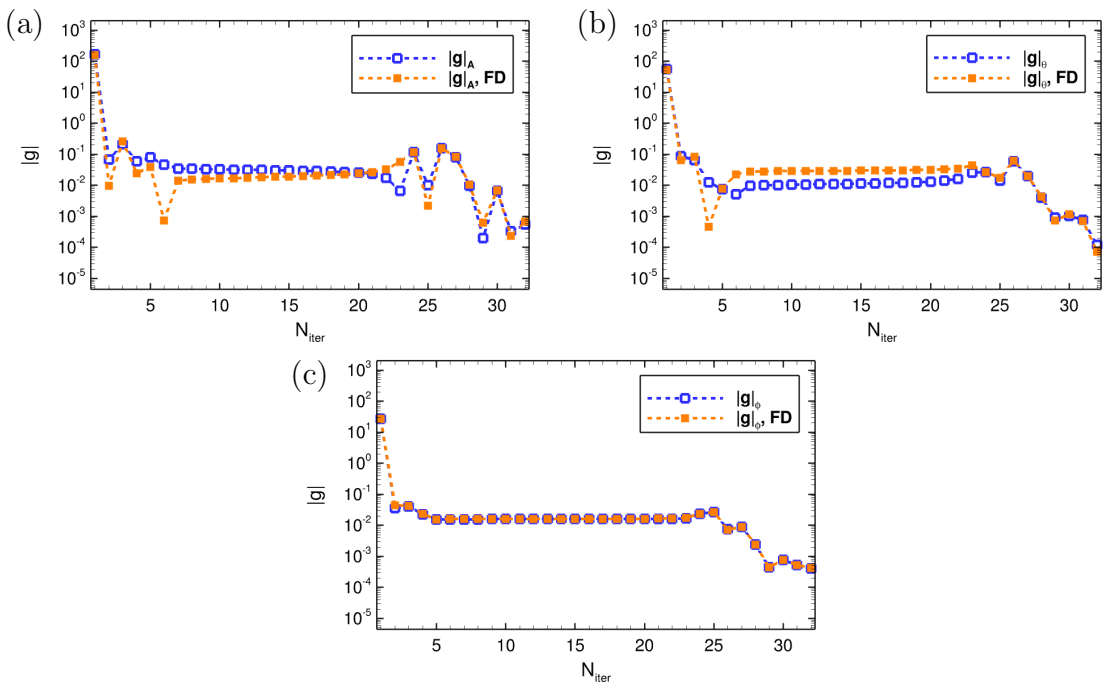

Figure 3.14: *Variation of the absolute value of gradient* $\boldsymbol{g}$ *with respect to each parameter in* $\boldsymbol{\gamma}$*: (a) the gradient of* A*; (b) the gradient of* θ*; (c) the gradient of* ϕ*. Blue dashed lines with hollow squares are the gradients given by adjoint equation, and orange dashed lines with solid squares are the gradients given by equation 3.19 numerically.*

converged. For g_ϕ, very good match was found at every iteration between the value from adjoint-ROM equation and the benchmark. It is worth noting that from about 4th iteration to 22nd iteration, the gradient was already low and nearly unchanged, which also corresponds to the very slow change of $\mathcal{J}$ and $\boldsymbol{\gamma}$ as shown in figure 3.13. That is the common challenge for all gradient based methods that sometimes the optimization would reach a low gradient region, which would take more iterations to move out to the real convergency. Such low gradient region would slow down the convergency of the optimization, and may even lead to a local optimum instead of the global optimum.

So far only one observation zone has been studied to evaluate the perturbation. Practically in either experiments or industrial designs, multiple probes or sensors are often used to detect the change of flow field. Therefore, with the same flow configuration and the same initial $\boldsymbol{\gamma}_t$, another observation zone Ω_{0b} was set in the area of $0 < x < 0.2$ and $0.4 < y < 0.8$, with the same size as Ω_0 but located above the moving airfoil. Accordingly the integration

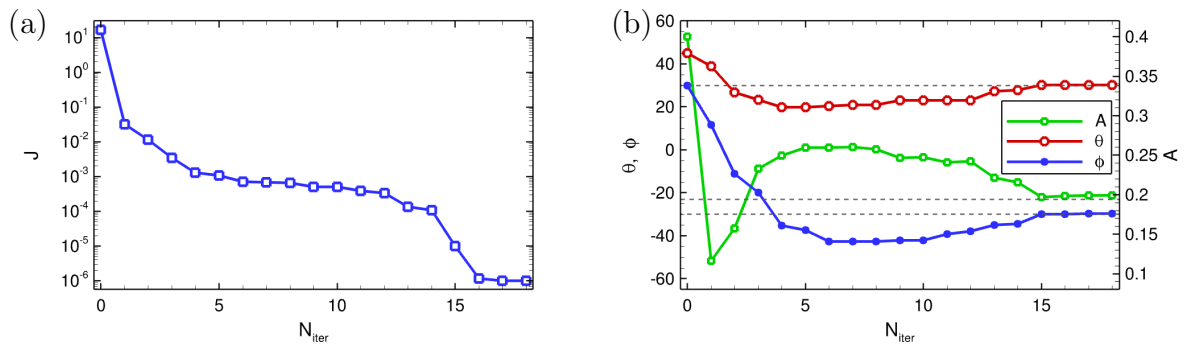

Figure 3.15: *Variation of the objective function $\mathcal{J}$ (a) and the variation of the control $\boldsymbol{\gamma} = [A, \theta, \phi]$ (b) when there are two observation zones. The horizontal dashed lines indicate the target control parameters $\boldsymbol{\gamma}_t = [0.2, 30°, -30°]$.*

in equation 3.18 was calculated in both observation zone. The results of optimal control are presented in figure 3.15. It can be found that with 2 observation zones, $\mathcal{J}$ can still be reduced by almost 6 orders of magnitudes, with all three control parameters recovered to the theoretical benchmarks at the end of the optimization. It is worth noting that with one more observation zone, much less iterations are required to reach the same level of convergency, as the number of iterations decreased from 32 to 18. It can be deduced that more probes may further enhance the controllability of the flow optimization and speed up the convergence.

3.3.3 Optimal force control of the flow past a moving cylinder

Lastly the adjoint-ROM was applied to aerodynamic force control of the flow past a 2D azimuthally oscillating cylinder. The control was imposed by changing the prescribed angular velocity $\boldsymbol{V}_\theta(t)$ of the cylinder:

$$\boldsymbol{V}_\theta(t) = \theta \sin(2\pi f t)\boldsymbol{e}_r, \tag{3.60}$$

where only the amplitude θ was controlled. Therefore, in this section below θ will be substituted by γ to avoid any confusion. The frequency f was fixed to be 0.1, and the Reynolds number was $Re = 200$, which was normalized by incoming flow velocity U and diameter of the cylinder D. The original flow was generated by DNS on a 60×60 domain (normalized by D), which was discretized by a 701×701 non-uniform Cartesian mesh. The numerical method for DNS was the same as the cases in prior sections. The flow configurations and a

snapshot of the flow field are presented in figure 3.16.

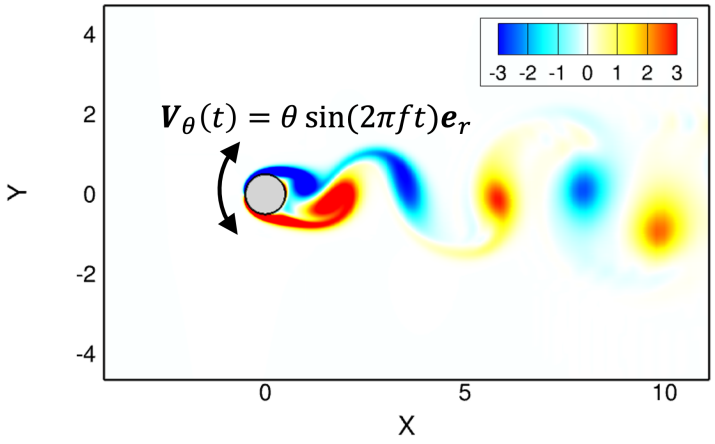

Figure 3.16: *The flow configurations and a snapshot when $\gamma = 0.25$ for the flow past a 2D azimuthally oscillating cylinder. The snapshot is contoured by vorticity field.*

With the given control parameter, the gradient function $\boldsymbol{g}$ can be derived accordingly:

$$\boldsymbol{g} = \int_T [\sum_i^N (-\frac{1}{\tau} \sum_m^N b_m \langle \boldsymbol{R}, \psi_m \boldsymbol{\Phi}_i \rangle a_i^*) + \sum_i^N (\sum_m^N \langle \boldsymbol{R} \cdot \nabla \psi_m, \psi_i \rangle b_m) b_i^*] \boldsymbol{V}_\gamma dt, \tag{3.61}$$

where

$$\boldsymbol{V}_\gamma = \sin(2\pi f t), \tag{3.62}$$

The "ROM-switching" strategy was used for this case, because the objective function (aerodynamic forces) studied here was more sensitive to the flow condition and control parameters. The library for the control consisted of seven different ROMs, generated with γ varying from 0.15 to 0.75 by step of 0.1. The aerodynamic forces of the flow when $\gamma = 0.25$ are shown in figure 3.17. It is found that the drag and lift given by the DNS, the direct projection of the original flow onto POD modes (L-DNS) and the ROM are very close to each other, especially for lift C_L, which demonstrates that the essence of the physics of the original flow can be captured accurately by ROMs. Moreover, the approximations of C_D and C_L by equation 3.33 are also compared to the real forces, which shows a good agreement with both forces. Similar results were also obtained for other γ in the chosen range. Therefore, the objective function equation 3.33 is valid and suitable for the force control problem.

Practically, after obtaining the force history, Fourier analysis was performed to determine the period T of the force in equation 3.18.

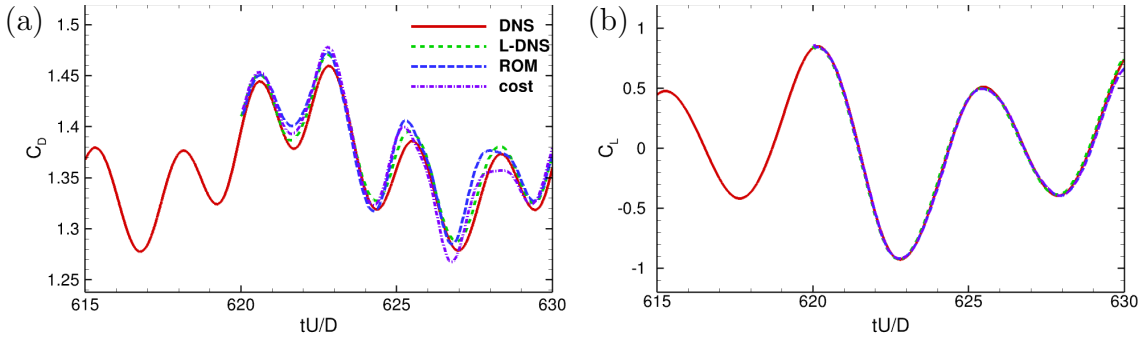

Figure 3.17: *The drag (a) and lift (b) coefficients for the cylinder flow when $\gamma = 0.25$, computed by DNS, L-DNS and ROM. C_D and C_L used in the objective function equation 3.18 are presented by purple dashed-dotted lines.*

Two initial control parameters, $\gamma^{(0)} = 0.25$ and $\gamma^{(0)} = 0.75$ were studied. The variations of the objective function $\mathcal{J}$ and corresponding γ after each iteration are presented in figure 3.18. For both initial control parameters, a local optimum was found by the adjoint-ROM approach to maximize the drag to $C_{D\max} = 1.418$. Correspondingly, the optimal solution $\boldsymbol{\gamma}_{\text{opt}} = 0.499$ was obtained for both cases, which shows the good convergence this approach can achieve with different initial control parameters.

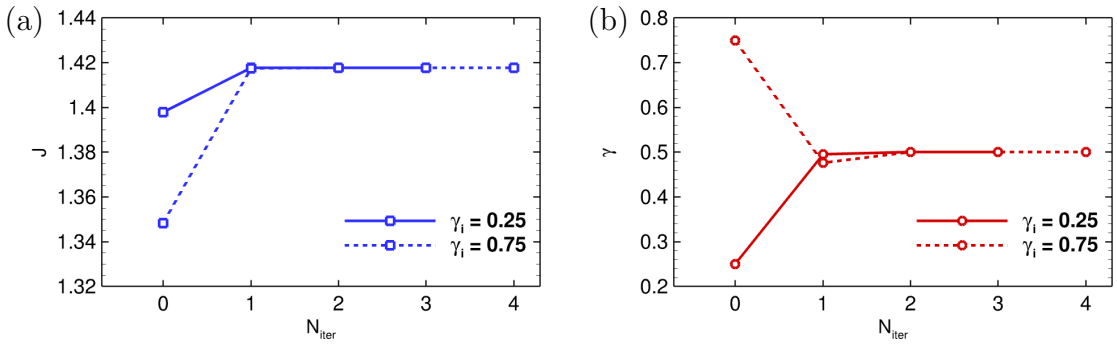

Figure 3.18: *The variations of the objective function $\mathcal{J}$ (a) and the control parameters γ (b) with the number of iterations. Initial controls are $\gamma^{(0)} = 0.25$ for solid lines and $\gamma^{(0)} = 0.75$ for dashed lines respectively.*

In order to validate the accuracy of the optimal solution calculated by the adjoint-ROM approach, a parametric study was performed by DNS as complement. As shown in figure

3.19, nine numerical simulations by DNS in total were run for γ in the range of $[0.15, 0.75]$, which indicated that a local maximum exists at $\gamma \approx 0.5$. It can be seen that the optimal solution given by the adjoint-ROM approach is quite accurate compared against the result of parametric study.

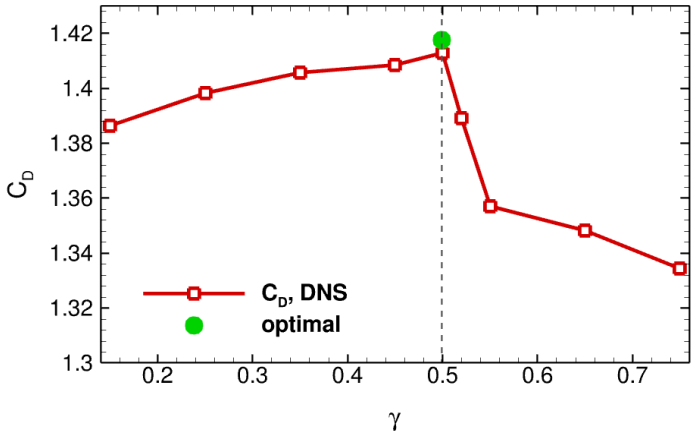

Figure 3.19: *The drag C_D obtained by DNS for different control parameters γ (red line and squares) and the optimal solution given by ROM-adjoint approach (green circle). The vertical dashed line indicates γ_{opt}.*

3.3.4 Computational cost analysis

A key motivation of the current study is reducing the computational cost of adjoint-based optimal control. Table 3.1 lists the computational time for the optimal control of all cases studied in the previous sections. A comparison is also made with the DNS-based adjoint optimal control method as used in previous works[80;100;101] as well as the adjoint-ROM approach developed in the present work. For the adjoint-ROM approach, online time is spent on the fast optimization process, while the offline time is spent in computing the database by DNS, as well as building ROMs. Note that the first three cases were optimized with "one-ROM" strategy when using adjoint-ROM while the last case was optimized with "ROM-switching" strategy. It can be found that by using the DNS-based optimization, the first three cases need over 10 hours or even days to search for the optimal solution, though it is already much more inexpensive compared to parametric study. For the forth case, the DNS-based

Table 3.1: *Computational time for using DNS-based adjoint approach and adjoint-ROM approach (including online and offline time) of different cases.*

Cases	DNS-based optimal control	adjoint-ROM (online)	adjoint-ROM (offline)
Oscillatory cylinder ($\gamma^{(0)} = 0.2$)	10h48m	13m10s	1h20m
2-DoF oscillatory cylinder	36h12m	14m22s	1h43m
3-DoF heaving-pitching NACA airfoil (one Ω_0)	52h25m	29m	2h
Force control on pitching cylinder ($\gamma^{(0)} = 0.25$)	N/A	32m	14h

optimization was not perform, but for a single DNS run, it took about 5 hours, not to mention that each iteration of line-search process requires several runs of DNS. In comparison, by using adjoint-ROM approach, the online computational time for all cases studied is less than or about 30 minutes. As can be seen in table 3.1, the reduction in computational time is 98.0% for the oscillating cylinder case, 99.3% for 2-DoF oscillating cylinder case, and 99.1%for 3-DOF moving airfoil case when using adjoint-ROM with "one-ROM" strategy compared to the DNS-based method. It is worth noting that the offline computational time of using adjoint-ROM is actually much longer than the online computational time. When using "one ROM" strategy, building the ROM takes one to two hours. When using "ROM-switching" strategy, since several ROMs are needed to construct the library of ROMs, about 14 hours are spent on offline computation. However, these computations are all conducted offline which will not add any burden to the optimization process. In a word, the huge savings in computational cost achieved through adjoint-ROM optimization methods with both strategies have shown great promise for the prospects of near-real-time flow control.

3.4 Concluding remarks

In this chapter an adjoint-ROM based optimal control method has been developed, which can handle the control introduced from moving solid boundaries. Two different strategies to guarantee the accuracy of ROMs during the optimization process have also been developed. The first strategy was "one-ROM" strategy. It was used to control the 2D flow past an oscillatory cylinder and a heaving-pitching NACA0012 airfoil perturbed by an abrupt change in the solid motion, with up to three control parameters. The optimization was realized by matching the velocity field in designated observation zones. This objective function can not only be used to easily evaluate the convergence of the adjoint method, but also be implemented for the stabilization of a perturbed flow deviating from the targeted working condition. For all the cases using "one-ROM" strategy, the target control parameters were able to recovered after few iterations, with a good convergence of the objective function. The gradients g derived by the adjoint method were compared with the gradients generated by finite difference method which are regarded as benchmarks. Good agreements between the gradients were achieved for all cases. The second strategy was "ROM-switching" strategy. It was used to optimize the aerodynamic force of the 2D flow past a moving cylinder performing rotational oscillation, with pitching amplitude as the control parameter. The accuracy of the adaptive ROM can be guaranteed by using different ROMs even when the control parameters change greatly during the control process. A new objective function to evaluate the drag and lift has been derived based on the force calculating approach introduced in Noca et al.[83], and the velocity decomposed with time coefficients and POD modes. It has also been demonstrated that the objective function developed accurately recreated the aerodynamic forces for use in optimal control strategies. The optimization has been carried out with two different initial controls, both of which have resulted in the same optimal solution that has been validated by the parametric study. The results presented in this study indicate that the adjoint-ROM based approach with "ROM-switching" strategy is effective to optimize aerodynamic properties like lift and drag.

The computational time on adjoint-ROM based optimal control process was studied and

compared to the time spent by using DNS-based approach. It is found that by using ROM-based approach with both strategies, the computational can be saved by over 90% for all cases, compared to DNS-based approach, let alone parametric study. The drastic saving in computational cost makes the present approach promising for the near-real-time flow control.

The case studied here by using "ROM-switching" strategy used seven ROMs for the construction of the library, which was shown to be sufficient. It can be inferred that if more control parameters are involved, more ROMs may be needed to keep the cost function accurate throughout the control process, therefore the offline time spent on preparing those ROMs may increase accordingly, which will eventually overshadow the benefit of time saving brought by using adjoint-ROM approach. In order to make the approach practical for more applications, more robust ROMs are desirable to guarantee the accuracy of cost function after each iteration while still keeping the computational cost low. One possible way to avoid generating many ROMs for different parameters is leveraging the interpolation between existing modes to generate adaptive ROMs at a new working condition. Some approaches, including the one based on Grassmann manifold which was carried out by Amsallem and Farhat[115], have shown limited success in the application to aeroelasticity. However, the application of these methods in the current study is beyond the scope of this dissertation.

Chapter 4

Adjoint-based gust mitigation

4.1 Introduction

In the past few decades, unmanned air vehicles (UAVs) have drawn great attention for their extensive applications in military surveillance, reconnaissance, and logistic support[116]. UAVs usually operate in a highly unsteady and unpredictable aerodynamic environment, such as gusty winds[117]. Due to their relatively small size and low flying speed, the disturbance of the gust may easily reach the same order of magnitude as the flight speed of the vehicles, making them more vulnerable to the impact of gust. In order to alleviate the negative influence of gusts on the aerodynamic performance of UAVs, an effective flow control strategy is desperately needed based on *in situ* unsteady flow conditions. However, the highly nonlinear transient flow in the gust-UAV interaction sophisticates the development of viable gust-mitigation approaches.

The seminal works by Küssner[118] and Von Karman and Sears[119] studied the interaction between a sharp-edged gust and a thin airfoil theoretically by using potential flow theory on the thin airfoil model. The model had limited success in predicting the lift changes during the gust encounter of large-scale aircraft, yet it was not accurate for the operating condition of modern UAVs[120], due to the assumptions of low gust ratios (GRs), inviscid flows, and attached flow boundaries. Since then, many experimental and numerical studies have

been conducted to better understand the mechanisms of the interaction between nonlinear, highly unsteady gusts and static or moving wings, which may ultimately aid in the design of effective gust mitigation strategies. Corkery et al.[121] studied the interactions between a steady top-hat shaped gust and a flat-plate wing by particle image velocimetry (PIV) me asurements. Similar wing models with different types of transverse gusts were also investigated experimentally and numerically by Badrya et al.[122] and Biler et al.[123]. The transient aerodynamic forces were found dependent on the effective angle of attack (AoA) at the leading edge. The impact of effective AoA as well as effective flow angle on the force coefficients when the static airfoil encounters the gust was also investigated experimentally by Smith et al.[120], Perrotta and Jones[124], and Stutz et al.[125], in which some updated theoretical models have been developed to predict the lift during gust interactions.

It can be inferred from the above results that changing the AoA of a static airfoil may reduce the aerodynamic force burst during the gust encounter, therefore the oscillating wing, which is able to adjust its effective AoA flexibly, becomes a natural choice for gust mitigation. In fact, some studie on flying birds and insects have already found that oscillating wings can play an important role in attenuating the disturbance of unsteady gusts[126–128], which has also inspired the design of UAVs[129]. Lian and Shyy[130] and Lian[131] used numerical simulation to investigate the effect of oscillating wings on alleviating the freestream gust fluctuation, and found multiple parameters may determine whether the gust can be mitigated. Poudel et al.[132] implemented the pitch-down maneuver and wing oscillation to mitigate the influence of a long-lived transverse gust on the NACA0012 airfoil numerically. The study found the optimal reduced pitching frequency to overcome the disturbance brought by the gust, but increasing the aerodynamic force at the same time, which might be undesirable for a stable flight. In the work of Andreu-Angulo and Babinsky[133], the unsteady pitch motion designed based on the classic theoretical solutions by Küssner[118] with unsteady flow modifications could achieve up to 90% lift alleviation during the top-hat transverse gust encounter modeled in a water tow tank. Some simple closed- and open-loop control strategies have been applied to mitigate gust effects with oscillating wings lately. Sedky et al.[134] and Sedky et al.[135] developed a closed-loop control strategy for lift regulation during a transverse gust encounter,

based on a proportional feedback controller with pitch acceleration input. The effective mitigation of the peak lift force was obtained. A follow-up study implemented and compared three different open-loop control maneuvers experimentally for the gust mitigation with respect to the lift force and pitch moments[136]. In the work of Pohl et al.[137] and Herrmann et al.[138], the gust mitigation was performed by a closed-loop control with an active trailing edge flap. The effectiveness of different control strategies at reducing lift fluctuations was validated by experimental results.

Due to the complexity of gust-wing interactions, numerous parameters have been found to have impact on the aerodynamic response of the oscillating wing, including plunging and pitching amplitude, effective and mean AoA, reduced frequency, and so on[117]. The parametric study is impractical or even impossible to find the optimal solution when facing a massive parametric space. The existing control schemes have had limited success in the gust mitigation, but to the best of author's knowledge, none of them has optimized multiple control parameters simultaneously. Additionally, usually different control strategies were required for different objectives in order to obtain better mitigation performance. All these challenges expose the constraints of existing approaches for addressing the gust mitigation problem.

As reviewed in chapter 1 and chapter 3, the non-cylindrical calculus enabled adjoint approach is able to optimize the aerodynamic performance of a moving airfoil, which shows its effectiveness on the optimization of flows with complicated FSI[80;88;100;101]. Therefore, it is natural to apply the similar approach to the gust mitigation with heaving-plunging wings, with reasonable modifications on specific control objectives.

In this work, the gust responses were evaluated by the deviation of the lift after gust encounter to the mean lift over one oscillating stroke. The mitigation of the gust was realized by minimizing an objective function to recover the mean lift while reducing the unsteadiness of the lift profile at the same time. The adjoint-based approach was implemented not only for a full-order model (FOM) of gust-wing interaction simulated by solving the Navier-Stokes equation with high fidelity, but also for the global POD-Galerkin projection based ROM with lower fidelity but extremely short computational time. The remainder of this chapter is

outlined as follows. The computational configurations, gust models, FOM- and ROM-based adjoint approach are introduced in section 4.2. The results of adjoint-based gust mitigation is presented in section 4.3, with the discussion on flow physics and control effectiveness. Section 4.4 gives the concluding remarks of this work.

4.2 Methodology

4.2.1 Gust models and computational setup

In this work, the high-fidelity DNS were carried out for the FOM of interactions between gusts and an NACA0012 airfoil. The airfoil was able to heave in vertical (y) direction as well as pitch about its 1/4 chord point harmonically, specifically given as follows:

$$\begin{aligned} Y(t) &= A\sin(2\pi ft), \\ \Theta(t) &= \theta\sin(2\pi ft+\phi)-\beta, \end{aligned} \tag{4.1}$$

where A is the heaving amplitude, θ the pitching amplitude, ϕ the phase delay angle between heaving and pitching motion, and β the mean AoA of the airfoil. In the rest of this paper, the control will be focused on these four parameters which may have impact on the aerodynamic force of wings. To describe the control parameters more concisely, they are summarized in a control vector $\boldsymbol{\gamma} = [A, \theta, \phi, \beta]$ for both 2D and 3D flows. The total degrees of freedom (DoF) of the control space were up to four.

As summarized in Jones[117], gusts can fall into one of following three categories: streamwise gust, transverse gust and vortical gust. In this work, a uniform streamwise gust and an unsteady transverse gust were studied. The streamwise gust was generated by changing the incoming flow velocity U^*, which also led to the change in the Reynolds number $Re = U^*c/\nu$, where ν is the kinetic viscosity of the flow, and c the chord length of the wing. The unsteady transverse gust was introduced by adding a jet of periodic vertical velocity profile ahead of the airfoil from a gust inlet at the bottom boundary of the computational domain. The

velocity of the unsteady transverse gust V_g is given by the following sinusoidal function:

$$V_g = \alpha[1 + \sin(2\pi f t)], \tag{4.2}$$

where α is the dimensionless gust intensity normalized by the incoming flow velocity, f is the frequency which is equal to the frequency of the airfoil oscillation. Details of the computational setup are introduced in the following sections for 2D and 3D configurations respectively.

The 2D and 3D simulations were performed by solving equation 1.3 numerically, with a second-order central difference scheme for spatial discretization, and a third-order Runge-Kutta/Crank-Nicolson scheme for time advancement. The numerical solver used in this work has been well validated in previous studies of the author's research group[80;100;101;139]. For the specific cases studied in this chapter, mesh independence study was still conducted, which is introduced in the following subsections. The time step Δt was constrained by fixing Courant-Friedrichs-Lewy (CFL) number to 0.1.

2D flow configuration

As depicted in figure 4.1, the 2D simulation was performed on a rectangular computational domain of $60c \times 12c$. The domain was discretized by a 901×676 non-uniform Cartesian mesh, with minimal mesh size $\Delta x_{\min} = 1.11 \times 10^{-2} c$. The mesh was fine and uniform in the near field of the airfoil and gradually stretched towards the far field. The transverse gust inlet was located at the bottom boundary in the range of $[-0.5c, 0.5c]$, with a width of $1c$. Initially, the quarter-chord point of the airfoil was located at the origin, and the distance between the bottom boundary and the rotating center of the airfoil was $1c$. A constant incoming flow U^* was imposed at the left boundary. When simulating streamwise gust condition, Neumann boundary condition was set for all other boundaries. As for the transverse gust condition, these boundary conditions remained almost the same except an inflow boundary imposed specifically at the gust inlet.

The study on grid independence for the flow past oscillating airfoil was conducted for

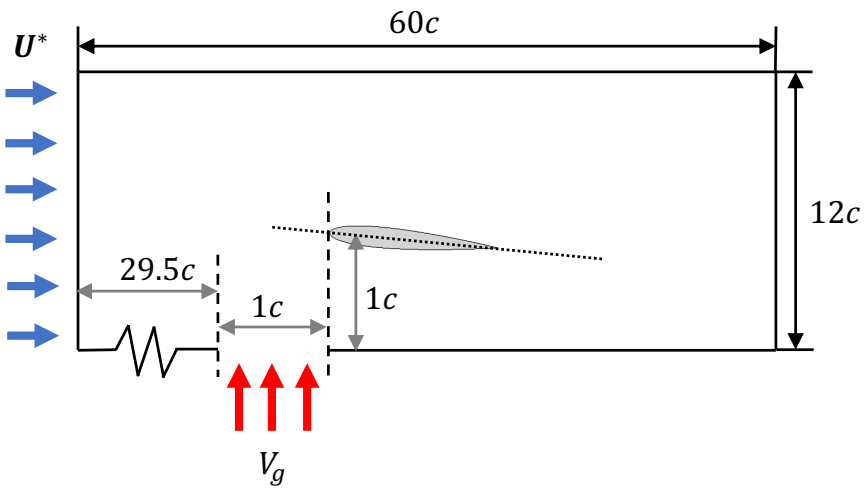

Figure 4.1: *An illustration of 2D computational domain. The dimension of the sketch is not to scale.*

three different meshes from coarse to dense at $Re = 200$ and with $\boldsymbol{\gamma} = [0.5, 5°, 0, 15°]$. The mean drag and lift coefficients were evaluated for each mesh under no-gust and transverse-gust conditions. As presented in table. 4.1, force coefficients obtained on different meshes become converged as mesh size becomes finer, with merely 0.5% difference in $\bar{C_D}$ and 0.1% difference in $\bar{C_L}$ for no-gust condition, and 1.0% difference in $\bar{C_D}$ and 0.2% difference in $\bar{C_L}$ for transverse-gust condition between 1201×676 mesh and 901×676 mesh. Therefore, 901×676 mesh was considered fine enough and was used for all 2D simulations.

Table 4.1: *Mean drag and lift coefficients obtained on different 2D meshes before and after transverse gust encounter.*

Mesh size	$\bar{C_D}$ (no gust)	$\bar{C_L}$ (no gust)	$\bar{C_D}$ (with gust)	$\bar{C_L}$ (with gust)
701×561, $\Delta x_{\min} = 1.49 \times 10^{-2}c$	0.395	0.659	0.500	1.119
901×676, $\Delta x_{\min} = 1.11 \times 10^{-2}c$	0.392	0.678	0.497	1.125
1201×676, $\Delta x_{\min} = 6.67 \times 10^{-3}c$	0.390	0.679	0.492	1.127

3D flow configuration

As shown in figure 4.2, the 3D simulation was performed in a $20c \times 5c \times 4c$ rectangular domain. The domain was discretized by a $601 \times 381 \times 161$ Cartesian mesh, with minimal mesh size $\Delta x_{\min} = 1.00 \times 10^{-2}$. The mesh was non-uniform in streamwise (x) and vertical (y) direction, which is fine and uniform in the near field of the airfoil, and stretched towards the far field. In spanwise (z) direction the mesh was uniform. The gust configurations were similar to the 2D counterparts. The streamwise gust was generated by changing the Reynolds number based on the incoming flow velocity, while the transverse gust was introduced by adding a vertical flow through a $1c \times 1c$ gust inlet located in the area of $x \in [-0.5c, 0.5c]$ and $z \in [-0.5c, 0.5c]$ at the bottom boundary. Initially, the quarter-chord axis of the airfoil was located in line with $x = 0$, with the vertical distance to the bottom boundary equal to $1c$. A constant incoming flow U^* was introduced from the upstream. Zero-flux boundary condition was set for other flow boundaries, except for the gust inlet when simulating transverse gust. The 3D NACA airfoil model is also presented in Fig. 4.2, with the aspect ratio equal to 2. The solid mesh consists of 17768 triangular elements on airfoil surface.

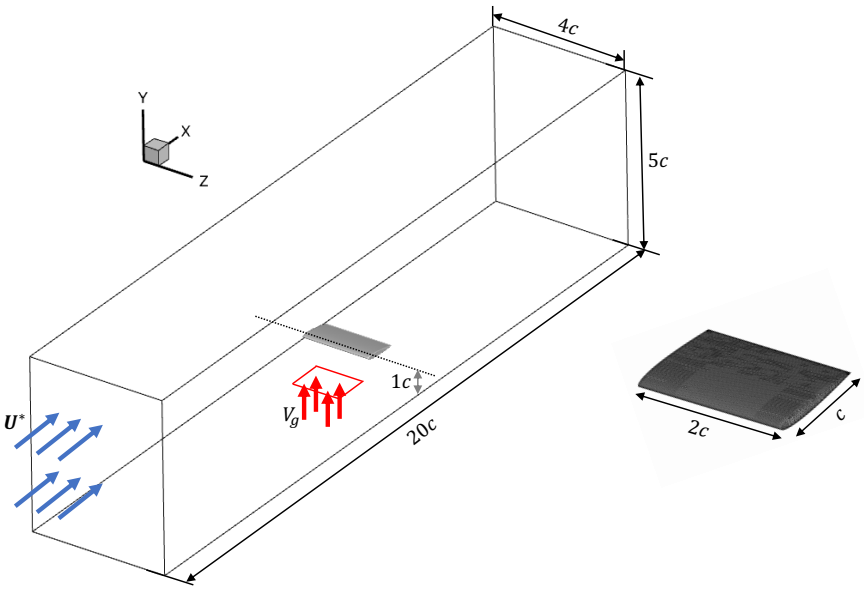

Figure 4.2: *An illustration of 3D computational domain and the 3D wing mesh.*

The 3D simulation was run in parallel on 20 CPU cores with the computational domain

being divided equally in spanwise (z) direction. The mesh independence study for the flow with gust in this work was performed on 3 different meshes as presented in table 4.2 at $Re = 100$ with $\boldsymbol{\gamma} = [0.2, 5°, 0, 15°]$, where the force coefficients were evaluated as did for 2D simulations. It was observed that between the finest mesh and $601 \times 381 \times 161$ mesh, the difference in both $\bar{C_D}$ and $\bar{C_L}$ is 0.2% for no-gust condition, while 0.7% in $\bar{C_D}$ and 0.1% in $\bar{C_L}$ for transverse-gust condition. The mesh-independent study indicated that $601 \times 381 \times 161$ mesh was fine enough thus used for all 3D simulations.

Table 4.2: *Mean drag and lift coefficients obtained on different 3D meshes before and after transverse gust encounter.*

Mesh size	$\bar{C_D}$ (no gust)	$\bar{C_L}$ (no gust)	$\bar{C_D}$ (with gust)	$\bar{C_L}$ (with gust)
$521 \times 361 \times 161$, $\Delta x_{\min} = 1.25 \times 10^{-2} c$	0.435	0.503	0.541	1.840
$601 \times 381 \times 161$, $\Delta x_{\min} = 1.00 \times 10^{-2} c$	0.435	0.506	0.538	1.846
$641 \times 401 \times 161$, $\Delta x_{\min} = 9.09 \times 10^{-3} c$	0.436	0.505	0.534	1.848

4.2.2 The assessment of gusts: objectives and gradients

It has been widely reported that during the gust-wing interaction, the aerodynamic force on the wing can change so largely that it may cause detrimental impact on the aerodynamic performance. In the present work, the goal of gust mitigation was trying to recover the original aerodynamic lift before gust encounter, while keep the unsteadiness of the lift as low as possible. Two different adjoint-based algorithms were developed for the gust mitigation: one using FOM to take advantage of its high fidelity to mitigate arbitrary gust with reasonable computational cost; and the other using ROM to treat specific weak gust due to its lower fidelity, while having the potential for real-time control because of its extremely short computational time. Generally, the objective function can be described by:

$$\mathcal{J} = \frac{1}{T} \int_T |C_L - \bar{C_{L0}}|^2 dt, \tag{4.3}$$

where $\bar{C_{L0}}$ is the target mean lift before gust encounter, and the lift coefficient is calculated by:

$$C_L = \frac{1}{D_0}\int_{\partial\Omega_s} \sigma_{2j} n_j ds, \tag{4.4}$$

where $D_0 = \rho U^{*2} c/2$. The concrete form of $\mathcal{J}$ was dependent on the model (FOM or ROM) used in the optimization, which will be corroborated in the following sections.

FOM-based adjoint approach

The objective function $\mathcal{J}$ with FOM is given by:

$$\mathcal{J} = \frac{1}{T}\int_T |\frac{1}{D_0}\int_{\partial\Omega_s} \sigma_{2i} n_i ds - \bar{C_{L0}}|^2 dt. \tag{4.5}$$

With $\mathcal{J}$, the similar adjoint equation 3.12 was solved, except that the boundary condition needs modification due to a different objective function:

$$u_i^* = -D_2 \delta_{2i} \quad \text{on} \quad \partial\Omega_s, \tag{4.6}$$

where

$$D_2 = 2(\frac{1}{D_0}\int_{\partial\Omega_s} \sigma_{2i} n_i ds - \bar{C_{L0}}). \tag{4.7}$$

The corresponding gradient with $\boldsymbol{\gamma} = [A, \theta, \phi, \beta]$ as control parameters is:

$$g_l = \frac{1}{T}\int_T D_2 \int_{\partial\Omega_s} [Z_{k,l}\frac{\partial \sigma_{2j}}{\partial x_j} n_k - \dot{V}_{i,l} - Z_{k,l}\frac{\partial u_i}{\partial x_k} Z_i^*] ds dt, \tag{4.8}$$

where the subscript l stands for each control parameter in $\boldsymbol{\gamma}$.

Adjoint-ROM approach

Combining equation 3.17 and 4.3, the objective function for ROM is derived as:

$$\mathcal{J} = \frac{1}{T}\int_T \sum_{i=0} \{\frac{da_i}{dt}[\hat{C}_{F1}(\boldsymbol{\Phi}_i) + \hat{C}_{F2}(\boldsymbol{\Phi}_i, t)] + a_i \hat{C}_{F3}(\boldsymbol{\Phi}_i) + \frac{1}{2}\sum_{j=0} a_i a_j \boldsymbol{S}_{ij} - \bar{C_{L0}})\} dt, \tag{4.9}$$

With this $\mathcal{J}$, similar to the derivation in chapter 3, the adjoint-ROM equation $\boldsymbol{\mathcal{F}}_i^*(\boldsymbol{q}_i^*) - \boldsymbol{\mathcal{R}}_i^* = 0$ is obtained:

$$\begin{aligned}
\boldsymbol{\mathcal{F}}_i^*(\boldsymbol{q}_i^*) &= \begin{bmatrix} \frac{da_i^*}{dt} + \sum_j [L'_{ji} + \sum_k (Q'_{jik} + Q'_{jki}) a_k] a_j^* \\ \\ \frac{db_i^*}{dt} + \sum_j [C'_{ji} - \sum_k (L^*_{jki} + \sum_m Q^*_{jkmi} a_m) a_k] a_j^* - \sum_m P'_{mi} b_m^* \end{bmatrix}, \\
\boldsymbol{\mathcal{R}}_i^* &= \begin{bmatrix} [\frac{1}{T} \sum_{j=0} a_j (\boldsymbol{S}_{ij} + \boldsymbol{S}_{ji}) - \frac{2}{T} \frac{\partial \hat{C}_{F2}}{\partial t} + \frac{2}{T} \hat{C}_{F3}](C_L - \bar{C_{L0}}) \\ 0 \end{bmatrix},
\end{aligned} \tag{4.10}$$

By solving equation 4.10 backward in time, the gradient g_l can be calculated as:

$$\boldsymbol{g}_l = \int_T [\sum_i^N (- \sum_m b_m C_{iml}) a_i^* + \sum_i^M (\sum_m b_m K_{iml}) b_i^*] \boldsymbol{V}_{l\gamma} dt, \quad l = 1, 2 \tag{4.11}$$

with l corresponding to the heaving ($l = 1$) and pitching ($l = 2$) motion respectively, coefficients C_{iml} and K_{iml} are given as:

$$\begin{aligned}
C_{im1} &= \frac{1}{\tau} \langle \boldsymbol{e}_y, (\psi_m \boldsymbol{\phi}_i) \rangle, \\
C_{im2} &= \frac{1}{\tau} \langle \boldsymbol{e}_r, (\psi_m \boldsymbol{\phi}_i) \rangle, \\
K_{im1} &= \langle (\boldsymbol{e}_y \cdot \psi_m), \boldsymbol{\phi}_i \rangle, \\
K_{im2} &= \langle (\boldsymbol{e}_r \cdot \psi_m), \boldsymbol{\phi}_i \rangle.
\end{aligned} \tag{4.12}$$

$\boldsymbol{V}_{l\gamma} = \frac{\partial \boldsymbol{V}_l}{\partial \boldsymbol{\gamma}}$. Based on equation 4.1,

$$\begin{aligned}
\boldsymbol{V}_{1\gamma} &= [2\pi f \cos(2\pi f t) \quad 0]^T, \\
\boldsymbol{V}_{2\gamma} &= [2\pi f \cos(2\pi f t + \phi) \quad - 2\pi f \theta \sin(2\pi f t + \phi)]^T.
\end{aligned} \tag{4.13}$$

It can be seen from the above derivation of $\boldsymbol{V}_{l\gamma}$, that the mean AoA β is cancelled out after the first-order differentiation, thus under the current framework of adjoint-ROM approach, the impact of β cannot be involved in $\boldsymbol{g}_l$. As a result, β is not controlled for ROM-based gust

mitigation. The total degrees of freedom are three with $\boldsymbol{\gamma} = [A, \theta, \phi]$. The conjugate gradient method was also implemented for adjoint-ROM optimization, as a result the optimal $\boldsymbol{\gamma}$ can be determined iteratively.

Special caution should be used when using the ROM-based approach. As discussed in chapter 3, almost all data-driven, projection-based ROM is highly sensitive to the change of flow condition, which means the ROM built upon the base flow may not be valid for the flow with strong perturbations of the gust, and *vice versa*. However, the ROM is usually adaptable for a flow condition that does not deviate too far from the one it is built upon, albeit the criterion to evaluate such deviation is usually empirical. In the present work, all ROMs for the gust mitigation were built upon the data of base flows throughout the optimization process. Therefore, only "weak" gusts were taken into account in this study to guarantee the accuracy of ROMs when control parameters changed. Moreover, *a posterori* validations were made for these gusts to guarantee the robustness of ROMs.

4.3 Results and discussion

In this section, the results of adjoint-based gust mitigation are presented together with the physical analysis and discussions. For 2D gusts, the streamwise gust and the transverse gust were mitigated by both FOM- and ROM-based adjoint approach. For 3D gusts, only FOM-based adjoint approach was applied, considering the offline computational cost to build the 3D ROM is high thus may overshadow the benefit of using ROM-based fast flow control method. The Reynolds number was kept modest for this study, thus the flows remained laminar even with the gust encounter.

4.3.1 2D streamwise gust mitigation

FOM-based control

The original flow without gust was generated at $Re = 200$ with an initial control $\boldsymbol{\gamma}_i = [0.5, 5°, 0, 15°]$ and oscillating frequency $f = 0.1$. The streamwise gust here was realized

by increasing the Reynolds number to 500, which was mimicking a gust from upstream. In order to prevent a deep dynamic stall event at high AoA, β was restricted in a range of $[-20°, 20°]$. The objective function was given by equation 4.3 with period $T = 10$. In this complete stroke, the airfoil is heaving upward first in $[0, T/2]$, and then downward in $[T/2, T]$. As presented in figure 4.3, the objective function was reduced by two orders of magnitude, or 97.8%, after 4 iterations of optimal control. The variations of $\boldsymbol{\gamma}$ are shown in figure 4.4. It can be observed that the heaving amplitude A and pitching amplitude θ were both reduced notably, with $A = 0.058$ and $\theta = -1.7°$ in the end. The negative final pitching amplitude indicated that the airfoil started pitching in the direction opposite to the initial pitching motion. On the contrary, the phase delay angle ϕ and the mean AoA β barely changed when the optimization was converged, which indicated that these two parameters were not as important as A and θ to mitigate the streamwise gust.

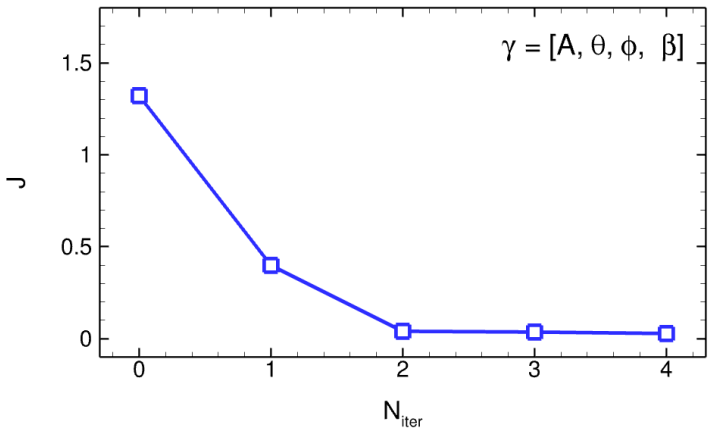

Figure 4.3: *The objective function $\mathcal{J}$ at every main iteration in the optimization.*

The lift coefficients of the base flow, the perturbed flow with streamwise gust, and the optimal flow are compared in figure 4.5. Initially, the lift profile was periodic, with the net mean lift $\bar{C_{L0}}$ equal to 0.678. After the airfoil encountered the streamwise gust, due to much higher incoming flow velocity, the mean lift on the airfoil was increased to 0.785, with higher peak value during the downward stroke. However, the lift profile remained nearly sinusoidal, implying no transition in the flow regime happened with the impact of gust. When the optimal control was applied, there was a decrease in the heaving and pitching amplitude,

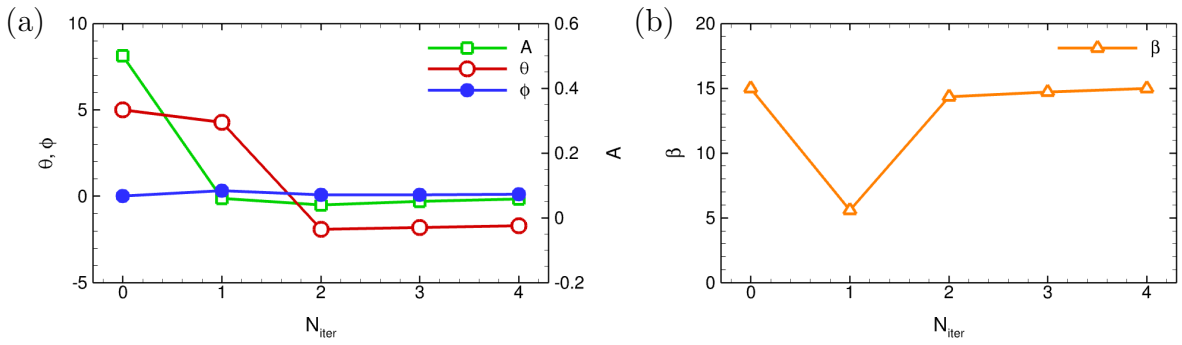

Figure 4.4: *The variations of 4 control parameters in $\boldsymbol{\gamma}$ at every main iteration. (a): The variations of A, θ, ϕ; (b) the variation of mean AoA β.*

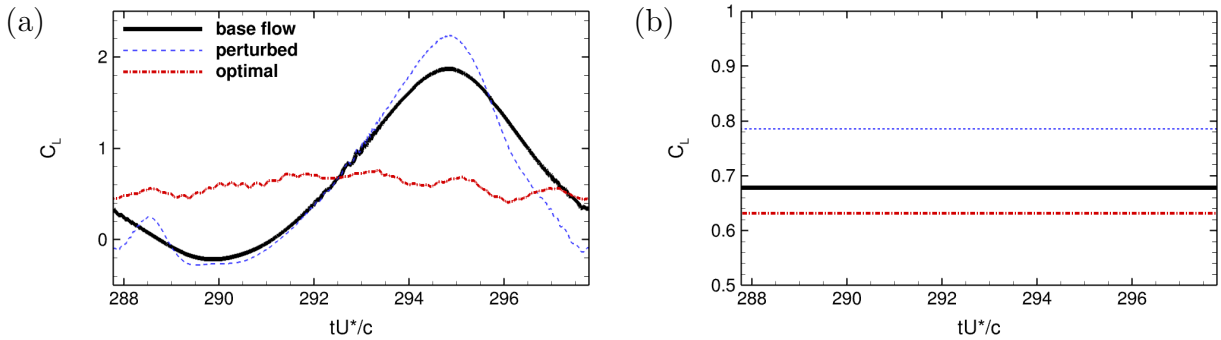

Figure 4.5: *The lift coefficients of the base flow, the perturbed flow with gust, and the optimal flow after gust mitigation. (a): Lift profile; (b): mean lift.*

causing the magnitude of C_L to decrease drastically, and the lift history was not sinusoidal anymore, but still oscillating. As a result, the mean lift was reduced to 0.632, as shown in figure 4.5 (b), with the deviation to $\bar{C_{L0}}$ decreasing from 15.8% to 6.8%. Although the mean lift of the base flow was not completely recovered, the deviation was still cut down by over half, which demonstrates the effectiveness of the gust mitigation approach. However, these results suggest that the fundamental flow of the optimal control case had shifted from the original, pre-gust case.

The flow fields at two typical time moments $tU^*/c = 290$ and $tU^*/c = 295$, where the airfoil was heaving upward and downward respectively, are presented in figure 4.6 for the base flow, the perturbed flow, and the optimal flow. It can be observed that for the base flow, the flow remained attached to both sides of the airfoil when it was heaving up and

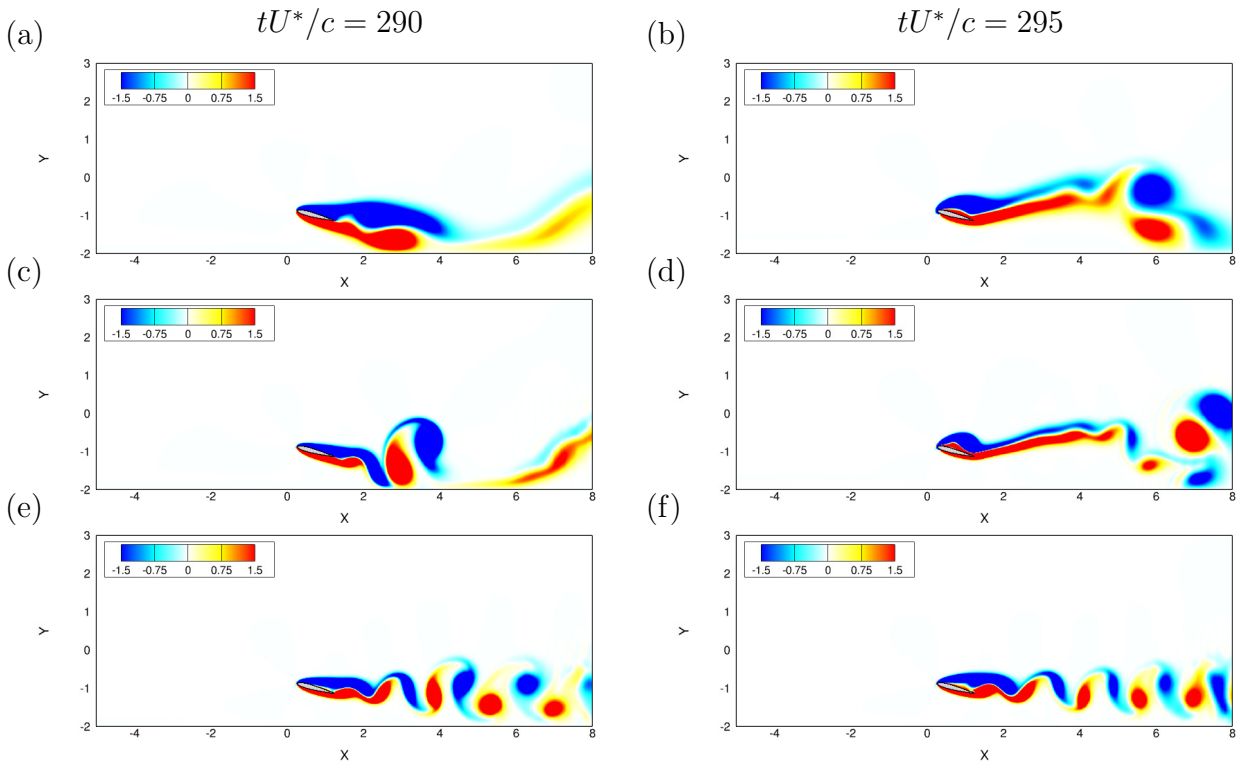

Figure 4.6: *The flow field contoured by vorticity at two typical time moments ($tU^*/c = 290$ and $tU^*/c = 295$) for: (a) and (b) base flows; (c) and (d) perturbed flows; (e) and (f) optimal flows*

pitching down, while the flow began to separate at the leading edge with the leading edge vortex (LEV) being formed as it was heaving down and pitching up. The generation of LEV led to a high instantaneous lift. After the streamwise gust occurred, the vortices were pairing earlier during the first half of the stroke, while the LEV was still generating and shed into the wake, with higher vorticity. The flow structures did not change dramatically with the impact of gust. However, after the optimization, the single vortex pair in the wake changed to a single vortex street, with much weaker heaving and pitching airfoil motion, as seen in figure 4.6 (e) and (f). Such wake structure directly resulted in a C_L with much lower magnitude, but the period of the wake had shifted significantly. The optimal control done here shows that when optimizing for mean lift, the results shifted to a case with almost no motion and did not align with the original oscillation behavior. Despite a big transition occurring in the flow field, the mean lift can still be recovered by over 50%, but care must be taken in defining the objective function in cases where the original motion must be preserved.

ROM-based control

The pre-gust flow was first generated by 2D DNS at $Re = 200$ with an initial control $\boldsymbol{\gamma}_i = [0.2, 5°, 0]$, a fixed mean AoA $\beta = 5°$, which was smaller than β in the FOM-based counterpart, and oscillating frequency $f = 0.1$. The streamwise gust was introduced by decreasing the incoming flow Reynolds number from 200 to 50, which mimicked a gust blown from the downstream. The reason to decrease Re instead of increasing it as done in FOM-based case is to guarantee the stability of the ROM during the whole optimization process. Then the global POD modes were extracted based on 300 snapshots with the flow under $Re = 50$ covering dimensionless time from $tU^*/c = 161$ to 171, for one wing stroke. With the first 20 leading POD modes and solid modes, a global POD-Galerkin ROM was built for the gust mitigation in lower-dimensional subspace. The accuracy of the ROM was validated first for the base flow as well as the perturbed flow with streamwise gust. Figure 4.7 compares the time coefficients of the first two POD modes between the direct linear projection of DNS data (L-DNS) and the solution of ROM equation, for the base flow and the perturbed flow with gust. The L-DNS results was regarded as the benchmarks as it was the direct projection of the high-fidelity FOM data. It is found that the time coefficients given by ROMs are in good agreement with both benchmarks for the base flow and the perturbed flow, which indicates that ROMs can appropriately capture the low-dimensional flow dynamics under various flow conditions. In addition, the lift coefficients computed by the DNS (FOM) and the ROM reconstruction are compared in figure 4.8. It can be seen that for both flows, the lift profiles reconstructed by ROMs can match the ones by DNS with good agreement, with only slight mismatch. Considering the objective function was the mean lift, the error between DNS and ROM results would be even smaller. Hereafter, only the ROM built upon the perturbed flow ($Re = 50$) was used for the gust mitigation.

The objective function was calculated by equation 4.9 with period $T = 10$. The variation of $\mathcal{J}$ during the gust mitigation iterations is presented in figure 4.9. It is found that the objective function was decreased by 79.8% after the optimal control was converged after 4 iterations. As shown in figure 4.20, to the achieve the optimal $\mathcal{J}$, the heaving amplitude A

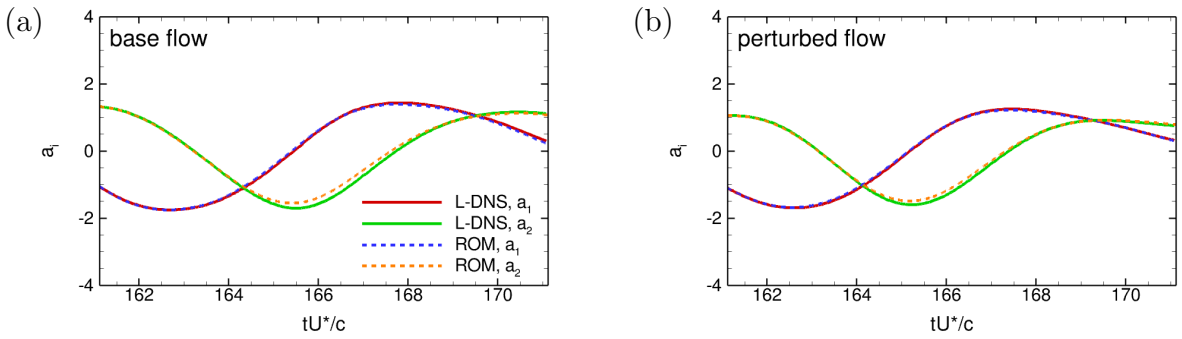

Figure 4.7: *The time coefficients a_i of the first two POD modes by the L-DNS and by solving ROM equation for: (a) the base flow; (b) the perturbed flow with gust.*

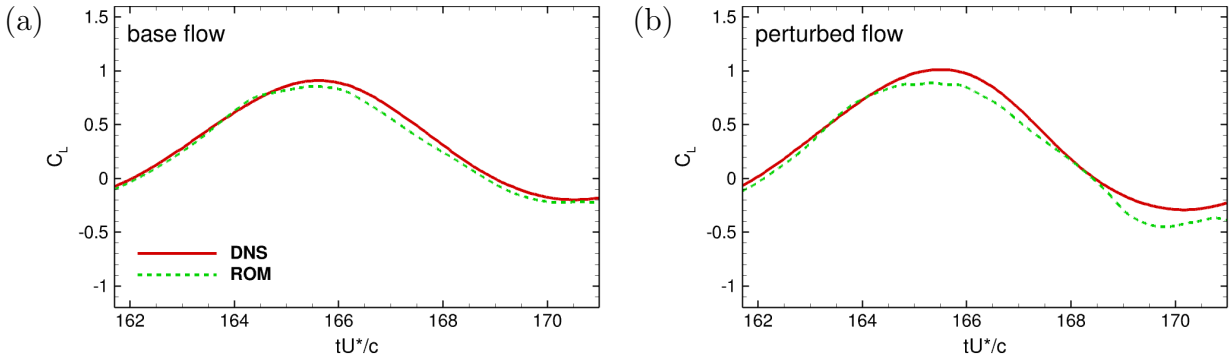

Figure 4.8: *The lift coefficients computed by DNS and ROM for: (a) the base flow; (b) the perturbed flow with gust.*

had to increase by 25.0%, and the pitching amplitude θ was also increased by 14.1%. The phase delay angle ϕ remained insignificant for the lift recovering with close to zero value throughout the control iterations.

Now that the optimal solution of the surrogate model in the subspace (ROM) was obtained, it is crucial to test if the optimal solution obtained in this subspace can precisely represent the optimal solution of the original high-order space (i.e. the FOM). The accuracy of the ROM for the optimal flow was assessed by comparing the lift coefficient computed by DNS and the one reconstructed by ROM as presented in figure 4.11. It can be found that the lift generated by ROM has a good agreement with the benchmark given by DNS. Therefore, it can be confirmed that ROM still kept accurate for the optimal flow, and the optimum of the surrogate model was exactly the one of the original model (FOM).

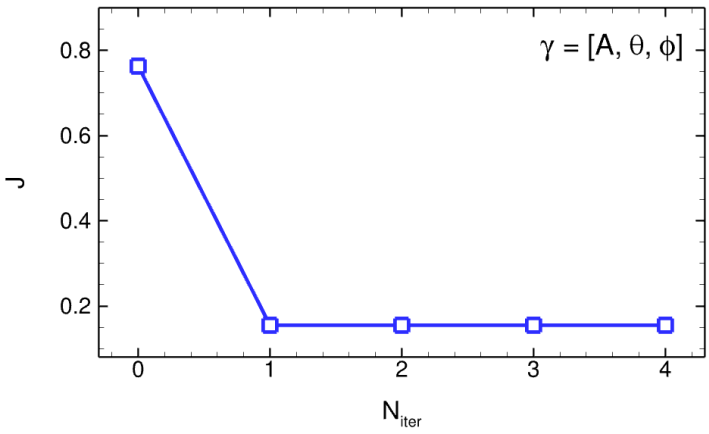

Figure 4.9: *The objective function $\mathcal{J}$ at every main iteration in the optimization.*

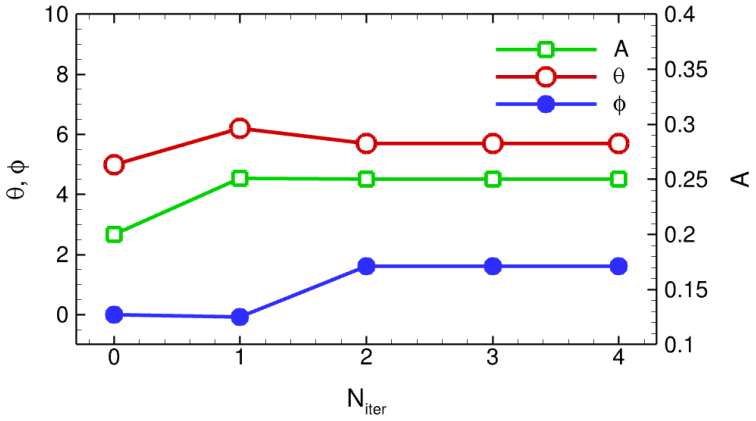

Figure 4.10: *The variations of 3 control parameters in $\boldsymbol{\gamma} = [A, \theta, \phi]$ at every main iteration.*

Lastly, the lift coefficients of the base flow, the perturbed flow with streamwise gust, and the optimal flow are compared in figure 4.12. The lift coefficients shown here were all reconstructed by corresponding ROMs. The initial lift was sinusoidal with a net mean lift $\bar{C_{L0}} = 0.281$. The weak streamwise gust altered the peak and valley values of the lift profile during one stroke period, which resulted in a mean lift of 0.241, or a 14.2% decrease, as shown in figure 4.12 (b). By increasing both A and θ, the mean lift was recovered very well, with $\bar{C_L} = 0.258$ at the end of the optimization, as shown in figure 4.12 (b). The deviation to $\bar{C_{L0}}$ was reduced from 14.2% to 8.2%. Although the deviation was cut down by nearly half, the ultimate mean lift was still lower than the target value, which implies the limited

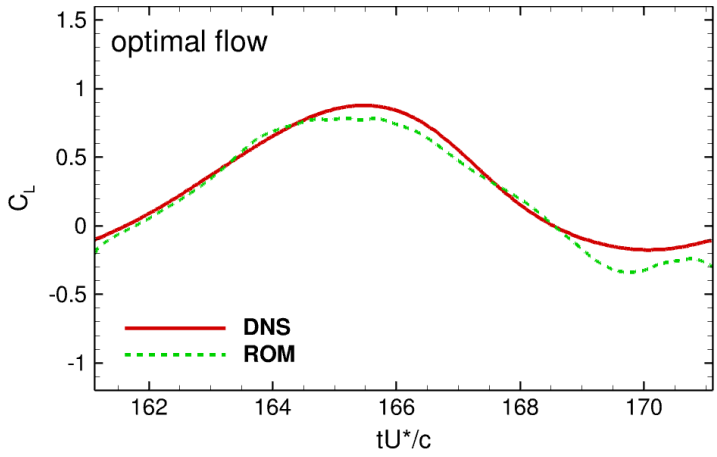

Figure 4.11: *The lift coefficients of the optimal flow after gust mitigation, computed by DNS (solid line) and reconstructed by ROM (dashed line) respectively.*

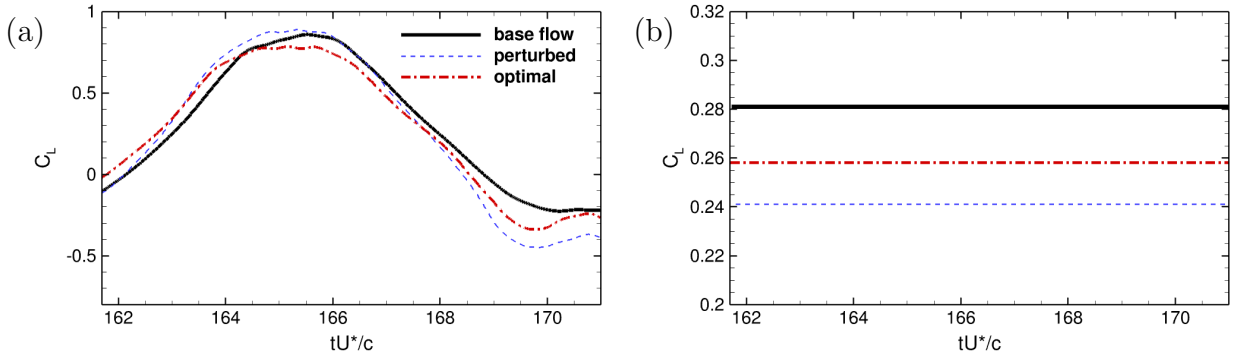

Figure 4.12: *The lift coefficients of the base flow, the perturbed flow with gust, and the optimal flow after gust mitigation. (a): Lift profile; (b): mean lift.*

controllability of adjoint-ROM approach.

It is worth noting that the airfoil had different optimal motion when the streamwise gust blew from different directions in FOM-based and ROM-based cases. Such different motion may be related to the strength of the gust as well as the controllability of different methods. In FOM-based case, the change of Re was large which led to a strong gust. Therefore, the airfoil may need to nearly stop its oscillation to reach a different flow regime in order to retain the pre-gust mean lift. In ROM-based case, the gust was much weaker, which resulted in relatively small deviation in the mean lift. Therefore the airfoil may still maintain its oscillating motion after the optimization.

4.3.2 2D transverse gust mitigation

FOM-based control

The original flow without gust was the same as the one used in the case of FOM-based streamwise gust mitigation. The transverse gust here was introduced by adding a periodic velocity profile from the bottom boundary with the gust intensity $\alpha = 1.0$ and $f = 0.1$ in equation 4.2. The mean AoA β was still restricted in a range of $[-20°, 20°]$ to prevent particularly deep dynamic stall events. The objective function was also given by equation 4.3 with period $T = 10$. The wing motion was the same as the case of streamwise gust mitigation. As presented in figure 4.13, the objective function was decreased by 61.3% when the optimal control was converged after 6 iterations. Correspondingly, the heaving amplitude A was decreased by 28.9% , as shown in figure 4.14 (a), with a final $A = 0.356$. The pitching amplitude θ remained small after optimization with a reduction of 46.8%. The phase delay angle ϕ still did not have significant impact on the gust mitigation. The mean AoA β was decreased from 15° to 3.2° in the end of the optimization, indicating that the airfoil tends to dive into the gust inlet to mitigate the lift surge caused by the gusty wind. Similar adjustment of the wing motion has also been found in previous research with different descriptions of gust effect under much higher Reynolds numbers[132].

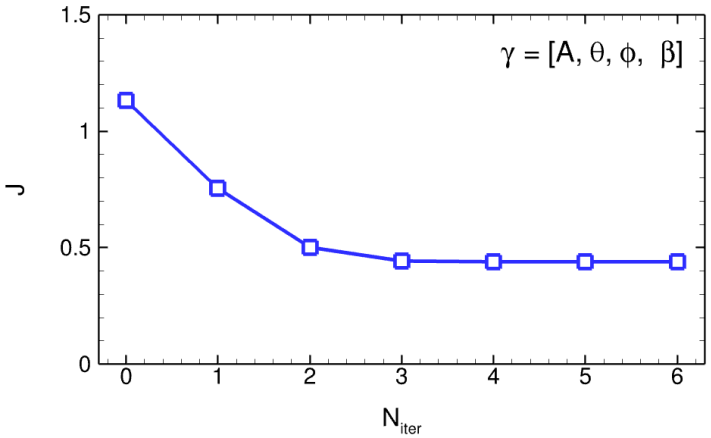

Figure 4.13: *The objective function $\mathcal{J}$ at every main iteration in the optimization.*

The lift coefficients of the base flow, the perturbed flow with transverse gust, and the

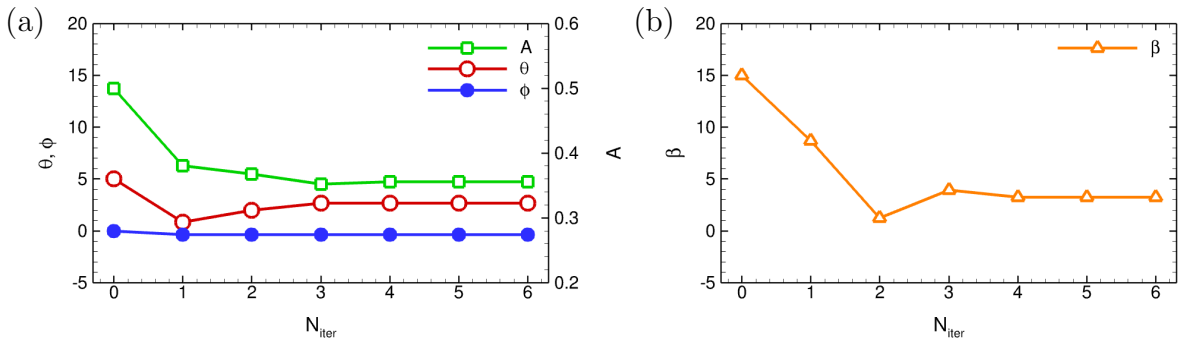

Figure 4.14: *The variations of 4 control parameters in $\boldsymbol{\gamma}$ at every main iteration. (a): The variations of A, θ, ϕ; (b) the variation of mean AoA β.*

optimal flow are compared in figure 4.15. The initial lift was the same as in FOM-based streamwise gust mitigation case with a net mean lift $\bar{C_{L0}} = 0.678$. The unsteady transverse gust significantly changed the lift profile during one stroke period. As seen in figure 4.15 (a), C_L was no longer purely sinusoidal, implying a transition in the flow dynamics was taking place due to the gust. Instead, the former lift valley without gust became a lift peak, followed by another lift plateau. As a result, the mean lift surged by 65.9%, as indicated by figure 4.15 (b). The optimal control managed to recover most of the initial mean lift by reducing A, θ, and β. This set of control parameters caused almost the entire lift curve to shift downward. However, the overall shape of the lift profile almost remained unchanged compared to the perturbed case, and the pre-gust lift profile was not retained by the optimization, just as observed in FOM-based streamwise gust mitigation case. After the optimization, the mean lift was reduced to 0.648, as shown in figure 4.15 (b), with the deviation to $\bar{C_{L0}}$ decreasing from 65.9% to 4.4%. The FOM-based gust mitigation approach was proved effective again by successfully reducing the deviation to $\bar{C_{L0}}$ by over 90%.

The flow fields at two typical time moments $tU^*/c = 290$ and $tU^*/c = 295$, where the airfoil was heaving upward and downward respectively, are presented in figure 4.16 for the base flow, the perturbed flow, and the optimal flow. It can be seen in figure 4.15 (a) that at these two time moments, the lift profile prior to the gust reached the minimum and the maximum respectively. The transverse gust not only shifted the phase of C_L, but also changed the moment where the maximal lift was generated. The base flow was the same as shown in figure

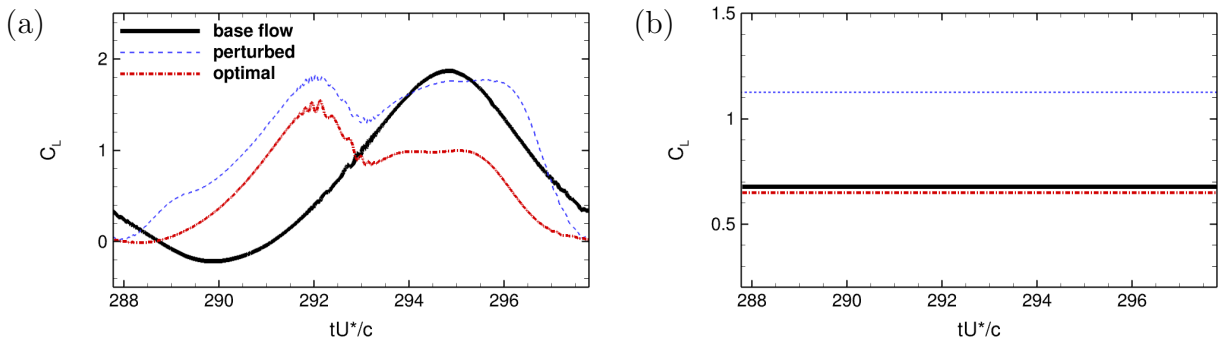

Figure 4.15: *The lift coefficients of the base flow, the perturbed flow with gust, and the optimal flow after gust mitigation. (a): Lift profile; (b): mean lift.*

4.16 (a) and (b). After the transverse gust blowing from the bottom boundary, the vortices were pairing earlier with stronger intensity during the upstroke. During the downstroke, part of the gust was merged into the wake, and interacted with the LEV separated from the airfoil surface. The gust changed the wake structure completely, with the upwash greatly enhancing the vortex shed from the trailing edge, and finally suppressing the shedding of LEV. As a result, the mean lift surged by the upwash effect, and the original vortex pairing disappeared. After the optimization, although no significant transition of vortex structures occurred, the vortex intensity was reduced with weaker vortex pairing in the wake as seen in figure 4.16 (e) and (f). The upwash caused by the gust was alleviated, resulting in a down shift of the lift profile as well as a much lower mean lift. It is demonstrated that when encountering a transverse gust, the stable flight can be maintained with merely slight change in the mean lift by controlling the maneuver of airfoil, without changing the flow structures and flow dynamics dramatically.

ROM-based control

The base flow was first generated by 2D DNS at $Re = 200$ with an initial control $\boldsymbol{\gamma}_i = [0.2, 5°, 0]$, a fixed mean AoA $\beta = 5°$, and oscillating frequency $f = 0.1$. The transverse gust was introduced by the similar way to the one used in FOM-based transverse gust mitigation case, but with a lower gust intensity $\alpha = 0.5$. Note that α is smaller than the one in FOM-based case in order to keep ROM accurate. Then the global POD modes were extracted

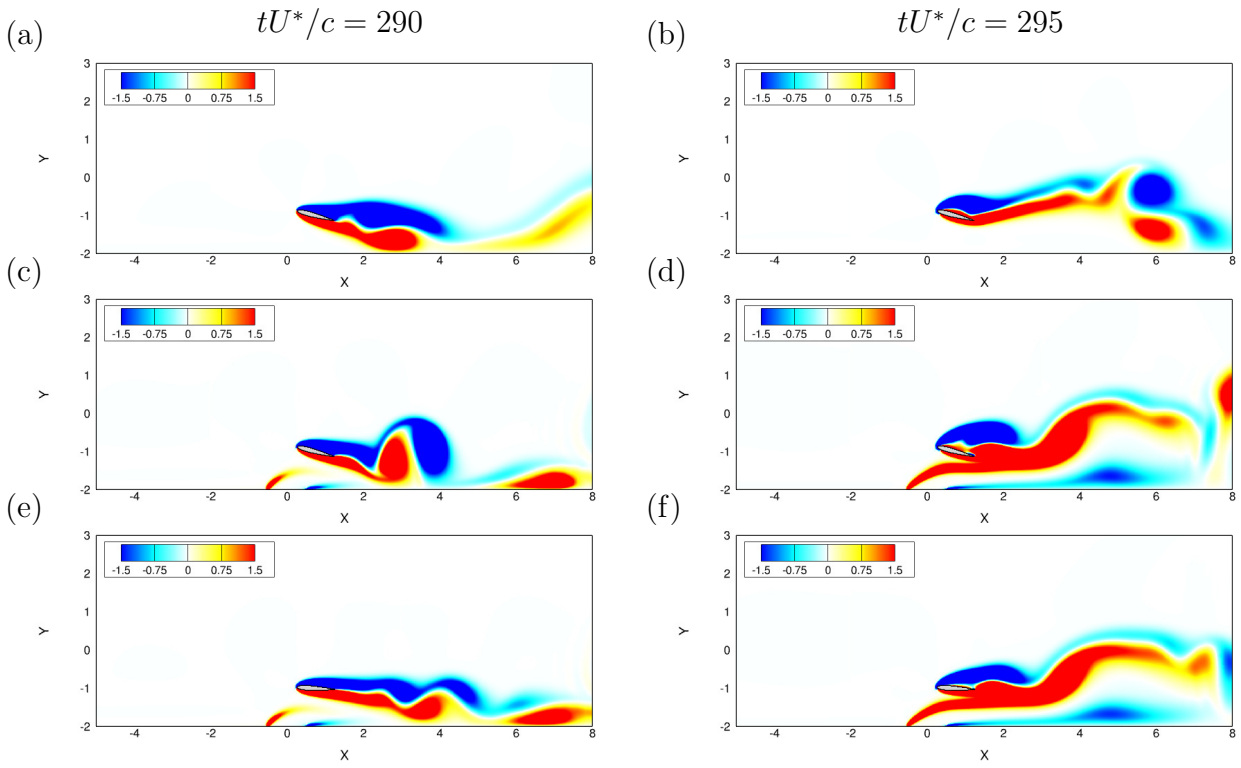

Figure 4.16: *The flow field contoured by vorticity at two typical time moments ($tU^*/c = 290$ and $tU^*/c = 295$) for: (a) and (b) base flows; (c) and (d) perturbed flows; (e) and (f) optimal flows*

based on 300 snapshots with transverse gust covering dimensionless time from $tU^*/c = 161$ to 171 for one wing stroke. With the first 20 leading POD modes and solid modes, a global POD-Galerkin ROM was built for the gust mitigation in lower-dimensional subspace. The accuracy of the ROM was validated first for the base flow as well as the perturbed flow with transverse gust. Figure 4.17 compares the time coefficients of the first two POD modes between the direct linear projection of DNS data (L-DNS) and the solution of ROM equation, for the base flow and the perturbed flow with gust. The L-DNS results can be regarded as the benchmarks. It is found that the time coefficients given by ROMs can match the benchmarks with high accuracy, which demonstrates ROMs can appropriately capture the low-dimensional flow dynamics. In addition, the lift coefficients computed by DNS and ROM reconstruction are also compared in figure 4.18. It can be seen that for both flows, the lift profiles reconstructed by ROMs can almost collapse onto the one by DNS, with only sight disagreements. Hereafter, only the ROM built upon the perturbed flow was used for

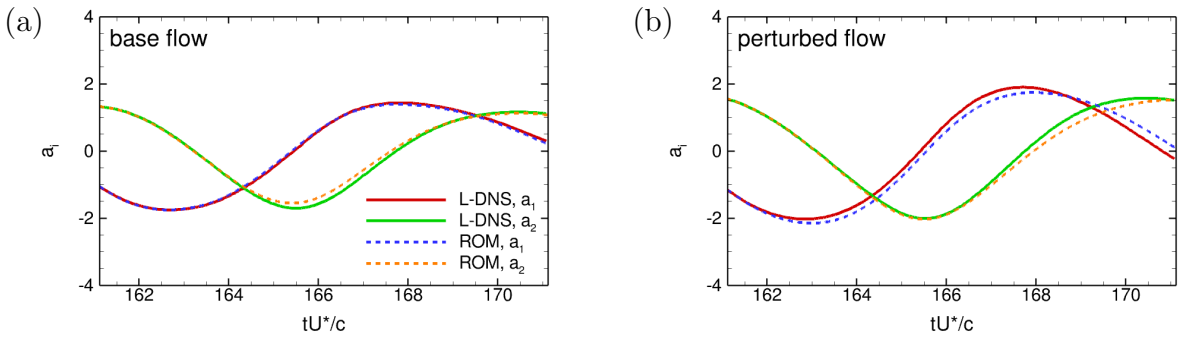

Figure 4.17: *The time coefficients a_i of the first two POD modes by the L-DNS and by solving ROM equation for: (a) the base flow; (b) the perturbed flow with gust.*

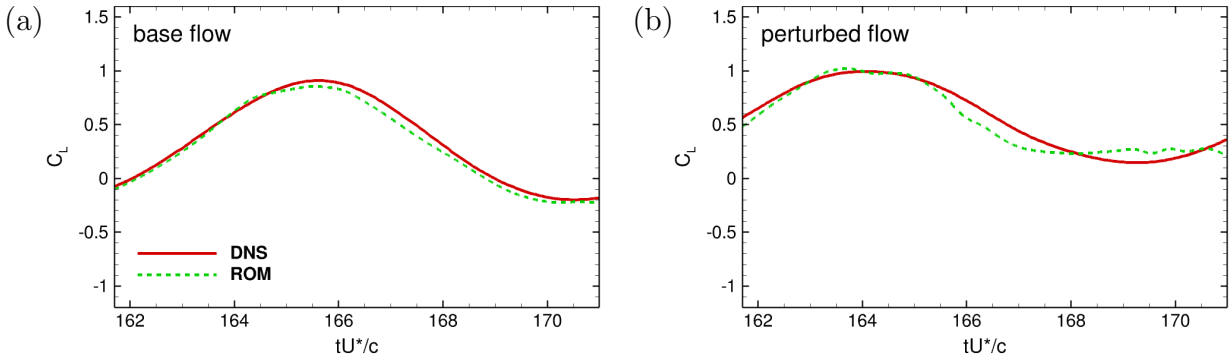

Figure 4.18: *The lift coefficients computed by DNS and ROM for: (a) the base flow; (b) the perturbed flow with gust.*

the mitigation.

The objective function was calculated by equation 4.9 with period $T = 10$. The variation of $\mathcal{J}$ during the gust mitigation iterations is presented in figure 4.19. It is found that the objective function was decreased by 26.9% when the optimal control was converged after 4 iterations. As shown in figure 4.20, the heaving amplitude A was decreased by 83.1%. Note that the negative final A indicates the opposite heaving direction. The pitching amplitude θ was also decreased by 37.9%. The phase delay angle ϕ kept nearly 0 throughout the optimization process, which shows it did not affect the outcome of lift recovering.

The entire on-the-fly mitigation was performed in the subspace, therefore it is intriguing to assess if the optimal solution obtained in this subspace can indeed represent the one of the original high-order space (i.e. the FOM). The accuracy of the ROM for the optimal flow

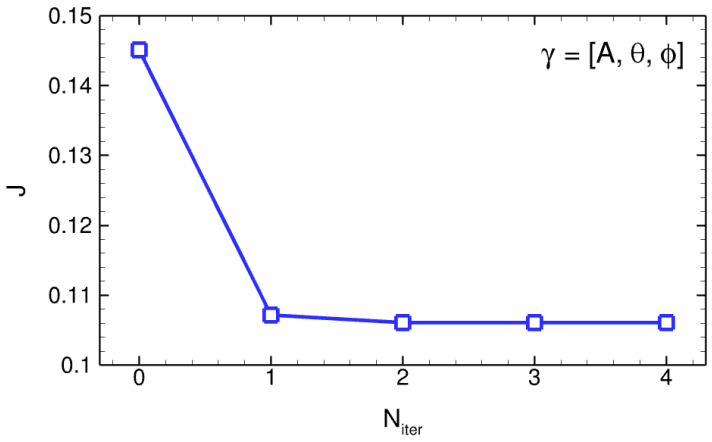

Figure 4.19: *The objective function $\mathcal{J}$ at every main iteration in the optimization.*

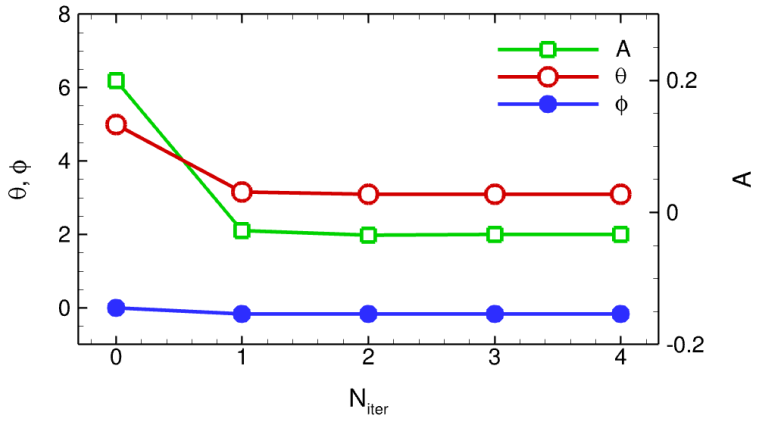

Figure 4.20: *The variations of 3 control parameters in $\boldsymbol{\gamma} = [A, \theta, \phi]$ at every main iteration.*

was evaluated by comparing the lift coefficient computed by DNS and the one reconstructed by ROM as presented in figure 4.21. It can be found that the lift generated by ROM is in good agreement with the benchmark given by DNS. Therefore, it can be confirmed that ROM still kept accurate for the optimal flow.

The lift coefficients of the base flow, the perturbed flow with transverse gust, and the optimal flow are compared in figure 4.22. Note that the lift coefficients shown here were all reconstructed by corresponding ROMs. The initial lift was sinusoidal with a net mean lift $\bar{C_{L0}} = 0.281$. The weak transverse gust altered the phase of the lift profile during one stroke period. with a higher minimum instantaneous lift. As a result, the mean lift was greatly

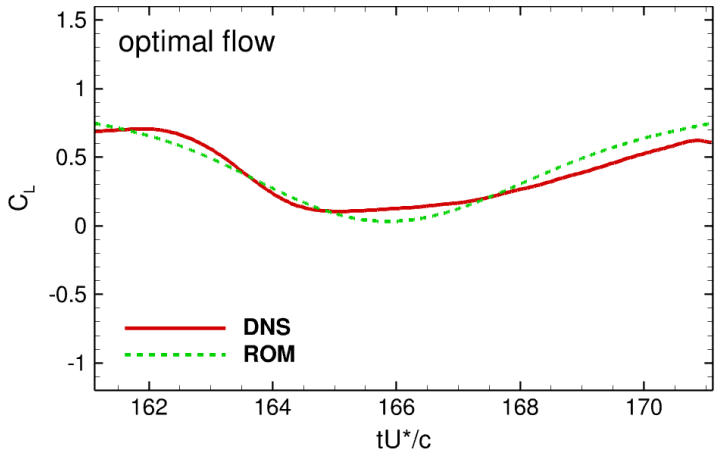

Figure 4.21: *The lift coefficients of the optimal flow after gust mitigation, computed by DNS (solid line) and reconstructed by ROM (dashed line) respectively.*

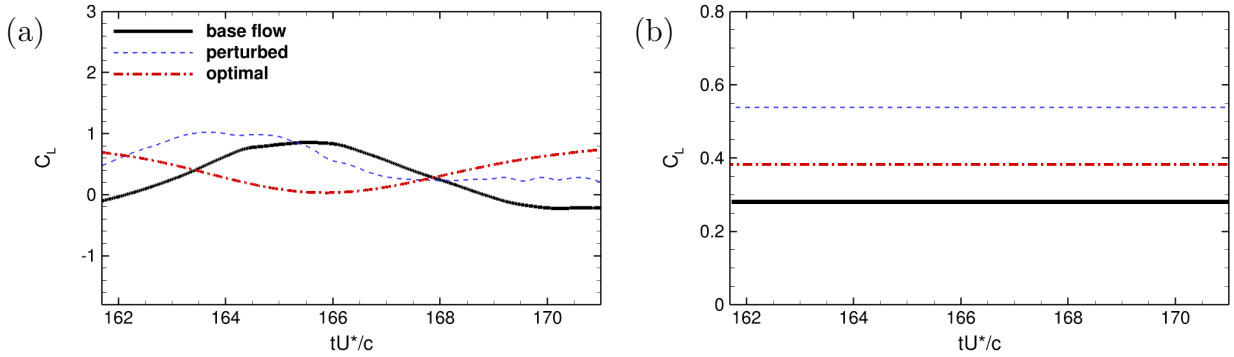

Figure 4.22: *The lift coefficients of the base flow, the perturbed flow with gust, and the optimal flow after gust mitigation. (a): Lift profile; (b): mean lift.*

increased by 91.8% as shown in figure 4.22 (b), despite the magnitude of C_L remaining small. The optimal control was able to alleviate the surge in $\bar{C_L}$ by reducing A and θ. With the optimal maneuver of the airfoil, the phase of the lift profile was changed with a near complete phase shift compared to the initial lift profile. However, the periodicity of the lift profile was still retained. After the optimization, the mean lift was reduced to 0.383, as shown in figure 4.22 (b), as the deviation to $\bar{C_{L0}}$ was reduced significantly from 91.8% to 36.3%, albeit still higher than the target mean lift of the initial flow. This suggests that the fidelity or accuracy of the ROM may limit the effectiveness of the optimization for control of mean lift.

4.3.3 3D streamwise gust mitigation

For this case, the original flow without gust was generated at $Re = 100$ with an initial control $\boldsymbol{\gamma}_i = [0.2, 5°, 0, 15°]$ with oscillating frequency fixed as $f = 0.5$. The streamwise gust was generated by increasing the Reynolds number of incoming flow to $Re = 350$. In order to prevent a severe dynamic stall event at high AoA, β was again restricted to a range of $[-20°, 20°]$. The objective function was given by equation 4.3 with period $T = 2$. In the motion for this case, the airfoil was heaving upward first in $[0, T/2]$, and then downward in $[T/2, T]$. The effect of mitigation is shown in figure 4.23 which presents the variation of the objective function at each iteration. The objective function was found to decrease by nearly two orders of magnitude, or 98.7%, after 2 iterations of optimal control. The corresponding variations of $\boldsymbol{\gamma}$ are shown in figure 4.24. It can be seen that the heaving amplitude A and pitching amplitude θ were both reduced to almost zero, with $A = -1.46 \times 10^{-3}$ and $\theta = -0.035°$. The phase delay angle ϕ remained nearly zero without much difference after the optimization. The mean AoA β was found almost unchanged, with a slight reduction from 15° to 13.8°. Comparing the change of $\boldsymbol{\gamma}$ against the 2D counterpart, it can be found that in both cases, in order to mitigate the streamwise gust, the amplitudes of oscillation needed to decrease to make the wing almost static. The similar change of $\boldsymbol{\gamma}$ for both cases implies the same mechanism might determine the effect of streamwise gust mitigation regardless whether the model is 2D or 3D.

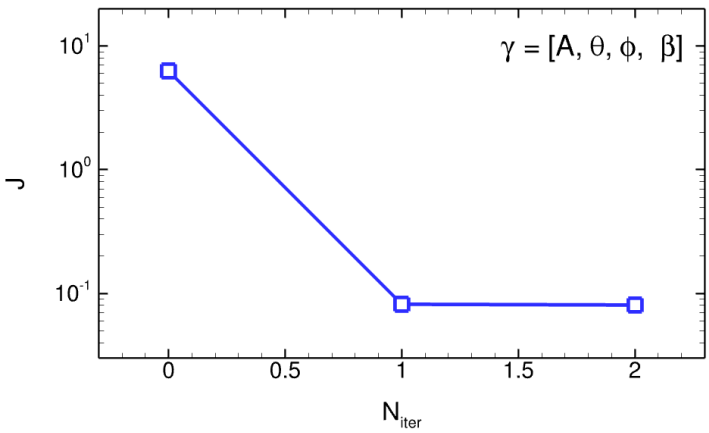

Figure 4.23: *The objective function $\mathcal{J}$ at every main iteration in the optimization.*

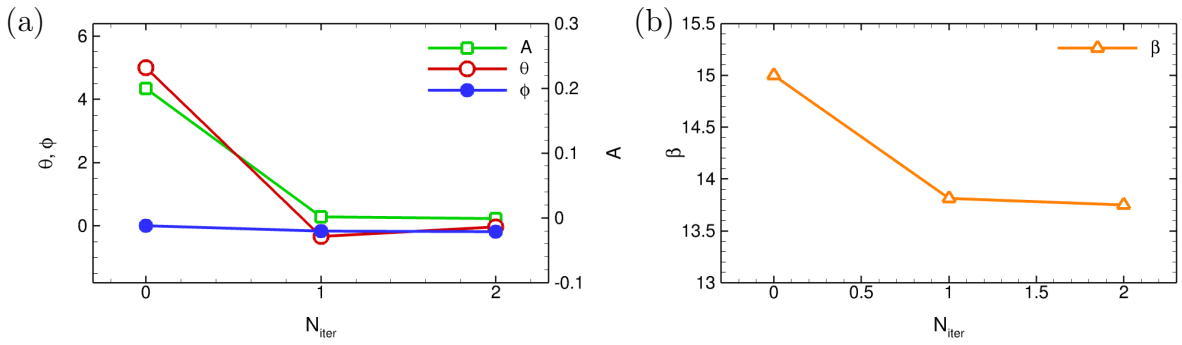

Figure 4.24: *The variations of 4 control parameters in $\boldsymbol{\gamma}$ at every main iteration. (a): The variations of A, θ, ϕ; (b) the variation of mean AoA β.*

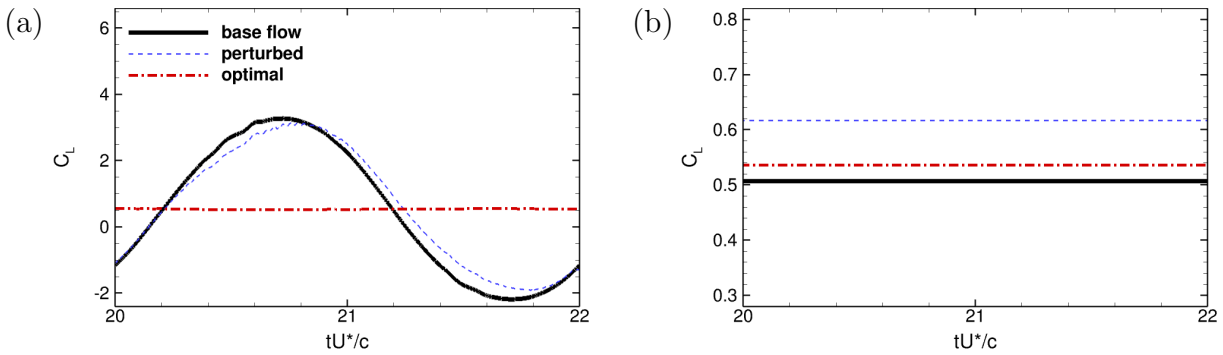

Figure 4.25: *The lift coefficients of the base flow, the perturbed flow with gust, and the optimal flow after gust mitigation. (a): Lift history; (b): mean lift. The small figure in (b) is the zoom-in view of the mean lift of base flow and optimal flow.*

The lift coefficients of the base flow, the perturbed flow with transverse gust, and the optimal flow are compared in figure 4.25. The initial lift profile was sinusoidal with a net mean lift $\bar{C_{L0}} = 0.507$. The streamwise gust from the upstream brought a small change to the lift profile, with slight phase alteration and slightly higher magnitude. As a result, the mean lift $\bar{C_L}$ appeared to increase by 21.7% to 0.617 as shown by figure 4.25 (b). By the optimal control with A, θ being reduced to almost zero, it can be found that the lift profile changed drastically from a periodic profile to a nearly steady profile. Correspondingly the initial mean lift of the base flow was recovered by 73.6% , as seen in figure 4.25 (b), to $\bar{C_L} = 0.536$ in the end of the optimization. The huge alteration of C_L profile indicates the flow regime might change as well, which will be studied by analyzing the flow structures as follow.

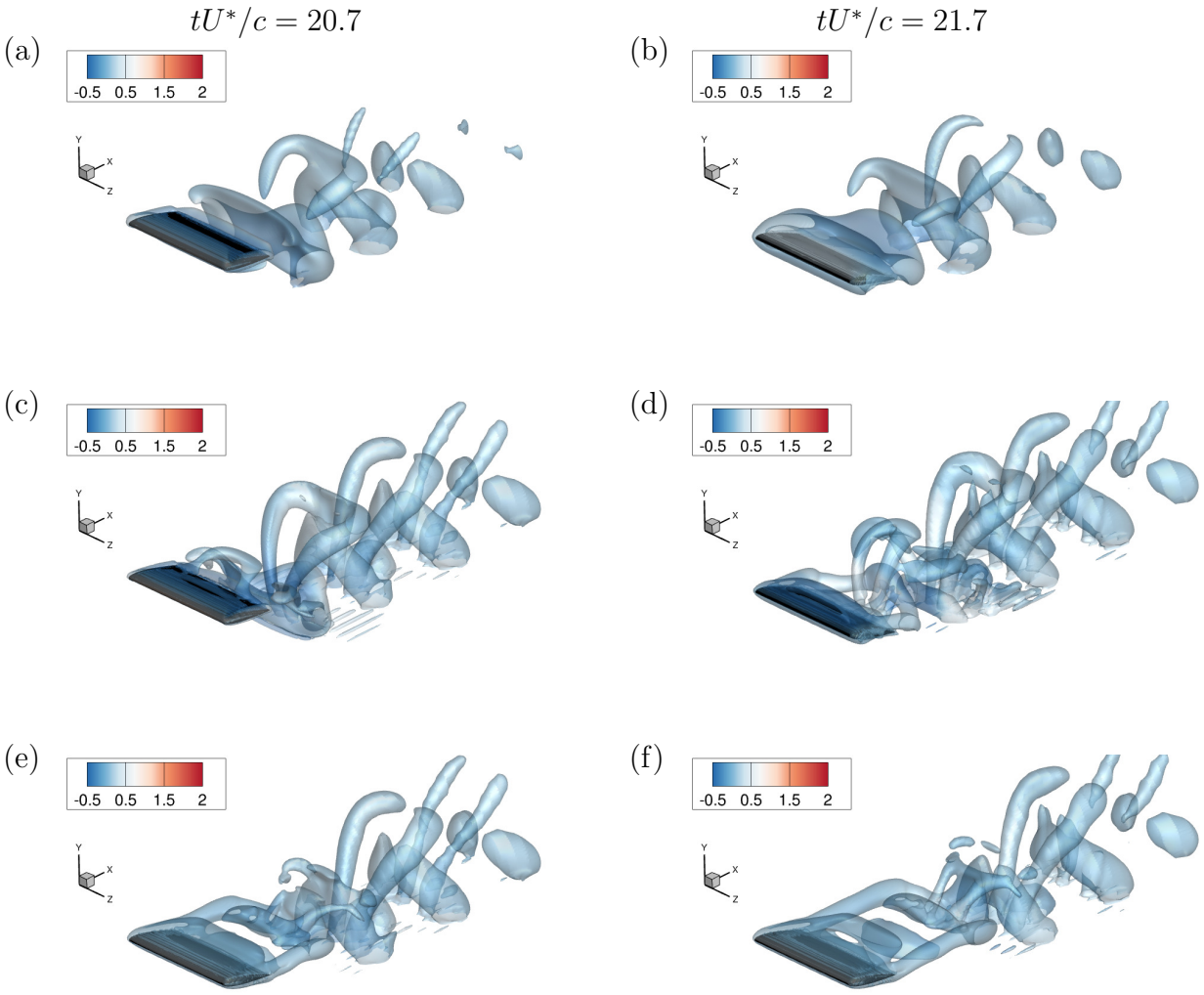

Figure 4.26: *The $Q = 0.1$ iso-surfaces contoured by vertical (y) velocity at two typical time moments ($tU^*/c = 20.7$ and $tU^*/c = 21.7$) for: (a) and (b) base flows; (c) and (d) perturbed flows; (e) and (f) optimal flows.*

The flow fields were rendered by $Q = 0.1$ iso-surfaces at two typical time moments $tU^*/c = 20.7$ and $tU^*/c = 21.7$, as presented in figure 4.26 for the base flow, the perturbed flow, and the optimal flow. In addition, the corresponding vorticity fields at the spanwise central plane $z = 0$ are illustrated in figure 4.27. It can be observed that for the base flow, due to the heaving and pitching wing motion, a single vortex street was formed even with a modest $Re = 100$. The flow separation already happened on the upper side of the wing as shown in figure 4.27, with a trailing edge vortex (TEV) rolling up simultaneously. As

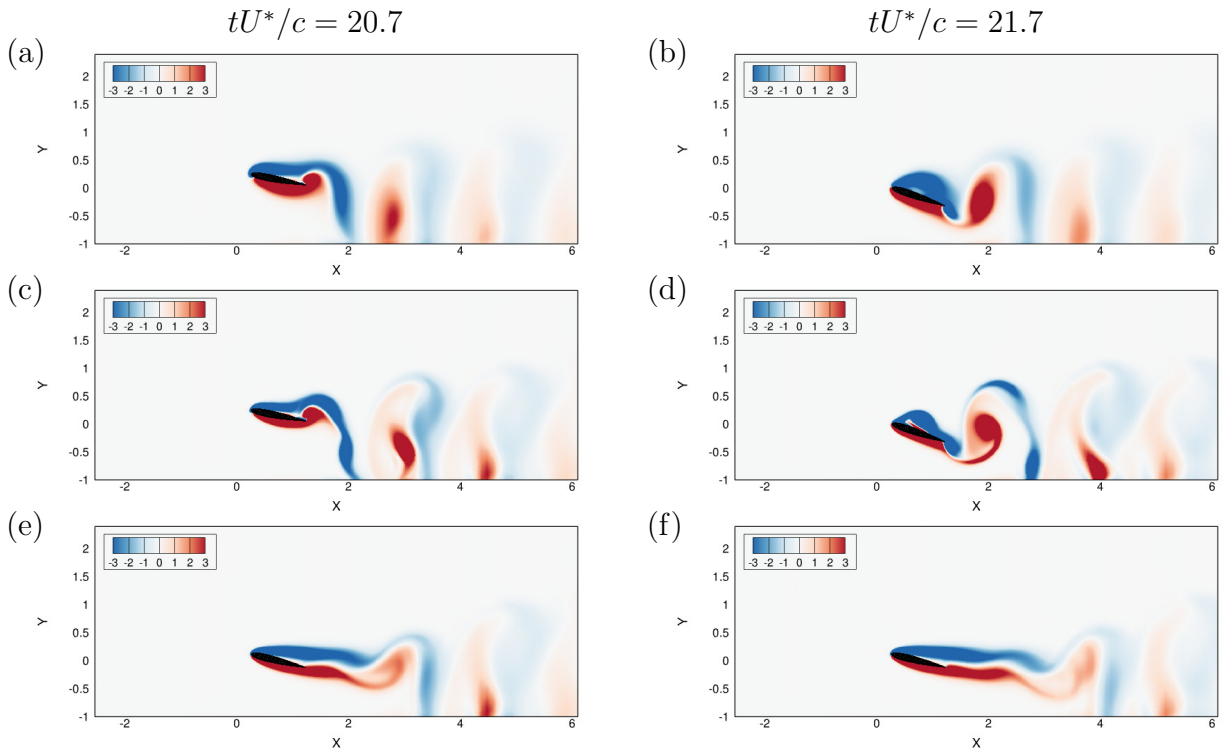

Figure 4.27: *The vorticity field on the $z = 0$ central plane at two typical time moments ($tU^*/c = 20.7$ and $tU^*/c = 21.7$) for: (a) and (b) base flows; (c) and (d) perturbed flows; (e) and (f) optimal flows.*

the streamwise gust blown from the upstream, stronger vortices of separation were generated with more active pairing in the wake. Such change of vortex structures led to a small increase in the mean lift. After the optimization, the wing almost stopped its oscillation, leading to the wake structure that the flow past a static bluff body would have, as seen in figure 4.27 (e) and (f). The flow reattachment and weaker vortex pairing in the wake may lead to the nearly steady C_L, which may further result in a good recovery and stabilization of the lift.

4.3.4 3D transverse gust mitigation

The original flow without gust was generated at $Re = 100$ with an initial control $\boldsymbol{\gamma}_i = [0.2, 5°, 0, 15°]$ and oscillating frequency $f = 0.5$. The transverse gust was generated in the way that was introduced in section 4.2, with $\alpha = 1.0$ and the same frequency as the wing oscillation $f = 0.5$. In order to prevent the severe dynamic stall event at high AoA, β was also restricted to a range of $[-20°, 20°]$. The objective function was still given by equation

4.3 with period $T = 2$. In this motion profile, the airfoil was able to heave upward first in $[0, T/2]$, and then downward in $[T/2, T]$. As shown in figure 4.28, the objective function was reduced by over two orders of magnitude, or 98.6%, after 6 iterations of optimal control. The corresponding variations of $\boldsymbol{\gamma}$ are shown in figure 4.29. It can be observed that the heaving amplitude A and pitching amplitude θ were both reduced notably, with $A = 0.021$ and $\theta = 0.13°$. The phase delay angle ϕ remained close to zero without much change after the optimization. The mean AoA β was found significantly reduced from 15° to −16.2°, which means the wing pitched down dramatically towards the direction of the gust inlet. Comparing the change of $\boldsymbol{\gamma}$ against the 2D counterpart, it can be found that in both cases, in order to stabilize the aerodynamic performance when the transverse gust occurred, the amplitudes of oscillation needed to decrease, with the whole wing pitching downward to the gust inlet to adjust the incident angle. The similar change of $\boldsymbol{\gamma}$ for both cases implies the same mechanism for the gust mitigation regardless of 2D or 3D model.

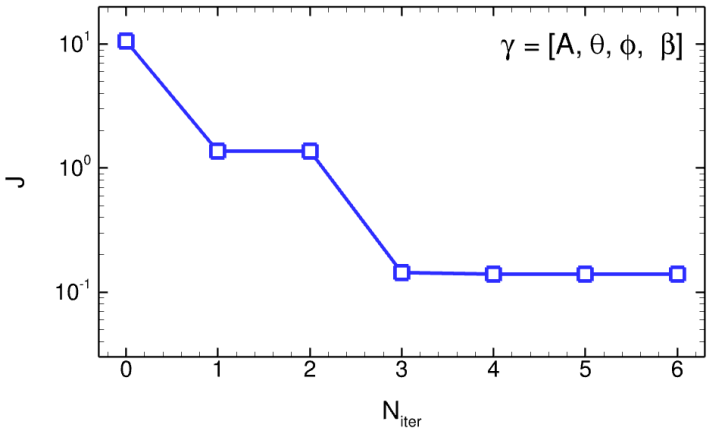

Figure 4.28: *The objective function $\mathcal{J}$ at every main iteration in the optimization.*

The lift coefficients of the base flow, the perturbed flow with transverse gust, the optimal flow are compared in figure 4.30. The initial lift was the same as in 3D streamwise gust mitigation case with a net mean lift $\bar{C_{L0}} = 0.507$. The unsteady transverse gust overall enhanced the lift greatly, with some shift in the phase of the lift profile as well. As a result, the mean lift appeared to surge by 256.5% as indicated by figure 4.30 (b). Such strong disturbance of the lift will definitely harm the flight stability of the 3D wing. However, by

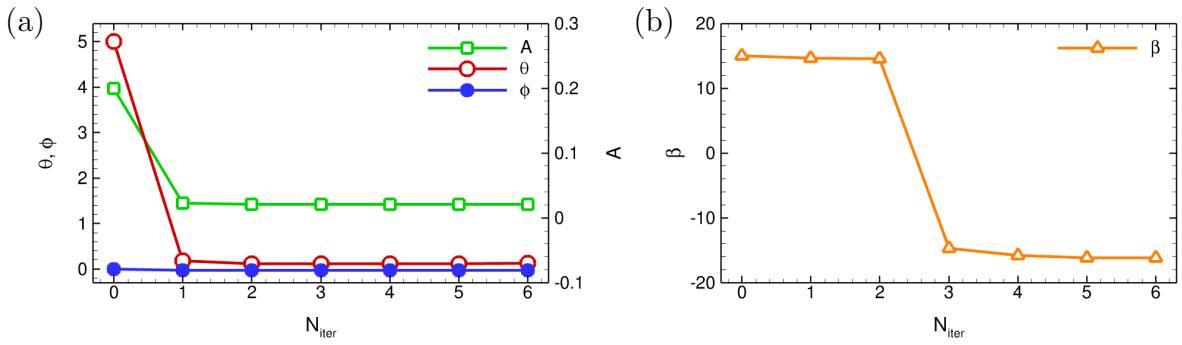

Figure 4.29: *The variations of 4 control parameters in $\boldsymbol{\gamma}$ at every main iteration. (a): The variations of A, θ, ϕ; (b) the variation of mean AoA β.*

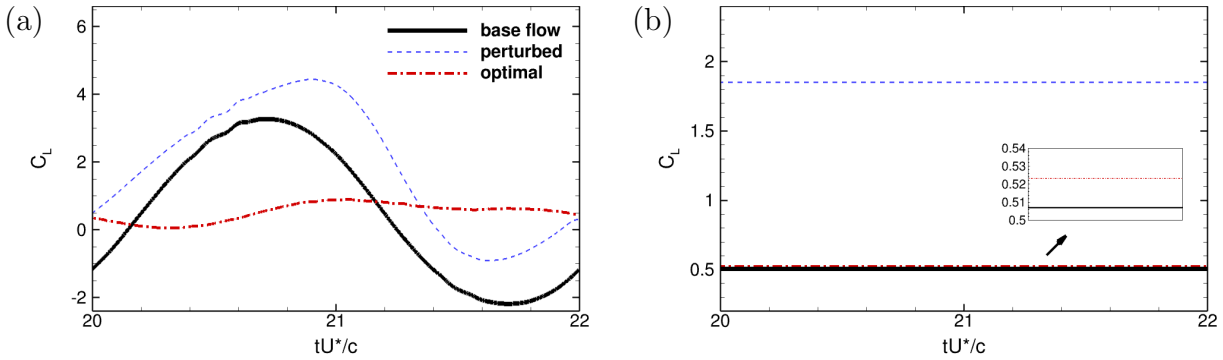

Figure 4.30: *The lift coefficients of the base flow, the perturbed flow with gust, and the optimal flow after gust mitigation. (a): Lift history; (b): mean lift. The small figure in (b) is the zoom-in view of the mean lift of base flow and optimal flow.*

the optimal control with reduced A, θ, and β, the initial mean lift of the base flow was able to be mostly recovered, as seen in figure 4.30 (b), to $\bar{C_L} = 0.523$ in the end of the optimization. The optimal maneuver of the wing can effectively reduce the force peak caused by the gust, with the whole lift profile becoming more steady. The optimal mean lift had a deviation to $\bar{C_{L0}}$ of only 3.2%, which demonstrates the high effectiveness of the present FOM-based gust mitigation approach on 3D models. Note that different than the 2D counterpart, the oscillation of the original lift was almost suppressed by the optimization, though the mean lift was well recovered.

The flow fields were rendered by $Q = 0.1$ iso-surfaces at two typical time moments $tU^*/c = 20.7$ and $tU^*/c = 21.7$, as presented in figure 4.31 for the base flow, the perturbed flow, and the optimal flow. In addition, the corresponding vorticity fields at the spanwise

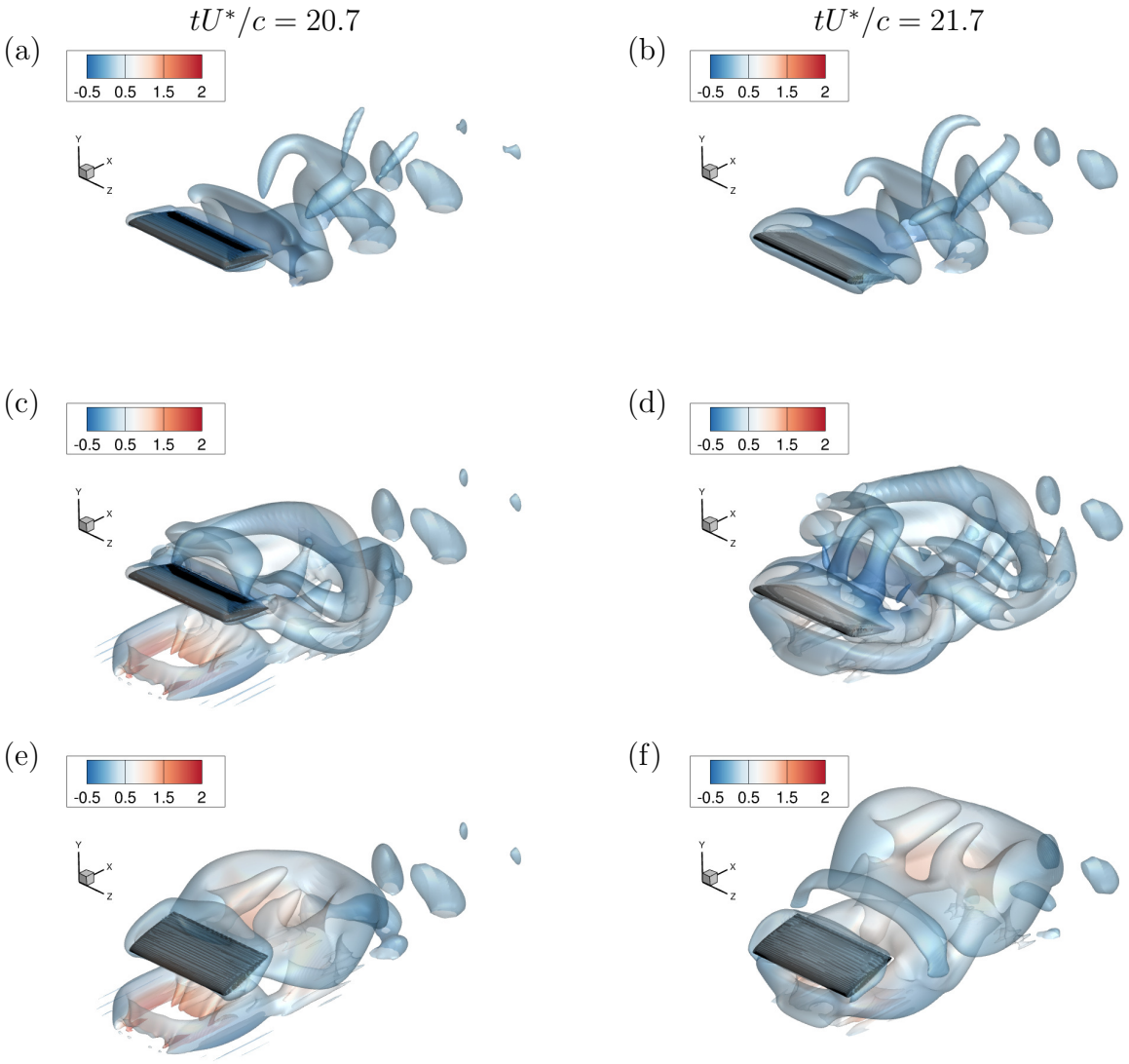

Figure 4.31: *The $Q = 0.1$ iso-surfaces contoured by vertical (y) velocity at two typical time moments ($tU^*/c = 20.7$ and $tU^*/c = 21.7$) for: (a) and (b) base flows; (c) and (d) perturbed flows; (e) and (f) optimal flows.*

central plane $z = 0$ are illustrated in figure 4.32. The base flow is the same as analyzed in 3D streamwise gust case. As the unsteady transverse gust with $\alpha = 0.5$ occurred, the vortex structures were altered significantly. A horseshoe-like LEV was generated with strong flow separation on the upper side of the wing. In addition, the LEV started intertwining with the gust, which resulted in a much more complicated flow structure in the near-solid field. The regular single vortex street disappeared, while the LEV and TEV shed from the wing began to pair. Eventually, the strong perturbation brought the upwash that was able to shifted

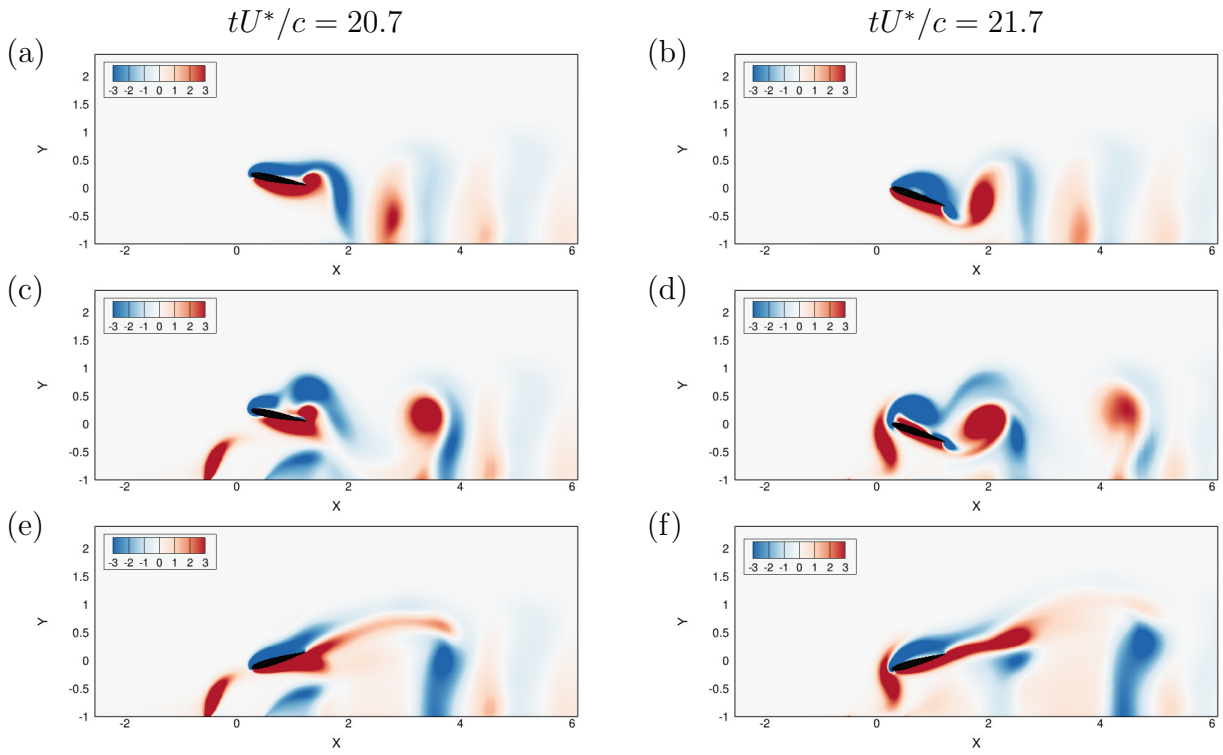

Figure 4.32: *The vorticity field on the $z = 0$ central plane at two typical time moments ($tU^*/c = 20.7$ and $tU^*/c = 21.7$) for: (a) and (b) base flows; (c) and (d) perturbed flows; (e) and (f) optimal flows.*

mean lift up by over 200%. However, after the optimization, the flow structures changed dramatically again, with LEV almost reattaching to the upper side of the wing. The TEV became weaker as well, as seen in figure 4.32 (e) and (f). The flow reattachment and weaker vortex pairing in the wake directly resulted in a more steady C_L with much lower magnitude. Similar to the 2D case, the upwash caused by the gust can still be mitigated effectively by pitching downward into the direction of the gust inlet, in order to generate more coherent flow structures on the wing instead of huge separation, which further resulted in a good recovery of the original mean lift, but with a dissimilar motion profile.

4.3.5 Computational cost estimation

The computational cost of adjoint-based optimal control was studied for the 2D and 3D streamwise gust mitigation cases by using FOM-based or ROM-based adjoint approach, as shown in table 4.3. An estimation was made on the computational time by using parametric

Table 4.3: *Computational time for DNS-based adjoint optimal control process and adjoint-ROM optimal control process of both 2D and 3D streamwise gust mitigation. The computational time of parametric study is estimated by assuming to have 5 attempts for each control parameter based on the online computational time of a single simulation. Only online computational time is presented.*

Case	Single simulation	Parametric study	Adjoint-based optimization
2D, FOM-based, 4DoF	45m	19d	8h12m
2D, ROM-based, 3DoF	10s	25m	3m
3D, FOM-based, 4DoF†	4h16m	111d	2d16h

† run in parallel with 20× CPU power.

study such that 5 simulations are presumably performed for each control parameter. As for a 4-DoF control, such assumption would result in 5^4 simulations in total to find the optimal solution by parametric study. For cases using FOM-based approach, 45 minutes are needed for a 2D simulation and over 4 hours are required for a 3D case even running with 20 CPU cores. Consequently the parametric study based on the aforementioned assumption would take weeks or even months, which is infeasible. As a comparison, the FOM-based adjoint approach is able to complete the optimization in about 8 hours for 2D case, and in less than 3 days for 3D case respectively. This results in over 98% computational time reduction was achieved for both cases, with the adjoint approach enabling flow optimization that is impossible by the parametric study. For the case using an adjoint-ROM approach, due to the fact that the online computational time of ROM simulation can be as short as seconds, the parametric study is feasible, yet it still takes too much time to conduct real-time control. By using adjoint-ROM approach with "one-ROM" strategy, the computational cost was further reduced by 88%, which allows for a quick optimization in 3 minutes and opens up a way for near-real-time control.

4.4 Concluding remarks

In this chapter, the adjoint-based flow optimization approach has been applied to gust mitigation for both 2D and 3D heaving-pitching wings. The streamwise gust and the transverse gust were studied separately, where the streamwise gust was introduced by changing the Reynolds number of the incoming flow, and the transverse gust was generated by imposing a sinusoidal velocity profile from a gust inlet at the bottom boundary of the computational domain. The FOM-based adjoint approach was used for all cases, while the adjoint-ROM approach developed in chapter 3 was used for 2D gusts only. The gust responses were evaluated by the deviation of the lift after gust encounter to the mean lift over one oscillating stroke. The mitigation of gust was realized by minimizing the objective function to recover the mean lift of the base flow, while keeping the lift profile as steady as possible.

The FOM-based adjoint approach was effective to mitigate relatively strong streamwise and transverse gusts for both 2D and 3D flows. For the streamwise gust, the wing tended to stop its oscillation to stay nearly static, in order to lower the extra lift caused by the gust blown from upstream, as well as to reduce the magnitude of the life profile for a more steady aerodynamic load. The phase delay angle ϕ and mean AoA β were found insignificant for the gust mitigation. After the optimization, a transition in the flow regime had occurred for both 2D and 3D flows, with wake structures very close to the one of flow past a fixed bluff body. The deviations to the mean lift of the base flow could be reduced by up to 73.6%. For the transverse gust, the wing tended to reduce its oscillating amplitudes, while pitching down towards the gust inlet with significantly lower β, to mitigate the great surge in lift when encountered the gust. The phase delay angle ϕ remain almost irrelevant to the effect of gust mitigation. For the 2D gust, the flow control did not change the flow dynamics with the gust completely. But by reducing the vortex intensity as well as the vortex pairing activity in the wake, the whole lift profile was shifted down to overcome the upwash caused by transverse gust, which further recovered the original mean lift by over 90%. For the 3D gust, however, a transition of the flow dynamics was observed, where the LEV and TEV generated from flow separation due to the gust started reattaching to the wing surface. The

flow reattachment and weaker vortex pairing in the wake resulted in a more steady lift profile with much lower magnitude, eventually led to a final mean lift with only 3.2% deviation to the original mean lift.

The adjoint-ROM approach was applied to cases with weak gusts, and without mean AoA as a control parameter. The "one-ROM" strategy was implemented for both gusts where a single ROM built upon the perturbed flow was used throughout the control process. The ROM was validated by comparing the lower-order flow dynamics by solving ROM equations as well as the lift reconstruction against the results of FOM. Overall, adjoint-ROM was able to effectively reduce the deviation of mean lift after gust encounter to the original mean lift by over 50%. It was realized by only controlling the lower-order flow dynamics, while the optimal mean lift achieved from this surrogate model was proved accurate compared to the one from FOM.

Chapter 5

Adjoint-based optimization for hydrofoil schooling

5.1 Introduction

It has been hypothesized for decades that the swimming animals may be able to make use of flows induced by neighboring swimmers when schooling, to obtain hydrodynamic benefits, including reducing the drag, boosting the thrust, and enhancing the propulsion efficiency[140–142]. Extensive work has been done attempting to elucidate the mechanisms of possible benefits brought by schooling, as well as to seek the ways of schooling that can lead to better hydrodynamic performance. Some pioneering work has been conducted by using living fishes and fish-like robotics, which has indicated that through the collective motion and vortex phase matching, individuals in a fish school can achieve hydrodynamic benefits such as lower energy consumption[143–145]. However, since the number of swimmers in a fish school is usually huge, the interactions between swimmers and surrounding flows are very complicated, which makes it difficult to thoroughly analyze the hydrodynamic performance by merely studying living fish school. Therefore, fish schooling has usually been studied by simplified models. Some work has modeled the fish school as self-propelled particles, which can only take into account limited hydrodynamics[146]. On the other hand, in order to

further investigate the complex hydrodynamic interaction of fish schooling, more work has been focusing on models consisting of moving foils and fins with different arrangements of formations. There have been two major categories of foil models that have attracted the most attention from fluid mechanics community. One is the rigid flapping hydrofoil model which could heave and pitch freely in a prescribed incoming flow. This model has been found to be effective at aiding in the identification of the fundamental mechanisms of fish schooling by enabling precise control, and measurement of motions and forces[147–154]. In addition, the experimental design and manufacturing of rigid foils is usually easier. Another model is the flexible undulatory hydrofoil model. This model is derived from the spine motion of fish[155], and has been widely used to study the effects of spatial arrangement of the fish and features of tail beating on the hydrodynamic performance of fish schooling[156–160]. In general, the hydrodynamic performance of schooling has been found highly dependent on the motion of each fish (i.e., flapping, undulating. etc.), as well as the spacing or the formation of the fish. These two major factors are reviewed as follow.

For rigid swimmers, most of the research focusing on the effect of hydrofoil motion has studied harmonic pitching for each foil. Boschitsch et al.[147] and Dewey et al.[148] investigated the propulsive performance of two unsteady hydrofoils schooling in in-line and side-by-side configurations experimentally. It was found that the thrust production as well as the propulsive efficiency was highly dependent on the phase delay angle between oscillating foils for both formations. Similar dependence of propulsive performance on the phase delay angle for side-by-side formation was also observed at higher Reynolds number (Re) by direct numerical simulations (DNS)[161]. Two flapping wings in tandem were studied experimentally and numerically at $Re = 5000$ by Lua et al.[150]. Two types of wing-wake interactions were discovered and a maximum thrust was achieved on the rear wing when the phase angle was $0°$. Multiple-foil schooling systems have been further studied with more foils. Yuan et al.[149] has reported that the propulsive performance of the triple-foil schooling system could be significantly enhanced compared to the dual-foil counterpart by choosing proper phase delays. In the work by Han et al.[153], triple-foil schooling system was also investigated numerically for both two-dimensional (2D) and three-dimensional (3D) models over a wide range of phase

delay angles between foils from 0° to 360°. It was found that by a comprehensive parametric study, some optimal phase delay angles can be determined in a limit parametric space to maximized the thrust as well as the propulsive efficiency. A 2D flapping motion including harmonic heaving and pitching for foils arranged in tandem formation was studied by Ji et al.[162]. An active learning method was leveraged to optimize the propulsion performance by tuning the flapping motion. Using flexible swimmers model, Gao and Triantafyllou[156] studied the effect of swimmer's caudal fin pitching on the reduction of self-propulsion energy in the wake of an upstream swimmer. Park and Sung[157] studied the schooling behavior of flexible fins by adding transverse heaving motion on the leading edge of each fin, while the body is passively driven by the surrounding fluid to undulate. It was found that the following swimmers were able to reduce the heaving amplitude to optimize the propulsive efficiency regardless whether they were schooling in triangle or diamond formation.

The geometrical arrangements of formation and the spacing distance between swimmers have also been found to play important roles in the hydrodynamic performance of schooling. For rigid hydrofoils, Newbolt Joel et al.[151] investigated uncoordinated swimmers experimentally, and found the flapping kinematics can control the locomotion of the follower in the wake which promoted group cohesion consequently. Heydari and Kanso[152] utilized vortex sheet model to analyze the locomotion dynamics of actively flapping swimmers. They found different heaving and pitching motion can lead to different cohesion behavior to achieve energetic benefits. The findings were consistent with many previous numerical and experimental results, which indicated such modulated formation is robust to the flapping mode. As for the flexible swimmer model, Hemelrijk et al.[163] numerically studied various schooling configurations, and suggested the optimal lateral distance for a diamond formation was 1.6 times the body length (BL) of swimmer. Daghooghi and Borazjani[164] studied the rectangular formation with 3D numerical simulation, and found the optimal power efficiency can be achieved with lateral distance equal to 0.4 BL. Recently, high-fidelity DNS was conducted by Pan and Dong[158] to investigate the density effect of a diamond formation on the hydrodynamic performance of schooling. It was found that dense school can achieve higher thrust production as well as higher propulsive efficiency. The lower spacing induced a wall effect to

generate an angled jet, which eventually benefitted the thrust production.

Despite ample studies having been conducted on the complex flow interactions of schooling, no study has demonstrated the conclusive reasoning for schooling behaviors. Moreover, the optimal motion or formation of swimmers that correspond to the optimal hydrodynamic performance is still too difficult to achieve using current numerical or experimental approaches, because of the huge parametric space fish school can have. Recently, Ji et al. [162] proposed leveraging an active learning method to optimize the propulsion performance as reviewed earlier. Nonetheless, the data training was still computationally expensive which may limit the size of parametric space the method can feasibly explore. Additionally, deep learning based approaches often struggle to provide comprehensive physical understandings of fluid flows.

As a different route, the adjoint-based approach is able to handle a large number of control parameters simultaneously without significantly increasing the computational cost, as reviewed and used in previous chapters. Therefore it became a natural choice to optimize the hydrodynamic performance of swimmer schooling. In this work, the high-fidelity numerical code used in previous chapters was upgraded to allow for multiple 2D solid bodies. The hydrodynamic performance was evaluated by the horizontal forces (i.e., drag or thrust) on following hydrofoils. For convenience, hereafter in this chapter "swimmers" will also be used to refer to hydrofoils. Both rigid and flexible swimmer models were studied for optimized swimmer motion and formation to achieve lowest drag or highest thrust at low Reynolds numbers. The adjoint-based approach was implemented with full-order model (FOM) simulated by solving incompressible Navier-Stokes equation. The remainder of this chapter is outlined as follows. Section 5.2 introduces the validation of the numerical code, computational configurations, swimmer models, and the objective functions. The results of optimizations of hydrofoil schooling are presented in section 5.3, with the discussion on flow physics. Section 5.4 draws the concluding remarks of this work.

5.2 Methodology

The flow past multiple solid bodies is still governed by the incompressible Navier-Stokes equation modified with the forcing term from the immersed boundary method (IBM) (equation 1.3). The 2D solver has been well validated for flows past a single solid body. In the present work, the 2D solver has been modified and upgraded to allow for simulations with multiple solid bodies. The new version of this 2D solver will be validated first in the following section.

5.2.1 Validation of numerical code

The validation of aforementioned numerical code was conducted by simulating the flow past dual fixed 2D cylinders in tandem and side-by-side formations. $Re = 200$ and two different distances ($L = 1.5D$ and $L = 3D$ where D is the diameter of each cylinder) between the centers of cylinders were studied. The flow domain had a size of $80D \times 60D$, which was discretized by a non-uniform Cartesian mesh. Two typical snapshots of the flow field with $L = 1.5D$ for both configurations are shown in figure 5.1. First the independence of the

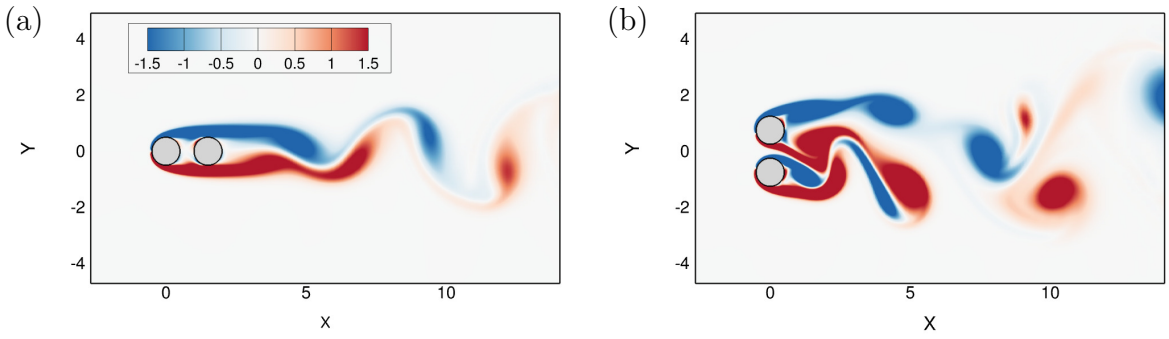

Figure 5.1: *The vorticity fields of flow past dual fixed tandem (a) and side-by-side (b) cylinders. $Re = 200$.*

results on mesh was examined by using three different grids for both arrangements, then a mesh with size of 1001×501 was implemented for all tandem cases and 901×801 for all side-by-side cases in this section. The Courant-Friedrichs-Lewy (CFL) number was fixed to 0.1 to guarantee the precision of all simulations. The quantitative validation on average

forces and Strouhal numbers is presented in table 5.3 for tandem cases and table 5.2 for side-by-side cases, compared to the results reported in Meneghini et al.[1] with the same flow configurations. It can be found that the average drag and lift for both configuration are overall in good agreement with benchmark results, with the biggest error in average drag and lift equal to 7.1% and 4.5% respectively. The Strouhal numbers (St) can also match the benchmarks well with the biggest error equal to 3.2%. The validation results indicate that the current numerical code is able to accurately resolve the flow dynamics with multiple solid bodies.

Table 5.1: *Drag and Strouhal numbers in the present work (before the slash) in comparison with Meneghini et al.[1] (after the slash) for tandem cases. Subscript "1" means the leading cylinder and "2" means the trailing cylinder.*

L	$\overline{C_{D1}}$	$\overline{C_{D2}}$	St_1	St_2
$1.5D$	1.092/1.06	$-0.193/-0.18$	0.163/0.167	0.163/0.167
$3D$	0.997/1.0	$-0.122/-0.08$	0.121/0.125	0.121/0.125

Table 5.2: *Forces in the present work (before the slash) in comparison with Meneghini et al.[1] (after the slash) for side-by-side cases. Subscript "1" means the upper cylinder and "2" means the lower cylinder.*

L	$\overline{C_{D1}}$	$\overline{C_{D2}}$	$\overline{C_{L1}}$	$\overline{C_{L2}}$
$1.5D$	1.39/1.32	1.39/1.32	0.418/0.40	$-0.418/-0.40$
$3D$	1.51/1.41	1.51/1.41	0.116/0.10	$-0.116/-0.10$

5.2.2 The objective function and the gradient

The main goal of this work is to optimize the horizontal hydrodynamic force on the trailing solid bodies. Therefore, the objective function $\mathcal{J}$ is defined accordingly as:

$$\mathcal{J} = \frac{1}{TD_0}\int_T \int_{\partial\Omega_s} \sigma_{1j} n_j ds dt, \tag{5.1}$$

where $D_0 = 1/2\rho U^{*2} D$, σ is the viscous stress, T is one period of solid oscillation, and subscript "1" represents the horizontal direction. Note that during the optimization, a positive $\mathcal{J}$ represents the drag, and a negative $\mathcal{J}$ represents the thrust. The FOM-based adjoint approach enabled by the non-cylindrical calculus toolbox, as implemented in chapter 4, was used here. Within this framework, the adjoint equation $\mathcal{N}^*(\boldsymbol{q})\boldsymbol{q}^* - \mathcal{F}^* = 0$ based on the objective function (equation 5.1) is given as:

$$\begin{aligned} \mathcal{N}^*(\boldsymbol{q})\boldsymbol{q}^* &= \begin{bmatrix} \frac{\partial u_j^*}{\partial x_j} \\ \\ \frac{\partial u_i^*}{\partial t} + u_j(\frac{\partial u_i^*}{\partial x_j} + \frac{\partial u_j^*}{\partial x_i}) + \nu\frac{\partial^* u_i^*}{\partial x_j^2} + \frac{\partial p^*}{\partial x_i} \end{bmatrix}, \\ \mathcal{F}^* &= 0, \quad \text{in} \quad \Omega \\ u_i^* &= -\delta_{1i}, \quad \text{on} \quad \partial\Omega_s. \end{aligned} \tag{5.2}$$

And the adjoint equation for transverse map velocity variable $\boldsymbol{Z}$ is:

$$Z_i^* = -(\sigma_{ij}^* n_j + u_j^* u_i n_i) \quad \text{on} \quad \partial\Omega_s. \tag{5.3}$$

By solving adjoint equation 5.2 and equation 5.3 the gradient g can be obtained as:

$$g = \frac{1}{TD_0}\int_T \int_{\partial\Omega_s} Z_k(-\frac{dZ_k^*}{dt} - Z_k^* \mathrm{div}_{\partial\Omega_s}\gamma - Z_i^*\frac{\partial u_i}{\partial x_k} + \frac{\partial \sigma_{1j}}{\partial x_j} n_k) ds dt. \tag{5.4}$$

The above derivation is focused on the optimization of a single solid body. In fact, it is more practical to consider the hydrodynamic performance of all solid bodies of interest at the same time. This was accomplished by introducing a weight w_i for each solid body of interest in the present work, where $w_i \in [0, 1]$ and $\sum_i w_i = 1$. With w_i multiple objectives might be optimized by the modified $\mathcal{J}$:

$$\mathcal{J} = \sum_i w_i \mathcal{J}_i, \tag{5.5}$$

where $\mathcal{J}_i$ has the form of equation 5.1. The gradient is modified accordingly as:

$$g = \sum_m w_m \frac{1}{TD_0} \int_T \int_{\partial\Omega_{s,m}} Z_{k,m}(-\frac{dZ^*_{k,m}}{dt} - Z^*_{k,m}\mathrm{div}_{\partial\Omega_{s,m}}\gamma - Z^*_{i,m}\frac{\partial u_i}{\partial x_k} + \frac{\partial \sigma_{1j}}{\partial x_j}n_k)dsdt. \quad (5.6)$$

5.2.3 Flow configurations and computational setup

The 2D simulation was performed on a rectangular computational domain of $40c \times 20c$, where c was the chord length of each hydrofoil. The domain was discretized by a 1001×501 non-uniform Cartesian mesh, with minimum mesh size $\Delta x_{\min} = 1.5 \times 10^{-2}c$. The mesh was fine and uniform in the near field of all hydrofoils in a $9c \times 6c$ area, and gradually stretched towards the far field. In the present work, two hydrofoil models were considered, as shown in figure 5.2. The first one was the rigid hydrofoils with NACA0012 shape, which can heave vertically and pitch azimuthally. This model has been popular in many previous studies[151–154] for its advantages of modeling fish fins and precision in control. Two different formations including triple foils in tandem (figure 5.2 (a)) and quadruple foils in diamond (figure 5.2 (b)) were studied for the rigid hydrofoil model. The second model was the flexible hydrofoils, which is able to heave vertically, and to undulate along its center line at the same time. This model has also been widely used previously[142;158;160;165], for it can mimic the fish-like swimming by introducing prescribed traveling wave kinematics on the swimmer. For both models, the horizontal tip-to-tail distance between neighboring swimmers is denoted as L. In the diamond formation, the vertical distance between the upper (or the lower) swimmer and the leading swimmer is denoted as H. The scenario of the research was set to be the schooling of multiple swimmers following a leading swimmer, where the leading swimmer is moving on its own regardless of the motion and location of followers, while all followers need to adjust their motion or location to gain hydrodynamic benefits. Therefore, the control is only enforced on followers. For the rigid swimmer model, both motion and formation of the following swimmers may be controlled, which resulted in the following equations to describe

the kinetics of each controllable swimmer:

$$\begin{aligned} X_i(t) &= \delta x_i, \\ Y_i(t) &= A_i \sin(2\pi f t) + \delta y_i, \\ \Theta_i(t) &= \theta_i \sin(2\pi f t + \phi_i), \end{aligned} \tag{5.7}$$

where A_i is the heaving amplitude, θ_i the pitching amplitude, ϕ_i the phase delay between the heaving and pitching motion, and δx_i and δy_i the displacement in horizontal and vertical direction. Subscript "i" stands for the number of swimmer. If three swimmers are studied, then $i = 2, 3$ because the leading swimmer will not be controlled. Then the control vector is given by $\boldsymbol{\gamma}_i = [A_i, \theta_i, \phi_i, \delta x_i, \delta y_i]$, of which the total dimension of the control can be up to 5 for each of the following swimmers.

For the flexible swimmer model, similar to some previous work[158;165;166], the carangiform undulating motion was modeled in the present work, considering that many sorts of carangiform fish have been reported to swim in schools, such as mackerel (*Scomber scombrus*). The following traveling wave function was used to prescribe the undulation:

$$z(x,t) = P(x) \cdot \sin[2\pi(\frac{x}{\lambda} - ft)], \tag{5.8}$$

where the position variables, x and z, are normalized by c already. Therefore locally $x = 0$ denotes the leading edge of the hydrofoil and $x = 1$ is the trailing edge. This wave function actually expresses the undulating motion of the midline of the hydrofoil, which can be regarded as the spine of swimmers. So $z(x,t)$ represents the lateral deviation of any point on the midline of the body at time moment t. λ is the wavelength of the traveling wave over an undulating body, which will be controlled. $P(x)$ is the amplitude envelope of a lateral motion and has a quadratic polynomial form:

$$P(x) = a_2 x^2 + a_1 x + a_0, \tag{5.9}$$

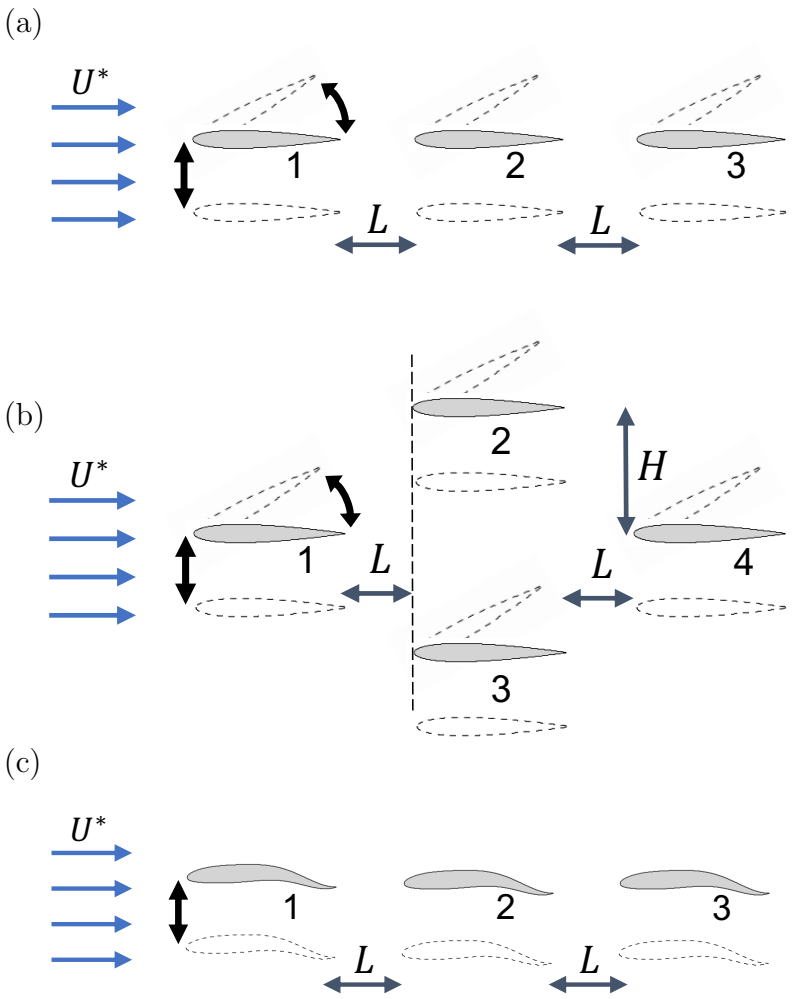

Figure 5.2: *The illustrations of two swimmer models studied in the present work. (a): Three Rigid hydrofoils that can heave vertically and pitch azimuthally in tandem formation. (b): Four Rigid hydrofoils that can heave vertically and pitch azimuthally in diamond formation. (c): Flexible hydrofoils that can heave vertically and have the carangiform undulating motion in tandem formation. Numbers are used in following sections to refer to corresponding swimmers.*

where $a_0 = 0.02$, $a_1 = -0.0825$, $a_2 = 0.1625$ as measured in experiments for the carangiform motion[167]. The amplitude envelope of the carangiform motion used in the present work is presented in figure 5.3 for different time moments in a tail-beat period. Besides undulation, the vertical heaving motion as well as the formation of following swimmers will still be studied for the flexible swimmer model. So combining equation 5.7 and equation 5.8, the control vector of each follow is given as $\boldsymbol{\gamma}_i = [A_i, \lambda_i, \delta x_i, \delta y_i]$, of which the total dimension of the control is up to 4 for each of the following swimmers.

The study on mesh independence for the flow past multiple swimmers was carried out for three different meshes from coarse to dense at $Re = 200$. The rigid hydrofoil model was used with three swimmers lining up in tandem formation. All three swimmers were heaving and pitching synchronously with $\boldsymbol{\gamma}_i = [0.5, 5°, 0]$. The mean drag and lift coefficients were evaluated for each mesh on every swimmer, and the Strouhal number of the wake was also compared. As presented in table 5.3, force coefficients obtained on different meshes would converge as mesh size becomes finer, with merely 0.5% difference in $\bar{C_{D1}}$, 0.75% difference in $\bar{C_{D2}}$, and 1.5% difference in $\bar{C_{D3}}$ between the most coarse mesh and mid-size mesh. Moreover, no difference was found in $\bar{C_D}$ between the mid-size mesh and the finest mesh. For all three meshes, C_L remained nearly zero or exactly zero, and St remained 0.1. Therefore, 1001×501 mesh was considered good enough and was used for simulations with rigid and flexible swimmer models.

It is worth noticing that practically when the formation of multiple swimmers is being optimized, a collision between swimmers might happen in control iterations to hamper the physical meaning of optimal solutions. Therefore, it is crucial to impose restrictions to constrain the possible displacement of each swimmer, and an effective algorithm to detect collision is needed. In the present work, the separating axis theorem (SAT) was utilized to detect if collision occurs between any swimmers during each iteration[168]. This algorithm has been proved effective and easy to implement for any 2D convex shapes, the details of which can be found in appendix D.

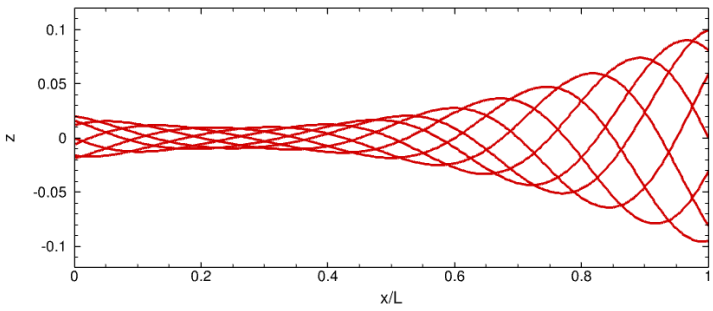

Figure 5.3: *Traveling wave amplitude of a carangiform motion at different time moments in one undulatory period.*

Table 5.3: *Mean drag and lift coefficients on each rigid hydrofoil and the Strouhal number on 3 different meshes*

	801×401 $\Delta x_{\min} = 2.14 \times 10^{-2}c$	1001×501 $\Delta x_{\min} = 1.5 \times 10^{-2}c$	1201×601 $\Delta x_{\min} = 8.0 \times 10^{-3}c$
$\bar{C_{D1}}$	0.199	0.20	0.20
$\bar{C_{D2}}$	0.132	0.133	0.133
$\bar{C_{D3}}$	0.134	0.132	0.132
$\bar{C_{L1}}$	-0.001	0.0	0.0
$\bar{C_{L2}}$	0.0	0.0	0.0
$\bar{C_{L3}}$	-0.001	0.0	0.0
St	0.1	0.1	0.1

5.3 Results and discussion

In this section, first the rigid swimmer model was used for optimization of triple swimmers in tandem formation and quadruple swimmers in diamond formation. Triple swimmers initially moving synchronously in tandem formation was studied for the optimized heaving and pitching motion to gain the lowest drag or the highest thrust on the followers. For the quadruple swimmers, only one trailing swimmer was controlled for not only the optimal motion but also the optimal formation. Then the flexible swimmer model was investigated on triple swimmers in tandem formation.

5.3.1 Heaving-pitching hydrofoils in tandem formation

Three rigid swimmers were initially performing harmonic heaving and pitching motion that was given by equation 5.7 at $Re = 200$, with horizontal spacing $L = 0.5c$. Formation was not controlled in this case therefore $\boldsymbol{\gamma}_i = [A_i, \theta_i, \phi_i]$. The total dimension of the control was 6 considering both following swimmers. For the initial synchronous motion, all three

swimmers had the same $\boldsymbol{\gamma}_i^{(0)} = [0.5, 5°, 0]$. In order to avoid erratic motion for each swimmer, prescribed limits were imposed to all control parameters, as $-1 \leq A_i \leq 1$, $-30° \leq \theta_i \leq 30°$, and $-90° \leq \phi_i \leq 90°$. The oscillating frequency was fixed as $f = 0.1$. All force coefficients were averaged in one oscillating stroke with $T = 10$. A snapshot of the initial flow field is presented in figure 5.4. It can be observed that the vortices shed from the swimmers ahead were almost blocked by the followers, and eventually merged into the wake after the third swimmer, resulting in a single vortex street. Net mean drag was found on all three swimmers, with the leader experiencing the highest drag $\bar{C_{D1}} = 0.2$, while the two followers experiencing almost the same drag as $\bar{C_{D2}} = 0.133$ and $\bar{C_{D3}} = 0.132$. The benefit of schooling can already be reflected on the lower drag the followers experience compared to the leader, even without any optimization.

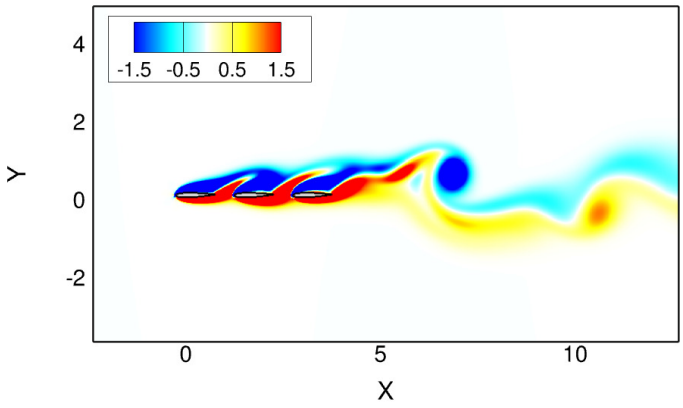

Figure 5.4: *The flow field of triple swimmers oscillating synchronously in tandem formation contoured by vorticity.*

Then optimization was carried out to minimize the drag of the following swimmers. The objective function 5.5 was utilized with different w_i. The control started with all weight on the 3rd swimmer to minimize its own drag, then with less weight on it but higher weight on the 2nd swimmer to evaluate the overall drag reduction on both followers. The variations of the objective function and control parameters are shown in figure 5.5 for different $\boldsymbol{w}$. When only $\bar{C_{D3}}$ was optimized, it can be clearly seen that not only the drag was minimized, but it was converted to thrust. The reduction on $\bar{C_{D3}}$ was 133%, with thrust coefficient $\bar{C_{T3}} = 0.044$

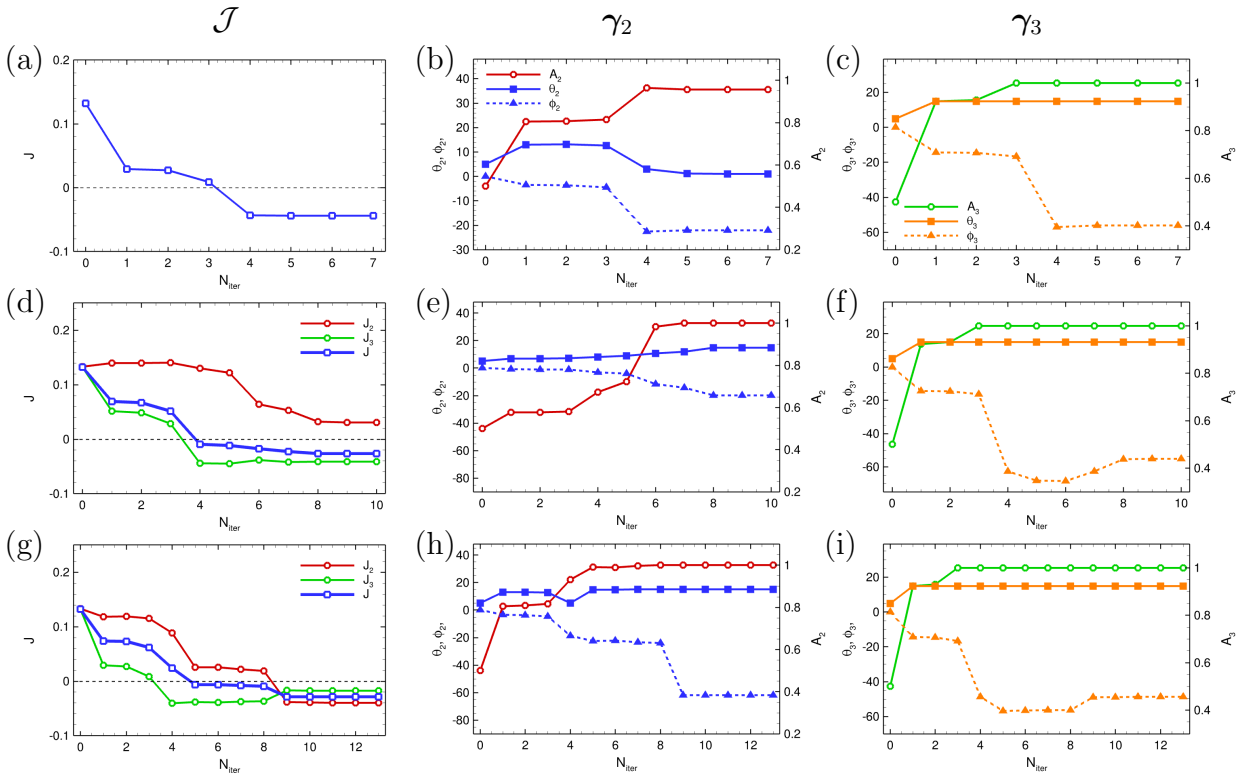

Figure 5.5: *The variations of objective function $\mathcal{J}$ (left column), $\boldsymbol{\gamma}_2$ (mid column) and $\boldsymbol{\gamma}_3$ (right column) with respect to optimization iterations. (a – c): $\boldsymbol{w} = [0, 0, 1]$; (d – f): $\boldsymbol{w} = [0, 0.2, 0.8]$; (g – i):$\boldsymbol{w} = [0, 0.5, 0.5]$.*

after 7 iterations. The 2nd swimmer increased the heaving amplitude to nearly upper limit, and decreased the phase delay. Pitching amplitude almost remained unchanged. The 3rd swimmer appeared to heave similarly to the 2nd one with $A_3 = 1$. ϕ_3 was decreased more to $-60°$ and θ_3 was increased to about $20°$. In order to gain thrust on the 3rd swimmer, two followers both attempted to move in a different pace than the leader, but themselves were still coordinating to swim with similar heaving amplitude and certain phase delay in oscillation. This optimal motion for the highest $\bar{C_{T3}}$ is not necessarily optimal for the 2nd swimmer. In fact, the mean drag on the 2nd swimmer was decreased by 12.4%. However, when w_2 was increased, better hydrodynamic performance can be achieved for the 2nd swimmer as well. As seen in figure 5.5 (d) and (g), the drag on the 2nd swimmer is reduced more by 76.8% when $w_2 = 0.2$ and $w_3 = 0.8$, and $\bar{C_{D2}}$ was eventually converted to net thrust when both w_2 and w_3 were 0.5. On the contrary, lower enhancement in thrust was observed for the 3rd swimmer as w_3 was decreased. To achieve lower drag or net thrust, θ_2 was found to increase

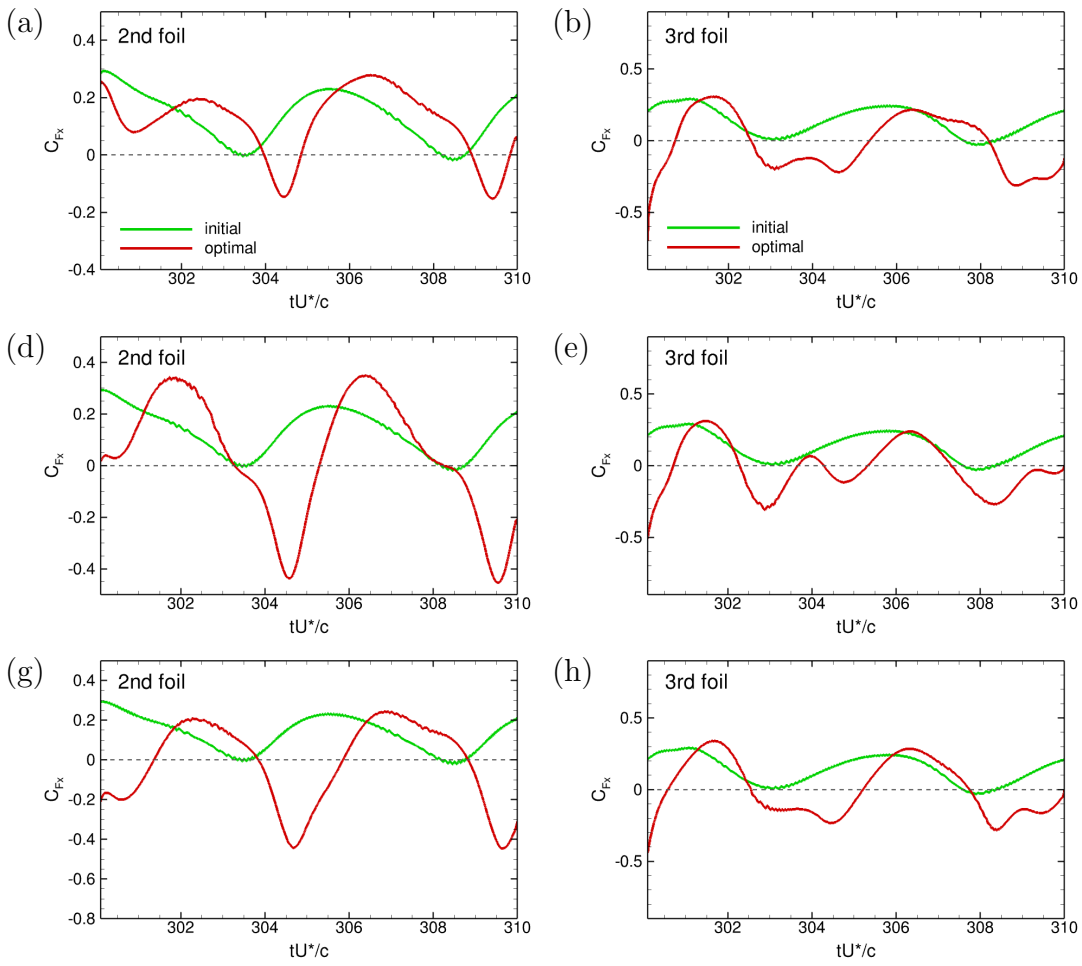

Figure 5.6: *The horizontal force coefficient profile in one oscillating stroke before and after optimization for the 2nd and the 3rd swimmer. (a – b):* $\boldsymbol{w} = [0, 0, 1]$*; (c – d):* $\boldsymbol{w} = [0, 0.2, 0.8]$*; (e – f):* $\boldsymbol{w} = [0, 0.5, 0.5]$.

to about 20°, with more decrease in ϕ_2 to lower than $-60°$. The 3rd swimmer did not change its moving pattern significantly. It is intriguing to find that when the drag on both followers are taken into account equally, they will move almost synchronously while having apparent different moving pattern than the leading swimmer.

The horizontal force coefficients before and after optimization in one oscillating stroke for two followers are compared in figure 5.6 with different $\boldsymbol{w}$. Before the optimization, the force on both followers was above zero for the most of time in one stroke, which resulted in net drag. Conversely, after the optimization, two following swimmers were experiencing high negative horizontal force (thrust) during over half of the stroke, regardless of $\boldsymbol{w}$. The

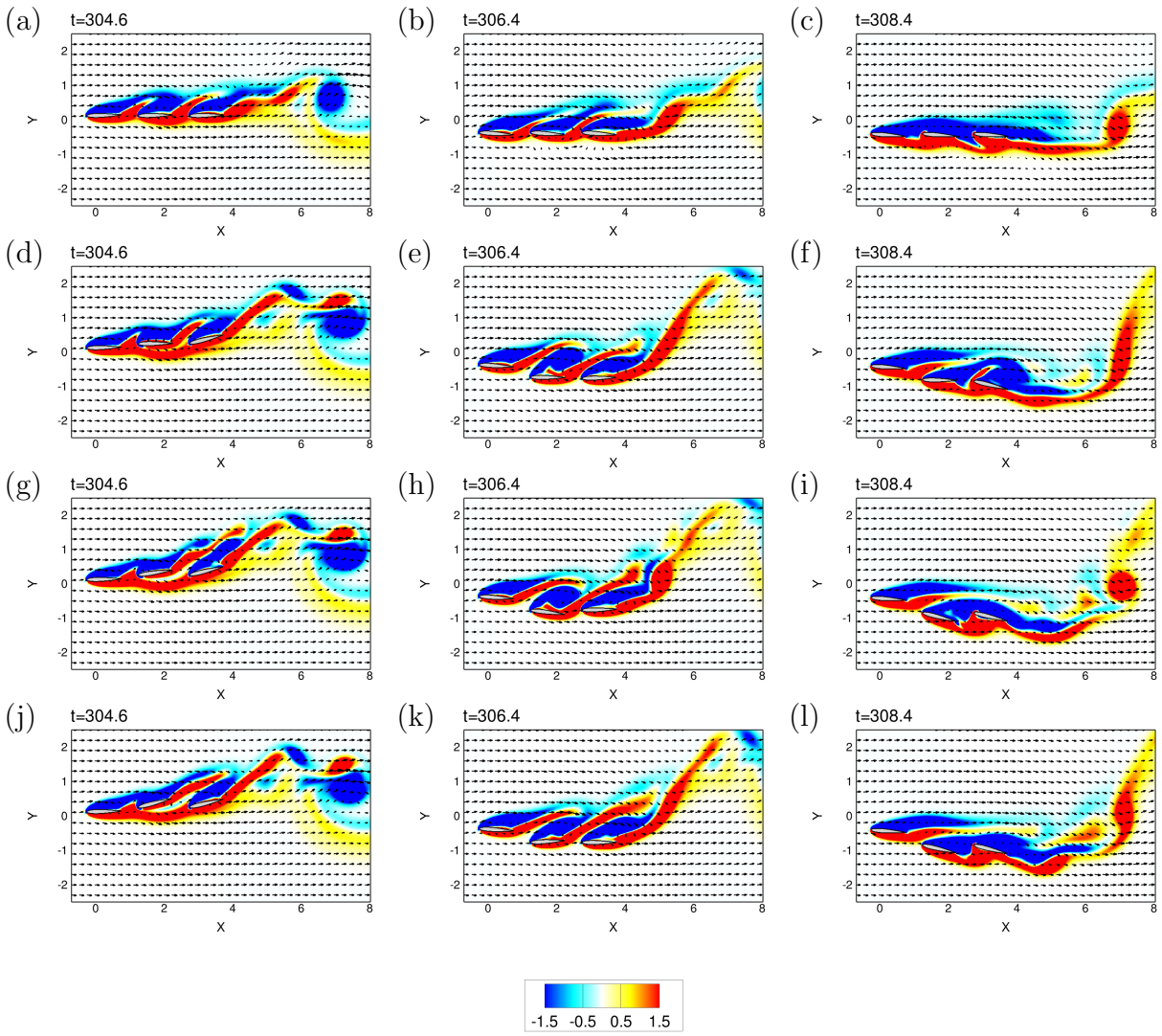

Figure 5.7: *The flow fields at three typical time moments contoured by vorticity. (a – c): The initial flow; (d – f): the optimal flow with* $\boldsymbol{w} = [0, 0, 1]$*; (g – i): the optimal flow with* $\boldsymbol{w} = [0, 0.2, 0.8]$*; (j – l): the optimal flow with* $\boldsymbol{w} = [0, 0.5, 0.5]$*. Black arrows indicate the velocity vectors.*

original periodic force profiles was found to change with more complicated oscillation of both following swimmers, as their heaving and pitching motion both altering significantly. The periodicity of the force profile was barely kept with smaller fluctuations.

To understand the change of the instantaneous horizontal force better, the corresponding flow fields contoured by vorticity were studied as shown in figure 5.7. To make a fair comparison, all pressures were normalized into the scale of $[-1, 1]$. Three typical time moments were chosen here. For the initial flow, at $tU^*/c = 304.6$ both swimmer experienced net drag

which has not reached the peak value yet. At $tU^*/c = 306.4$, C_{Fx3} was almost peaking with C_{Fx2} declining yet still positive. At $tU^*/c = 308.4$, horizontal forces on both swimmers were close to zero. As seen in figure 5.7 (a) to (c), the drag was higher when leading edge vortex (LEV) was separated from two following swimmers, while drag was close to zero when LEVs reattached to the hydrofoil. Due to low pitching amplitude, the flow separation caused by LEV was not strong. After the optimization, stronger flow separation at leading edges of both followers appeared more frequently at the same 3 time moments regardless of $\boldsymbol{w}$. LEVs with higher vorticity brought significant enhancement in negative pressure on the upper surface when swimmers pitching down, which resulted in strong suction to pull both swimmers forward. At the same time, the vortices shed from the leader with clockwise rotation (i.e. positive vorticity) increased the positive pressure on the lower leading edge as well, which further boosted the thrust of the following swimmers. At $tU^*/c = 308.4$, both following swimmers were pitching up. The flow reattachment on the 2nd swimmer may correlated to the instantaneous drag increase. While for the 3rd swimmer, the vortices shed from the 2nd swimmer appeared to bring enough suction in the region ahead of it, which eventually may lead to a "tow effect" on the 3rd swimmer to pull it forward, thus an instantaneous thrust was still observed in figure 5.6 (h). In a word, the following swimmers managed to take advantage of the vortex-swimmer interaction when swim off of the pace of the leading swimmer while with nearly synchronous heaving motion for more thrust generation by performing the optimal heaving and pitching motion.

5.3.2 Heaving-pitching hydrofoils in diamond formation

To study the diamond formation, four rigid swimmers were initially performing harmonic heaving and pitching motion that was governed by equation 5.7 at $Re = 200$ in diamond formation. $L = 0.5c$ and $H = 0.5c$ were set for the initial formation. In this case, the motion and the position of the forth (4th) swimmer (left most swimmer in figure 5.2 (b)) was controlled for the optimal horizontal force. That led to 5 control parameters in total with $\boldsymbol{\gamma}_4 = [A_4, \theta_4, \phi_4, \delta x_4, \delta y_4]$. Initially all four swimmers were performing synchronous motion

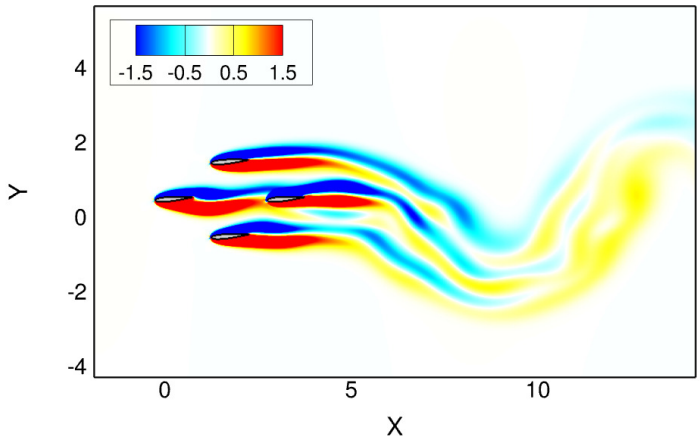

Figure 5.8: *The flow field of quadruple swimmers oscillating synchronously in diamond formation contoured by vorticity.*

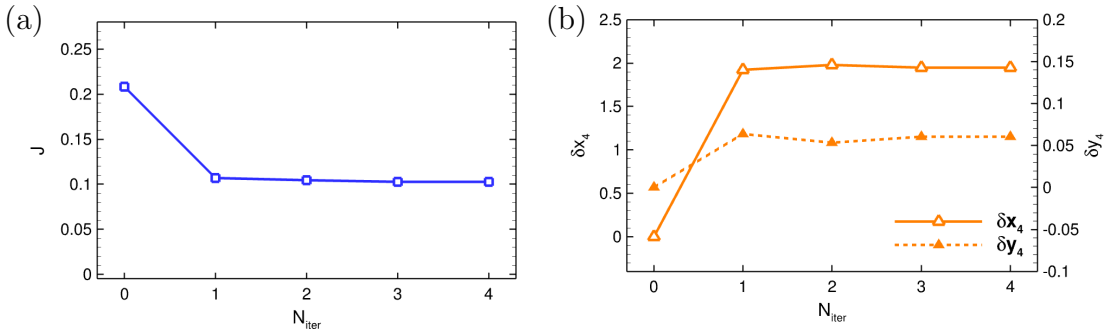

Figure 5.9: *The variation of objective function $\mathcal{J}$ (a) and the variations of δx_4 and δy_4 (b) with respect to optimization iterations.*

with $\boldsymbol{\gamma}_i^{(0)} = [0.5, 5°, 0, 0, 0]$. Constraints were also set on $\boldsymbol{\gamma}_4$ to prevent erratic motion, with as $-1 \leq A_4 \leq 1$, $-30° \leq \theta_4 \leq 30°$, and $-90° \leq \phi_4 \leq 90°$. The oscillating frequency f was still set to be 0.1, and the period of one oscillating stroke was $T = 10$. A snapshot of the initial flow field is shown in figure 5.8. It can be observed that the 4th swimmer was basically moving in the wake of the first swimmer, while the wake of upper and lower swimmer did not have much impact on the 4th swimmer. Due to wider distance between the 1st and the 4th swimmer than the previous triple-hydrofoil setup, however, the drag on the 4th swimmer remained at $\bar{C_{D4}} = 0.208$, similar to the drag a single swimmer was experiencing.

Two optimal controls was performed on the 4th swimmer. The first control was to optimize the formation of the 4th swimmer only (i.e. $\boldsymbol{\gamma}_4 = [\delta x_4, \delta y_4]$), and the second control

was to conduct the full control with $\boldsymbol{\gamma}_4 = [A_4, \theta_4, \phi_4, \delta x_4, \delta y_4]$ to seek both optimal motion and formation. The reason for designing two controls was to evaluate the benefit brought by both optimal motion and formation compared to the result with only one factor (the formation in this study) being optimized. The variations of the objective function and control parameters in the first and second control are present in figure 5.9 and figure 5.10, respectively. It can be seen that mean drag on the 4th swimmer was reduced by 50.7% to $\bar{C_{D4}} = 0.102$ in the first control. To achieve such drag reduction, the 4th swimmer shifted backward by about $2c$ and upward by about $0.05c$ as indicated in figure 5.9 (b). In comparison, $\bar{C_{D4}}$ was reduced by 64.0% to 0.075 when both motion and formation were controlled. As shown in figure 5.10 (c), the 4th swimmer was still placed farther from the three leading swimmer by about $1.9c$ in horizontal direction, while barely changed its position in vertical direction with $\delta y_4 = -3.5 \times 10^{-3} c$. The change of formation is almost the same as the first control results. As seen in figure 5.10 (b), the optimal pitching motion was able to further reduce the drag by 13.3%, with a higher pitching amplitude and the phase delay angle of $-9.3°$. The heaving amplitude A_4 was found insignificant in terms of the drag reduction.

The horizontal force coefficients on the 4th swimmer before and after optimization for both controls are shown in figure 5.11. Initially the drag profile was periodic due to the synchronous oscillation of all four swimmers. The whole profile remained above zero, as the 4th swimmer was experiencing net drag all the times during one stroke. After the optimization, the periodicity of the force profile was similar, with apparently lower peak drag values for both controls. The 4th swimmer was found to experience instantaneous thrust (negative C_{Fx}) for longer time during one stroke when both motion and formation were optimized, which resulted in an overall lower mean drag compared to optimizing formation only.

The flow fields contoured by vorticity at three typical time moments are plotted in figure 5.12 for the initial flow and the optimal flows of two different controls. It can observed that initially when the vortex shed from the 1st swimmer impinged with the 4th swimmer (at $tU*/c = 316$ and 319), the drag on the 4th swimmer was increasing. The drag was lower when the vortex shed from the leader merged into the wake of the 4th swimmer, without

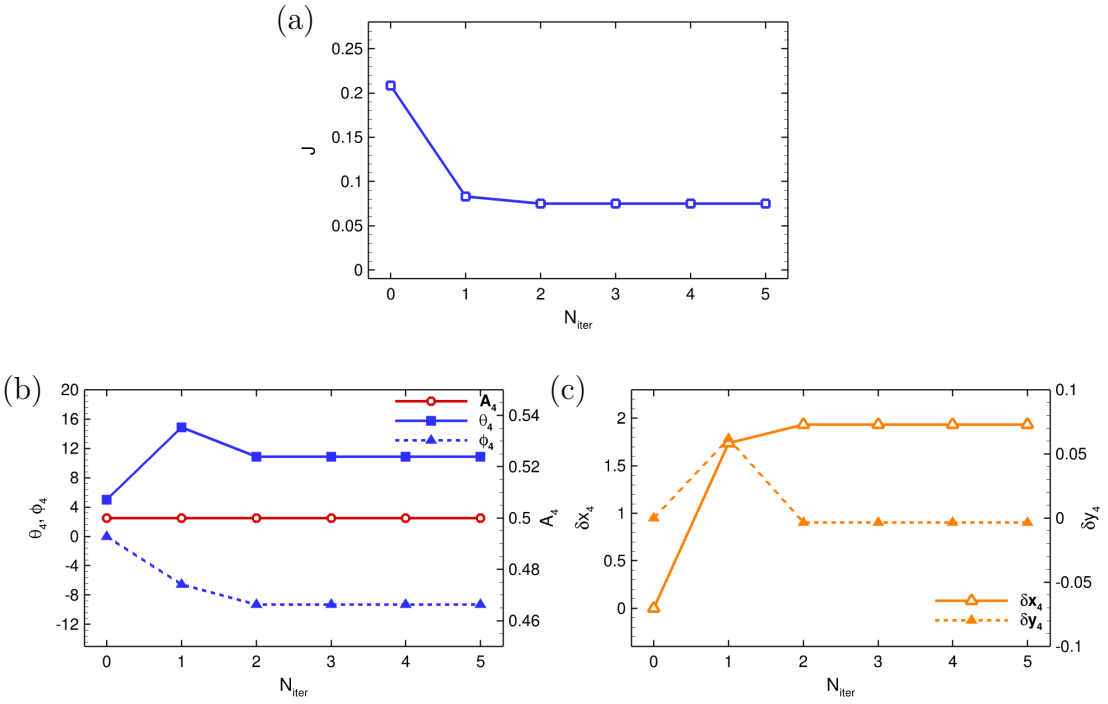

Figure 5.10: *The variation of objective function $\mathcal{J}$ (a), the variations of A_4, θ_4 and ϕ_4 (b), and the variations of δx_4 and δy_4 (c) with respect to optimization iterations.*

direct impingement onto its surface. After the optimization, the 4th swimmer positioned itself farther from three swimmers ahead. By doing so, the wake of the 1st swimmer had even weaker impact on the 4th swimmer. However, it started frequently interacting with the wake of upper and lower swimmer. As observed in figure 5.12 (d) and (g), when all swimmers were moving in downstroke, a strong separation vortex was generated on the lower surface of the 4th swimmer and then shed downstream. The wake of the 2nd swimmer merged into vortex which was about to separate at the leading edge of the 4th swimmer, resulting in the reattachment on the upper surface and a stronger wake. As swimmers kept in downstroke (figure 5.12 (e) and (h)), the 4th swimmer moved into the wake of the 3rd swimmer, and the flow separation happened on the upper surface instead. During the downstroke, net drag was found on the 4th swimmer. When swimmers moved in upstroke, at $tU^*/c = 319$ the flow separation occurred on both sides of the 4th swimmer, which may result in temporary negative C_{Fx} (i.e. thrust). The optimal pitching motion only changed flow structures slightly, therefore the extra drag reduction the 4th swimmer can obtained was limited, which was

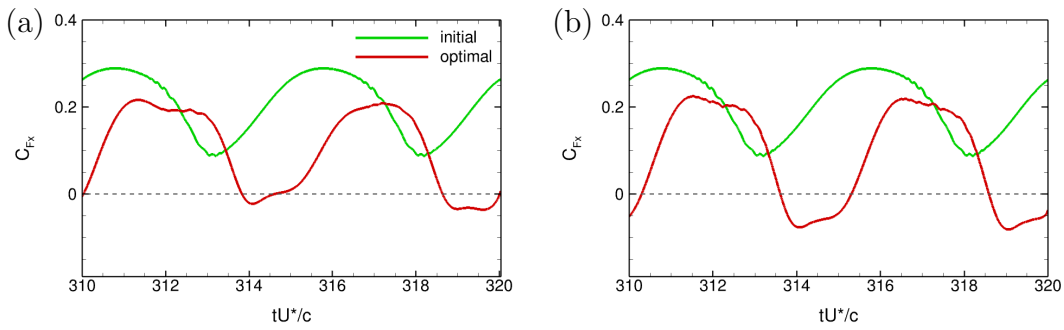

Figure 5.11: *The horizontal force coefficient profile in one oscillating stroke before and after optimization for the 4th swimmer. (a): Optimization on* $\boldsymbol{\gamma}_4 = [\delta x_4, \delta y_4]$*; (b): optimization on* $\boldsymbol{\gamma}_4 = [A_4, \theta_4, \phi_4, \delta x_4, \delta y_4]$.

already reflected in figure 5.9 and figure 5.10.

5.3.3 Heaving-undulating hydrofoils in tandem formation

To study undulating motions, three flexible swimmers were initially performing harmonic undulation governed by equation 5.8 at $Re = 200$ in tandem formation as illustrated in figure 5.2 (c), with horizontal spacing $L = 0.5c$. The swimmers were able to heave freely in vertical direction as well. The control was focused on motion first, then based on the optimal motion, an optimization on the position was carried out. Both optimizations aimed at optimizing the horizontal force on the 3rd swimmer. For the optimization on motion the control parameters were $\boldsymbol{\gamma}_i = [A_i, \lambda_i]$ with $i = 2, 3$, leading to 4 control parameters in total. For the initial synchronous undulation, all three swimmers had the same $\boldsymbol{\gamma}_i^{(0)} = [0, 1.0]$. A snapshot of the initial flow field is presented in figure 5.13. The frequency was fixed to be 0.2. A_i was limited in a range of $[-1, 1]$, and the wave length λ_i was constrained in a range of [0.1, 2]. All force coefficients were averaged in one undulating period with $T = 5$. With only undulation, initially flow attached onto three swimmers with no shedding vortex forming in the wake. The two followers initially had net mean drag as $\bar{C_{D2}} = 0.122$ and $\bar{C_{D3}} = 0.104$.

The variations of $\mathcal{J}$ and $\boldsymbol{\gamma}_i$ during the motion optimization are shown in figure 5.14. Two different weights were used to construct $\mathcal{J}$. The first case had a weight $\boldsymbol{w} = [0, 0, 1]$ which only optimized the 3rd swimmer. The second case had equal weight on both following

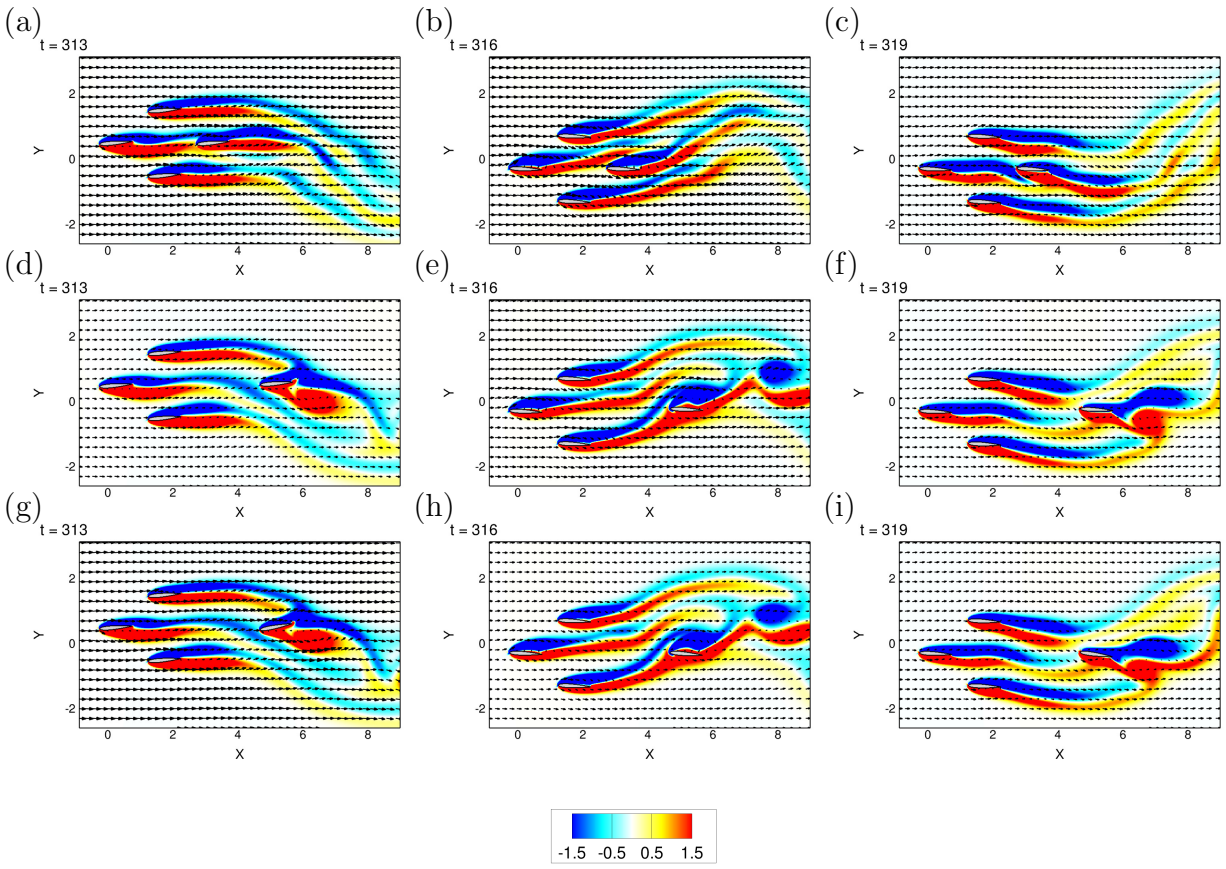

Figure 5.12: *The flow fields at three typical time moments contoured by vorticity. (a – c): The initial flow; (d – f): the optimal flow after optimization on* $\boldsymbol{\gamma}_4 = [\delta x_4, \delta y_4]$ *; (g – i): the optimal flow after optimization on* $\boldsymbol{\gamma}_4 = [A_4, \theta_4, \phi_4, \delta x_4, \delta y_4]$*. Black arrows indicate the velocity vectors.*

swimmers with $\boldsymbol{w} = [0, 0.5, 0.5]$. It can be seen that when only the 3rd swimmer was controlled, $\bar{C_{D3}}$ was effectively reduced by 279%, then eventually converted to a net thrust equal to 0.187. However, $\bar{C_{D2}}$ was conversely increased by 12.3% to 0.137, which implies the 2nd swimmer indeed sacrificed its own hydrodynamic performance to aid the 3rd swimmer. The heaving amplitude of both following swimmers increased drastically, as $A_2 = 0.41$ and $A_3 = 1.0$, which was the upper limit set for heaving amplitude. The wave length of the undulation also altered, with λ_2 decreased by 68.9% and λ_3 increased by 47.1%. When both following swimmers were controlled, $\boldsymbol{\gamma}$ changed differently yet with the similar trend to achieve optimal hydrodynamic performance as indicated in figure 5.14 (d – f) . In terms of heaving motion, both A_2 and A_3 were increased to 0.88 and 1.0 respectively. The 2nd swimmer had to heave more actively to achieve higher thrust. While for the undulation,

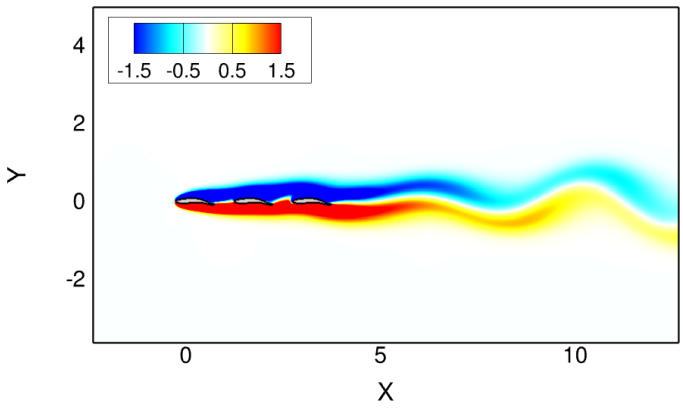

Figure 5.13: *The flow field of triple flexible swimmers undulating synchronously in tandem formation contoured by vorticity.*

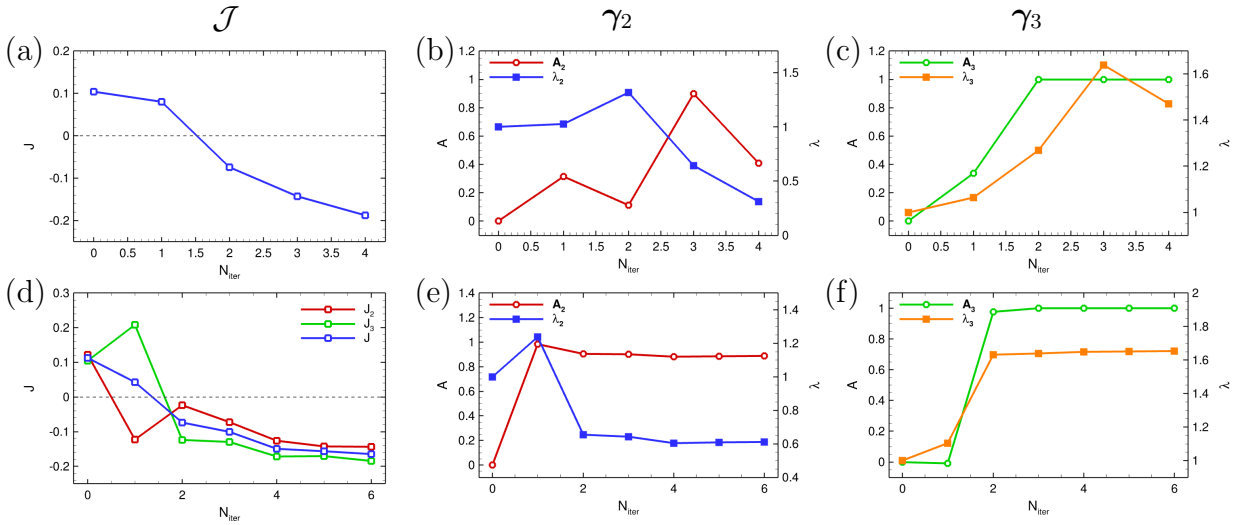

Figure 5.14: *The variations of objective function $\mathcal{J}$ (left column), $\boldsymbol{\gamma}_2$ (mid column) and $\boldsymbol{\gamma}_3$ (right column) with respect to optimization iterations. (a – c): $\boldsymbol{w} = [0, 0, 1]$; (d – f): $\boldsymbol{w} = [0, 0.5, 0.5]$.*

λ_2 decreased by 39.0% and λ_3 increased by 65.2%. By conducting optimal motion, both following swimmers obtained drag-to-thrust conversion to boost the final thrust to $C_{T2} = 0.124$ and $C_{T3} = 0.181$ respectively. The thrust on the 3rd swimmer achieved by equal weight in $\mathcal{J}$ was merely slightly lower than controlling the 3rd swimmer only, showing that $\boldsymbol{w} = [0, 0.5, 0.5]$ was a better choice for both following swimmers.

With the optimal motion, the horizontal force profile in one period was also changed dramatically for following swimmers, as plotted in figure 5.15. Initially with synchronous

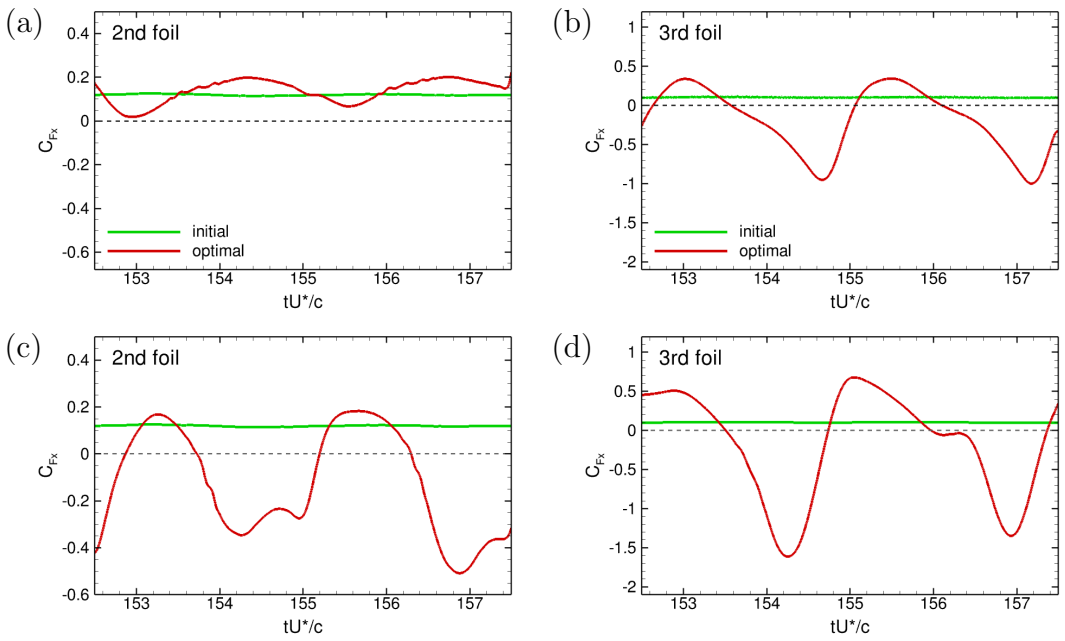

Figure 5.15: *The horizontal force coefficient profile in one undulating period before and after motion optimization for the 2nd (left column) and the 3rd (right column) swimmer. (a – b):* $\boldsymbol{w} = [0, 0, 1]$*; (c – d):* $\boldsymbol{w} = [0, 0.5, 0.5]$.

undulation, C_{Fx} was almost steady. However, after the optimization it became highly unsteady with bigger magnitudes, which implies drastic change has also taken place in the flow field. When only controlling the force of the 3rd swimmer, the 2nd swimmer still experienced net drag in one period, while the 3rd swimmer spent over half of the period experiencing negative C_{Fx}, which resulted in the net thrust over one period. When having equal weight for both following swimmers, the 2nd swimmer had a significantly different force profile with much time having negative C_{Fx}. The force profile of the 3rd swimmer overall had the similar shape to the previous case, with larger peak and valley values of C_{Fx}.

The flow structures before and after the optimization are shown in figure 5.16. It can be seen that strong vortex shedding was induced by heaving motion and undulating with different wave lengths after the optimization for both weight settings. LEVs were generated on both 2nd and 3rd swimmer, and were strengthened by the wake of the leading swimmer. Active and complex vortex pairing occurred in the wake of the 3rd swimmer. There was not only the interaction between the LEV and the TEV of the 3rd swimmer, but also the vortex

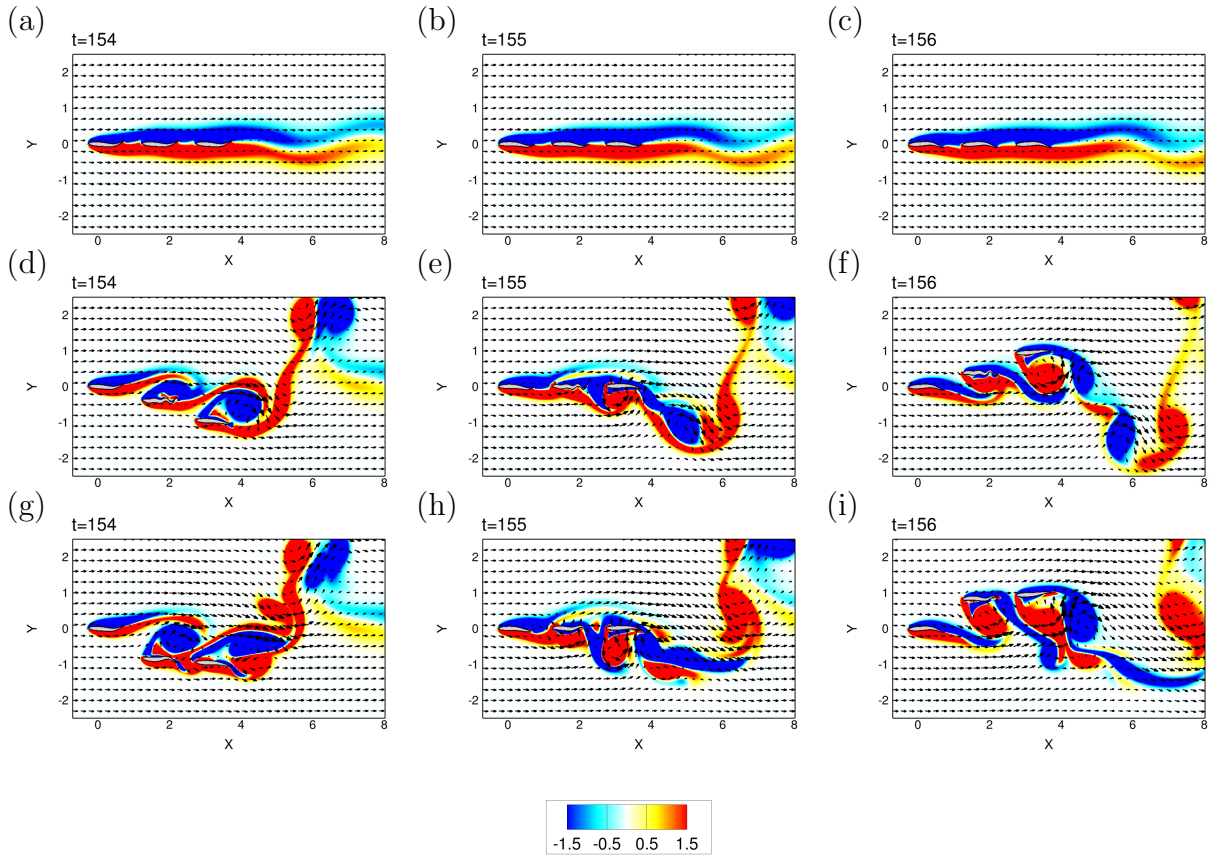

Figure 5.16: *The flow fields at three typical time moments contoured by vorticity. (a – c): The initial flow; (d – f): the optimal flow after optimization on* $\boldsymbol{\gamma}_i = [A_i, \lambda_i]$ *with* $\boldsymbol{w} = [0, 0, 1]$*; (g – i): the optimal flow after optimization on* $\boldsymbol{\gamma}_i = [A_i, \lambda_i]$ *with* $\boldsymbol{w} = [0, 0.5, 0.5]$*. Black arrows indicate the velocity vectors.*

shed from the 2nd swimmer intertwining with vortices separated from the 3rd swimmer. When the 2nd swimmer was not optimized, the very short wave length of its undulation had negative impact on the LEV shedding, making it noticeably less intense, which might impede the interaction with the wake of the leading swimmer, and eventually hamper its own hydrodynamic performance in terms of the drag. When both swimmers were controlled, they moved with similar pace to generate strong LEVs. The suction effect brought by these more active LEVs may contribute greatly to the thrust enhancement.

Based on the motion optimization results, the formation of the following swimmers were further optimized. Here only $\boldsymbol{w} = [0, 0.5, 0.5]$ was studied, considering it was able to achieve drag-to-thrust conversion for both following swimmers. The optimization started right with the optimal motion obtained in the previous control, with $A_2 = 0.89$, $A_3 = 1.0$, $\lambda_2 = 0.61$,

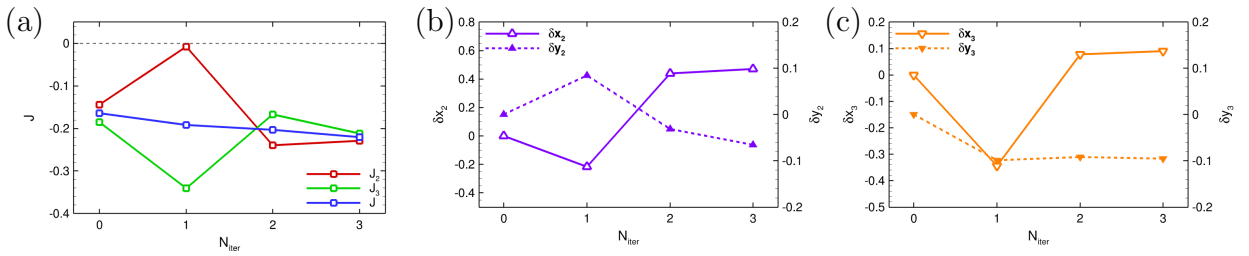

Figure 5.17: *The variations of objective function $\mathcal{J}$ (a), $\boldsymbol{\gamma}_2$ (b) and $\boldsymbol{\gamma}_3$ (c) with* $\boldsymbol{w} = [0, 0.5, 0.5]$.

and $\lambda_3 = 1.65$ being fixed for the formation control. Therefore, the control vector can be written as $\boldsymbol{\gamma}_i = [\delta x_i, \delta y_i]$ where $i = 2, 3$. The variation of the objective function as well as $\boldsymbol{\gamma}_i$ is shown in figure 5.17. It can be seen that formation control can effectively keep minimizing the objective function including C_{Fx} for both following swimmers based on the optimal motion. The overall thrust of following swimmers increased by 37.5%, with $\bar{C_{T2}}$ increased to 0.229 and $\bar{C_{T3}}$ to 0.212. As seen in figure 5.17, the 2nd swimmer was located farther from the leading swimmer by $0.47c$, and was moved downward by $0.065c$ from its equilibrium position. The 3rd swimmer was also positioned backward by $0.091c$, and was moved downward by $0.095c$ from its equilibrium position. The most significant formation change occurred to the horizontal position of the 2nd swimmer, which led to big alterations in horizontal force profile too as shown in figure 5.18.

As shown in figure 5.18 (a), much lower minimal C_{Fx} values were achieved on the 2nd swimmer, which were approximately two times the minimal C_{Fx} after motion optimization. Peak values of C_{Fx} were increased as well, nevertheless the change of force profile still contributed to 59.0% enhancement of mean thrust on the 2nd swimmer. Contrary to the 2nd swimmer, the amplitude of C_{Fx3} became smaller after the optimization on position, but the average thrust in one undulating period still managed to increase by 14.6% when both following swimmers were positioned in the optimal manner.

The flow structures after the formation optimization at the same time moments are shown in figure 5.19. It can be seen that the vortex structures were quite different than the ones shown in figure 5.16 (g – i) due to the change of distance between swimmers. The impact of the vortices shed from the leading swimmer was weaker when both following swimmers were

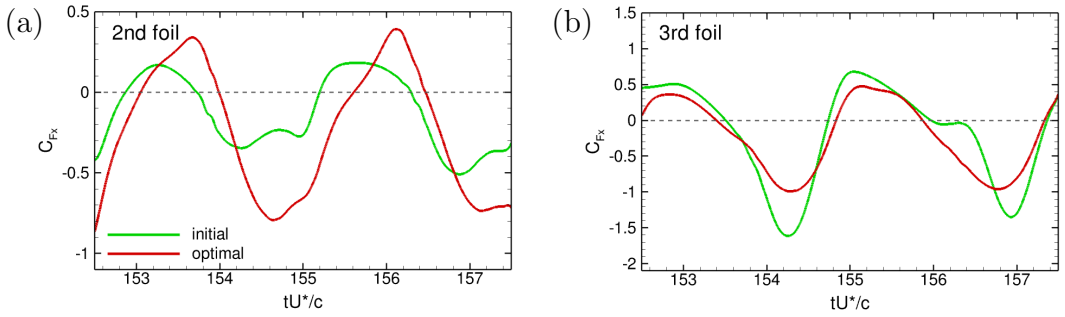

Figure 5.18: *The horizontal force coefficient profile in one undulating period before and after formation optimization for the 2nd (a) and the 3rd (b) swimmer with* $\boldsymbol{w} = [0, 0.5, 0.5]$.

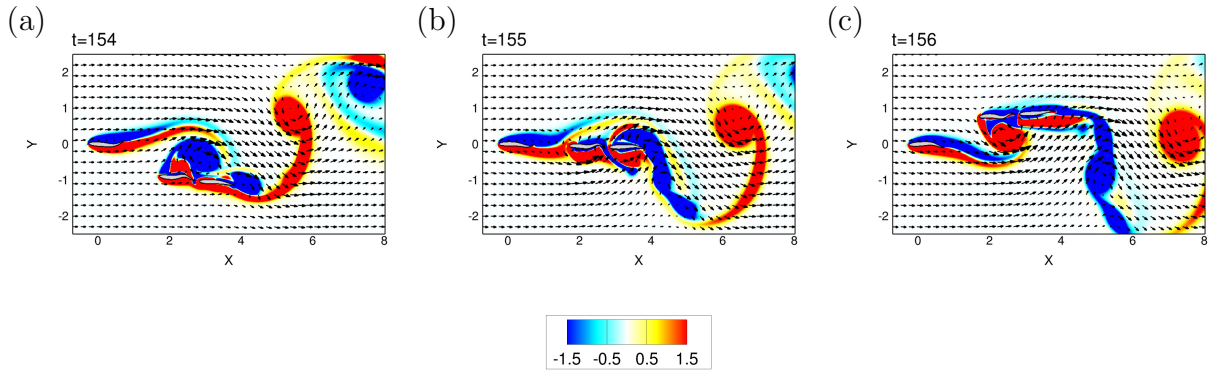

Figure 5.19: *The flow fields at three typical time moments contoured by vorticity with* $\boldsymbol{w} = [0, 0.5, 0.5]$. *Black arrows indicate the velocity vectors.*

placed farther into the downstream. The LEV of the 2nd swimmer had higher intensity to provide more suction ahead of the solid body, resulting in stronger thrust. Smaller distance between two following swimmers made them interact with the wake as a single moving bluff body, inducing a single vortex street instead of the complicated vortex pairing after optimization on the motion. It can be inferred that for the following swimmers, moving as one swimmer and closer the spacing between them to depress the vortex generated in the gap may contribute to the mean thrust enhancement.

5.4 Concluding remarks

In this chapter, the FOM-based adjoint approach was first extended to handle 2D flows with multiple moving solid bodies, the hydrofoil schooling was optimized with respect to

the drag on the following swimmers by controlling the motion and formation of them. The optimization was performed under modest Reynolds number equal to 200. Two models of swimmer were investigated: the rigid hydrofoil with the capability of heaving and pitching; and the flexible hydrofoil with the capability of heaving and undulating in a carangiform manner. Triple swimmers in tandem and quadruple swimmers in a diamond formation were studied as the initial spatial arrangement of swimmers for the rigid hydrofoil model. The flexible hydrofoil model was only focused on triple swimmers in tandem formation. All swimmers were moving synchronously before the optimization to mimic the schooling behavior.

For triple rigid swimmers in tandem formation, the motion of the following two swimmers was optimized. It was found that by increasing the heaving and pitching amplitude, as well as by decreasing the phase delay angle between heaving and pitching motion, the hydrodynamic drag was greatly reduced and could be converted to thrust for all 3 different weight functions used in objective functions. The higher the weight in the weight function on a particular swimmer, the more drag reduction or thrust enhancement was obtained. Two following swimmers experienced high thrust generation after the optimization. Oscillations on the force profile were closely related to the interaction between separation vortices induced by the optimized motion and the following swimmers. A strong negative pressure region at the leading edge of both following swimmers resulted in the suction effective to eventually boost the thrust.

For quadruple rigid swimmers in a diamond formation, the motion and the formation of the 4th swimmer was optimized for its own hydrodynamic performance. Up to 64.0% drag reduction was achieved on the 4th swimmer, by moving into the wake of the 2nd and the 3rd swimmer to benefit from more active and stronger LEVs and TEVs. The optimal formation contributed more to the drag reduction than the optimal heaving and pitching motion in this case.

For the case with three flexible swimmers in tandem formation, the heaving and undulating motion of the following two swimmers, as well as their position, was optimized. Drag-to-thrust conversion was achieved first by motion optimization when both following

swimmers were weighted equally. It was found to be beneficial for thrust enhancement to increase the heaving amplitude of both swimmers thus making them oscillating more synchronously in vertical direction. As for the undulating motion, the 2nd swimmer decreased its wave length while the 3rd swimmer increased the wave length for better thrust production. Based on the optimal motion, the optimization on the position further boosted the thrust to move the 2nd and 3rd swimmers farther from the leader, while positioning themselves closer. The optimal formation was able to strengthen the LEV of the 2nd swimmer to generate more suction effect, while decreasing the vortex interaction between the following swimmers for a more coherent vortex structure in the wake.

An estimation of the computational cost saving by the adjoint-based approach can be made briefly here. All optimizations made in this chapter were converged in less than 3 days, with a single DNS or adjoint simulation taking about 1 hour. Considering the dimension of the parametric space, the brute-force parametric study can easily cost weeks or even months of computation. Therefore the adjoint-based approach is still able to drastically reduce the computational cost.

Chapter 6

Conclusions

The present work was dedicated to simplifying, analyzing, and optimizing complex flows with fluid-structure interactions (FSI). First a global proper orthogonal decomposition (POD) and Galerkin projection based reduced-order model (ROM) has been developed to capture the essential physics of a FSI system with moving solid boundaries, while significantly lowering the dimension of the original flow system. Two different descriptions for moving solid domains led to two types of global POD-Galerkin ROMs, namely CSDM and DSDM ROMs. Both ROMs were able to handle numerical and experimental two-dimensional (2D) and three-dimensional (3D) data, and have shown adequate accuracy in the reconstruction of flow dynamics and structures, and in the prediction of key aerodynamic properties including drag and lift force. The computational cost of solving ROMs, especially DSDM-ROMs, was extremely low compared to DNS or other CFD methods, which allowed for rapid reductions in computational time required.

Based upon the ROM and the conventional adjoint approach, a new adjoint approach has been development to achieve fast flow control and optimization of an FSI system with moving solid boundaries. Two different strategies, "one-ROM" strategy and "ROM-switching" strategy, have been designed to guarantee the accuracy of ROMs during the optimization process. These new adjoint-ROM approaches have been applied to the stabilization of the flow field as well as the aerodynamic force optimization of 2D flows past oscillating cylinders

and NACA0012 airfoils. The adjoint-ROM approach has been proved effective and fast, of which the computational cost saving may make a near-real-time flow control possible.

The adjoint-based approach has also been applied to analyzing two complicated FSI problems for the first time. First the adjoint-based flow optimization approach has been applied to the gust mitigation for both 2D and 3D heaving-pitching wings. The streamwise gust and the transverse gust were studied separately. The full-order model (FOM) based adjoint approach was used for all cases, while the adjoint-ROM approach developed in the present work was used for 2D gusts only. The objective function was designed to minimize the difference between the instantaneous lift in one oscillating period and the mean lift of the original flow. Optimal control of the wing motion was achieved to not only recover the original mean aerodynamic force on the wing, but sometimes alleviated the unsteadiness of the force history. It was found that the optimal wing motion may transition the flow into different flow regime to retain the mean lift. The FOM-based was able to handle relatively strong gusts with very good recovery of the lift force, and the computational cost was reasonably low. The adjoint-ROM was able to mitigate weak gusts with lower controllability, and the optimization took significantly less computational time.

Then the FOM-based adjoint approach was extended to optimize the schooling of multiple hydrofoils. The drag force of 2D rigid and flexible hydrofoils in different arrangements of formation was optimized by controlling the motion and formation of the trailing hydrofoils. Significant drag reduction and drag-to-thrust conversion were achieved by adjoint-based optimization. The analysis of vortex-structure interactions of optimal schooling shed some light upon the hydrodynamic mechanisms of fish schooling behavior.

Future work can be focused on the following four problems:

1. Currently the time scale τ in ROMs is empirically determined with the help of a loose theoretical estimation. A more rigorous rule to pick valid τ will greatly simplify the offline preparation of building ROM.

2. Algorithms to further guarantee the robustness of ROMs in adjoint-ROM optimization with less offline computation will be desirable.

3. The gust mitigation cases so far have only considered flows with modest Reynolds numbers. However, many gusts taking place in real life are highly chaotic and even turbulent. Therefore the extension of the current method to turbulent flow will be meaningful to guide the design of aircrafts.

4. The current flow optimization on 2D schooling behavior opens up a way for more realistic 3D case. The adjoint-approach is more powerful in 3D scenario considering the huge computational time reduction. Some preliminary studies have been conducted[169], and it shows the capability to optimize more complicated 3D schooling by the present approach.

Bibliography

[1] J. R. Meneghini, F. Saltara, C. L. R. Siqueira, and J. A. Ferrari. Numerical simulation of flow interference between two circular cylinders in tandem and side-by-side arrangements. *Journal of Fluids and Structures*, 15(2):327–350, 2001.

[2] Boyce E. Griffith and Neelesh A. Patankar. Immersed methods for fluid–structure interaction. *Annual Review of Fluid Mechanics*, 52(1):421–448, 2020. ISSN 0066-4189.

[3] Matthias Heil and Andrew L Hazel. Fluid-structure interaction in internal physiological flows. *Annual Review of Fluid Mechanics*, 43:141–162, 2011. ISSN 0066-4189.

[4] Earl H. Dowell and Kenneth C. Hall. Modeling of fluid-structure interaction. *Annual Review of Fluid Mechanics*, 33:445, 2001. ISSN 00664189.

[5] Yuri Bazilevs, Kenji Takizawa, and Tayfun E Tezduyar. *Computational Fluid-Structure Interaction: Methods and Applications*. John Wiley & Sons, 2013.

[6] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37:239–261, 2005. ISSN 0066-4189.

[7] Charles S Peskin. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics*, 10(2):252–271, 1972. ISSN 0021-9991.

[8] Charles S Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252, 1977. ISSN 0021-9991.

[9] MF McCracken and CS Peskin. A vortex method for blood flow through heart valves. *Journal of Computational Physics*, 35(2):183–205, 1980. ISSN 0021-9991.

[10] Charles S Peskin. The fluid dynamics of heart valves: experimental, theoretical, and computational methods. *Annual Review of Fluid Mechanics*, 14(1):235–259, 1982. ISSN 0066-4189.

[11] Ming-Chih Lai and Charles S Peskin. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *Journal of Computational Physics*, 160(2):705–719, 2000. ISSN 0021-9991.

[12] J Mohd-Yusof. For simulations of flow in complex geometries. *Annual Research Briefs*, 317, 1997.

[13] E. A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *Journal of Computational Physics*, 161(1):35–60, 2000. ISSN 00219991.

[14] R Mittal, C Bonilla, and HS Udaykumar. Cartesian grid methods for simulating flows with moving boundaries. *Computational Methods and Experimental Measurements-XI*, pages 557–566, 2003.

[15] Kunihiko Taira and Tim Colonius. The immersed boundary method: A projection approach. *Journal of Computational Physics*, 225(2):2118–2137, 2007. ISSN 0021-9991.

[16] Rajat Mittal, Haibo Dong, Meliha Bozkurttas, FM Najjar, Abel Vargas, and Alfred von Loebbecke. A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *Journal of Computational Physics*, 227(10):4825–4852, 2008. ISSN 0021-9991.

[17] Hong Zhao, Jonathan B. Freund, and Robert D. Moser. A fixed-mesh method for incompressible flow–structure systems with finite solid deformations. *Journal of Computational Physics*, 227(6):3114–3140, 2008. ISSN 00219991.

[18] Tao Yang, Mingjun Wei, and Hong Zhao. Numerical study of flexible flapping wing propulsion. *AIAA Journal*, 48(12):2909–2915, 2010.

[19] Tao Yang. *Numerical study of flexible flapping wings.* Thesis, New Mexico State University, 2012.

[20] Min Xu. *Understanding flapping-wing aerodynamics through adjoint-based approach.* Thesis, New Mexico State University, 2014.

[21] Wei Zhang. *Linear Structure of Nonlinear Dynamic Systems via Koopman Decomposition.* Thesis, Kansas State University, 2019.

[22] Charles HK Williamson. Vortex dynamics in the cylinder wake. *Annual Review of Fluid Mechanics*, 28(1):477–539, 1996. ISSN 0066-4189.

[23] Lawrence Sirovich. Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987. ISSN 0033-569X.

[24] Gal Berkooz, Philip Holmes, and John L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1):539–575, 1993. ISSN 0066-4189.

[25] Karen Willcox and Jaime Peraire. Balanced model reduction via the proper orthogonal decomposition. *AIAA Journal*, 40(11):2323–2330, 2002. ISSN 0001-1452.

[26] Clarence W Rowley. Model reduction for fluids, using balanced proper orthogonal decomposition. *International Journal of Bifurcation and Chaos*, 15(03):997–1013, 2005. ISSN 0218-1274.

[27] Moritz Sieber, C. Oliver Paschereit, and Kilian Oberleithner. Spectral proper orthogonal decomposition. *Journal of Fluid Mechanics*, 792:798–828, 2016. ISSN 0022-1120.

[28] Aaron Towne, Oliver T Schmidt, and Tim Colonius. Spectral proper orthogonal decomposition and its relationship to dynamic mode decomposition and resolvent analysis. *Journal of Fluid Mechanics*, 847:821–867, 2018. ISSN 0022-1120.

[29] Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010. ISSN 1469-7645.

[30] Kunihiko Taira, Steven L. Brunton, Scott T. M. Dawson, Clarence W. Rowley, Tim Colonius, Beverley J. McKeon, Oliver T. Schmidt, Stanislav Gordeyev, Vassilios Theofilis, and Lawrence S. Ukeiley. Modal analysis of fluid flows: an overview. *AIAA Journal*, 55(12):4013–4041, 2017. ISSN 0001-1452.

[31] John Leask Lumley. The structure of inhomogeneous turbulent flows. *Atmospheric Turbulence and Radio Wave Propagation*, pages 166–178, 1967.

[32] Steven L Brunton and J Nathan Kutz. *Data-driven science and engineering: machine learning, dynamical systems, and control*. Cambridge University Press, 2022. ISBN 1009098489.

[33] Bernd R Noack, Paul Papas, and Peter A Monkewitz. The need for a pressure-term representation in empirical galerkin models of incompressible shear flows. *Journal of Fluid Mechanics*, 523:339–365, 2005. ISSN 1469-7645.

[34] Jean-Christophe Loiseau, Steven Brunton, and Bernd Noack. *From the POD-Galerkin method to sparse manifold models*. 2018.

[35] Max D Gunzburger. *Perspectives in flow control and optimization*. SIAM, 2002. ISBN 089871527X.

[36] Herrmann Schlichting and Klaus Gersten. *Boundary-layer theory*. Springer Science & Business Media, 2003. ISBN 3540662707.

[37] Ismail H. Tuncer and Mustafa Kaya. Optimization of flapping airfoils for maximum thrust and propulsive efficiency. *AIAA Journal*, 43(11):2329–2336, 2005. ISSN 0001-1452.

[38] Thomas R. Bewley, Parviz Moin, and Roger Temam. Dns-based predictive control of turbulence: an optimal benchmark for feedback algorithms. *Journal of Fluid Mechanics*, 447:179–225, 2001. ISSN 0022-1120.

[39] Mingjun Wei and Jonathan B Freund. A noise-controlled free shear flow. *Journal of Fluid Mechanics*, 546:123–152, 2006. ISSN 1469-7645.

[40] Byung Joon Lee, Mattia Padulo, and Meng-Sing Liou. Non-sinusoidal trajectory optimization of flapping airfoil using unsteady adjoint approach. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 1312, 2011.

[41] Martin Jones and Nail K. Yamaleev. Adjoint-based shape and kinematics optimization of flapping wing propulsive efficiency. In *43rd Fluid Dynamics Conference*, Fluid Dynamics and Co-located Conferences. American Institute of Aeronautics and Astronautics, 2013.

[42] Thomas R Bewley. Flow control: new challenges for a new renaissance. *Progress in Aerospace sciences*, 37(1):21–58, 2001. ISSN 0376-0421.

[43] Mordukhaĭ Moiseevich Vaĭnberg, Leonid Vitalevich Kantorovich, and Gleb Pavlovich Akilov. *Variational methods for the study of nonlinear operators*. Holden-Day, 1964.

[44] Elijah Polak. *Computational methods in optimization: a unified approach*, volume 77. Academic press, 1971. ISBN 008096091X.

[45] John A Ekaterinaris and Max F Platzer. Computational prediction of airfoil dynamic stall. *Progress in Aerospace Sciences*, 33(11-12):759–846, 1998.

[46] Rohit Jain, Arnaud Le Pape, Amanda Grubb, Michel Costes, François Richez, and Marilyn Smith. High-resolution computational fluid dynamics predictions for the static and dynamic stall of a finite-span oa209 wing. *Journal of Fluids and Structures*, 78: 126–145, 2018.

[47] Anupam Sharma and Miguel Visbal. Numerical investigation of the effect of airfoil thickness on onset of dynamic stall. *Journal of Fluid Mechanics*, 870:870–900, 2019.

[48] Earl H Dowell and Kenneth C Hall. Modeling of fluid-structure interaction. *Annual Review of Fluid Mechanics*, 33(1):445–490, 2001.

[49] C. W. Rowley and D. R. Williams. Dynamics and control of high-Reynolds-number flow over open cavities. *Annual Review of Fluid Mechanics*, 38:251–276, 2006.

[50] Philip Holmes, John L Lumley, Gahl Berkooz, and Clarence W Rowley. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge university press, 2012. ISBN 1107008255.

[51] Mingjun Wei and Clarence W. Rowley. Low-dimensional models of a temporally evolving free shear layer. *Journal of Fluid Mechanics*, 618:113–134, 2009.

[52] Mingjun Wei, Bashar R Qawasmeh, Matthew Barone, Bart G van Bloemen Waanders, and Lin Zhou. Low-dimensional model of spatial shear layers. *Physics of Fluids*, 24 (1):014108, 2012.

[53] Xia Ma and George Em Karniadakis. A low-dimensional model for simulating three-dimensional cylinder flow. *Journal of Fluid Mechanics*, 458:181–190, 2002.

[54] Bernd R Noack, Konstantin Afanasiev, Marek Morzyński, Gilead Tadmor, and Frank Thiele. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *Journal of Fluid Mechanics*, 497:335–363, 2003.

[55] Stefan G Siegel, Juergen Seidel, Casey Fagley, DM Luchtenburg, Kelly Cohen, and Thomas McLaughlin. Low-dimensional modelling of a transient cylinder wake using double proper orthogonal decomposition. *Journal of Fluid Mechanics*, 610:1–42, 2008.

[56] Bernd R Noack and Robert K Niven. Maximum-entropy closure for a galerkin model of an incompressible periodic wake. *Journal of Fluid Mechanics*, 700:187–213, 2012.

[57] Clarence W Rowley, Tim Colonius, and Richard M Murray. Model reduction for compressible flows using pod and galerkin projection. *Physica D: Nonlinear Phenomena*, 189(1-2):115–129, 2004.

[58] Bashar R Qawasmeh and Mingjun Wei. Low-dimensional models for compressible temporally developing shear layers. *Journal of Fluid Mechanics*, 731:364–393, 2013.

[59] Chao Zhang, Zhenhua Wan, and Dejun Sun. Model reduction for supersonic cavity flow using proper orthogonal decomposition (POD) and galerkin projection. *Applied Mathematics and Mechanics*, 38(5):723–736, 2017.

[60] Kaushik Kumar Nagarajan, Sintu Singha, Laurent Cordier, and Christophe Airiau. Open-loop control of cavity noise using proper orthogonal decomposition reduced-order model. *Computers & Fluids*, 160:1–13, 2018.

[61] Gilead Tadmor, Oliver Lehmann, Bernd R Noack, and Marek Morzyński. Galerkin models enhancements for flow control. In *Reduced-Order Modelling for Flow Control*, pages 151–252. Springer, 2011.

[62] Mingjun Wei and Tao Yang. A global approach for reduced-order models of flapping flexible wings. AIAA paper 2010-5085, the 5th Flow Control Conference, Chicago, IL, 2010.

[63] Haotian Gao and Mingjun Wei. Global model reduction for flows with moving boundary. In *52nd Aerospace Sciences Meeting*, page 0222, 2014.

[64] Haotian Gao and Mingjun Wei. Domain decomposition in POD-Galerkin projection for flows with moving boundary. In *54th AIAA Aerospace Sciences Meeting*, page 1102, 2016.

[65] GC Lewin and H Haj-Hariri. Reduced-order modeling of a heaving airfoil. *AIAA Journal*, 43(2):270–283, 2005.

[66] Bernd R Noack, Gilead Tadmor, and Marek Morzynski. Actuation models and dissipative control in empirical Galerkin models of fluid flows. In *Proceedings of the 2004 American Control Conference*, volume 6, pages 5722–5727. IEEE, 2004.

[67] Gilead Tadmor and Bernd R Noack. Dynamic estimation for reduced Galerkin models of fluid flows. In *Proceedings of The 2004 American Control Conference*, volume 1, pages 746–751. IEEE, 2004.

[68] JSR Anttonen, PI King, and PS Beran. POD-based reduced-order models with deforming grids. *Mathematical and Computer Modelling*, 38(1-2):41–62, 2003.

[69] John SR Anttonen, Paul I King, and Philip S Beran. Applications of multi-POD to a pitching and plunging airfoil. *Mathematical and Computer Modelling*, 42(3-4):245–259, 2005.

[70] Victor Troshin, Avi Seifert, David Sidilkover, and Gilead Tadmor. Proper orthogonal decomposition of flow-field in non-stationary geometry. *Journal of Computational Physics*, 311:329–337, 2016.

[71] Victor Troshin and Avraham Seifert. Modeling of a pitching and plunging airfoil using experimental flow field and load measurements. *Experiments in Fluids*, 59(1):6, 2018.

[72] Victor Troshin and Avi Seifert. Data-driven modeling of pitching and plunging tandem wings in hovering flight. *Experiments in Fluids*, 60(3):39, 2019.

[73] Tso-Kang Wang and Kourosh Shoele. Geometrically weighted modal decomposition techniques. *Journal of Fluid Mechanics*, 911:A41, 2021. ISSN 0022-1120.

[74] Erwan Liberge and Aziz Hamdouni. Reduced order modeling method via proper orthogonal decomposition (POD) for flow around an oscillating cylinder. *Journal of Fluids and Structures*, 26(2):292–311, 2010.

[75] Erwan Liberge, Mustapha Benaouicha, and Aziz Hamdouni. Proper orthogonal decomposition investigation in fluid structure interaction. *European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique*, 16(3-4):401–418, 2007.

[76] R. Glowinski, T. W. Pan, T. I. Hesla, and D. D. Joseph. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 25:755–794, 1999.

[77] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, and T.-W. Pan. A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 26(9):1509 – 1524, 2000.

[78] Haotian Gao, Mingjun Wei, and John Hrynuk. Data-driven rom for the prediction of dynamic stall. In *2018 Fluid Dynamics Conference*, page 3094, 2018.

[79] Mark Sussman, Peter Smereka, and Stanley Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational physics*, 114(1): 146–159, 1994. ISSN 0021-9991.

[80] Min Xu, Mingjun Wei, Tao Yang, and Young S Lee. An embedded boundary approach for the simulation of a flexible flapping wing at different density ratio. *European Journal of Mechanics-B/Fluids*, 55:146–156, 2016.

[81] R. Ghias, R. Mittal, and H. Dong. A sharp interface immersed boundary method for compressible viscous flows. *Journal of Computational Physics*, 225:528–553, 2007.

[82] F Noca, D Shiels, and D Jeon. Measuring instantaneous fluid dynamic forces on bodies, using only velocity fields and their derivatives. *Journal of Fluids and Structures*, 11 (3):345–350, 1997.

[83] F Noca, D Shiels, and D Jeon. A comparison of methods for evaluating time-dependent fluid dynamic forces on bodies, using only velocity fields and their derivatives. *Journal of Fluids and Structures*, 13(5):551–578, 1999.

[84] John T. Hrynuk and Douglas G. Bohl. The effects of leading-edge tubercles on dynamic stall. *Journal of Fluid Mechanics*, page A5, 2020.

[85] Thomas J Mueller and James D DeLaurier. An overview of micro air vehicle aerodynamics. *Fixed and Flapping Wing Aerodynamics For Micro Air Vehicle Applications*, 195:1–9, 2001.

[86] H Dong, R Mittal, and FM Najjar. Wake topology and hydrodynamic performance of low-aspect-ratio flapping foils. *Journal of Fluid Mechanics*, 566:309, 2006.

[87] Zongxian Liang, Haibo Dong, and Mingjun Wei. Computational analysis of hovering hummingbird flight. In *48th AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition*, page 555, 2010.

[88] Kun Jia. *Optimization of flexible wings and vorticity transfer in a leading-edge vortex due to spanwise bending*. Thesis, Kansas State University, 2021.

[89] Jamie M Anderson, K Streitlien, DS Barrett, and Michael S Triantafyllou. Oscillating foils of high propulsive efficiency. *Journal of Fluid mechanics*, 360:41–72, 1998.

[90] Manoochehr M Koochesfahani. Vortical patterns in the wake of an oscillating airfoil. *AIAA Journal*, 27(9):1200–1205, 1989.

[91] Gordon J Berman and Z Jane Wang. Energy-minimizing kinematics in hovering insect flight. *Journal of Fluid Mechanics*, 582:153, 2007.

[92] Pat Trizila, Chang-Kwon Kang, Hikaru Aono, Wei Shyy, and Miguel Visbal. Low-Reynolds-number aerodynamics of a flapping rigid flat plate. *AIAA Journal*, 49(4): 806–823, 2011.

[93] Mehdi Ghommem, Muhammad R Hajj, Dean T Mook, Bret K Stanford, Philip S Beran, Richard D Snyder, and Layne T Watson. Global optimization of actively morphing flapping wings. *Journal of Fluids and Structures*, 33:210–228, 2012.

[94] Ismail H Tuncer and Mustafa Kaya. Optimization of flapping airfoils for maximum thrust and propulsive efficiency. *AIAA Journal*, 43(11):2329–2336, 2005.

[95] Matthew Culbreth, Yves Allaneau, and Antony Jameson. High-fidelity optimization of flapping airfoils and wings. In *29th AIAA Applied Aerodynamics Conference*, page 3521, 2013.

[96] Max Gunzburger. Adjoint equation-based methods for control problems in incompressible, viscous flows. *Flow, Turbulence and Combustion*, 65(3):249–272, 2000.

[97] Siva K Nadarajah and Antony Jameson. Optimum shape design for unsteady flows with time-accurate continuous and discrete adjoint method. *AIAA Journal*, 45(7): 1478–1491, 2007. ISSN 0001-1452.

[98] Marwan Moubachir and Jean-Paul Zolesio. *Moving shape analysis and control: applications to fluid structure interactions*. Chapman and Hall/CRC, 2006. ISBN 0429146094.

[99] Bartosz Protas and Wenyuan Liao. Adjoint-based optimization of pdes in moving domains. *Journal of Computational Physics*, 227(4):2707–2723, 2008. ISSN 0021-9991.

[100] Min Xu and Mingjun Wei. Using adjoint-based optimization to study kinematics and deformation of flapping wings. *Journal of Fluid Mechanics*, 799:56–99, 2016. ISSN 0022-1120.

[101] Min Xu, Mingjun Wei, Chengyu Li, and Haibo Dong. Adjoint-based optimization for thrust performance of three-dimensional pitching–rolling plate. *AIAA Journal*, 57(9): 3716–3727, 2019.

[102] Mingjun Wei and Clarence W Rowley. Low-dimensional models of a temporally evolving free shear layer. *Journal of Fluid Mechanics*, 618:113–134, 2009.

[103] Elnaz Rezaian and Mingjun Wei. A hybrid stabilization approach for reduced-order models of compressible flows with shock-vortex interaction. *International Journal for Numerical Methods in Engineering*, 121(8):1629–1646, 2020.

[104] Haotian Gao, Mingjun Wei, and Kun Jia. Model adaptation of an improved global pod-galerkin model. In *AIAA Scitech 2019 Forum*, page 1898, 2019.

[105] Bolun Xu, Haotian Gao, Mingjun Wei, and John Hrynuk. Pod-galerkin projection rom for the flow passing a rotating elliptical airfoil. In *AIAA Aviation 2020 Forum*, page 3082, 2020.

[106] Bolun Xu, Haotian Gao, Mingjun Wei, and John Hrynuk. Global pod-galerkin roms for fluid flows with moving solid structures. *AIAA Journal*, 60(3):1400–1414, 2021.

[107] Alexandra Tallet, Cyrille Allery, and Cédric Leblond. Optimal flow control using a POD-based reduced-order model. *Numerical Heat Transfer, Part B: Fundamentals*, 70 (1):1–24, 2016. ISSN 1040-7790.

[108] Mourad Oulghelou and Cyrille Allery. Optimal control based on adaptive model reduction approach to control transfer phenomena. *AIP Conference Proceedings*, 1798 (1):020119, 2017. ISSN 0094-243X.

[109] M. Oulghelou and C. Allery. A fast and robust sub-optimal control approach using reduced order model adaptation techniques. *Applied Mathematics and Computation*, 333:416–434, 2018. ISSN 0096-3003.

[110] K. Ito and S. S. Ravindran. A reduced-order method for simulation and control of fluid flows. *Journal of Computational Physics*, 143(2):403–425, 1998. ISSN 0021-9991.

[111] S. S. Ravindran. Reduced-order adaptive controllers for fluid flows using POD. *Journal of Scientific Computing*, 15(4):457–478, 2000. ISSN 1573-7691.

[112] S. S. Ravindran. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 34 (5):425–448, 2000. ISSN 0271-2091.

[113] Michel Bergmann, Laurent Cordier, and Jean-Pierre Brancher. Optimal rotary control of the cylinder wake using proper orthogonal decomposition reduced-order model. *Physics of Fluids*, 17(9):097101, 2005.

[114] Michel Bergmann and Laurent Cordier. Optimal control of the cylinder wake in the laminar regime by trust-region methods and POD reduced-order models. *Journal of Computational Physics*, 227(16):7813–7840, 2008.

[115] David Amsallem and Charbel Farhat. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA Journal*, 46(7):1803–1813, 2008. ISSN 0001-1452.

[116] Anya R. Jones, Oksan Cetiner, and Marilyn J. Smith. Physics and modeling of large flow disturbances: discrete gust encounters for modern air vehicles. *Annual Review of Fluid Mechanics*, 54(1):469–493, 2022. ISSN 0066-4189.

[117] Anya R. Jones. Gust encounters of rigid wings: taming the parameter space. *Physical Review Fluids*, 5(11):110513, 2020.

[118] Herbert G Küssner. Zusammenfassender bericht über den instationären auftrieb von flügeln. *Luftfahrtforschung*, 13(12):410–424, 1936. ISSN 0368-7643.

[119] T. H. Von Karman and W. R. Sears. Airfoil theory for non-uniform motion. *Journal of the Aeronautical Sciences*, 5(10):379–390, 1938.

[120] Zachary F. Smith, Anya R. Jones, and John T. Hrynuk. *Micro Air Vehicle Scale Gust-Wing Interaction in a Wind Tunnel*. AIAA SciTech Forum. American Institute of Aeronautics and Astronautics, 2018.

[121] S. J. Corkery, H. Babinsky, and J. K. Harvey. On the development and early observations from a towing tank-based transverse wing–gust encounter test rig. *Experiments in Fluids*, 59(9):135, 2018. ISSN 1432-1114.

[122] Camli Badrya, Hülya Biler, Anya R. Jones, and James D. Baeder. Effect of gust width on flat-plate response in large transverse gust. *AIAA Journal*, 59(1):49–64, 2020. ISSN 0001-1452.

[123] Hülya Biler, Girguis Sedky, Anya R. Jones, Murat Saritas, and Oksan Cetiner. Experimental investigation of transverse and vortex gust encounters at low reynolds numbers. *AIAA Journal*, 59(3):786–799, 2020. ISSN 0001-1452.

[124] Gino Perrotta and Anya R. Jones. Unsteady forcing on a flat-plate wing in large transverse gusts. *Experiments in Fluids*, 58(8):101, 2017. ISSN 1432-1114.

[125] Colin Stutz, John Hrynuk, and Douglas Bohl. Investigation of static wings interacting with vertical gusts of indefinite length at low reynolds numbers. *Experiments in Fluids*, 63(5):82, 2022. ISSN 1432-1114.

[126] J. T. Vance, I. Faruque, and J. S. Humbert. Kinematic strategies for mitigating gust perturbations in insects. *Bioinspiration & Biomimetics*, 8(1):016004, 2013. ISSN 1748-3190.

[127] Alex Fisher, Sridhar Ravi, Simon Watkins, Jon Watmuff, Chun Wang, Hao Liu, and Phred Petersen. The gust-mitigating potential of flapping wings. *Bioinspiration & Biomimetics*, 11(4):046010, 2016. ISSN 1748-3190.

[128] Jorn A. Cheney, Jonathan P. J. Stevenson, Nicholas E. Durston, Jialei Song, James R. Usherwood, Richard J. Bomphrey, and Shane P. Windsor. Bird wings act as a suspension system that rejects gusts. *Proceedings of the Royal Society B: Biological Sciences*, 287(1937):20201748, 2020.

[129] S. H. Abbasi and A. Mahmood. Bio-inspired gust mitigation system for a flapping wing uav: modeling and simulation. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 41(11):524, 2019. ISSN 1806-3691.

[130] Yongsheng Lian and Wei Shyy. *Aerodynamics of low Reynolds number plunging airfoil under gusty environment*. Aerospace Sciences Meetings. American Institute of Aeronautics and Astronautics, 2007.

[131] Yongsheng Lian. *Numerical study of a flapping airfoil in gusty environments*. Fluid

Dynamics and Co-located Conferences. American Institute of Aeronautics and Astronautics, 2009.

[132] Naresh Poudel, Meilin Yu, and John T. Hrynuk. Gust mitigation with an oscillating airfoil at low reynolds number. *Physics of Fluids*, 33(10):101905, 2021. ISSN 1070-6631.

[133] Ignacio Andreu-Angulo and Holger Babinsky. Mitigation of airfoil gust loads through pitch. *AIAA Journal*, 60(9):5273–5285, 2022. ISSN 0001-1452.

[134] Girguis Sedky, Anya R. Jones, and Francis D. Lagor. Lift regulation during transverse gust encounters using a modified goman–khrabrov model. *AIAA Journal*, 58(9):3788–3798, 2020. ISSN 0001-1452.

[135] Girguis Sedky, Francis D. Lagor, and Anya Jones. Unsteady aerodynamics of lift regulation during a transverse gust encounter. *Physical Review Fluids*, 5(7):074701, 2020.

[136] Girguis Sedky, Antonios Gementzopoulos, Ignacio Andreu-Angulo, Francis D. Lagor, and Anya R. Jones. Physics of gust response mitigation in open-loop pitching manoeuvres. *Journal of Fluid Mechanics*, 944:A38, 2022. ISSN 0022-1120.

[137] Johannes E. Pohl, Rolf Radespiel, Benjamin Herrmann, Steven L. Brunton, and Richard Semaan. Gust mitigation through closed-loop control. i. trailing-edge flap response. *Physical Review Fluids*, 7(2):024705, 2022.

[138] Benjamin Herrmann, Steven L. Brunton, Johannes E. Pohl, and Richard Semaan. Gust mitigation through closed-loop control. ii. feedforward and feedback control. *Physical Review Fluids*, 7(2):024706, 2022.

[139] Tao Yang, Mingjun Wei, and Hong Zhao. Numerical study of flexible flapping wing propulsion. *AIAA Journal*, 48(12):2909–2915, 2010.

[140] D Weihs. Hydromechanics of fish schooling. *Nature*, 241(5387):290–291, 1973.

[141] R Gopalkrishnan, Michael S Triantafyllou, George S Triantafyllou, and D Barrett. Active vorticity control in a shear flow using a flapping foil. *Journal of Fluid Mechanics*, 274:1–21, 1994.

[142] James C Liao. A review of fish swimming mechanics and behaviour in altered flows. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1487):1973–1993, 2007.

[143] C. Liao James, N. Beal David, V. Lauder George, and S. Triantafyllou Michael. Fish exploiting vortices decrease muscle activity. *Science*, 302(5650), 2003.

[144] Stefano Marras and Maurizio Porfiri. Fish and robots swimming together: attraction towards the robot demands biomimetic locomotion. *Journal of The Royal Society Interface*, 9(73):1856–1868, 2012.

[145] Liang Li, Máté Nagy, Jacob M. Graving, Joseph Bak-Coleman, Guangming Xie, and Iain D. Couzin. Vortex phase matching as a strategy for schooling in robots and in fish. *Nature Communications*, 11(1):5408, 2020.

[146] Audrey Filella, François Nadal, Clément Sire, Eva Kanso, and Christophe Eloy. Model of collective fish behavior with hydrodynamic interactions. *Physical Review Letters*, 120(19):198101, 2018.

[147] Birgitt M. Boschitsch, Peter A. Dewey, and Alexander J. Smits. Propulsive performance of unsteady tandem hydrofoils in an in-line configuration. *Physics of Fluids*, 26 (5):051901, 2014.

[148] Peter A. Dewey, Daniel B. Quinn, Birgitt M. Boschitsch, and Alexander J. Smits. Propulsive performance of unsteady tandem hydrofoils in a side-by-side configuration. *Physics of Fluids*, 26(4):041903, 2014.

[149] Chengjun Yuan, Geng Liu, Yan Ren, and Haibo Dong. Propulsive performance and vortex interactions of multiple tandem foils pitching in line. In *45th AIAA Fluid Dynamics Conference*, page 3220, 2015.

[150] K. B. Lua, H. Lu, X. H. Zhang, T. T. Lim, and K. S. Yeo. Aerodynamics of two-dimensional flapping wings in tandem configuration. *Physics of Fluids*, 28(12):121901, 2016.

[151] W. Newbolt Joel, Jun Zhang, and Leif Ristroph. Flow interactions between uncoordinated flapping swimmers give rise to group cohesion. *Proceedings of the National Academy of Sciences*, 116(7):2419–2424, 2019.

[152] Sina Heydari and Eva Kanso. School cohesion, speed and efficiency are modulated by the swimmers flapping motion. *Journal of Fluid Mechanics.*, 922:A27, 2021.

[153] Pan Han, Yu Pan, Geng Liu, and Haibo Dong. Propulsive performance and vortex wakes of multiple tandem foils pitching in-line. *Journal of Fluids and Structures*, 108: 103422, 2022.

[154] Xingjian Lin, Jie Wu, Liming Yang, and Hao Dong. Two-dimensional hydrodynamic schooling of two flapping swimmers initially in tandem formation. *Journal of Fluid Mechanics*, 941:A29, 2022.

[155] George V. Lauder. Fish locomotion: recent advances and new directions. *Annual Review of Marine Science*, 7(1):521–545, 2015.

[156] Amy Gao and Michael S. Triantafyllou. Independent caudal fin actuation enables high energy extraction and control in two-dimensional fish-like group swimming. *Journal of Fluid Mechanicss*, 850:304–335, 2018.

[157] Sung Goon Park and Hyung Jin Sung. Hydrodynamics of flexible fins propelled in tandem, diagonal, triangular and diamond configurations. *Journal of Fluid Mechanics*, 840:154–189, 2018.

[158] Yu Pan and Haibo Dong. Computational analysis of hydrodynamic interactions in a high-density fish school. *Physics of Fluids*, 32(12):121901, 2020.

[159] P. Han, G. V. Lauder, and H. Dong. Hydrodynamics of median-fin interactions in fish-like locomotion: Effects of fin shape and movement. *Physics of Fluids*, 32(1), 2020.

[160] Huiyang Yu, Xi-Yun Lu, and Haibo Huang. Collective locomotion of two uncoordinated undulatory self-propelled foils. *Physics of Fluids*, 33(1):011904, 2021. ISSN 1070-6631.

[161] Ahmet Gungor and Arman Hemmati. Wake symmetry impacts the performance of tandem hydrofoils during in-phase and out-of-phase oscillations differently. *Physical Review E*, 102(4):043104, 2020.

[162] Tingwei Ji, Fan Jin, Fangfang Xie, Hongyu Zheng, Xinshuai Zhang, and Yao Zheng. Active learning of tandem flapping wings at optimizing propulsion performance. *Physics of Fluids*, 34(4):047117, 2022.

[163] C. K. Hemelrijk, D. A. P. Reid, H. Hildenbrandt, and J. T. Padding. The increased efficiency of fish swimming in a school. *Fish and Fisheries*, 16(3):511–521, 2015.

[164] Mohsen Daghooghi and Iman Borazjani. The hydrodynamic advantages of synchronized swimming in a rectangular pattern. *Bioinspiration & Biomimetics*, 10(5):056018, 2015.

[165] Valentina Di Santo, Elsa Goerig, K. Wainwright Dylan, Otar Akanyeti, C. Liao James, Theodore Castro-Santos, and V. Lauder George. Convergence of undulatory swimming kinematics across a diversity of fishes. *Proceedings of the National Academy of Sciences*, 118(49):e2113206118, 2021.

[166] Muhammad Saif Ullah Khalid, Junshi Wang, Haibo Dong, and Moubin Liu. Flow transitions and mapping for undulating swimmers. *Physical Review Fluids*, 5(6):063104, 2020.

[167] JJ Videler and F Hess. Fast continuous swimming of two pelagic predators, saithe (Pollachius virens) and mackerel (Scomber scombrus): a kinematic analysis. *Journal of Experimental Biology*, 109(1):209–228, 1984. ISSN 1477-9145.

[168] Stefan Gottschalk. Separating axis theorem. 1996.

[169] Daniel Colgan, Bolun Xu, Mingjun Wei, and John Hrynuk. *3D vortical structure of multiple moving spheroids under adjoint-based optimal control*. AIAA SciTech Forum. American Institute of Aeronautics and Astronautics, 2023.

Appendix A

The semi-implicit scheme to discretize ROM equations

The semi-implicit scheme for temporal integration of ROM equation 2.3 is addressed here. With 2nd-order Adams-Bashforth scheme considered for convective and forcing terms, and 2nd-order Crank-Nicholson scheme for viscous term, equation 2.3 can be discretized as:

$$\frac{a_i^{n+1} - a_i^n}{\Delta t} = \frac{1}{2}(3EXP_i^n - EXP_i^{n-1}) + \frac{1}{2}[(\sum_{j=0}^{N} L_{ij}a_j)^n + (\sum_{j=0}^{N} L_{ij}a_j)^{n+1}], \tag{A.1}$$

where

$$EXP_i \equiv \sum_{j=0}^{N}\sum_{k=0}^{N} Q_{ijk}a_j a_k + C_i. \tag{A.2}$$

With some rearrangement of terms the equation can be given as:

$$a_i^{n+1} - (\frac{\Delta t}{2}\sum_{j=0}^{N} L_{ij}a_j)^{n+1} = \frac{\Delta t}{2}(3EXP_i^n - EXP_i^{n-1} + \sum_{j=0}^{N} L_{ij}a_j^n) + a_i^n, \tag{A.3}$$

which can be written as a linear system with noting $[\frac{\Delta t}{2}(3EXP_i^n - EXP_i^{n-1} + \sum_{j=0}^{N} L_{ij}a_j^n) + a_i^n]$ as RHS_i:

$$\mathbf{Ax} = \mathbf{b}, \tag{A.4}$$

where $\mathbf{A}$ is an $N \times (N+1)$ matrix:

$$\mathbf{A} = \begin{bmatrix} -\frac{\Delta t}{2}L_{10} & 1-\frac{\Delta t}{2}L_{11} & -\frac{\Delta t}{2}L_{12} & \cdots & \cdots & -\frac{\Delta t}{2}L_{1N} \\ -\frac{\Delta t}{2}L_{20} & -\frac{\Delta t}{2}L_{21} & 1-\frac{\Delta t}{2}L_{22} & \cdots & \cdots & -\frac{\Delta t}{2}L_{2N} \\ \vdots & & \ddots & & \vdots & \\ -\frac{\Delta t}{2}L_{N0} & -\frac{\Delta t}{2}L_{N1} & \cdots & \cdots & \cdots & 1-\frac{\Delta t}{2}L_{NN} \end{bmatrix} \tag{A.5}$$

$\mathbf{x}$ an $(N+1)$ dimensional column vector:

$$\mathbf{x} = [a_0 \quad a_1 \quad a_2 \quad \cdots \quad a_N]^T \tag{A.6}$$

and $\mathbf{b}$ an N dimensional right-hand side vector:

$$b_i = RHS_i \tag{A.7}$$

Since a_0 is corresponding to mean flow which is fixed as 1 constantly, $\mathbf{A}$ and right-hand side vector $\mathbf{b}$ can be degenerated to:

$$\mathbf{A}' = \begin{bmatrix} 1-\frac{\Delta t}{2}L_{11} & -\frac{\Delta t}{2}L_{12} & \cdots & \cdots & -\frac{\Delta t}{2}L_{1N} \\ -\frac{\Delta t}{2}L_{21} & 1-\frac{\Delta t}{2}L_{22} & \cdots & \cdots & -\frac{\Delta t}{2}L_{2N} \\ \vdots & & \ddots & & \vdots \\ -\frac{\Delta t}{2}L_{N1} & \cdots & \cdots & \cdots & 1-\frac{\Delta t}{2}L_{NN} \end{bmatrix} \tag{A.8}$$

and

$$b'_i = RHS_i - (\frac{\Delta t}{2}L_{i0}a_0), \tag{A.9}$$

where $\mathbf{A}'$ is an $N \times N$ square matrix and $\mathbf{b}'$ as new right-hand side vector. The new linear system $\mathbf{A}'\mathbf{x} = \mathbf{b}'$ is solved by **MKL** of **Intel Fortran**.

Appendix B

Calculate aerodynamic force without pressure

The method to calculate the aerodynamical forces without using pressure is presented here briefly. The fundamental equation for the aerodynamic forces from flow-field quantities is the momentum equation in integral form, which can be written as:

$$\boldsymbol{F} = -\frac{d}{dt}\int_V \boldsymbol{u}dV + \oint_S \hat{\boldsymbol{n}}\cdot[-p\boldsymbol{I} - \boldsymbol{uu} + \boldsymbol{T}]dS - \oint_{S_b} \hat{\boldsymbol{n}}\cdot(\boldsymbol{uu})dS, \tag{B.1}$$

where V is an arbitrary fixed control volume (CV) which must contain the solid body at any snapshot and S is the surface surrounding it (for 2D case it is the contour instead), S_b is the surface of the solid body. According to Noca et al.[83], equation B.1 can be converted by using several identity relations to:

$$\boldsymbol{F} = -\frac{d}{dt}\int_V \boldsymbol{u}dV + \oint_S \hat{\boldsymbol{n}}\cdot\boldsymbol{\Gamma}dS - \oint_{S_b} \hat{\boldsymbol{n}}\cdot(\boldsymbol{uu})dS, \tag{B.2}$$

where $\mathbf{\Gamma}$ is a second-order tensor having the form of:

$$\begin{aligned}\mathbf{\Gamma} = &\frac{1}{2}\boldsymbol{u}^2\boldsymbol{I} - \boldsymbol{u}\boldsymbol{u} - \frac{1}{N_d - 1}\boldsymbol{u}(\boldsymbol{x} \times \boldsymbol{\omega}) + \frac{1}{N_d - 1}\boldsymbol{\omega}(\boldsymbol{x} \times \boldsymbol{u}) \\ &- \frac{1}{N_d - 1}[(\boldsymbol{x} \cdot \frac{\partial \boldsymbol{u}}{\partial t})\boldsymbol{I} - \boldsymbol{x}\frac{\partial \boldsymbol{u}}{\partial t}] + \frac{1}{N_d - 1}[\boldsymbol{x} \cdot (\nabla \cdot \boldsymbol{T})\boldsymbol{I} - \boldsymbol{x}(\nabla \cdot \boldsymbol{T})] + \boldsymbol{T},\end{aligned} \tag{B.3}$$

where N_d is the dimension of the problem, which can be 2 or 3 here. Now the aerodynamic forces can be computed without knowing the pressure field.

As addressed in Noca et al.[83], the raw data from experiments often suffer from unexpected vibrations of experimental devices, which may lead to a bit noisy force computation results. In order to reduce such unphysical high-frequency oscillations, a 5th-order Butterworth scheme as used in Noca et al.[83] was implemented to filter the experimental data, which was realized numerically by a built-in function in **Matlab**.

Appendix C

An alternative way to derive adjoint-ROM equations and gradient

This derivation starts with the same objective function $\mathcal{J}$, and can derive exactly the same form of adjoint equations and gradient as the method used chapter 3 without using the method of variation.

Using the same $\mathcal{J}$, and considering the whole problem as a constrained optimization, $\mathcal{J}$ can be rewritten as:

$$\tilde{\mathcal{J}} = \int_{T_0}^{T_1} \int_{\Omega_0} |\boldsymbol{u} - \boldsymbol{u_0}|^2 d\Omega dt + \sum_i \int_T \boldsymbol{q}_i^* \cdot [\boldsymbol{\mathcal{F}}_i(\boldsymbol{q}_i) - \boldsymbol{\mathcal{R}}_i] dt, \tag{C.1}$$

where $\boldsymbol{q}_i^*$ is the Lagrangian multiplier, and the restriction of this optimization should be that the governing equation $\boldsymbol{\mathcal{F}}_i(\boldsymbol{q}_i) - \boldsymbol{\mathcal{R}}_i = 0$ is satisfied.

The modified $\tilde{\mathcal{J}} \equiv \tilde{\mathcal{J}}(\boldsymbol{q}_i, \boldsymbol{q}_i^*, \boldsymbol{\gamma})$. When the extrema of $\mathcal{J}$ are reached, the following

conditions must be satisfied:

$$\frac{\partial \tilde{\mathcal{J}}}{\partial \boldsymbol{q}_i} = 0 \tag{C.2}$$

$$\frac{\partial \tilde{\mathcal{J}}}{\partial \boldsymbol{q}_i^*} = 0 \tag{C.3}$$

$$\frac{\partial \tilde{\mathcal{J}}}{\partial \boldsymbol{\gamma}} = g(\boldsymbol{\gamma}) \tag{C.4}$$

Solving equation C.3 will recover the governing equations immediately. Solving equation C.2 will generate two groups of equations:

$$\frac{\partial \tilde{\mathcal{J}}}{\partial a_i} = 0 \tag{C.5}$$

$$\frac{\partial \tilde{\mathcal{J}}}{\partial b_i} = 0 \tag{C.6}$$

With integration by parts, equation C.5 yields:

$$\frac{\partial}{\partial a_i} \sum_i \int_T [\int_{\Omega_0} 2(\sum_j a_j \boldsymbol{\Phi}_j - \sum_j a_{0j} \boldsymbol{\Phi}_{0j}) a_i \boldsymbol{\Phi}_i - \dot{a}_i^* - \sum_j L'_{ji} a_j^* - \sum_j \sum_k (Q'_{jik} + Q'_{jki}) a_k a_j^*] dt = 0 \tag{C.7}$$

Let all the terms in the integral be 0 and

$$\begin{aligned} K_{ij} &= \int_{\Omega_0} \boldsymbol{\Phi}_i \cdot \boldsymbol{\Phi}_j d\Omega, \\ M_{ij} &= \int_{\Omega_0} \boldsymbol{\Phi}_{0i} \cdot \boldsymbol{\Phi}_j d\Omega, \end{aligned} \tag{C.8}$$

and with the help of relations:

$$\begin{aligned} \frac{\partial a_j}{\partial a_i} &= \delta_{ij} \\ \frac{\partial b_j}{\partial b_i} &= \delta_{ij}, \end{aligned} \tag{C.9}$$

then it can be derived that

$$\dot{a}_i^* + \sum_j L'_{ji} a_j^* + \sum_j \sum_k (Q'_{jik} + Q'_{jki}) a_k a_j^* - 2 \sum_j (K_{ji} a_j - M_{ji} a_{0j}) = 0 \tag{C.10}$$

Similarly, by solving equation C.6, it can be derived that:

$$\dot{b}_i^* + \sum_j [C'_{ji} - \sum_k (L^*_{jki} + \sum_m Q^*_{jkmi} a_m) a_k] a_j^* - \sum_j P'_{ji} b_j^* = 0, \tag{C.11}$$

where

$$\begin{aligned} C'_{ij} &= \frac{1}{\tau} \langle \boldsymbol{V}, \boldsymbol{\Phi}_i \psi_j \rangle \\ P'_{ij} &= \langle \boldsymbol{V} \cdot \nabla \psi_j, \psi_i \rangle \end{aligned} \tag{C.12}$$

It can be found easily that equation C.10 and equation C.11 are identical to the adjoint equations in chapter 3.

By solving equation C.4, the gradient can be obtained as the optimality of the system. As mentioned the exact form of gradient varies for different forms of controls. Use one-degree-of-freedom control of a 2D oscillatory cylinder flow for example, the solid velocity

$$\boldsymbol{V} = 2\pi f \gamma \sin(2\pi f t) \boldsymbol{e}_y. \tag{C.13}$$

Plug it into equation C.4, the gradient is obtained as:

$$g(\gamma) = \int_T [\sum_i^N (- \sum_m b_m C'_{im}) a_i^* + \sum_i^M (\sum_m b_m P'_{im}) b_i^*] \cdot V_\gamma dt, \tag{C.14}$$

where

$$\begin{aligned} C'_{ij} &= \frac{1}{\tau} \langle \boldsymbol{e}_y, \boldsymbol{\Phi}_i \psi_j \rangle \\ P'_{ij} &= \langle \boldsymbol{e}_y \cdot \nabla \psi_j, \psi_i \rangle, \end{aligned} \tag{C.15}$$

and $V_\gamma = 2\pi f \sin(2\pi f t)$.

Appendix D

Separating axis theorem to detect collision between swimmers

The separating axis theorem (SAT) is a mathematical algorithm to determine if two 2D convex polygons intersect or not[168], which can be adopted to detect collision between 2D swimmers. The basic idea is to check if there is an axis that separates the two swimmers. If such an axis exists, then two swimmers will not collide into each other. The algorithm can be summarized as follow by pseudo code:

Algorithm 2: The collision detection function using SAT in iteration k.

```
Function checkCollision(swimmer i, swimmer j):
    for each edge in swimmer i:
        axis = perpendicular(edge) # get the perpendicular axis to the edge
        min1, max1 = projectSwimmer(axis, swimmer i) # project every vertex of
        swimmer i onto the axis
        min2, max2 = projectSwimmer(axis, swimmer j) # project every vertex of
        swimmer j onto the axis
        if (min1 > max2 or min2 > max1): # checking collision
            return False # two swimmers are separated by the axis, no collision
    Repeat above steps for swimmer j
    return True # no separating axis is found, collision will happen
```

This algorithm can only work for 2D convex shapes, so it is not viable for 3D or concave shapes. However, the extension to these shapes is beyond the scope of this work.