

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE
ENGINEERING AND TECHNOLOGY

**LOAD ESTIMATION USING DIGITAL PARTICLE IMAGE VELOCIMETRY
DATA FOR FLAPPING WING AERODYNAMICS**

M.Sc. THESIS

Onur PAÇA

Department of Aeronautics and Astronautics Engineering

Aeronautics and Astronautics Engineering Program

Thesis Advisor: Prof. Dr. N. L. Okşan ÇETİNER YILDIRIM

JUNE, 2015

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE
ENGINEERING AND TECHNOLOGY

**LOAD ESTIMATION USING DIGITAL PARTICLE IMAGE VELOCIMETRY
DATA FOR FLAPPING WING AERODYNAMICS**

M.Sc. THESIS

**Onur PAÇA
(511131166)**

Department of Aeronautics and Astronautics Engineering

Aeronautics and Astronautics Engineering Program

Thesis Advisor: Prof. Dr. N. L. Okşan ÇETİNER YILDIRIM

JUNE, 2015

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**ÇIRPAN KANAT AERODİNAMİĞİ UYGULAMALARI İÇİN, PARÇACIK
İZLEYEREK HIZ BELİRLEME YÖNTEMİ İLE ELDE EDİLEN VERİLERDEN
KUVVET TAHMİNİ**

YÜKSEK LİSANS TEZİ

**Onur PAÇA
(511131166)**

Uçak-Uzay Mühendisliği Anabilim Dalı

Uçak-Uzay Mühendisliği Yüksek Lisans Programı

Tez Danışmanı: Prof. Dr. N. L. Okşan ÇETİNER YILDIRIM

HAZİRAN 2015

Onur Paça, a M.Sc. student of ITU **Institute of Graduate School of Science Engineering and Technology** student ID 511131166, successfully defended the thesis/dissertation entitled “**LOAD ESTIMATION USING DIGITAL PARTICLE IMAGE VELOCIMETRY DATA FOR FLAPPING WING AERODYNAMICS**”, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Dr. N.L. Okşan ÇETİNER YILDIRIM**
İstanbul Technical University

Jury Members : **Prof. Dr. Fırat Oğuz EDİS**
İstanbul Technical University

Ast. Prof. Dr. Uğur Oral ÜNAL
İstanbul Technical University

Date of Submission : 04 May 2015

Date of Defense : 29 May 2015

To my parents,

FOREWORD

This study is supported by TUBITAK Grant 112M682 and it is conducted within the context of FP7 Project 605151 - “Non-Intrusive Optical Pressure and Loads Extraction for Aerodynamic Analysis (NIOPLEX)”.

I would like to thank my supervisor Prof. Dr. Okşan Çetiner for giving me valuable advice and support always when needed. Other important persons considering my thesis are Onur Son, Dr. İdil Fenerciođlu, Emre Turam, Berk Zalođlu, Cihad Kse and Ferhat Karakaş whom I like to give my special thanks for supporting me.

June 2015

Onur PAÇA
Astronautical Engineer

TABLE OF CONTENTS

	<u>Page</u>
FOREWORD	ix
TABLE OF CONTENTS	xi
ABBREVIATIONS	xiii
LIST OF TABLES	xv
LIST OF FIGURES	xvii
SUMMARY	xix
ÖZET	xxi
1. INTRODUCTION	1
2. LITERATURE REVIEW	3
2.1 Load Estimation	3
2.2 Plunge Motion	6
3. FORCE ESTIMATION ALGORITHM	9
3.1 Theoretical Considerations	9
3.1.1 Unsteady Term	10
3.1.2 Flux Term	10
3.1.3 Pressure Term	11
3.2 Numerical Considerations	15
3.3 Details of the Code	17
4. DETAILS OF THE EXPERIMENTAL DATA UNDERTAKEN	19
4.1 General Information About Flow and Motion System	19
4.2 Sensitivity Analysis Cases	20
4.3 Different Plunge Motion Cases	23
5. RESULTS OF SENSITIVITY ANALYSIS	27
5.1 Selection of Run Variations	27
5.2 Spatial and Temporal Resolution	32
6. RESULTS FOR DIFFERENT PLUNGE CASES	39
7. CONCLUSION	53
7.1 Concluding Remarks	53
7.2 Recommendations for Further Studies	54
REFERENCES	55
APPENDICES	59
APPENDIX A.1	59
CURRICULUM VITAE	78

ABBREVIATIONS

CCD	: Charged Coupled Device
CD	: Drag Coefficient
CL	: Lift Coefficient
CPU	: Central Processing Unit
DPIV	: Digital Particle Image Velocimetry
FTC	: Fluid Trajectory Correlation
HO	: Higher Order
LO	: Lower Order
İTÜ	: İstanbul Teknik Üniversitesi
MAVs	: Micro Aerial Vehicles
NaN	: Not-a-Number
Nd:Yag	: Neodymium-Doped Yttrium Aluminium Garnet
PIV	: Particle Image Velocimetry
Re	: Reynolds Number
RTT	: Reynolds Transport Theorem
TAL	: Trisonik Araştırma Laboratuvarı
TR-PIV	: Time-Resolved Particle Image Velocimetry
St	: Strouhal Number
VI	: Virtual Interface

LIST OF TABLES

	<u>Page</u>
Table 4.1: Experimental Parameters for Added Mass Case	25
Table 5.1: Sensitivity Analysis Cases Work	33
Table 6.2: Trends of CD and CL accuracy	51

LIST OF FIGURES

	<u>Page</u>
Figure 2.1 : Drag/Thrust as a function of h and k.....	7
Figure 3.1 : Control Volume Representation.	9
Figure 3.2 : Interpolation grid for velocity.....	13
Figure 3.3 : Pressure gradient integration grid.	14
Figure 4.1 : Experimental setup.....	19
Figure 4.2 : The motion of the airfoil and the instants the velocity fields are acquired.	21
Figure 4.3 : The raw PIV image obtained when the airfoil is at the maximum of its plunge motion (low spatial resolution).	22
Figure 4.4 : The raw PIV image obtained when the airfoil is at the maximum of its plunge motion (high spatial resolution).	23
Figure 4.5 : Investigated test cases.	24
Figure 5.1 : HO vs. LO Unst. Surf. Instantaneous Results for Re 5000 freq 8 Hz....	27
Figure 5.2 : HO vs. LO Unst. Surf. Phase Averaged Results for Re 5000 freq 8 Hz.	28
Figure 5.3 : HO vs. LO Unst. Surf. Moving Ave Results for Re 5000 freq 8 Hz.	28
Figure 5.4 : HO vs. LO Unst. Line Instantaneous Results for Re 5000 freq 8 Hz.	29
Figure 5.5 : HO vs. LO Unst. Line Phase Averaged Results for Re 5000 freq 8 Hz.	29
Figure 5.6 : HO vs. LO Unst. Line Moving Ave Results for Re 5000 freq 8 Hz.	29
Figure 5.7 : Unst. Line vs. Unst Surf. HO Inst. Results for Re 5000 freq 8 Hz.....	30
Figure 5.8 : Unst. Line vs. Unst Surf. HO Phase Ave. Results for Re 5000 freq 8 Hz.	30
Figure 5.9 : Unst. Line vs. Unst Surf. HO Moving Ave. Results for Re 5000 freq 8 Hz.	31
Figure 5.10 : Unst. Line vs. Unst Surf. LO Inst. Results for Re 5000 freq 8 Hz.	31
Figure 5.11 : Unst. Line vs. Unst Surf. LO Phase Ave. Results for Re 5000 freq 8 Hz.	31
Figure 5.12 : Unst. Line vs. Unst Surf. LO Moving Ave. Results for Re 5000 freq 8 Hz.	32
Figure 5.13 : Re 5000 freq 8 Hz Instantaneous.	34
Figure 5.14 : Re 10000 freq 8 Hz Instantaneous.	34
Figure 5.15 : Re 5000 freq 8 Hz Period Averaged.	34
Figure 5.16 : Re 10000 freq 8 Hz Period Averaged.	35
Figure 5.17 : Re 5000 freq 15 Hz Instantaneous.	35
Figure 5.18 : Re 10000 freq 15 Hz Instantaneous.	35
Figure 5.19 : Re 5000 freq 15 Hz Period Averaged.	36
Figure 5.20 : Re 10000 freq 15 Hz Period Averaged.	36
Figure 5.21 : Re 5000 freq 80 Hz Instantaneous.	37
Figure 5.22 : Re 5000 freq 80 Hz Ave Correlation.	37
Figure 6.1 : CL for Case 1 [k=5.01 Re=1250 f=0.1 h _{amp} /c=0.50].	39
Figure 6.2 : CD for Case 1 [k=5.01 Re=1250 f=0.1 h _{amp} /c=0.50].....	40
Figure 6.3 : CL for Case 2 [k=5.01 Re=2500 f=0.2 h _{amp} /c=0.25].	40
Figure 6.4 : CD for Case 2 [k=5.01 Re=2500 f=0.2 h _{amp} /c=0.25].....	41

Figure 6.5 : CL for Case 3 [k=1.00 Re=3125 f=0.05 h _{amp} /c=1.00].....	41
Figure 6.6 : CD for Case 3 [k=1.00 Re=3125 f=0.05 h _{amp} /c=1.00].	42
Figure 6.7 : CL for Case 4 [k=1.25 Re=5000 f=0.10 h _{amp} /c=0.50].....	42
Figure 6.8 : CD for Case 4 [k=1.25 Re=5000 f=0.10 h _{amp} /c=0.50].	43
Figure 6.9 : CL for Case 5 [k=2.50 Re=5000 f=0.20 h _{amp} /c=0.25].....	43
Figure 6.10 : CD for Case 5 [k=2.50 Re=5000 f=0.20 h _{amp} /c=0.25].	44
Figure 6.11 : CL for Case 6 [k=0.63 Re=10000 f=0.10 h _{amp} /c=0.50].....	44
Figure 6.12 : CD for Case 6 [k=0.63 Re=10000 f=0.10 h _{amp} /c=0.50].....	45
Figure 6.13 : CL for Case 7 [k=1.25 Re=10000 f=0.20 h _{amp} /c=0.25].....	45
Figure 6.14 : CD for Case 7 [k=1.25 Re=10000 f=0.20 h _{amp} /c=0.25].....	46
Figure 6.15 : CL for Case 8 [k=1.25 Re=12500 f=0.05 h _{amp} /c=1.00].....	46
Figure 6.16 : CD for Case 8 [k=1.25 Re=12500 f=0.05 h _{amp} /c=1.00].....	47
Figure 6.17 : CL for Case 9 [k=1.00 Re=3129 f=0.05 h _{amp} /c=1.00].....	47
Figure 6.18 : CD for Case 9 [k=1.00 Re=3129 f=0.05 h _{amp} /c=1.00].	48
Figure 6.19 : CL for Case 10 [k=1.00 Re=3129 f=0.10 h _{amp} /c=0.50].....	48
Figure 6.20 : CD for Case 10 [k=1.00 Re=3129 f=0.10 h _{amp} /c=0.50].....	49
Figure 6.21 : CL for Case 11 [k=4.00 Re=3129 f=0.20 h _{amp} /c=0.25].....	49
Figure 6.22 : CD for Case 11 [k=4.00 Re=3129 f=0.20 h _{amp} /c=0.25].....	50

LOAD ESTIMATION USING DIGITAL PARTICLE IMAGE VELOCIMETRY DATA FOR FLAPPING WING AERODYNAMICS

SUMMARY

The current study aims to develop an in-house code to estimate loads for a flapping wing using Digital Particle Image Velocimetry (DPIV) data. For the sake of simplicity, pure plunge motion is considered and experiments were performed using different space and time resolutions to assess the performance of the load estimation. Then, the developed code has been tested for different plunge motion cases. Although the study is focused on the load estimation, it make uses of a large quantity of DPIV and direct force measurement data.

In order to calculate the force acting on the flapping airfoil, Reynolds Transport Theorem (RTT) can be utilized. As the velocity field is provided by the DPIV data in time, in order to compute the forces acting on the body, one has to obtain the pressure field around it. Present study includes the estimation of forces by using the pressure field which is obtained by the explicit integration of planar pressure gradients. The pressure gradients are obtained by the use of Navier-Stokes Equations in two-dimensional form. In Navier-Stokes Equations, material derivative is required to compute the pressure gradients. In order to compute the material derivative, two methods are used succesively in this work, namely Eulerian and Lagrangian approaches. The thesis describes the algorithm in detail with its variations. As a result higher order differentiation and surface integral for unsteady term yielded a better performance in force prediction.

According to the sensitivity analyses, a spatial resolution of 3% of the chord and a field of view covering all the amplitude of motion gave a satisfactory results with a temporal resolution expressed as 40 vector fields per cycle of motion. In terms of DPIV post-processing, average correlation was found to reduce the noise as it is expected. On the other hand, the estimated load matched better the direct measurement when it was phase averaged and filtered. It should be noted that the compared direct measurement results were also 1 Hz filtered.

For all the investigated results, both for the sensitivity analyses and general plunge cases, lift coefficient variation was in general predicted well, in agreement with the

direct force measurements, however the estimated drag coefficient variation had a very large discrepancy with the direct force measurements. This might be due to the neglected viscous term. Consequently, the fact disabled thrust/drag analyses for plunge cases. Estimation of the lift force coefficient variation exhibited a nearly perfect match even on the instantaneous results when the plunge motion has the lowest frequency and the largest amplitude.

**ÇIRPAN KANAT AERODİNAMİĞİ UYGULAMALARI İÇİN, PARÇACIK
İZLEYEREK HIZ BELİRLEME YÖNTEMİ İLE ELDE EDİLEN
VERİLERDEN KUVVET TAHMİNİ
ÖZET**

Mevcut çalışmanın amacı, çırpma (düşey salınım) hareketi yapan bir kanadın üzerine etkiyen aerodinamik kuvvetlerin tahmini için bilgisayar programı geliştirmektir. Kuvvet tahmini için Dijital Parçacık Takip Ederek Hız Belirleme (DPIV - *Digital Particle Image Velocimetry*) yöntemiyle kanadın etrafındaki hız alanı elde edilmiştir. Basit olması amacıyla, kanada yalnızca düşey salınım hareketi yaptırılmıştır. Kuvvet tahmininin başarısını sınamak amacıyla farklı zaman ve mekan çözünürlüklerinde deneyler gerçekleştirilmiş ve kuvvet tahmininin bu çözünürlüklere hangi ölçüde bağlı olduğu belirlenerek hassasiyet analizi gerçekleştirilmiştir. Daha sonra hazırlanan yazılım farklı düşey salınım parametrelerine sahip olan kanat hareketleri için çalıştırılmıştır. Bu çalışma her ne kadar kuvvet tahminine odaklanmış olsa da büyük miktarda DPIV verileri ve kuvvet ölçüm sonuçlarını kullanmaktadır.

Çırpın kanada etki eden aerodinamik kuvvetlerin hesaplanması için Reynolds Taşınım Teoremi (RTT – *Reynolds Transport Theorem*) kullanılabilir. RTT'nin kullanılabilmesi için hız alanı ve hız alanının zaman ve uzay türevleri ile basınç alanı (ayrıca sıkıştırılabilir durumda yoğunluk alanı) bilinmelidir. Hız alanı DPIV yöntemiyle elde edildiğine göre hız alanının zaman ve uzay türevleri de geometri ve zaman bilgileri sayesinde sayısal olarak elde edilebilir. Fakat basınç alanının elde edilmesi için daha uzun bir işlemler dizisi gereklidir. Mevcut çalışmada, basınç alanı hız alanı kullanılarak elde edilmiştir. Bunun için önce hız alanında düzlemsel basınç gradyenleri bulunmuş daha sonra kapalı bir şekilde integre edilerek kuvvet hesabı için gerekli basınç katkısı hesaplanmıştır. Bu düzlemsel basınç gradyenleri, Navier-Stokes Denklemleri kullanılarak elde edilmiştir. Navier-Stokes Denklemleri yardımıyla maddesel türev ve basınç gradyeni arasında ilişki kurulmuştur. Maddesel türev hesabı için Euler yaklaşımı ve Lagrange yaklaşımı art arda kullanılmıştır. Bu tezde, kuvvet hesabı algoritması tüm varyasyonlarıyla ayrıntılı bir şekilde anlatılmıştır. Sonuç olarak yukarıda bahsedilen hız alanının sayısal türevi için yüksek mertebeli ve düşük mertebeli yaklaşımlar karşılaştırılmış ve yüksek mertebeli işlemin daha iyi sonuç verdiği görülmüştür. Ayrıca RTT içindeki zamana bağlı terimin yüzey integrali ve

çizgi integrali olarak hesaplanmaları karşılaştırılmış ve yüzey integralinin daha iyi sonuç verdiği görülmüştür.

Kuvvet tahmini analizi önce bir test durumu için uygulanmıştır. Bu test durumu deneyleri FP7 Project 605151 - “Non-Intrusive Optical Pressure and Loads Extraction for Aerodynamic Analysis (NIOPLEX) - *Aerodinamik Analizler için Akışa Müdahalesiz Optik Yöntemlerle Basınç ve Yük Çıkarımı*” kapsamında İstanbul Teknik Üniversitesi (İTÜ) Trisonik Araştırma Laboratuvarı’nda (TAL) yapılmıştır. Hassasiyet analizi ile ilgili deneyler için NACA0012 kodlu simetrik ve %12 kalınlık oranına sahip kanat profili kullanılmıştır. Değişik zaman ve uzay çözünürlüklerinde gerçekleştirilen hassasiyet analizi sonuçlarına göre 8 Hz’lik bir veri alma frekansı 15 Hz’lik durumdan daha iyi sonuç vermektedir. Öte yandan 80 Hz’lik zaman çözünürlüğü ise 8 Hz ve 15 Hz’lik durumlardan çok daha kötü bir sonuç vermiştir. Dolayısıyla 8 Hz’lik zaman çözünürlüğü tatmin edici bir doğrulukta kuvvet tahmini yapılabilmesi için yeterlidir. 8 Hz’lik data alma işlemi çarpma hareketinin bir periyodunun 40 vektör alanıyla ifade edilmesi anlamına gelmektedir. Uzay çözünürlüğünde ise veter uzunluğunun %3’ünün yeterli olduğu görülmüştür. Bununla beraber hız alanı olarak tüm çarpma hareketinin kapsanması ve bir periyodun 40 vektör alanıyla ifade edilmesi halinde kuvvet tahmini başarılı sonuç vermektedir.

DPIV hız alanlarını kuvvet tahmini için daha da uygun hale getirmek üzere hız alanı elde edildikten sonra averaj korelasyon işlemi de uygulanmıştır. Ham veriye uygulanan bu işlem 80 Hz’lik durumda denenmiş ve averaj korelasyonun kuvvet tahmin sonuçlarındaki gürültüyü oldukça azalttığı görülmüştür. Öte yandan tahmin edilen kuvvet sonuçlarının faz averajı alındığında ve sonrasında bir periyotluk veriye filtre uygulandığı zaman ölçülen kuvvetlere çok daha yakın sonuçlar verdiği görülmüştür. Ayrıca karşılaştırma amaçlı ölçülen kuvvetlerin de 1 Hz’lik filtreye tabi tutulduğu göz önüne alınmalıdır.

Hasasiyet analizi deneyleri gibi değişik düşey salınım hareketi deneyleri de TAL’da yapılmıştır. Bu deneyler için hasasiyet analizi deneylerinden farklı olarak bir düz levha kullanılmıştır. Düz levha kullanılmasının sebebi zahiri kütle etkilerini analitik biçimde hesaplayabilmek ve böylece kanada etki eden net aerodinamik kuvveti belirleyebilmektir. Bu düz levhanın veter uzunluğu 30 cm ve kanat açıklığı 10 cm’dir. Tüm düşey salınım hareketi deneylerinde bir periyot 40 vektör alanıyla ifade edilmiş

ve toplamda her varyasyon için 30 periyot veri alınmıştır. Böylece gerçek zamanlı olarak toplam 1200 vektör alanı mevcuttur. Hem hassasiyet analizi deneyleri hem de daha sonraki düşey hareket deneylerinin kuvvet tahminleri incelendiğinde taşıma katsayısı tahmininin ölçülen değerlerle çok yakın sonuçlar verdiği görülmüştür. Fakat sürüklenme katsayısı tahminlerinde ise ölçülen değerler ve tahmin sonucu arasında çok fazla fark olduğu gözlemlenmiştir. Bu durumun RTT’de ihmal edilen viskoz terime bağlı olabileceğinden düşünülmektedir. Sonuç olarak, sürüklenme katsayısı tahminleri göz ardı edildiğinde taşıma katsayısı tahmininin gerçek zamanlı hız alanlarıyla hesaplandığında bile en düşük frekanslı ve en yüksek genlikli çarpma hareketi durumlarında neredeyse mükemmel bir biçimde ölçülen değerlerle eşleştiği görülmüştür.

1. INTRODUCTION

Biological inspiration is a recent interest to enhance the performance of the next generation of small-scale Micro Aerial Vehicles (MAVs) over existing fixed and rotary wing systems. Researchers aim to employ the unsteady mechanisms of moving, flapping and/or deforming, flexible wings to overcome the unfavorable aerodynamic conditions, therefore the unsteady flows in low Reynolds number regimes gain increasing attention in recent years. These phenomena are also highly relevant in more traditional fields; e.g the blades of helicopter or wind turbine rotors, or aircraft or marine propellers under unsteady conditions. In the aerodynamic analysis of the associated flow phenomena, the non-intrusive pressure and loads determination methods provide a convenient way to extract these data and relate them to the observed flow and wing-deformation phenomena. Currently, Particle Image Velocimetry (PIV) is the major diagnostic technique to obtain the mean flow field and turbulent fluctuations. In 2013, an FP7 project entitled “Non-intrusive Optical Pressure and Loads Extraction for Aerodynamic Analysis” and with an acronym of NIOPLEX has been started with a workpackage dedicated to unsteady wing aerodynamics. The main objective and ambition for the work in the NIOPLEX project is to develop and assess flow-diagnostic techniques that enable a comprehensive aerodynamics analysis to be obtained, through the simultaneous measurement of the surface pressure, the flow field and the pressure distribution inside an unsteady flow (van Oudheusden, 2013).

The current study aims to develop an in-house code to estimate loads for a flapping wing using Digital Particle Image Velocimetry (DPIV) data. For the sake of simplicity, pure plunge motion is considered and experiments were performed using different space and time resolutions to assess the performance of the load estimation. Then, the developed code has been tested for different plunge motion cases. Although the study is focused on the load estimation, it make uses of a large quantity of DPIV and direct force measurement data.

The overall objectives of the study can be summarized as follows:

- Develop an in-house code for load estimation of bodies in motion

- Determine the spatial and temporal resolutions needed for the DPIV experiments for an accurate load estimation
- Compare force estimation results with direct force measurement data and determine the shortcomings of the estimation algorithm

In order to compute the forces acting on the body, one has to obtain the pressure field around it. Present study includes the estimation of forces by using the pressure field which is obtained by the explicit integration of planar pressure gradients. The pressure gradients are obtained by the use of Navier-Stokes Equations in two-dimensional form. In Navier-Stokes Equations, material derivative is required to compute the pressure gradients. In order to compute the material derivative, two methods are used succesively in this work, namely Eulerian and Lagrangian approaches. The algorithm with its variations are presented in the third chapter after the second chapter devoted to the litterature review. The forth chapter presents the details of the experiments performed and it is followed by the sensitivy analysis results given in the fifth chapter. While Chapter 6 includes all the results covering load estimations for different plunge motion cases, Chapter 7 gives the concluding remarks and recommendations for further studies.

2. LITERATURE REVIEW

2.1 Load Estimation

The fact that flapping foils could lend themselves to both thrust generation and energy extraction from the surrounding fluids significantly have drawn attention in the last decade. Although the basic mechanism of thrust generation is readily known (Jones et al., 1996), investigations for a complete understanding of the generated forces with the flapping motions are still required. One of the many possible ways to obtain the forces acting on a flapping wing is direct force measurements. However, unless otherwise identified by an encoder, the load cells measure the total of forces acting on the foil in the experimental medium. This results in a more important problem when an experiment is performed in a water channel, since the measured forces are constituted mainly of the non-circulatory forces due to the fact that medium in which the foil moves is water. In flapping foil research where the airfoil has large acceleration, the contribution of the effect of the added mass on the measured data might even be higher than the contribution of the circulatory forces of the airfoil (Rival et al., 2009). Hence the identification of the forces acting on flapping foils through non-intrusive methods is necessary.

Particle Image Velocimetry has become an important non-intrusive flow measurement technique since late 1990's. It has rapidly developed into a trusted and versatile technique for flow field measurement, capable of delivering a detailed experimental characterization of the flow in terms of (large) ensembles of instantaneous velocity fields (van Oudheusden, 2009). On the other hand, measuring the aerodynamic forces experienced by a body is of major interest when dealing with flow control applications and strain gauges are widely used for steady flow configurations whereas piezo-electric devices appear as an alternative solution for unsteady flow configurations (David et al., 2009). As those techniques are limited to a specific range of loads, in parallel with the development of PIV, researchers started to estimate loads based on PIV data and using integration of flow variables inside and around a control volume surrounding the body (Noca et al., 1997, 1999; Unal et al., 1997). As this approach allows a direct link between flow behavior and force mechanisms, it is found to be particularly powerful besides the fact that the load characterization is performed in a

non-intrusive manner (Jardin et al., 2009). Further details on the development of load estimation can be found in the review paper of van Oudheusden (2013).

Load estimation by utilizing the velocity field which is obtained by DPIV method has drawn so much attention in recent years (Violato et. al, 2011; Mohebbian and Rival, 2012; Tronchin et al., 2015). Since all the numerical computations somewhat impair the accuracy of the procedure by truncation errors, there is a significant endeavour present in literature to minimize these errors with different methods. These errors should be taken into account both in the calculation of material derivative and in the integration of the information about the pressure, whether it is an explicit or implicit integration. The most prominent method is to apply the Eulerian and Lagrangian approaches successively for attaining the material derivative by using the former as an initial information for the computation of the latter. If we overview the research topic in the past decade:

Liu and Katz (2006) used a two camera and four-exposure PIV system to obtain the instantaneous pressure fields and material accelerations. They applied the Eulerian and Lagrangian methods for a synthetic flow case that was generated using MATLAB. The synthetic PIV images had $2048 \text{ pixels} \times 2048 \text{ pixels}$ resolution and $32 \text{ pixels} \times 32 \text{ pixels}$ interrogation window. They proposed an omni-directional integration method for getting pressure field from planar pressure gradient.

Kurtuluş et al. (2006) investigated the flow around a square cylinder with TR-PIV in order to estimate the unsteady forces acting on it. They presented the terms in the force equation, namely unsteady, flux and pressure and emphasized which term is dominant in which component of the force exerted on the cylinder. It was stated that the convective term is more significant than the others in lift coefficient and the pressure term in the drag coefficient.

van Oudheusden et al. (2007) calculated the integral forces around various geometries using PIV velocity data. They addressed the calculation of time-mean pressure data and the evaluation of forces using velocity ensemble data.

de Kat et al. (2008) performed the instantaneous pressure field determination with the same Eulerian and Lagrangian approach and compared the computation with the measured pressure information that they acquired using microphones.

Violato et al. (2010) applied Eulerian and Lagrangian approximations for the computation of pressure field around a rod-airfoil geometry using the 3D velocity field that they acquired using TR-TOMO PIV with a temporal separation of 5 kHz and showed that when in need of a relative precision error of maximum 10% the Eulerian method requires a better temporal separation than the Lagrangian method.

de Kat and van Oudheusden (2011) presented the fundamentals for performing the Eulerian and Lagrangian considerations and tested with a synthetic flow case. They proposed that the interrogation windows must be set to be 5 times smaller than the turbulent flow structures and the temporal separation must be 10 times higher than the flow frequency. They also performed stereoscopic and tomographic PIV experiments around a square cylinder.

All of the aforementioned studies apply the Eulerian and Lagrangian approximations successively. The Eulerian method requires the calculation of convective and local derivatives. The convective derivatives are calculated using the velocity information in adjacent time steps. The local contribution is calculated within every time step at the spatial nodes. However, for the Lagrangian perspective these studies track the same group of particles and calculate the velocity difference of the same group of fluid particles by using the Taylor Series. The Eulerian calculation is used as an input for the Lagrangian method. The second approach to calculate the material derivative is to track the same group of particles by estimating a path for those particles. There are also a number of studies conducted with this approach.

In 2012, Novara and Scarano presented their work based on the combination of PIV and 3D Particle Tracking Velocimetry (3D-PTV) for a better accuracy (Novara and Scarano, 2012). This investigation showed that the reconstruction of particles paths is favorable for the reduction of errors. They implemented this procedure on the synthetic vortex-ring field.

Lynch and Scarano (2013) proposed the Fluid Trajectory Correlation (FTC). In FTC method, the temporal derivative is calculated using the polynomial path that is fitted for the fluid particles. In 2014, the same authors compared Eulerian and Lagrangian consecutive calculation with the FTC using four-pulse TOMO-PIV data and showed

that the latter technique is more accurate regarding the errors (Lynch and Scarano, 2014).

Recently, Tronchin et al. (2015) studied loads and pressure fields calculated using velocity fields obtained with a scanning tomography PIV. The experiments were performed to investigate the flow around a NACA0012 airfoil for a flapping motion in a water tank at a Reynolds number of 1000.

2.2 Plunge Motion

Due to the simplicity of the motion, the earliest theories concerning flapping wing flight are related to purely heaving airfoils. The theory of thrust generation using flapping foils was first proposed by Knoller (1909) and Betz (1912) and then experimentally confirmed by Katzmayr (1922). The Knoller-Betz theory states that a harmonically plunging wing in a freestream flow results in generation of an effective angle of attack and when the airfoil is oscillated at sufficiently high amplitude and frequency, the downstream velocity distribution becomes jet-like and thus is indicative of a net thrust on the airfoil.

Jones et al. (1998) demonstrated the Knoller-Betz effect by performing flow visualization experiments in the wake of a purely plunging airfoil model and according to positions of shed vortices, they classified their results as drag producing, neutral, thrust producing and dual-mode thrust producing categories. They defined the non-dimensional plunge velocity by the multiplication of the reduced frequency (k) and the non-dimensional plunging amplitude (h) which essentially has the same meaning as Strouhal number in characterizing the type of vortex shedding in the wake of a flapping airfoil. As given in Figure 1, plunge amplitude times reduced frequency value is plotted as a constant dividing line between the drag producing and thrust producing parameters for a single plunging airfoil.

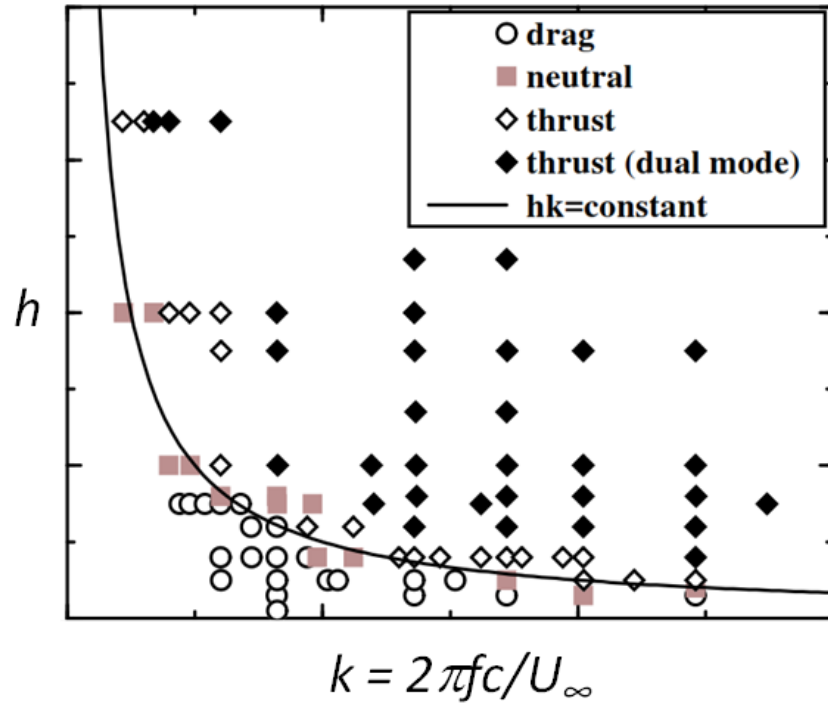


Figure 2.1: Drag/Thrust as a function of h and k (Jones et al. 1996)

In a previous study, Fenercioglu and Cetiner (2012) also observed similar occurrence for the threshold of thrust producing wake. They categorized the flow structures around and in the near wake of the airfoil based on different independent parameters, obtained from quantitative flow field measurements using DPIV (Digital Particle Image Velocimetry) method in a water channel for a pitching and plunging airfoil in a range applicable to Micro Air Vehicles (MAVs). In the preceding study, average thrust force and efficiency estimate from the DPIV velocity data by simply using the wake excess velocity and the momentum theorem expressed for a control volume bounded by a control surface, using the steps provided by Anderson et al. (1998) (Fenercioglu [2010]). The same water channel facility and the same airfoil kinematics were used in recent studies by Karakas et al. (2014) and Caylan et al. (2014) where simultaneous direct force acquisitions were performed in addition to DPIV measurements. One of the major problems encountered in direct force acquisition in water channels is the added mass effect and the ratio of this added mass force to the total measured force needs to be revealed in order to comment on the net force acting on a flapping airfoil, therefore the efficiency of flapping. In the aforementioned previous studies, the added mass effect was prevailing particularly for the cases with higher thrust production and

lessens for the cases with lower thrust and higher efficiency values. In those studies the airfoil's combined flapping motion was dominated by the plunging component, as also given in Fenercioglu and Cetiner (2014), thus the focus of the the experiments run for laod estimation is on simple plunging motion to take into account the contribution of the added mass in the direct force measurements.

3 FORCE ESTIMATION ALGORITHM

3.1 Theoretical Considerations

In order to calculate the force acting on the flapping airfoil, Reynolds Transport Theorem (RTT) can be utilized:

$$\vec{F}(t) = -\rho \iiint_V \frac{\partial \vec{V}}{\partial t} dV - \rho \iint_S (\vec{V} \cdot \vec{n}) \vec{V} dS - \iint_S p \vec{n} dS + \iint_S \vec{\tau} dS \quad (3.1)$$

RTT must be applied for a fixed control volume that encompasses the body. Control volume selection is very important for the calculation of unsteady force term. In DPIV procedure the regions that are close to the edges of the images may produce bad vectors, thus the control volume should not be selected too close to the edges. On the other hand, the control surface should be far enough from the airfoil so that it is not interrupted by the geometry. The control volume is selected regarding these points. A sample control volume is sketched in Figure 3.1.

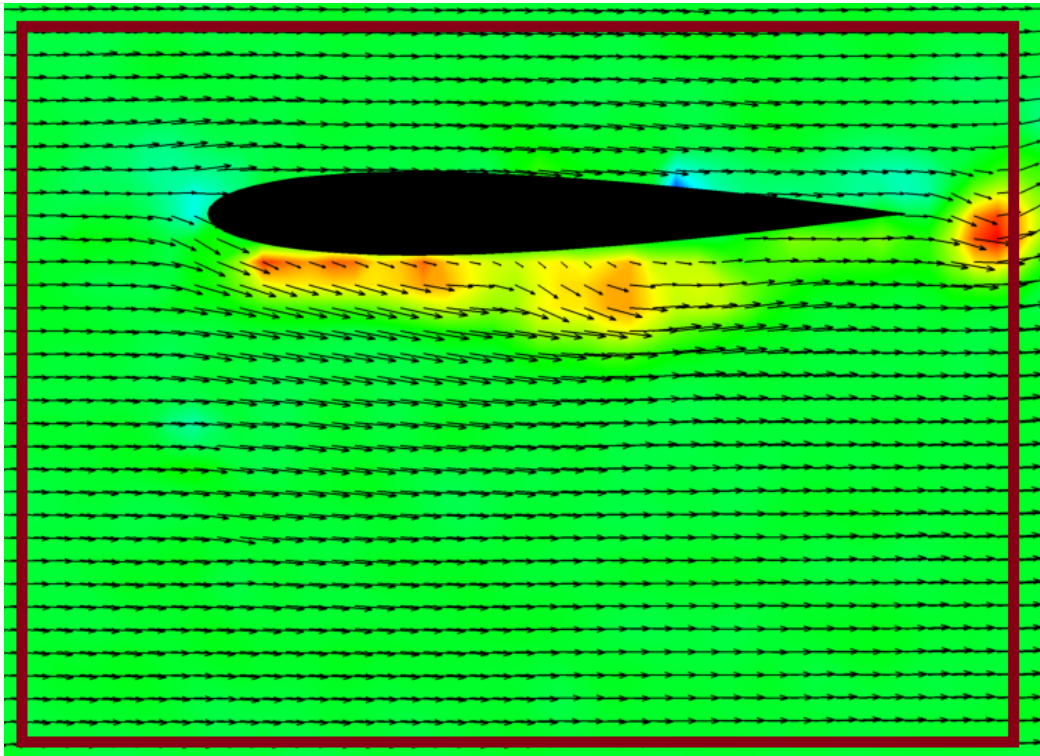


Figure 3.1: Control Volume Representation

After applying Equation 3.1 to a control volume around the flapping wing the the RTT is rewritten. Unsteady forces acting on the body are given by Kurtuluş et al. (2006):

$$\begin{bmatrix} D \\ L \end{bmatrix} = -\rho \iint_V \begin{bmatrix} \frac{\partial u}{\partial t} dx dy \\ \frac{\partial v}{\partial t} dx dy \end{bmatrix} + \rho \oiint_S \begin{bmatrix} -u^2 dy + uv dx \\ -uv dy + v^2 dx \end{bmatrix} + \oint_S \begin{bmatrix} -p dy \\ p dx \end{bmatrix} \quad (3.2)$$

In Equation 3.2, unsteady drag and lift forces are given for a two dimensional case. In the following subchapters, the terms on the right hand side are explained in more detail.

3.1.1 Unsteady Term

In Equation 3.2, the first term on the right hand side is unsteady term which can be calculated with the velocity field provided by DPIV experiments. As can be seen in Equation 3.2 the unsteady term is a surface integral, however, it can be rewritten as a line integral through the control surfaces as given by Mohebbian and Rival (2012):

$$\iiint_V \frac{\partial \vec{V}}{\partial t} dV = \frac{\partial}{\partial t} \iint_S \vec{r}(\vec{V} \cdot \vec{n}) dS \quad (3.3)$$

Both line integration and surface integration are calculated in this study for comparison. As described in the next chapter, higher and lower order differentiations are aslo compared. So, with two different unsteady term calculation methods and two different numerical differentiation orders there are 4 different calculation procedures for each case.

For this purpose, the velocities at different time steps are used for the derivation in time. The numerical schemes are given in the following chapters. The numerical differentiation introduces some truncation errors. The integral is calculated within the control volume at each cell surface.

3.1.2 Flux Term

The second term on the right hand side is the flux term which can also be calculated using the velocity field. For this calculation the velocities at the vicinage grids at each time step are used. The flux term is calculated on the control surfaces. The integration

is carried out by in counter-clockwise direction. Also, the normal vectors on the edges are taken into account.

3.1.3 Pressure Term

Finally, the last term is the pressure term and since the pressure field is not provided by DPIV it takes a long procedure to calculate the last term. The pressure field must be calculated from the velocity field information using the flow constituent equations, namely momentum equations (Navier-Stokes). The momentum equation, with neglecting the gravitational term, can be given as follows in vectoral form:

$$\vec{\nabla} p = -\rho \frac{D\vec{V}}{Dt} + \mu \nabla^2 \vec{V} \quad (3.4)$$

where p is pressure, \vec{V} is velocity vector, ρ is density and μ is dynamic viscosity. $\vec{\nabla}$ and ∇^2 are gradient and curl operators respectively.

The explicit form of the momentum equation in x -direction:

$$\frac{\partial p}{\partial x} = -\rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (3.5)$$

The explicit form of the momentum equation in y -direction:

$$\frac{\partial p}{\partial y} = -\rho \left(\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right) + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (3.6)$$

In the explicit form of these equations the first term on the right hand side in paranthesis are convective derivatives and the other three terms are local derivatives.

The term on the left hand side is the pressure gradient which constitutes the main goal of this study. The first term on the right hand side is the material derivative and the last term is the viscous term. The viscous term is neglected as stated by Violato et al. (2010) when $Re > 1000$. The Eulerian approach to material derivative in vectoral form is:

$$\frac{D\vec{V}}{Dt} = \frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \vec{\nabla}) \vec{V} \quad (3.7)$$

The explicit form of the material derivative in x -direction:

$$\frac{Du}{Dt} = \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \quad (3.8)$$

The explicit form of the material derivative in y -direction:

$$\frac{Dv}{Dt} = \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \quad (3.9)$$

The Eulerian derivatives can be computed with the velocities at the predefined nodes which are provided by DPIV measurements. Numerical schemes are used for the derivatives. The details of the numerical work is given in the next section.

For the Lagrangian approach the velocities of the same group of particles are needed in a certain timespan:

$$\left. \frac{D\vec{V}}{Dt} \right|_x = \frac{\vec{V}(\mathbf{x}_{t+\Delta t}, t + \Delta t) - \vec{V}(\mathbf{x}_{t-\Delta t}, t - \Delta t)}{2\Delta t} \quad (3.10)$$

where the \mathbf{x} is the position vector of the fluid particles. At each time step the velocities and spatial coordinates are known at certain grid points. The spatial positions of each fluid particle can be found by using Taylor Series. Since the Taylor Series is invoked in locating the coordinates it will be highly unlikely for the fluid particles to end up at regular predefined grid points. Once the positions of the fluid particles are known at adjacent temporal separations the velocities of the particles at those new positions can be interpolated by using the velocities and coordinates at the regular grid points. The Taylor Series for the positions of the particles:

$$\mathbf{x}_{t+\Delta t} = \mathbf{x}_t + \dot{\mathbf{x}}_t \Delta t + \frac{1}{2} \ddot{\mathbf{x}}_t \Delta t^2 + O(\Delta t^3) \quad (3.11)$$

$$\mathbf{x}_{t-\Delta t} = \mathbf{x}_t - \dot{\mathbf{x}}_t \Delta t + \frac{1}{2} \ddot{\mathbf{x}}_t \Delta t^2 - O(\Delta t^3) \quad (3.12)$$

The Taylor Series expansion is used with forward and backward time steps for using a central numerical differentiation scheme. In the above expansions the first derivative of the position with respect to time is velocity whereas the second derivative is the acceleration. As mentioned before the Eulerian acceleration is used as an input for the

Lagrangian acceleration computation. After the first iteration for calculating the Lagrangian derivative using Eulerian derivative as an input for the Taylor Series, the iterative procedure is resumed by the output of the first iteration. So, for the second iteration the Lagrangian derivative is used for the Lagrangian calculation. The number of iteration is limited in each case due to high CPU-time.

For the interpolation purpose 4 closest nodes in the vicinity of the final destination of the fluid particles are used. As can be seen In Figure 3.2 the corresponding areas are calculated and the reciprocal of these areas are used as the weighting parameter for each node, so that the closest node has the biggest weight and the furthest node has the smallest contribution in the weighted averaging.

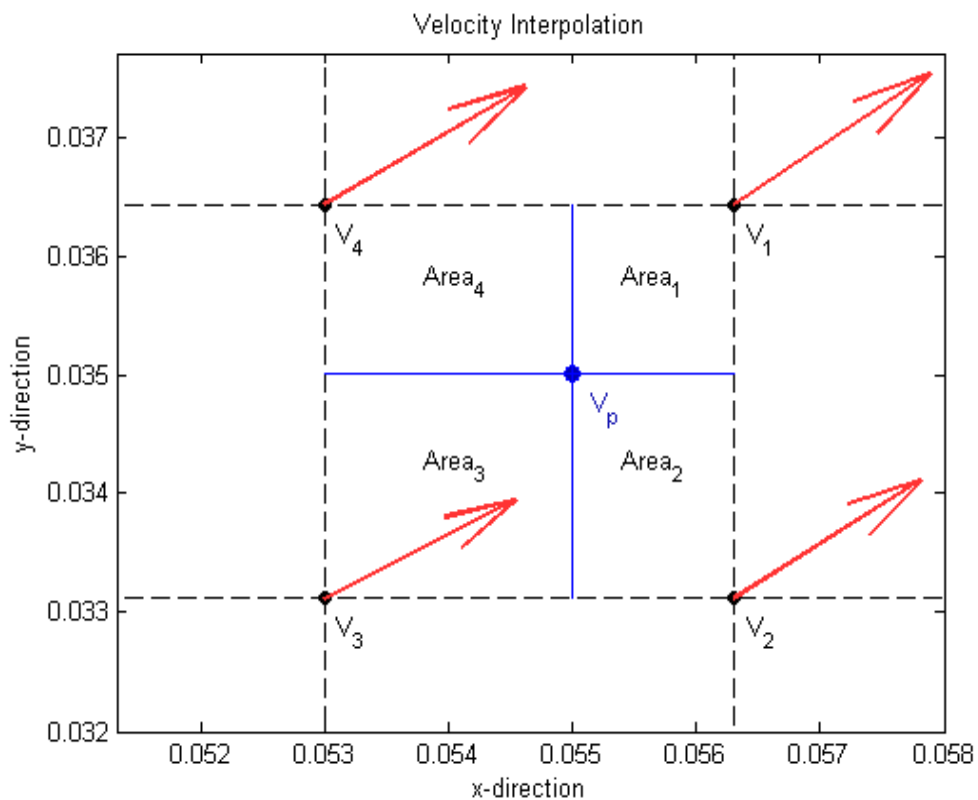


Figure 3.2: Interpolation grid for velocity

If the final spatial coordinate of the tracked particle is outside of the flow domain this approach is not applicable hence, in such cases the Eulerian acceleration is used instead of the Lagrangian one.

After obtaining the material derivative in Lagrangian form, the pressure gradients can be computed easily. As it can be seen in Equation 3.4 the pressure gradient is equal to the sum of material derivative and viscous term. The viscous term in the right hand side of the Equation 3.5 and Equation 3.6 can be computed easily by numerical differentiation on regular nodes in hand. The viscous term in pressure gradient calculation is taken into account whereas the viscous term in the RTT is omitted.

At this stage the pressure gradients in each direction on the 2D plane are known. There are several ways to integrate these pressure gradients. In this study an explicit scheme is used. Since the pressure gradient is known at each node the pressure of an arbitrary grid point P_{ij} can be obtained by the four different nodes that lie in upstream of the P_{ij} . These nodes are sketched in Figure 3.3

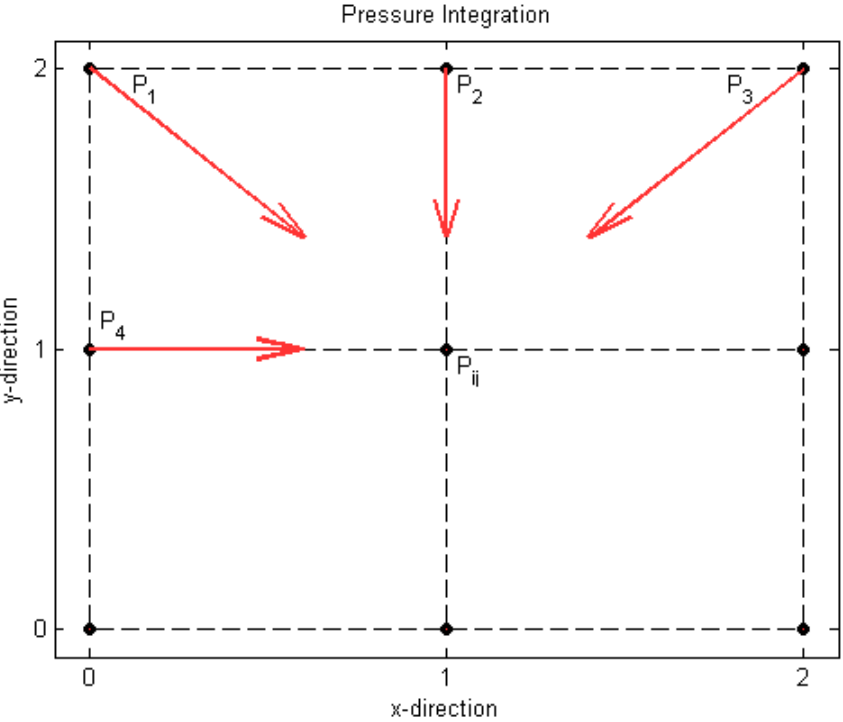


Figure 3.3: Pressure gradient integration grid

In order to take advantage of the four different pressure information in the upstream of the P_{ij} these contributions are averaged and thus the pressure at P_{ij} is calculated.

The contributions of the nodes 4 already calculated nodes are given as follows by Baur and Kongeter (1998):

$$\Delta p = \int dp = \int \left[\frac{\partial p(x, y)}{\partial x} dx + \frac{\partial p(x, y)}{\partial y} dy \right] \quad (3.13)$$

$$\Delta P_{i,j} = \left[\frac{\partial P}{\partial x} \Big|_{i,j} \Delta x + \frac{\partial P}{\partial y} \Big|_{i,j} \Delta y \right] \quad (3.14)$$

$$P_{i,j} = \frac{1}{4} [(P_1 + \Delta P_1) + (P_2 + \Delta P_2) + (P_3 + \Delta P_3) + (P_4 + \Delta P_4)] \quad (3.15)$$

In this case Δx and Δy are spatial resolutions in x and y directions respectively. Since the meshspace is equidistant in this study Δx and Δy are equal.

After the integration of the pressure gradients the forces acting on the airfoil can be computed as it was given in Equation 3.2

3.2 Numerical Considerations

As stated before, the force estimation procedure requires the calculation of the derivatives in spatial and temporal directions. These are done by truncating Taylor Series. In current study, the affects of the order of numerical derivation is investigated with different cases. There are two group of force prediction implementation. One of them is higher order (HO) case and the other is the lower order (LO) case. The alteration of numerical order is also reflected to the Lagrangian calculation, so in higher order case the Lagrangian approach is performed with more points that are tracked in time. In both cases there are two alternative sets of derivations in case of lack of grid points both in time and space.

The lower order formulae, forward, backward and central, are given as follows:

$$f'_i = \frac{f_{i+1} - f_i}{h} \quad (3.16)$$

$$f'_i = \frac{f_i - f_{i-1}}{h} \quad (3.17)$$

$$f'_i = \frac{f_{i+1} - f_{i-1}}{2h} \quad (3.18)$$

The higher order formulae, forward, backward and central are given as follows:

$$f'_i = \frac{-f_{i+2} + 2f_{i+1} - 3f_i}{2h} \quad (3.19)$$

$$f'_i = \frac{f_{i-2} - 2f_{i-1} + 3f_i}{2h} \quad (3.20)$$

$$f'_i = \frac{-f_{i-3} + 9f_{i-2} - 45f_{i-1} + 45f_{i+1} - 9f_{i+2} + f_{i+3}}{60h} \quad (3.21)$$

If there is available data for f_{i-3} and f_{i+3} the Equation 3.21 is used, otherwise the order is dropped by one and Equation 3.22 is used:

$$f'_i = \frac{f_{i-2} - 8f_{i-1} + 8f_{i+1} - f_{i+2}}{12h} \quad (3.22)$$

For the HO case, the Lagrangian procedure is carried out for more further points in time:

$$\mathbf{x}_{t+\Delta t} = \mathbf{x}_t + \dot{\mathbf{x}}_t \Delta t + \frac{1}{2} \ddot{\mathbf{x}}_t \Delta t^2 + O(\Delta t^3) \quad (3.23)$$

$$\mathbf{x}_{t+2\Delta t} = \mathbf{x}_t + \dot{\mathbf{x}}_t 2\Delta t + 2\ddot{\mathbf{x}}_t \Delta t^2 + O(\Delta t^3) \quad (3.24)$$

$$\mathbf{x}_{t+3\Delta t} = \mathbf{x}_t + \dot{\mathbf{x}}_t 3\Delta t + 4\ddot{\mathbf{x}}_t \Delta t^2 + O(\Delta t^3) \quad (3.25)$$

$$\mathbf{x}_{t-\Delta t} = \mathbf{x}_t - \dot{\mathbf{x}}_t \Delta t + \frac{1}{2} \ddot{\mathbf{x}}_t \Delta t^2 - O(\Delta t^3) \quad (3.26)$$

$$\mathbf{x}_{t-2\Delta t} = \mathbf{x}_t - \dot{\mathbf{x}}_t 2\Delta t + 2\ddot{\mathbf{x}}_t \Delta t^2 - O(\Delta t^3) \quad (3.27)$$

$$\mathbf{x}_{t-3\Delta t} = \mathbf{x}_t - \dot{\mathbf{x}}_t 3\Delta t + 4\ddot{\mathbf{x}}_t \Delta t^2 - O(\Delta t^3) \quad (3.28)$$

For the LO case, the Lagrangian trackings are done as it is given in Equation 3.11 and Equation 3.12.

3.3 Details of the Code

In this section, the details about the algorithm are presented. The whole script can be found in Appendix A. These points are important to comprehend how the code works:

- In the first part flow, airfoil and experimental parameters must be invoked as user defined inputs, such as Reynolds Number, chord length, sampling frequency and the name of the case.
- After the input section the dataset can be loaded into the procedure from the textfiles. In order to take the data in a user defined MATLAB function that is called "*PIVdata_readin*" is used. This function reads the dataset succesively from the folders that should be choosen by the user.
- "*X*", "*Y*", "*U*", "*V*" and "*Vort*" variables are the grid coordinates, velocities and the vorticity.
- "*airpos*" variable is a matrix that marks the position of the airfoil. The nodes with zero velocities shows the airfoil position. These nodes are flagged by not-a-number (NaN) property of the MATLAB.
- "*dx*" and "*dy*" are mesh spaces in *x* and *y* directions.
- Control volume is selected in the flow domain. "*LoEdge*", "*UpEdge*", "*LeEdge*" and "*RiEdge*" are the lower, upper, leftmost and the rightmost boundaries of the control volume.
- The next section is the implementation of the Eulerian calculation with numerical derivatives.
- "*grads*" is the user defined function for higher order Eulerian calculation.
- "*grads2*" is the user defined function for lower order Eulerian calculation.
- "*EulerAccX*" and "*EulerAccY*" are higher order Eulerian Accelerations in *x* and *y* directions.
- "*EulerAccX2*" and "*EulerAccY2*" are lower order Eulerian Accelerations in *x* and *y* directions.
- After the calculation of the Eulerian acceleration the next step is to calculate the Lagrangian acceleration.
- "*LagrangeAcceleration*" is the user defined function for higher order Lagrangian calculation.

- “*LagrangeAcceleration2*” is the user defined function for lower order Lagrangian calculation.
- “*LagrangeAccX*” and “*LagrangeAccY*” are lower order Lagrangian Accelerations in x and y directions.
- “*LagrangeAccX2*” and “*LagrangeAccY2*” are lower order Lagrangian Accelerations in x and y directions.
- When the material derivative is present pressure gradients are computed easily by summing the material accelerations with the viscous contribution. Then the pressure gradients are integrated by the user-defined function called “*PressGradInteg*”.
- “*UnsteadyTermX*” and “*UnsteadyTermY*” are unsteady terms in the Equation 3.2 in x and y directions.
- “*FluxTermX*” and “*FluxTermY*” are flux terms in the Equation 3.2 in x and y directions.
- “*PresTermX*” and “*PresTermY*” are pressure terms in the Equation 3.2 in x and y directions.
- Calculated forces are non-dimensionalized by $\frac{1}{2}cU_{\infty}^2$, because the calculated pressure was already divided by density.
- There are four different sets of aerodynamic coefficients calculated.
 - “*cL*” and “*cD*” are obtained by surface unsteady term integration and higher order calculation.
 - “*cL2*” and “*cD2*” are obtained by line unsteady term integration and higher order calculation.
 - “*cL3*” and “*cD3*” are obtained by surface unsteady term integration and lower order calculation.
 - “*cL4*” and “*cD4*” are obtained by line unsteady term integration and lower order calculation.

4 DETAILS OF THE EXPERIMENTAL DATA UNDERTAKEN

4.1 General Information About the Flow and Motion System

There are two sets of experimental cases to be investigated in the current study both of which are conducted at the free surface water channel in Trisonic Laboratory of Istanbul Technical University, Faculty of Aeronautics and Astronautics. The dimensions of the test section of the channel are 1010 mm (width) and 790 mm (height). The facility can provide a uniform freestream with low turbulent intensity within a range of 5 mm/s to 14 mm/s.

A NACA0012 and a flat plate model are used and manufactured from PlexiGlass material. They are mounted in a cantilevered arrangement inside the water channel between two end plates to reduce the free surface and end effects. The mounting beam is connected to a pitch motor which itself is connected to a linear table which allows for the plunging motion. The models have a chord (c) of 10cm and span (s) of 30cm. The experimental setup is shown in Figure 4.1.

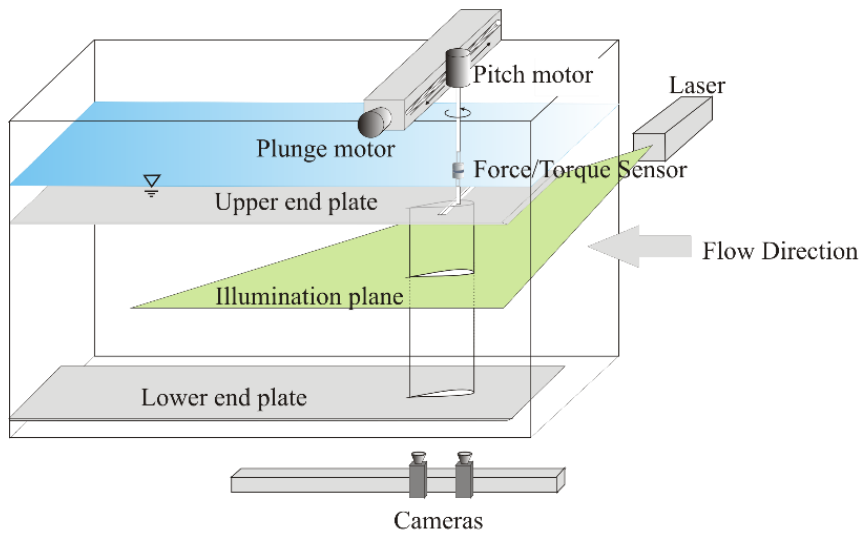


Figure 4.1: Experimental setup

The flow is illuminated by a dual cavity Nd:Yag laser (max. 120mJ/pulse) and the water is seeded with silver coated hollow glass spheres with a mean diameter of 10 μm . The velocity fields around and in the near wake of the models are obtained using two 10-bit cameras with 1600×1200 pixels resolution, positioned underneath the water channel. Recorded images are stitched using an in-house code and then

interrogated using a double frame, cross-correlation technique with a window size of 64×64 pixels and 50% overlapping in each direction.

Force and moments acting on the plunging flat plate are measured using a six-component ATI NANO-17 IP68 Force/Torque (F/T) sensor. The sensor is attached to the vertical cantilevered mounting beam of the test model, oriented with its cylindrical z-axis normal to the pitch-plunge plane. The plunge motion of the models is accomplished with Kollmorgen/Danaher Motion AKM54K servo motor which was connected to a computer via ServoSTAR S700 digital servo amplifier. Motor motion profiles are generated by a signal generator Labview VI (Virtual Instrument) for the given amplitude and frequency. The same VI triggered the PIV system at the beginning of the third motion cycle of the airfoil and synchronization is achieved using a National Instruments PCI-6601 timer device. Another Labview VI is used simultaneously to acquire force/moment data with a sampling frequency of 1000 Hz. The sinusoidal plunging motion of the airfoil is given as;

$$h(t) = h_{amp} \cos(2\pi ft)$$

where $h(t)$ is the linear plunge motion, transverse to the freestream velocity, h_{amp} is the plunge amplitude, f is the plunging frequency.

The reduced frequency (k) and the Strouhal number (St) are defined as:

$$k = \frac{\pi f c}{U_{\infty}}$$

$$St = \frac{2f h_{amp}}{U_{\infty}}$$

where U_{∞} is the freestream velocity.

4.2 Sensitivity Analysis Cases

These experiments are conducted within the context of the FP7 Project 605151 - “Non-Intrusive Optical Pressure and Loads Extraction for Aerodynamic Analysis (NIOPLEX)” to investigate the effects of temporal and spatial resolution for pressure and force estimation from PIV measurements. In these cases plunge motion of a NACA0012 is investigated with different spatial and temporal resolutions.

The parameters of the sinusoidal plunge motion are as follows:

$$h_{\text{amp}} = 0.25c$$

$$f = 0.2\text{Hz}$$

The reduced frequencies, St and Re numbers are:

$$k = 1.257, St = 0.1 (Re = 10000)$$

$$k = 2.513, St = 0.2 (Re = 5000)$$

In terms of temporal resolution, the PIV sampling rate is chosen to be either 8 Hz or 15 Hz.

The motion of the airfoil (airfoil represented in a 50% scale) and the instants the velocity fields are acquired are shown in Figure 4.2 for the PIV sampling frequency of 8Hz.

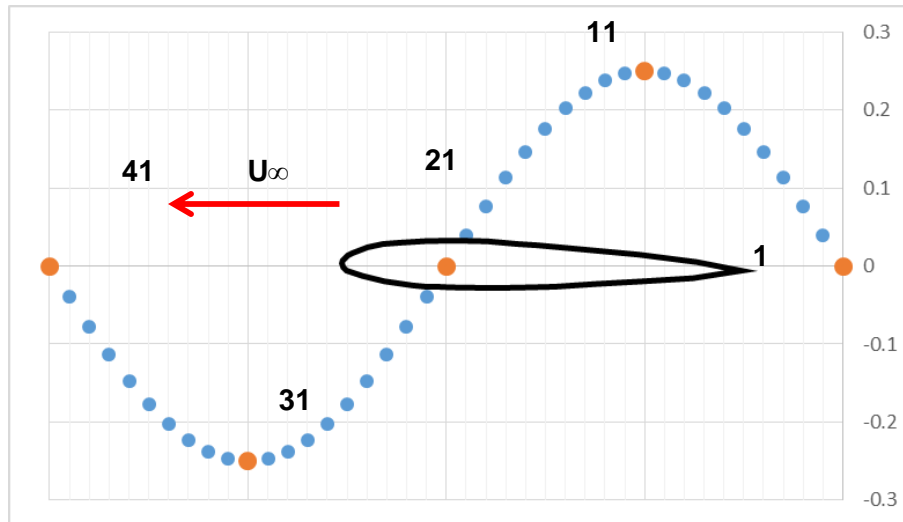


Figure 4.2: The motion of the airfoil and the instants the velocity fields are acquired

In order to accomplish the sensitivity analysis, the experiments are performed in two stages. The details are given as follows:

The flow field around the airfoil can be captured with a single CCD camera using a 60mm lens. The raw PIV image given in Figure 4.3 is obtained when the airfoil is at the maximum of its plunge motion (Position numbered as 11 in Figure 4.2).

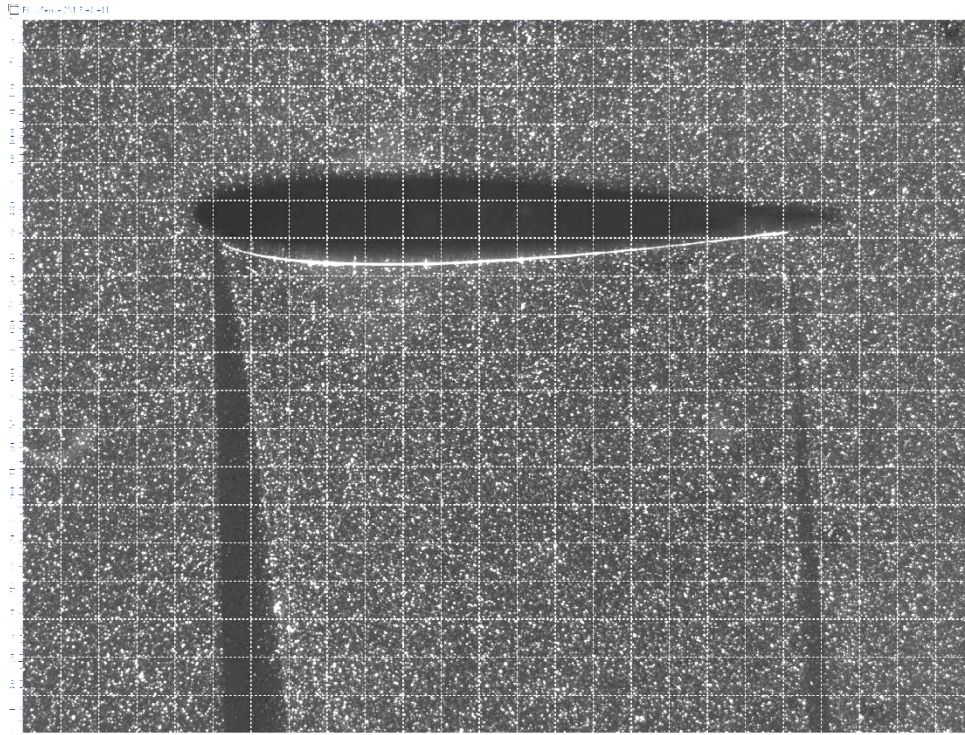


Figure 4.3: The raw PIV image obtained when the airfoil is at the maximum of its plunge motion (low spatial resolution)

With the use of a 60mm lens, the field of view covers all the plunge motion of the airfoil and includes 3-5 grid points below and above for the extreme positions of the plunge motion. The resulting vector field resolution is $3.4 \text{ mm} \times 3.4 \text{ mm}$. At this spatial resolution the PIV data is acquired at two different sampling frequency, i.e, 8Hz ($\Delta t=0.125\text{s}$) and 15Hz ($\Delta t=0.067\text{s}$). The lower one allows capturing the flow at interested position of the sinusoidal motion, the larger one is the maximum frequency that the PIV system allows. For 8 Hz sampling frequency, 200 velocity fields are acquired for 5 periods of motion and for 15 Hz sampling frequency 225 velocity fields are acquired for 3 periods of motion.

In order to increase the spatial resolution, two 10-bit cameras with 1600×1200 pixels resolution are positioned underneath the water channel and two 105mm lenses are used. Two images from the two cameras are stitched before interrogation using two marker points in the illumination plane. The flow field is focused on the upper portion of the plunge motion as seen in Figure 4.4 where a raw PIV image is given for the maximum of the airfoil's plunge motion (Position numbered as 11 in Figure 4.2). The final grid resolution of velocity vectors is $1.6 \text{ mm} \times 1.6 \text{ mm}$ in the plane of the flow.

In this setup, in order to observe the effect of temporal resolution, first, a sampling frequency of 8Hz is used and 200 velocity fields are acquired for 5 periods of motion. Then with a same sampling frequency of 4Hz, running 20 sets of experiments, the PIV synchronization is shifted using $t = 2T + n \times 0.0125s$ ($n=1,2, \dots, 20$) and in total 4000 velocity fields are acquired with a temporal resolution of $\Delta t=0.0125s$ (corresponding to a sampling frequency of 80Hz).

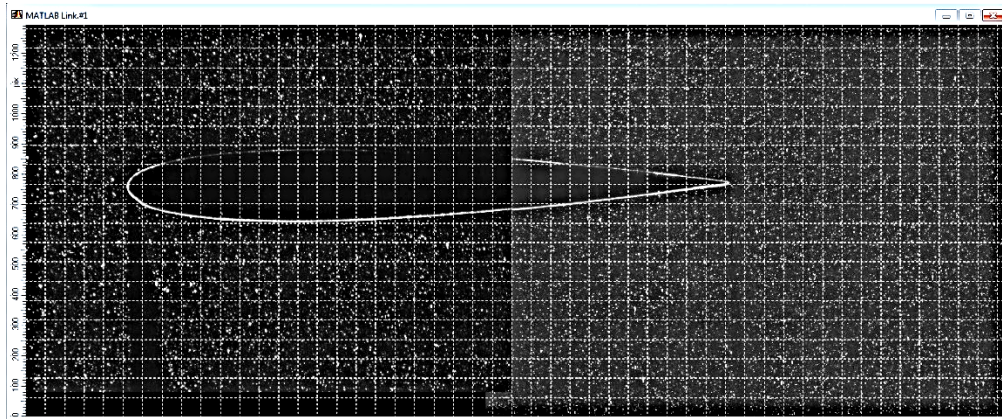


Figure 4.4: The raw PIV image obtained when the airfoil is at the maximum of its plunge motion (high spatial resolution)

4.3 Different Plunge Motion Cases

In this set of experiments, a flat plate is used in order to determine the value of the added mass contribution in the total force measurements to reveal the net force acting on a flapping foil.

The occurrence of Drag and Thrust producing wake for a plunging airfoil as a function of the plunging amplitude and the reduced frequency is re-illustrated in Figure 4.5 based on the data by Jones et al. (1996). The test case points for the present study are also marked with triangular symbols on the same plot.

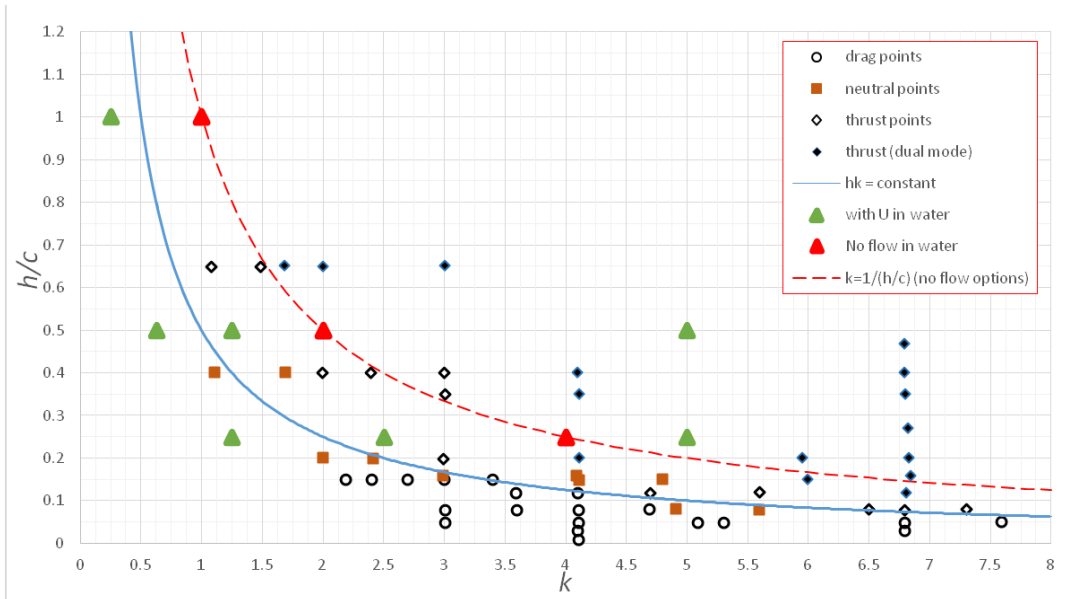


Figure 4.5: Investigated test cases

The investigated points as given in the plot include both drag cases and thrust cases placed on both sides of the dividing hk constant line as well as cases of interest to the EU-FP7 project NIOPLEX. The plunge-only equivalent cases of the initially pitching and plunging airfoil as tested in Fenercioglu and Cetiner [2012] were also included. The tests were also repeated without the free-stream velocity, those special cases lie $k = 1/(h/c)$ constant line. The test case parameters are given in Table 4.1.

Table 4.1: Experimental Parameters for Added Mass Case

Case #	Re	f [Hz]	k	h_{amp}/c	St
1	1250	0.1	5.01	0.50	0.8
2	2500	0.2	5.01	0.25	0.4
3	3125	0.05	1.00	1.00	0.3
4	5000	0.1	1.25	0.50	0.2
5	5000	0.2	2.50	0.25	0.2
6	10000	0.1	0.63	0.50	0.1
7	10000	0.2	1.25	0.25	0.1
8	12500	0.05	0.25	1.00	0.1
9	3129	0.05	1.00	1.00	0.2
10	3129	0.1	2.00	0.50	0.3
11	3129	0.2	4.00	0.25	0.3

For each case a total of 1200 velocity fields have been acquired during 30 cycles of plunge motion. Therefore, for each cycle of motion, 40 velocity fields are used to estimate the loads.

5 RESULTS OF SENSITIVITY ANALYSIS

5.1 Selection of Run Variations

The developed code has been tested in two cases to identify the effect of using high order terms in the integration and using surface or line integration for the unsteady term. Figures 5.1, 5.2 and 5.3 show the difference in using high or low order terms in the integration for the cases where surface integration is adopted for unsteady term. Figure 5.1 includes the results obtained from instantaneous PIV data acquired for $Re=5000$ and with a sampling frequency of 8 Hz. Although the estimation results are highly noisy, the periodicity is well captured. Five period estimation results are then phase averaged and represented in Figure 5.2. It is seen that some of the noise characteristics are repeated periodically. The data fits better to the measurement results when the later is filtered using a 3 point moving average (see Figure 5.3).

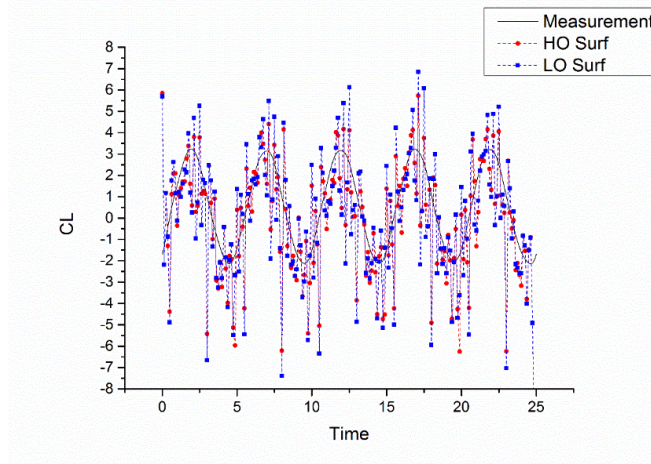


Figure 5.1: H.O. vs. L.O. Unst. Surf. Instantaneous Results for Re 5000 freq 8 Hz

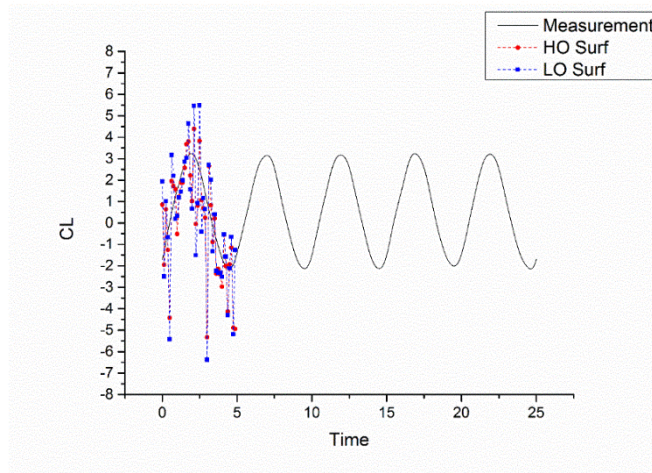


Figure 5.2: H.O. vs. L.O. Unst. Surf. Phase Averaged Results for Re 5000 freq 8 Hz

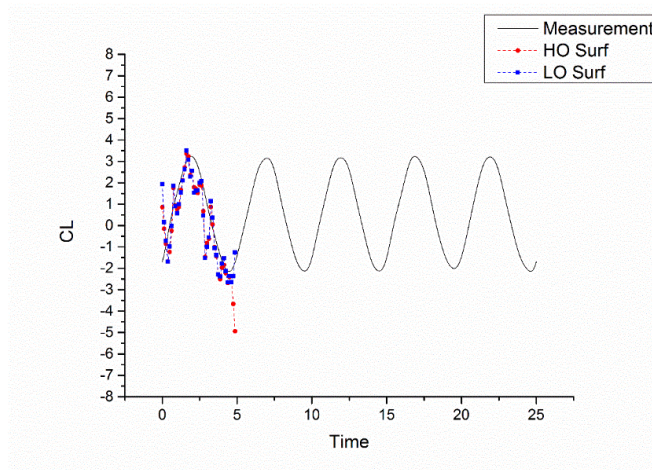


Figure 5.3: H.O. vs. L.O. Unst. Surf. Moving Ave. Results for Re 5000 freq 8 Hz

The same comparison procedure has been done for line integration of the unsteady term. Figures 5.4, 5.5 and 5.6 show the difference in using high or low order terms in the integration for the cases where line integration is adopted for unsteady term.

Although the phase averaged and filtered results do not clearly show the difference between the use of high or low order terms in the integration, the fluctuations in the instantaneous load estimation is higher when low order terms are used. Therefore for the analyses of different plunge motions, high order terms will be used in the integration.

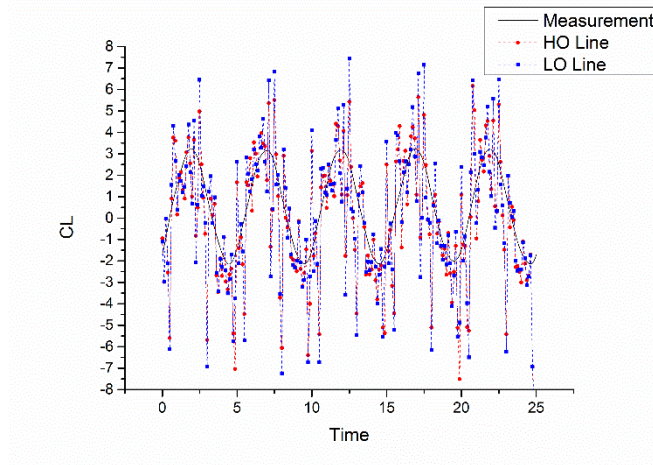


Figure 5.4: H.O. vs. L.O. Unst Line Instantaneous Results for Re 5000 freq 8 Hz

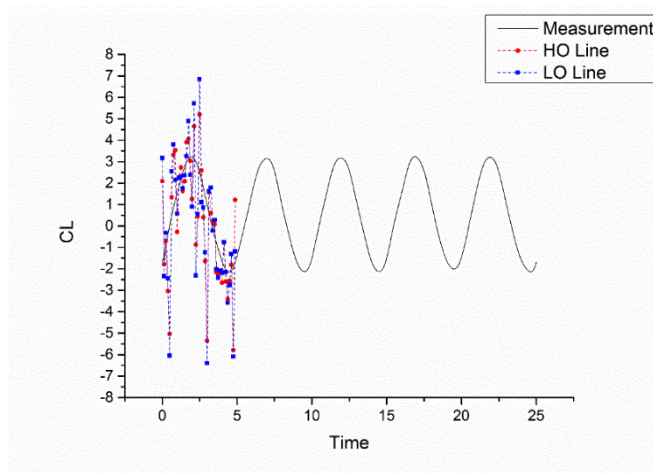


Figure 5.5: H.O. vs. L.O. Unst Line Phase Ave. Results for Re 5000 freq 8 Hz

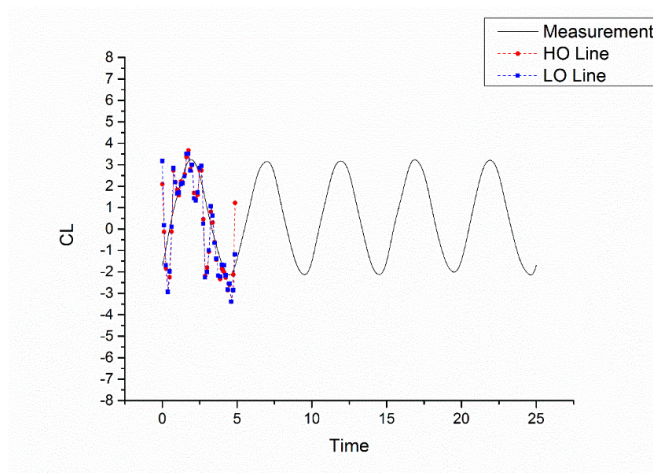


Figure 5.6: H.O. vs. L.O. Unst Line Moving Ave. Results for Re 5000 freq 8 Hz

The effect of using either surface or line integration for the unsteady term has been visualized in the graphics below. Figures 5.7, 5.8 and 5.9 have higher order integration schemes. Whereas Figures 5.10, 5.11 and 5.12 have lower order integration schemes. Figures 5.7 and 5.10 present the instantaneous estimation results, Figures 5.8 and 5.11 show the phase average results using 5 periods of information and finally, Figures 5.9 and 5.12 are for the filtered results.

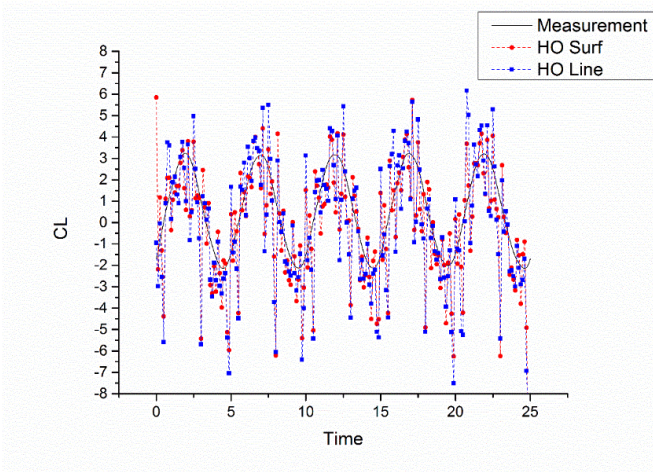


Figure 5.7: Unst Line vs. Unst Surf H.O. Inst. Results for Re 5000 freq 8 Hz

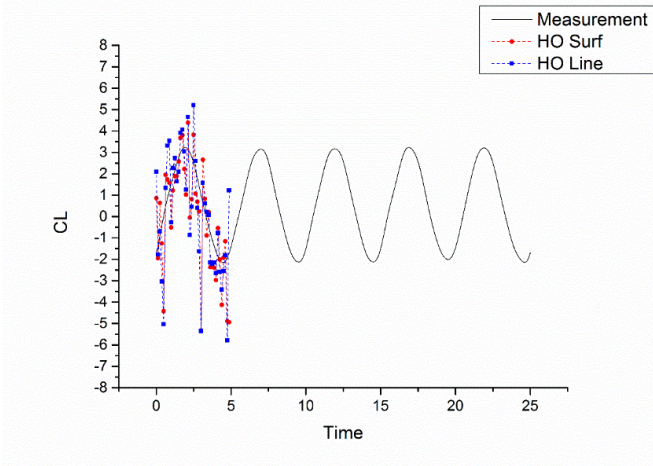


Figure 5.8: Unst Line vs. Unst Surf H.O. Phase Ave. Results for Re 5000 freq 8 Hz

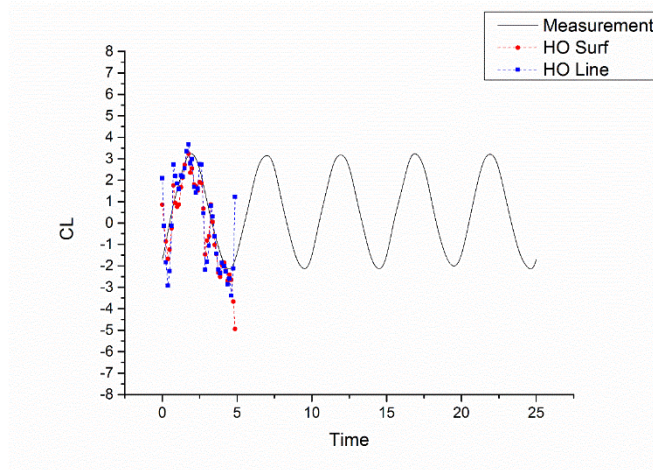


Figure 5.9: Unst Line vs. Unst Surf H.O. Moving Ave Results for Re 5000 freq 8 Hz

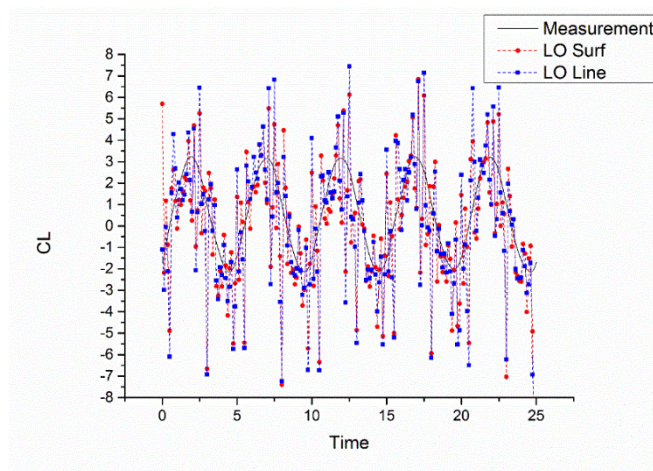


Figure 5.10: Unst Line vs. Unst Surf L.O. Inst. Results for Re 5000 freq 8 Hz

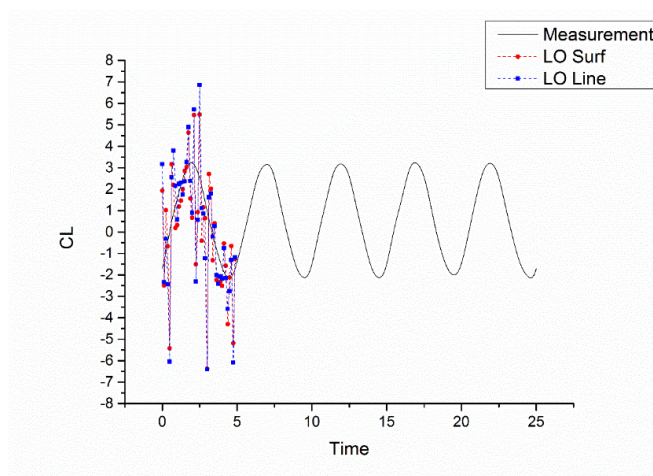


Figure 5.11: Unst Line vs. Unst Surf L.O. Phase Ave. Results for Re 5000 freq 8 Hz

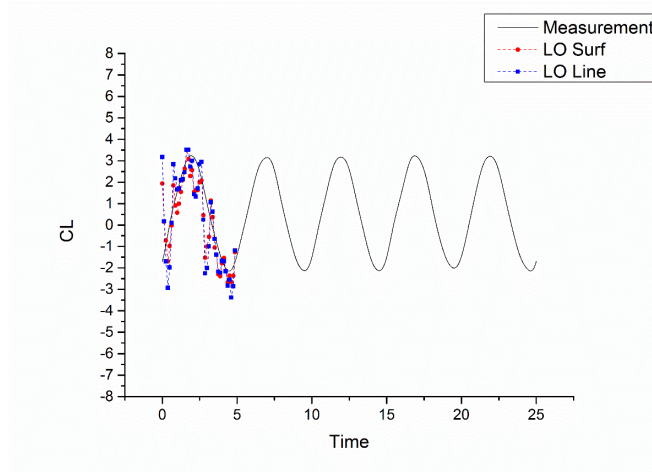


Figure 5.12: Unst Line vs Unst Surf LO Moving Ave Results for Re 5000 freq 8 Hz

As can be seen in Figure 5.7 to Figure 5.12 the surface integration for unsteady term results better compared to the line integration. Although not presented here, it is also seen that the line integration is more sensitive to control volume selection.

Thus, higher order integration and surface integration for unsteady term is selected for force prediction.

5.2 Spatial and Temporal Resolution

In order to determine the spatial and temporal resolution to be used in DPIV measurements, two cases defined in §4.2 have been studied for different DPIV image acquisition/post-processing settings given in Table 5.1.

The last column in Table 5.1 indicates cases where an average correlation algorithm is used when processing DPIV data. Average correlation is mainly used to reduce the noise in the correlation, the cross-correlation planes belonging to the same instant in a periodic motion are summed up to obtain a single velocity field instead of obtaining the instantaneous velocity fields and averaging those.

Table 5.1: Sensitivity Analysis Cases Work

Freq, f	Spat. Res.	Instantaneous	Ave Corr.
8 Hz [200 Imgs 5 Periods]	3.32 mm × 3.32 mm	200/200	X
15 Hz [225 Imgs 3 Periods]	3.32 mm × 3.32 mm	225/225	X
80 Hz [4000 Imgs 10 Periods]	1.60 mm × 1.60 mm	400/4000	400/400

In Table 5.1 the **green** cells represents that the force prediction procedure is carried out for both $Re=5000$ and $Re=10000$. The **red** cells shows that the work is done for only $Re=5000$. The **black** cells represent that no work is available for those configurations.

In this case, the vector field averaging have been also tested to enhance the performance of the force prediction. It is compared with the phase averaging over the calculated force coefficients and it is seen that there is not a considerable difference between averaging the predicted coefficients and vector field averaged calculations, therefore the vector field averaging cases are not presented herein.

Figures 5.13 to 5.20 show the results for 8Hz and 15Hz sampling frequencies for two cases, $Re=5000$ and $Re=10000$ respectively. The results are shown as both instantaneous load estimations and phase averaged load estimations on available periods of motion.

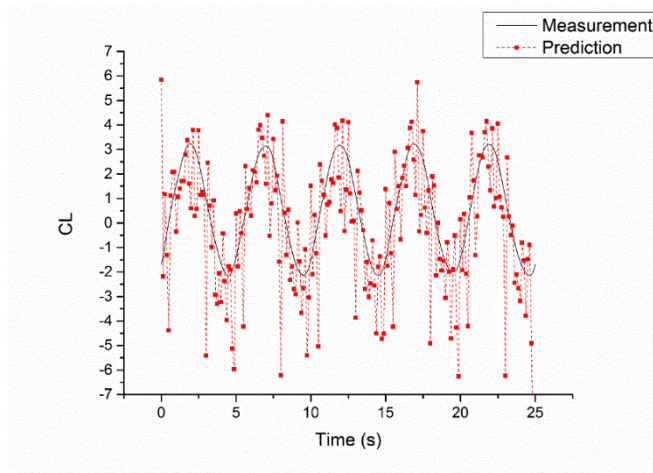


Figure 5.13: Re 5000 freq 8 Hz Instantaneous

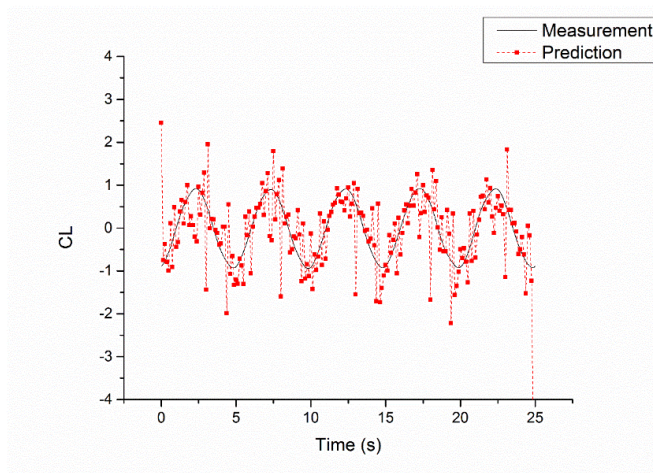


Figure 5.14: Re 10000 freq 8 Hz Instantaneous

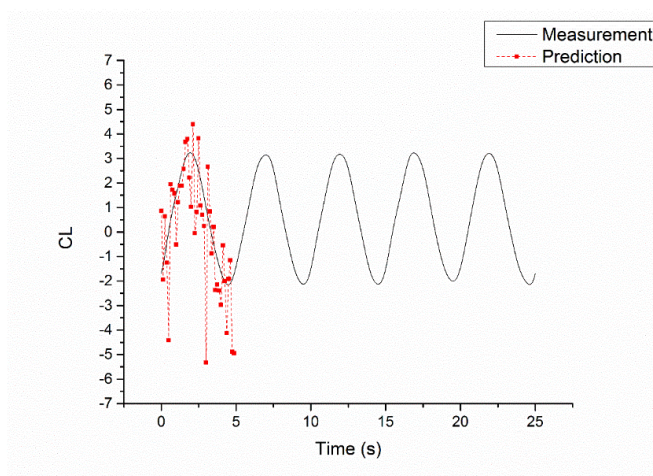


Figure 5.15: Re 5000 freq 8 Hz Period Averaged

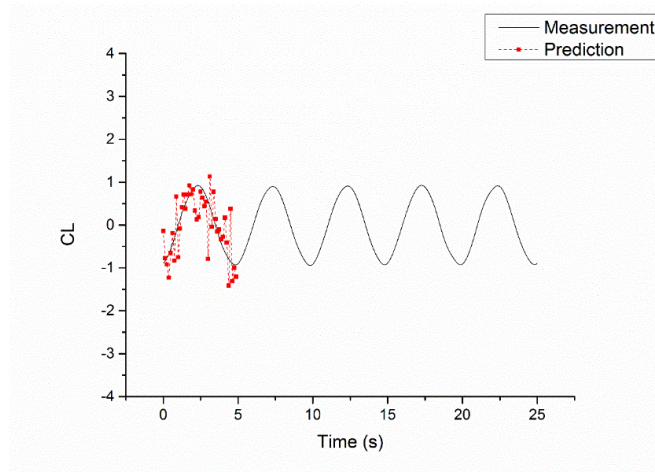


Figure 5.16: Re 10000 freq 8 Hz Period Averaged

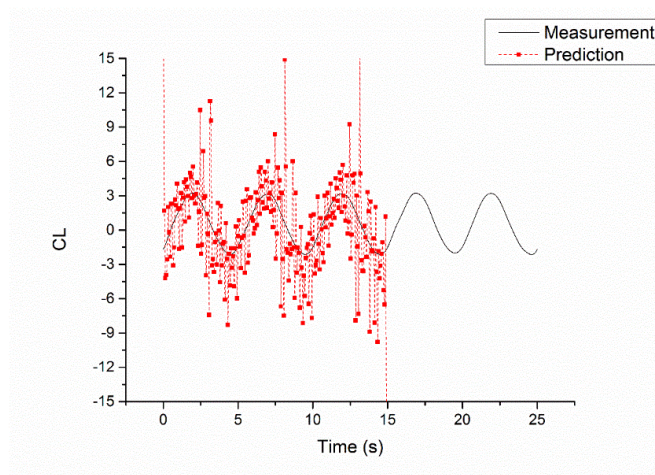


Figure 5.17: Re 5000 freq 15 Hz Instantaneous

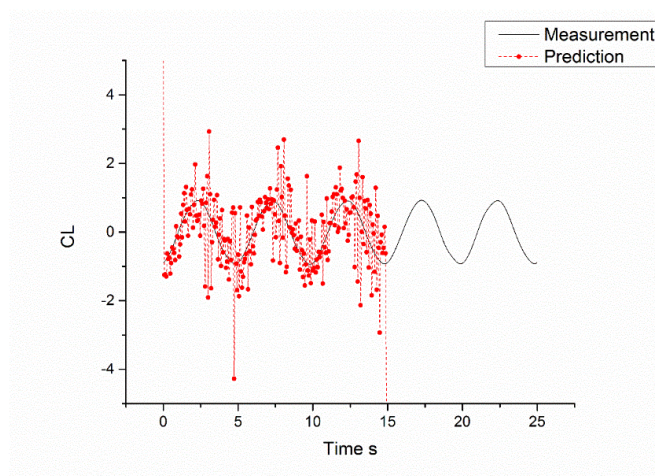


Figure 5.18: Re 10000 freq 15 Hz Instantaneous

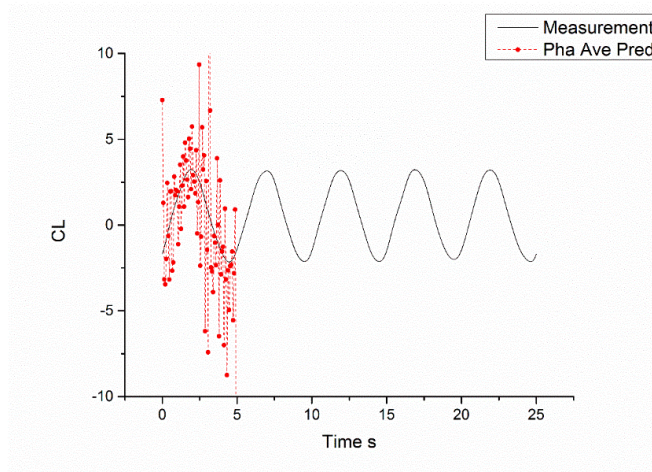


Figure 5.19: Re 5000 freq 15 Hz Period Averaged

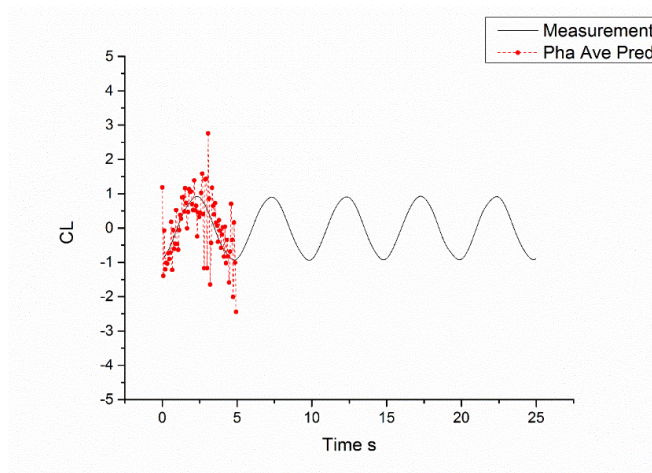


Figure 5.20: Re 10000 freq 15 Hz Period Averaged

In all cases, for instantaneous load estimations, there is a remarkable discrepancy at the beginning and at the end of the process. This is due to the lack of temporal information at the following or preceding time steps. It is clear that phase averaging decreases the fluctuation level for both cases regardless of the sampling frequency. A sampling frequency of 8 Hz yields better results compared to that of 15 Hz. A much higher sampling frequency results are shown in Figure 5.21 and Figure 5.22. As the frequency of motion is low, higher sampling frequencies introduces more fluctuations/noise.

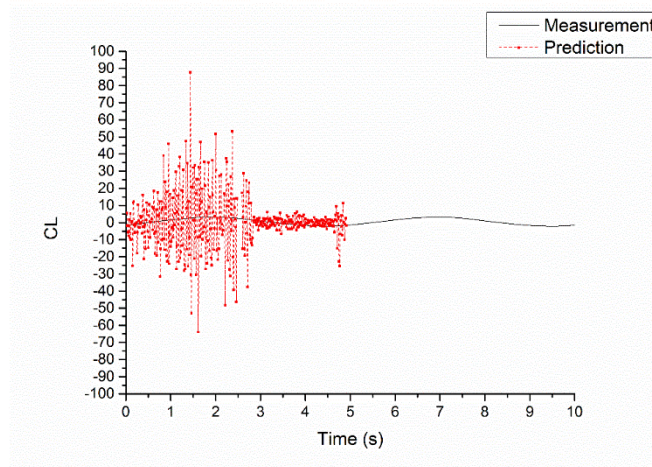


Figure 5.21: Re 5000 freq 80 Hz Instantaneous

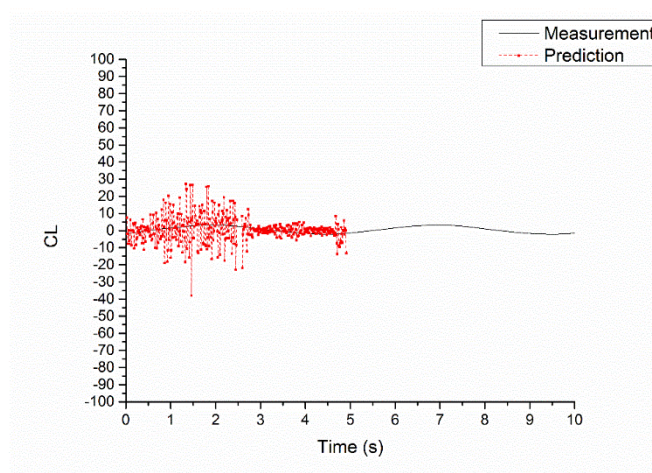


Figure 5.22: Re 5000 freq 80 Hz Ave Correlation

Figure 5.22 shows the load estimation results when average correlation is used in the post-processing of DPIV data. It is clear that average correlation decreases the fluctuations considerably. In any case, a sampling frequency of 80 Hz does not yield an acceptable force prediction results compared to the sampling frequency of 8 Hz and 15 Hz. Therefore, increasing temporal and spatial resolutions dramatically do not necessarily give better performance. A spatial resolution of 3% of the chord and a field of view covering all the amplitude of motion give a satisfactory results with a temporal resolution expressed as 40 vector fields per cycle of motion.

6. RESULTS FOR DIFFERENT PLUNGE CASES

Load estimation code is used to generate lift and drag coefficient variation for the general plunge cases which were described in §4.3. CL and CD are given for 11 different cases. Instantaneous results are given from Figure 6.1 through Figure 6.22.

Case-1:

The kinematics of this case is adopted from Fenercioğlu (2010) and is the same as A1 flow categorization, except for the existence of pitch motion that is found to be insignificant. The average velocity and vorticity fields, not presented herein, indicate jet like flow, which is indicative of thrust, in accordance with the highest thrust obtained in this case. However, the value of the lift coefficient is misleading since the fluctuation amplitude around zero is very high due to the high acceleration, therefore added-mass effect. The free-stream velocity is also the lowest one in all cases, which amplifies the measurement error associated. Although the estimated lift periodicity is in accordance with the measurement, the apparent amplitude is slightly lower than the measured. Estimated drag has a very large amplitude and even the average is not meaningful as it indicates drag instead of thrust.

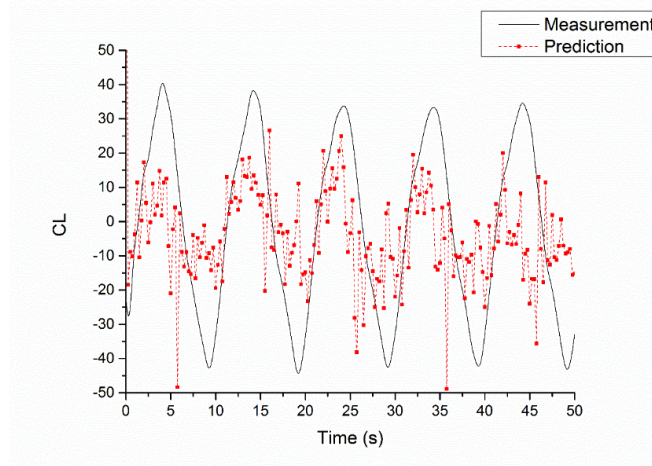


Figure 6.1: CL for Case 1 [$k=5.01$ $Re=1250$ $f=0.1$ $h_{amp}/c=0.50$]

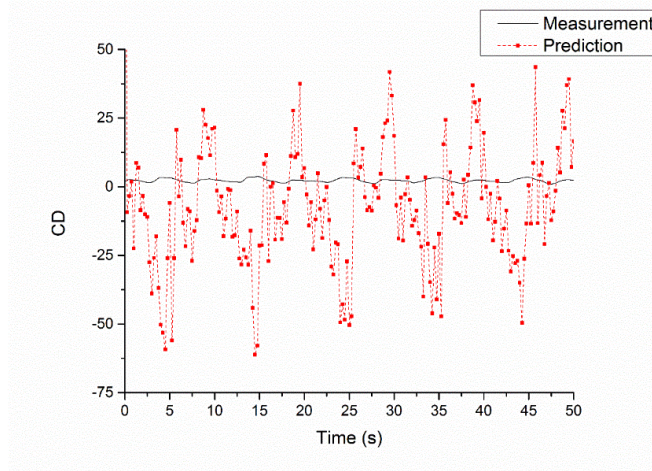


Figure 6.2: CD for Case 1 [$k=5.01$ $Re=1250$ $f=0.1$ $h_{amp}/c=0.50$]

Case-2:

The kinematics of this case is adopted from Fenercioğlu (2010) and is the same as B2 flow categorization, except for the existence of pitch motion that is found to be insignificant. The average velocity and vorticity fields, not presented herein, indicate jet like flow, which is indicative of thrust, the jet width is narrower in accordance with the smaller plunge amplitude. The thrust is also less than that of the Case-1. Similar to the results obtained for Case-1, although the estimated lift periodicity is in accordance with the measurement, the apparent amplitude is slightly lower than the measured. Estimated drag has again a very large amplitude and even the average is not meaningful as it indicates drag instead of thrust.

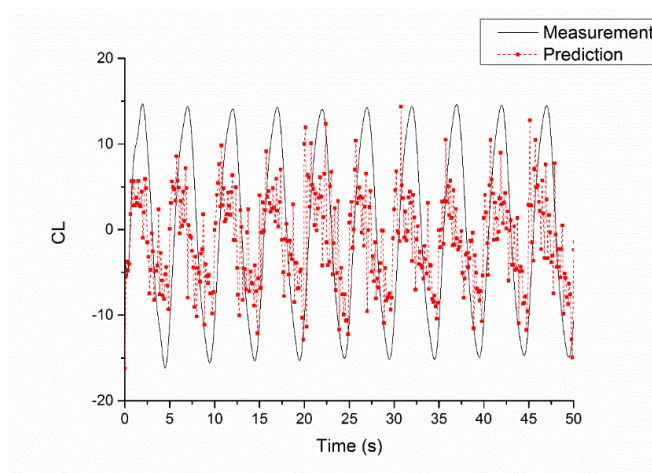


Figure 6.3: CL for Case 2 [$k=5.01$ $Re=2500$ $f=0.2$ $h_{amp}/c=0.25$]

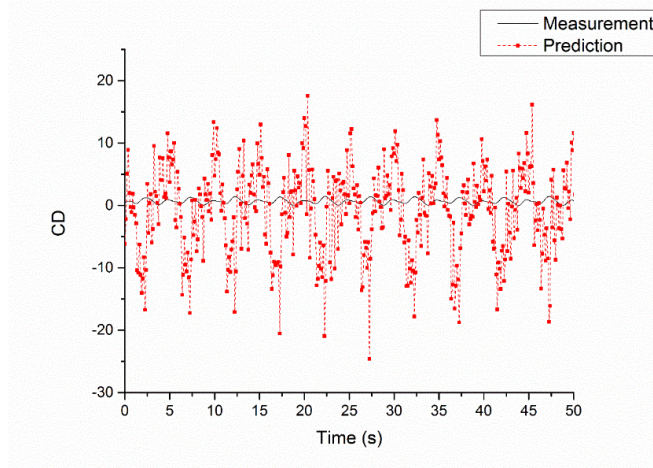


Figure 6.4: CD for Case 2 [$k=5.01$ $Re=2500$ $f=0.2$ $h_{amp}/c=0.25$]

Case-3:

The kinematics of this case is selected to be in thrust production side of the $kh=const.$ line for the lowest plunge frequency investigated. The case is significant as it has an equivalent no flow counterpart. The average velocity and vorticity fields, not presented herein, indicate a less intense jet like flow, however the jet width is wider compared to that of the Case-2. As a result, this plunge motion yields a similar thrust coefficient as in the Case-2. Although there are still some fluctuations, the estimated lift variation is very successful, the amplitude is also in agreement with the measurement. slightly lower than the measured. However, there is not an improvement in the drag estimation, it has a very large amplitude.

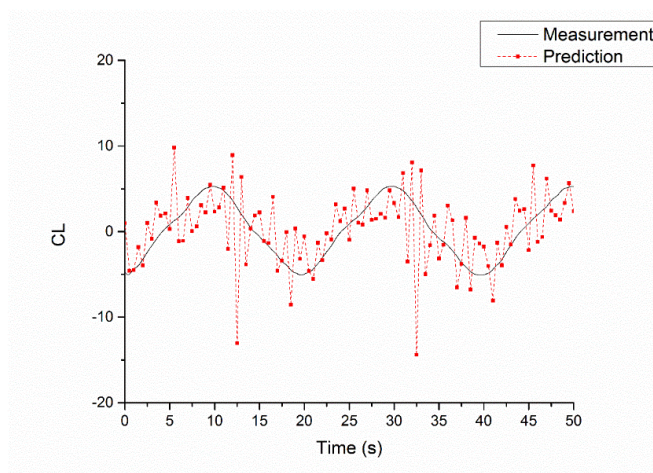


Figure 6.5: CL for Case 3 [$k=1.00$ $Re=3125$ $f=0.05$ $h_{amp}/c=1.00$]

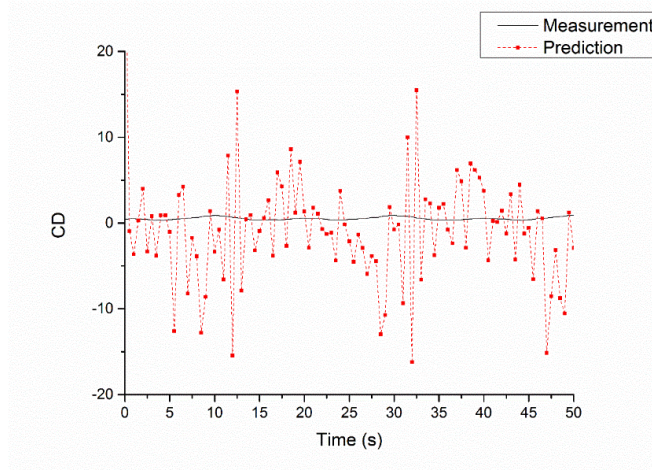


Figure 6.6: CD for Case 3 [$k=1.00$ $Re=3125$ $f=0.05$ $h_{amp}/c=1.00$]

Case-4:

The kinematics of this case is selected to be in thrust production side of the $kh=const.$ line to be able to compare with the case on the other side of the $kh=const$ line, which has the same kinematics of Category E of Fenercioğlu (2010) except for the existence of pitch motion. The averaged velocity field, not presented herein, indicate a jet like flow in the wake. The thrust coefficient is relatively low, in accordance with the expectations as the case is very close to the thrust-drag switch line. Estimated lift coefficient has a comparable amplitude as the measured one, however there is some disagreement in the periodicity which cannot be regarded as fluctuations.

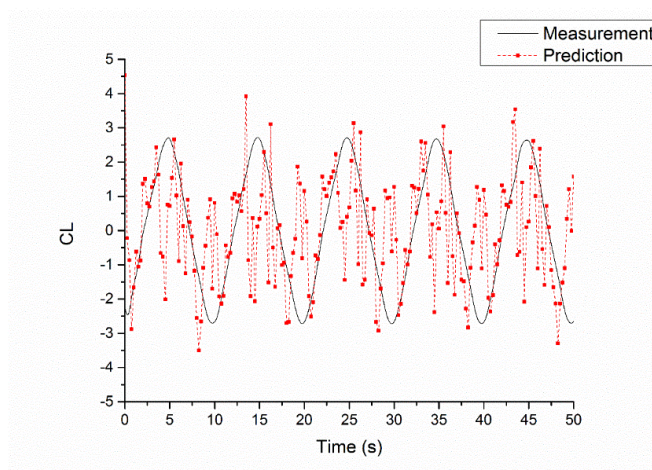


Figure 6.7: CL for Case 4 [$k=1.25$ $Re=5000$ $f=0.10$ $h_{amp}/c=0.50$]

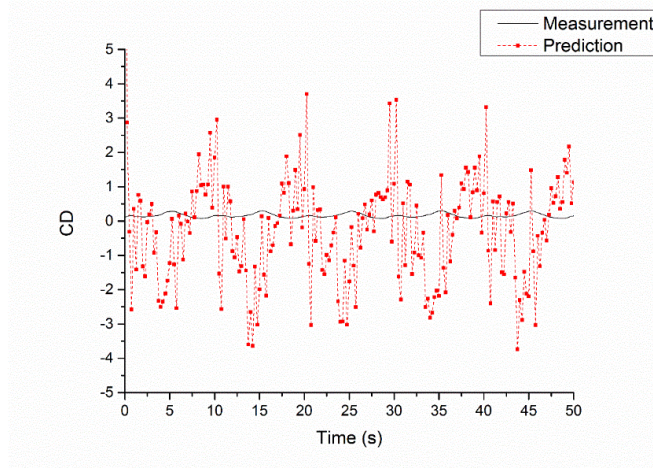


Figure 6.8: CD for Case 4 [$k=1.25$ $Re=5000$ $f=0.10$ $h_{amp}/c=0.50$]

Case-5:

The kinematics of this case is adopted from Fenercioğlu (2010) and is the same as D flow categorization, except for the existence of pitch motion that is found to be insignificant. In line with the wide and less intense jet like velocity formation, not presented herein, the thrust is comparable with that of the Case-4. Estimated lift coefficient totally disagree with the measurement, there is no apparent periodicity. On the other hand, although the estimated drag preserves its disagreement with the measurement as in the previous cases, the average value, by chance, is the closest to the measured among all the cases, in terms of both absolute value and sign.

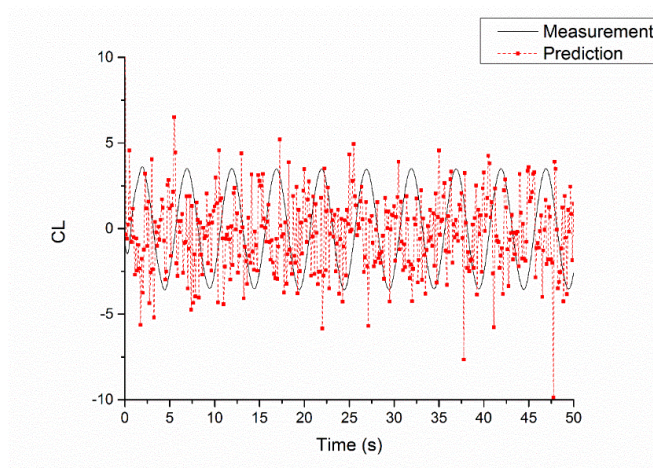


Figure 6.9: CL for Case 5 [$k=2.50$ $Re=5000$ $f=0.20$ $h_{amp}/c=0.25$]

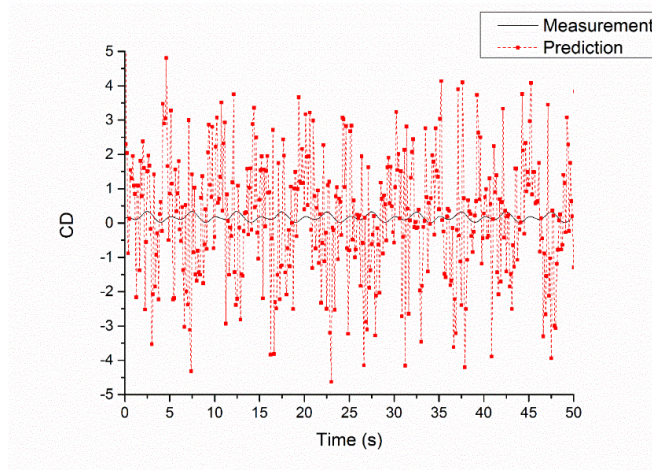


Figure 6.10: CD for Case 5 [$k=2.50$ $Re=5000$ $f=0.20$ $h_{amp}/c=0.25$]

Case-6:

The kinematics of this case is adopted from Fenercioğlu (2010) and is the same as E flow categorization, except for the existence of pitch motion that is found to be insignificant. Although the averaged velocity in the wake, not presented herein, does not show any jet-like formation, the average of the measured force coefficient in the free-stream is positive with a low value. It should be noted that force measurement results involve large errors for small values. The estimated lift coefficient variation is mostly following the periodicity of the measurement results with high fluctuations especially around the peak values. Although the fluctuations in drag estimation are high, for the first time, the apparent amplitude and periodicity with a phase of π are in agreement with the measurement.

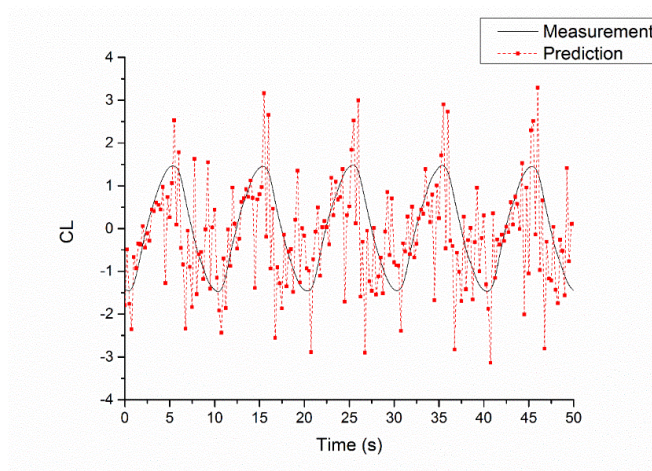


Figure 6.11: CL for Case 6 [$k=0.63$ $Re=10000$ $f=0.10$ $h_{amp}/c=0.50$]

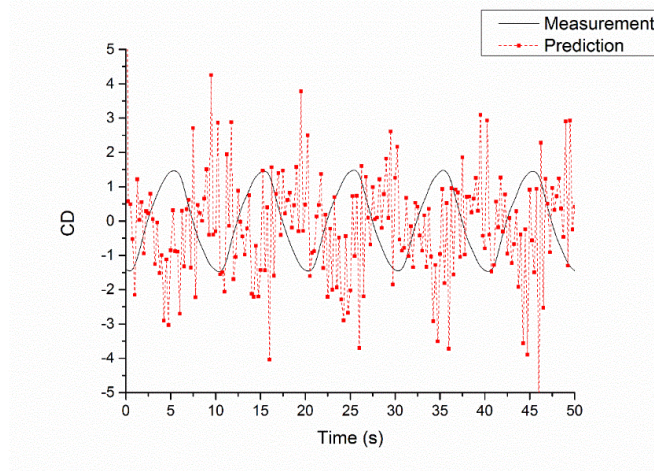


Figure 6.12: CD for Case 6 [$k=0.63$ $Re=10000$ $f=0.10$ $h_{amp}/c=0.50$]

Case-7:

The kinematics of this case is adopted from NIOPLEX, and it is in drag occurrence side of the $kh=\text{const.}$ line to be able to compare with the case on the other side of the $kh=\text{const}$ line, which has the same kinematics of Category D of Fenercioğlu (2010) except for the existence of pitch motion. The averaged velocity field, not presented herein, does not show a jet-like formation in the wake. The force measurement results indicate a comparable value as in the Case-6. Although there are high fluctuations, the estimated lift coefficient variation agrees with the measurement to some degree as the apparent amplitude is similar and the periodicity is present with a phase of π with respect to the measured variation. Although the average values are considered to be close, the estimated drag coefficient variation is not successful as in many cases.

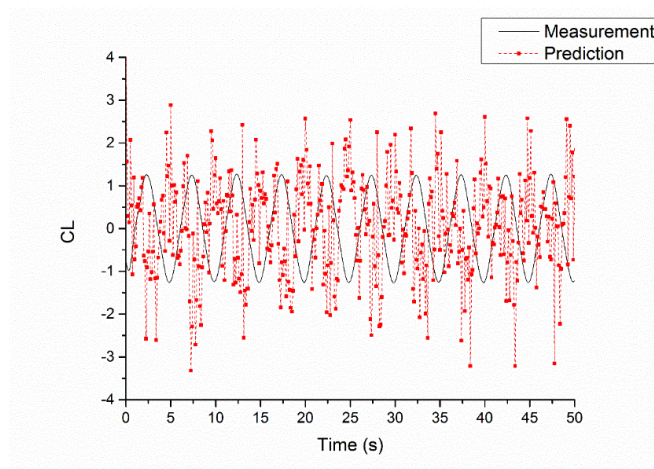


Figure 6.13: CL for Case 7 [$k=1.25$ $Re=10000$ $f=0.20$ $h_{amp}/c=0.25$]

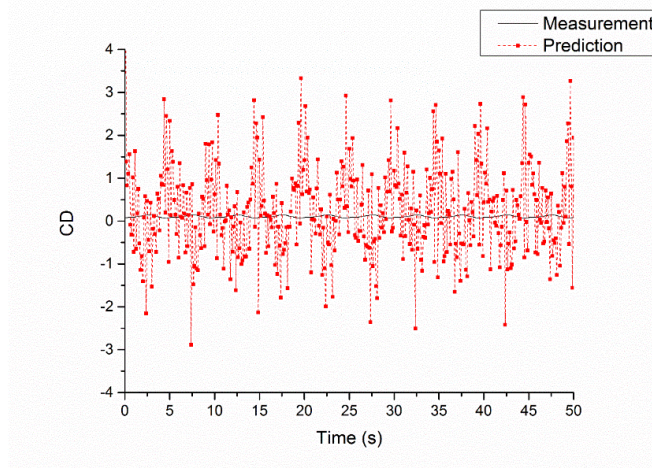


Figure 6.14: CD for Case 7 [$k=1.25$ $Re=10000$ $f=0.20$ $h_{amp}/c=0.25$]

Case-8:

The kinematics of this case is selected to be in drag occurrence side of the $kh=const.$ line for the lowest plunge frequency investigated. The averaged velocity field, not presented herein, does not show a jet-like formation in the wake. The force measurement results indicate a comparable value as in the two previous cases. The estimated lift coefficient variation is considered to be very successful except a couple of instants in a period in terms of both amplitude and periodicity. The estimated variation of drag coefficient still does not match at all the measurement results.

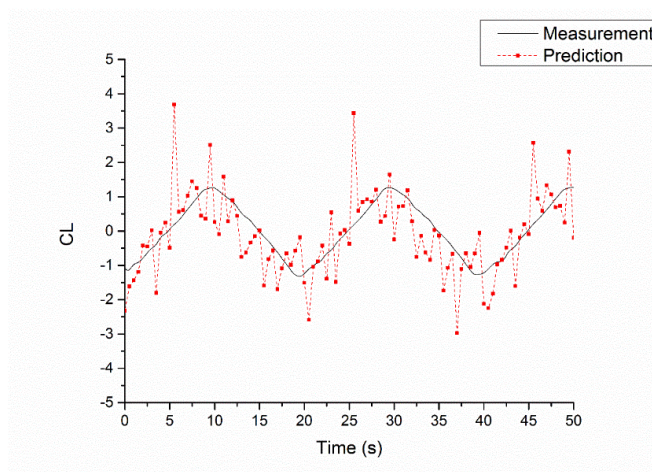


Figure 6.15: CL for Case 8 [$k=1.25$ $Re=12500$ $f=0.05$ $h_{amp}/c=1.00$]

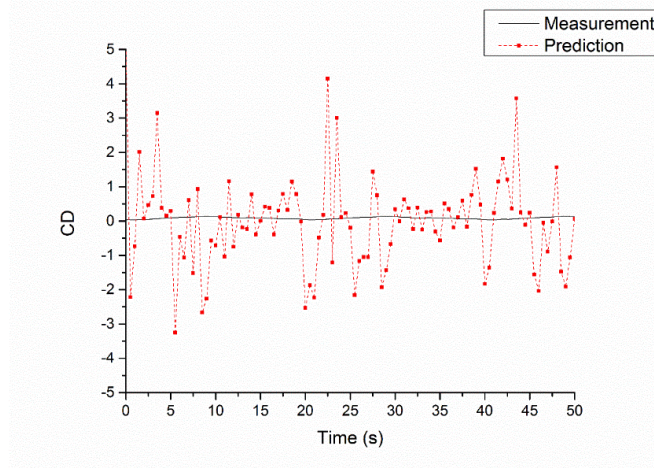


Figure 6.16: CD for Case 8 [$k=1.25$ $Re=12500$ $f=0.05$ $h_{amp}/c=1.00$]

Case-9:

Case-9 has the same kinematics as Case-3; however, the flat plate plunges in a stagnant fluid (water). The averaged velocity field, not presented herein, does not indicate a jet-like formation. According to the measurements, the averaged force coefficient also indicates drag occurrence. The estimated lift coefficient variation is considered to be very successful in terms of both amplitude and periodicity. The estimated variation of drag coefficient still does not match at all the measurement results.

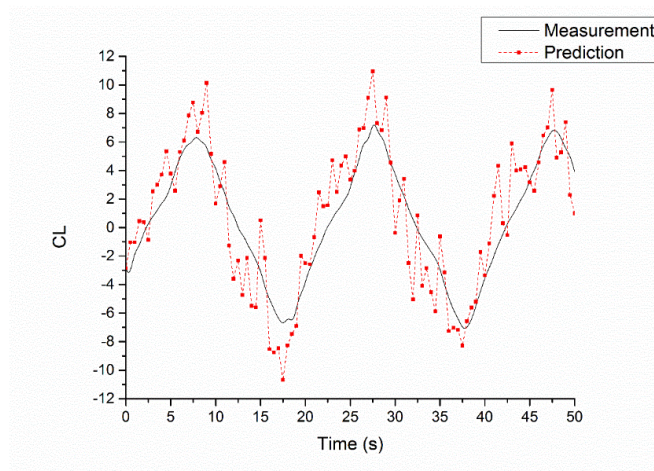


Figure 6.17: CL for Case 9 [$k=1.00$ $Re=3129$ $f=0.05$ $h_{amp}/c=1.00$]

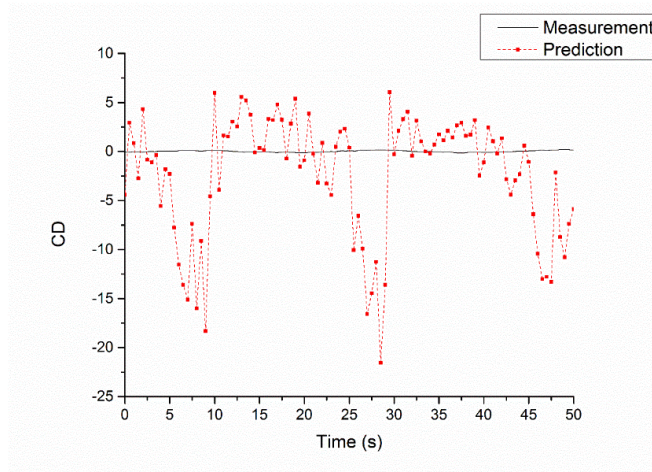


Figure 6.18: CD for Case 9 [$k=1.00$ $Re=3129$ $f=0.05$ $h_{amp}/c=1.00$]

Case-10:

Case-10 is also performed in stagnant fluid (water), it is located on $k-h$ plot in the thrust production side for $f=0.1$ Hz. The averaged velocity field, not presented herein, indicates thrust production with the occurrence of jet-like formation. The estimated lift coefficient variation is considered to be very successful in terms of both amplitude and periodicity. The estimated variation of drag coefficient still does not match at all the measurement results.

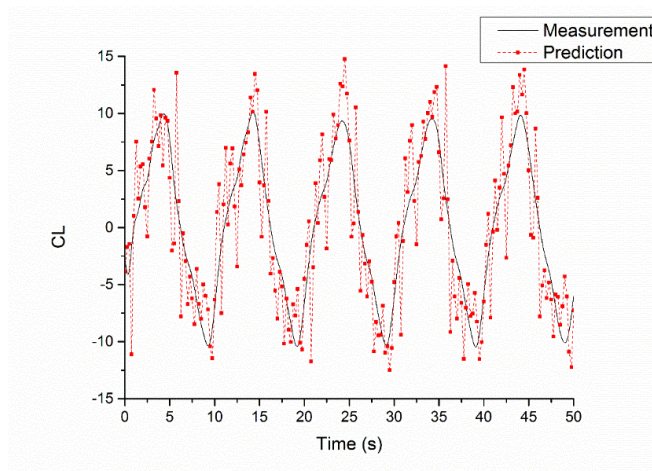


Figure 6.19: CL for Case 10 [$k=1.00$ $Re=3129$ $f=0.10$ $h_{amp}/c=0.50$]

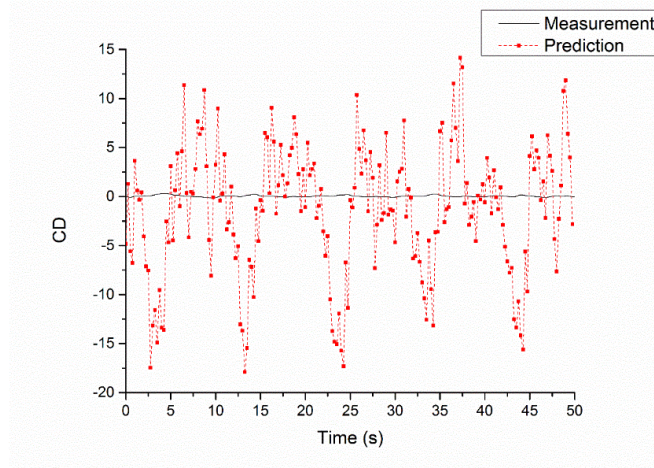


Figure 6.20: CD for Case 10 [$k=1.00$ $Re=3129$ $f=0.10$ $h_{amp}/c=0.50$]

Case-11:

Case-11 is also performed in stagnant fluid (water), it is located on $k-h$ plot in the thrust production side for the highest frequency. The averaged velocity field, not presented herein, indicates thrust production with the occurrence of jet-like formation. The jet-like formation has a narrower width and higher intensity in comparison with the previous case in accordance with the increase in the measured thrust force coefficient. Although the estimated lift coefficient variation is not as successful as in the previous two cases, it follows the periodicity of the measurement, undershooting the amplitude especially for the positive peak and presenting a phase lag in the descending part of the sinus. The estimated variation of drag coefficient still does not match at all the measurement results.

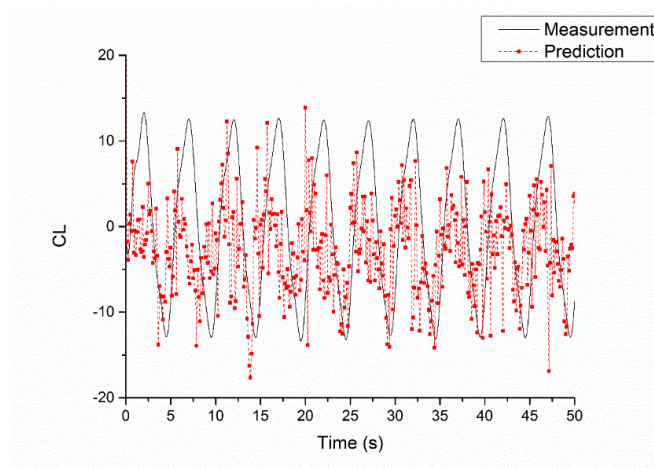


Figure 6.21: CL for Case 11 [$k=4.00$ $Re=3129$ $f=0.20$ $h_{amp}/c=0.25$]

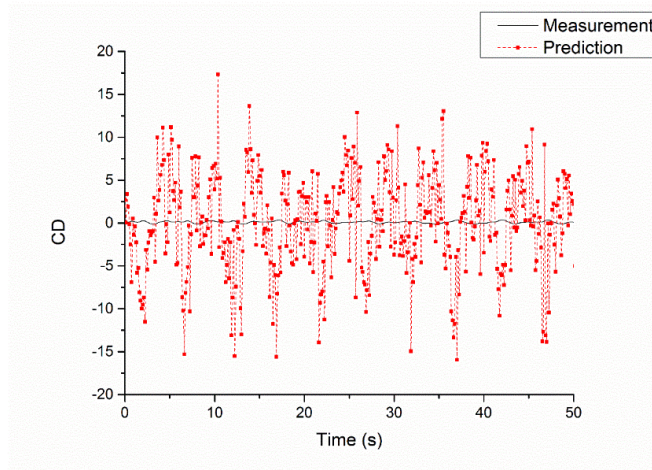


Figure 6.22: CD for Case 11 [$k=4.00$ $Re=3129$ $f=0.20$ $h_{amp}/c=0.25$]

Overall, CL prediction has an acceptable accuracy whereas, there are very large fluctuations with respect to the measurement and therefore a total disagreement in CD prediction.

Through Figure 6.1 to Figure 6.22, it is clear that CD predictions show a remarkable discrepancy with the measured values. However, in order to check the agreement between the average values and comment on drag/thrust occurrence if possible, the obtained values are reported in Table 6.1.

Table 6.1 also includes a grading to show how successful is the CL prediction. The accuracy of CL is graded in a scale from 1 to 3. The grading is done according to following principles:

- If the magnitude of prediction is accurate 1 point is given.
- If the first half of the sine wave is accurate 1 point is given and 0.5 point is given if there is partial accuracy.
- If the second half of the sine wave is accurate 1 point is given and 0.5 point is given if there is partial accuracy.

Table 6.2 : Trends of CD and CL accuracy

Case #	Re	f [Hz]	k	h_{amp}/c	CD _{ave} meas	CD _{ave} pred	CL acc
1	1250	0.1	5.01	0.50	2.07541	-7.01852	1
2	2500	0.2	5.01	0.25	0.65392	-0.97113	1.5
3	3125	0.05	1.00	1.00	0.56296	-0.56464	3
4	5000	0.1	1.25	0.50	0.15225	-0.3951	2
5	5000	0.2	2.50	0.25	0.14796	0.17444	1
6	10000	0.1	0.63	0.50	-0.00108	-0.24018	1.5
7	10000	0.2	1.25	0.25	0.10279	0.19979	1
8	12500	0.05	0.25	1.00	0.08381	-0.06449	3
9	3129	0.05	1.00	1.00	-0.01385	-1.26764	3
10	3129	0.1	2.00	0.50	0.09859	6.72167	3
11	3129	0.2	4.00	0.25	0.18095	-0.01324	1.5

According to the grading, the best performance is obtained when the motion has the lowest frequency and the largest amplitude.

Although Case-5 and Case-7 are identical to the cases performed for sensitivity analysis, satisfactory results could not be obtained and the fact requires further investigation especially on the experimental data in terms of both PIV and direct force measurement comparisons.

7. CONCLUSION

The current study gives the details of the development of an in-house code to estimate loads for a flapping wing using Digital Particle Image Velocimetry (DPIV) data. For the sake of simplicity, pure plunge motion is considered and experiments were performed using different space and time resolutions to assess the performance of the load estimation. Then, the developed code has been tested for different plunge motion cases.

7.1 Concluding Remarks

As presented in Chapter 5.1, the higher order differentiation and surface integral for unsteady term yields a better performance in force prediction. Also, sensitivity analyses shows that a sampling frequency of 8 Hz gives better results than a sampling frequency of 15 Hz, whereas a sampling frequency of 80 Hz gives worse, noisest results compared to lower sampling frequencies used. As a result of the sensitivity analyses, a sampling frequency of 8 Hz is found to be satisfactory which corresponds to 40 vector fields per cycle of motion. A spatial resolution of 3% of the chord and a field of view covering all the amplitude of motion give a satisfactory results with a temporal resolution expressed as 40 vector fields per cycle of motion. In terms of DPIV post-processing, average correlation is found to reduce the noise as it is expected. On the other hand, the estimated load matches better the direct measurement when it is phase averaged and filtered. It should be noted that the compared direct measurement results are also 1 Hz filtered.

For all the investigated results, both for the sensitivity analyses and general plunge cases, lift coefficient variation is in general predicted well, in agreement with the direct force measurements, however the estimated drag coefficient variation has a very large discrepancy with the direct force measurements. This may be due to the neglected viscous term. Consequently, the fact disabled thrust/drag analyses for plunge cases. Estimation of the lift force coefficient variation exhibits a nearly perfect match even

on the instantaneous results when the plunge motion has the lowest frequency and the largest amplitude.

7.2 Recommendations for Further Studies

For further studies, the effect of the viscous term should be considered, the term can be calculated on the control surfaces which may be optimized by a more careful selection of the control volume. Since the line integration of the unsteady term is also sensitive to the control volume selection, it will be possible to obtain a better result in line integration for the unsteady term.

In this study, the pressure field is obtained by the explicit integration of planar pressure gradients. An implicit integration could give better results in pressure field calculation.

Fianlly, in higher order schemes used in this study, the spatial differentiation is one order higher than its temporal counterpart. This could be altered by increasing the temporal differentiation order.

REFERENCES

- Anderson, J. M., Streitlen, K., Barrett, D. S., Triantafyllou, M. S.,** (1998). Oscillating Foils of High Propulsive Efficiency, *Journal of Fluids Mechanics*, 360, 41-72.
- Baur, T. and Köngeter, J.,** (1999). PIV with High Temporal Resolution for the Determination of Local Pressure Reductions from Coherent Turbulence Phenomena, *3rd International Workshop on Particle Image Velocimetry*, University of California-Santa Barbara, USA, September 16-18.
- Betz, A.,** (1912). Ein Beitrag zur Erklärung des Segelfluges, *Zeitschrift für Flugtechnik und Motorluftschiffahrt*, 3, 269-272.
- Çaylan, U., Seçkin, S., Köse, C., Son, O., Zaloğlu, B., Fenercioğlu, İ., Çetiner, O.,** (2014). Çırpan Kanatta Üç Boyutluluk Etkisi, *III. Ulusal Havacılıkta İleri Teknolojiler Konferansı*, Hava Harp Okulu, İstanbul, Turkey, June 18-19.
- David, L., Jardin, T., Farcy, A.,** (2009). On the non-intrusive evaluation of fluid forces with the momentum equation approach. *Measurement Science and Technology*, 20(9) doi:10.1088/0957-0233/20/9/095401
- de Kat, R., van Oudheusden, B. W., Scarano, F.,** (2008). Instantaneous Planar Pressure Field Determination Around a Square-Section Cylinder Based on Time-Resolved Stereo-PIV, *14th International Symposium on Applications of Laser Techniques to Fluid Mechanics*, Lisbon, Portugal, July 07-10.
- de Kat, R., van Oudheusden, B. W.,** (2012). Instantaneous Planar Pressure Determination from PIV in Turbulent Flow, *Experiments in Fluids*, 52(5), 1089-1106.
- Fenercioğlu İ., Çetiner, O.,** (2012). Categoriization of Flow Structures Around a Pitching and Plunging Airfoil, *Journal of Fluids and Structures*, 31, 92-102.
- Fenercioğlu İ., Çetiner, O.,** (2014). Effect of Unequal Flapping Frequencies on Flow Structures, *Aerospace Science and Technology*, 35, 39-53.
- Fenercioğlu İ.,** (2010). *Experimental Investigation of Flow Structures Around an Oscillating Airfoil in Steady Current* (Doctoral dissertation).
- Jardin, T., David, L., Farcy, A.,** (2009). Characterization of Vortical Structures and Loads Based On Time-Resolved PIV for Asymmetric Hovering Flapping Flight. *Experiments in Fluids*, 46(5), 847-857.
- Jones, K. D., Dohring, C. M., Platzer, M. F.** (1996). Wake Structures Behind Plunging Airfoils: A Comparison of Numerical and Experimental Results, *34th Aerospace Sciences Meeting Exhibit*, Reno, Nevada, USA, January 15-18.

- Jones, K. D., Dohring, C. M. and Platzer, M. F.**, (1998). An Experimental and Computational Investigation Of the Knoller-Betz Effect, *AIAA Journal*, 36(7), 1240-1246.
- Karakaş, F., Paça, O., Köse, C., Son, O., Zaloğlu, B., Fenercioğlu, İ., Çetiner, O.**, (2014). Çırpan Kanatta Kanat Profilinin Etkisi, *III. Ulusal Havacılıkta İleri Teknolojiler Konferansı*, Hava Harp Okulu, İstanbul, Turkey, June 18-19.
- Knoller, R.**, (1909). Die Gesetze des Luftwiderstandes Flug- und Motortechnik (Wien), 3, 1-7
- Katzmayr, R.**, (1922). Effect of Periodic Changes of Angle of Attack on Behaviour of Airfoils, NACA Report No. 147, Oct. 1922. (translated from *Zeitschrift für Flugtechnik und Motorluftschiffahrt*, Mar. 31, pp. 80-82, and Apr. 13, 1922, pp. 95-101).
- Kurtuluş, D. F., Scarano, F., David, L.**, (2007). Unsteady Aerodynamic Forces Estimation on a Square Cylinder by TR-PIV, *Experiments in Fluids*, 42(2), 185-196.
- Liu, X., Katz, J.**, (2006). Instantaneous Pressure and Material Acceleration Measurements using a Four-Exposure PIV System, *Experiments in Fluids*, 41(2), 227-240.
- Lynch, K. P. and Scarano, F.**, (2013). A High-Order Time-Accurate Interrogation Method for Time-Resolved PIV. *Measurement Science and Technology*, 24(3) doi: 10.1088/0957-0233/24/3/035305
- Lynch, K. P. and Scarano, F.**, (2014). Material Acceleration Estimation by Four-Exposure Tomo-PIV. *Measurement Science and Technology*, 25(3) doi:10.1088/0957-0233/25/8/084005
- Mohebbian, A., Rival, D.**, (2012). Assesment of the Derivative-Moment Transformation Method for Unsteady-Load Estimation, *Experiments in Fluids*, 53(2), 319-330.
- Novara, M., Scarano, F.**, (2012). Lagrangian Acceleration Evaluation for Tomographic PIV: A Particle-Tracking Based Approach, 14th *International Symposium on Applications of Laser Techniques to Fluid Mechanics*, Lisbon, Portugal, July 09-12.
- Noca F., Shiels, D., Jeon, D.**, (1999). 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.
- Noca F., Shiels, D., Jeon, D.**, (1997). Measuring Instantaneous Fluid Dynamic Forces on Bodies, Using Only Velocity Fields and Their Derivatives, *Journal of Fluids and Structures*, 11(3), 345-350.
- Rival, D., Manejev, R., Tropea, C.**, (2010). Measurement of Parallel Blade-Vortex Interaction at Low Reynolds Numbers, *Experiments in Fluids*, 49(1), 89-99.

- Rival D., Prangemeier T., Tropea C.,** (2009). The influence of Airfoil Kinematics on the Formation of Leading-Edge Vortices in Bio-Inspired Flight, *Experiments in Fluids*, 46(5), 823-833.
- Tronchin, T., David. L., Farcy, A.,** (2015). Loads and Pressure Evaluation of the Flow Around a Flapping Wing from Instantaneous 3D Velocity Measurements, *Experiments in Fluids*, 56(7), doi: 10.1007/s00348-014-1870-x
- Ünal. M. F., Lin, J. C., Rockwell. D.,** (1997). Force Prediction by PIV Imaging: A Momentum-Based Approach, *Journal of Fluids and Structures*, 11, 965-971.
- van Oudheusden, B. W.,** (2013). PIV-based Pressure Measurement. *Measurement Science and Technology*, 24(3) doi:10.1088/0957-0233/24/3/032001.
- van Oudheusden, B. W.,** (2009). Recent Advances in Particle Image Velocimetry, *von Karman Institute for Fluid Dynamics [Lecture Series]*, Rhode-Saint Genese, Belgium, January 26-30.
- van Oudheusden, B. W., Scarano, F., Roosenboom, E. W. M., Casimiri, E. W. F., Souverein, L. J.,** (2007). Evaluation of Integral and Pressure Fields from Planar Velocimetry Data for Incompressible and Compressible Flows, *Experiments in Fluids*, 43(2-3), 153-162.
- Violato, D., Moore, P., Scarano, F.,** (2011). Lagrangian and Eulerian Pressure Fields Evaluation of Rod-Airfoil from Time-Resolved Tomographic PIV. *Experiments in Fluids*, 50(4), 1057-1070.

APPENDICES

APPENDIX A.1: MATLAB Script

```
%Main Body-----
clc; clear all; close all;
tStart=tic;
%Parameters-----
vel_length='m';
grid_length='mm';
des_length='m';
CaseName='E_Case7';
wrtename='E_Case7.mat';
ReyNu=10000;
FreeS=ReyNu/100000;
SampFreq=8;

LE_coordline=5;
dt=1/SampFreq;
chord=0.1;
nu=(FreeS*chord)/ReyNu;
%PIV data Input-----
[FileName,PathName,FilterIndex]=uigetfile('*.TEC','MultiSelect','on');
addpath(PathName);
tReadIn=tic;
t_count=length(FileName);
for t=1:t_count
    [X,Y,U(:, :, t),V(:, :, t),Vort(:, :, t),x_count,y_count,...
        airpos(:, :, t),origin_x(t),origin_y(t),geom(:, :, t)] ...
        = PIVdata_readin(FileName{t},grid_length,...
            vel_length,des_length,LE_coordline);t
end
dx=X(2)-X(1); dy=Y(2)-Y(1);

min_i=zeros(t_count,1); max_i=zeros(t_count,1);
min_j=zeros(t_count,1); max_j=zeros(t_count,1);
for t=1:t_count
    [i_wing,j_wing]=find(isnan(airpos(:, :, t)));
    i_wing=unique(i_wing); j_wing=unique(j_wing);
    min_i(t)=min(i_wing); max_i(t)=max(i_wing);
    min_j(t)=min(j_wing); max_j(t)=max(j_wing);
end
min_i=unique(min_i); max_i=unique(max_i);
min_j=unique(min_j); max_j=unique(max_j);
min_i=min(min_i); max_i=max(max_i);
min_j=min(min_j); max_j=max(max_j);

LoEdge=min_i-10; UpEdge=max_i+10;
LeEdge=min_j-10; RiEdge=max_j+10;

while LoEdge<=3
    LoEdge=LoEdge+1;
end
while LeEdge<=3
    LeEdge=LeEdge+1;
end
while UpEdge>=y_count-2
    UpEdge=UpEdge-1;
end
while RiEdge>=x_count-2
    RiEdge=RiEdge-1;
end
CtrlVolume=[LoEdge UpEdge LeEdge RiEdge];
tReadIn=toc(tReadIn)
%Eulerian Material Acceleration Computation-----
tEuler=tic;
[dudx,dvdx,dudy,dvdy,dudt,dvdt] = grads(U,V,airpos,dx,dy,dt);
[dudx2,dvdx2,dudy2,dvdy2,dudt2,dvdt2] = grads2(U,V,dx,dy,dt);
EulerAccX=dudt+U.*dudx+V.*dudy;
EulerAccY=dvdt+U.*dvdx+V.*dvdy;
EulerAccX2=dudt2+U.*dudx2+V.*dudy2;
EulerAccY2=dvdt2+U.*dvdx2+V.*dvdy2;
tEuler=toc(tEuler)
```

```

tLagrange=tic;
%Lagrangian Material Acceleration Computation-----
[LagrangeAccX,LagrangeAccY] =
LagrangeAcceleration(EulerAccX,EulerAccY,U,V,airpos,X,Y,dt,10^-5,25);
[LagrangeAccX2,LagrangeAccY2] =
LagrangeAcceleration2(EulerAccX2,EulerAccY2,U,V,airpos,X,Y,dt,10^-5,25);
toc(tLagrange)

tPres=tic;
%Pressure Gradient and Poisson Eq. Computation-----
dpx=zeros(y_count,x_count,t_count);
dpy=zeros(y_count,x_count,t_count);
Pressure=zeros(y_count,x_count,t_count);
dpx2=zeros(y_count,x_count,t_count);
dpy2=zeros(y_count,x_count,t_count);
Pressure2=zeros(y_count,x_count,t_count);

for t=1:t_count
    dpx(LoEdge:UpEdge,LeEdge:RiEdge,t)=-
LagrangeAccX(LoEdge:UpEdge,LeEdge:RiEdge,t)+4*nu.*del2(U(LoEdge:UpEdge,LeEdge:RiEdge,
t),dx,dy);
    dpy(LoEdge:UpEdge,LeEdge:RiEdge,t)=-
LagrangeAccY(LoEdge:UpEdge,LeEdge:RiEdge,t)+4*nu.*del2(V(LoEdge:UpEdge,LeEdge:RiEdge,
t),dx,dy);
    dpx2(LoEdge:UpEdge,LeEdge:RiEdge,t)=-
LagrangeAccX2(LoEdge:UpEdge,LeEdge:RiEdge,t)+4*nu.*del2(U(LoEdge:UpEdge,LeEdge:RiEdge
,t),dx,dy);
    dpy2(LoEdge:UpEdge,LeEdge:RiEdge,t)=-
LagrangeAccY2(LoEdge:UpEdge,LeEdge:RiEdge,t)+4*nu.*del2(V(LoEdge:UpEdge,LeEdge:RiEdge
,t),dx,dy);
    sprintf('Grad %d',t)
end

for t=1:t_count

Pressure(:, :,t)=PressGradInteg(Pressure(:, :,t),dpx(:, :,t),dpy(:, :,t),dx,dy,airpos(:
,:,t),CtrlVolume);

Pressure2(:, :,t)=PressGradInteg(Pressure2(:, :,t),dpx2(:, :,t),dpy2(:, :,t),dx,dy,airp
os(:, :,t),CtrlVolume);
    sprintf('Pres %d',t)
end

tPres=toc(tPres)
%Calculate Reynolds Transport Theorem Terms-----
tForce=tic;
FluxTermX=zeros(t_count,1);
FluxTermY=zeros(t_count,1);
PresTermX=zeros(t_count,1);
PresTermY=zeros(t_count,1);
PresTermX2=zeros(t_count,1);
PresTermY2=zeros(t_count,1);
UnsteadyTermX=zeros(t_count,1);
UnsteadyTermY=zeros(t_count,1);
UnsteadyTermX2=zeros(t_count,1);
UnsteadyTermY2=zeros(t_count,1);

for t=1:t_count
    for i=LoEdge:UpEdge-1
        for j=LeEdge:RiEdge-1
            if ~isnan(dudt(i,j,t))
                UnsteadyTermX(t)=UnsteadyTermX(t)+dudt(i,j,t)*dx*dy;
            end
            if ~isnan(dvdt(i,j,t))
                UnsteadyTermY(t)=UnsteadyTermY(t)+dvdt(i,j,t)*dx*dy;
            end
        end
    end
end

for t=1:t_count
    %Left Edge---Upper to Lower
    for i=UpEdge:-1:LoEdge-1
        FluxTermX(t)=FluxTermX(t)-U(i,LeEdge,t)*U(i,LeEdge,t)*dy;
        FluxTermY(t)=FluxTermY(t)-V(i,LeEdge,t)*U(i,LeEdge,t)*dy;
        PresTermX(t)=PresTermX(t)-Pressure(i,LeEdge,t)*dy;
    end
end

```

```

        PresTermX2(t)=PresTermX2(t)-Pressure2(i,LeEdge,t)*dy;
        UnsteadyTermX2(t)=UnsteadyTermX2(t)-X(LeEdge)*U(i,LeEdge,t)*dy;
        UnsteadyTermY2(t)=UnsteadyTermY2(t)-0.5*Y(i)*U(i,LeEdge,t)*dy;
    end
%   Lower Edge---Left to Right
for j=LeEdge:RiEdge-1
    FluxTermX(t)=FluxTermX(t)-U(LoEdge,j,t)*V(LoEdge,j,t)*dx;
    FluxTermY(t)=FluxTermY(t)-V(LoEdge,j,t)*V(LoEdge,j,t)*dx;
    PresTermY(t)=PresTermY(t)-Pressure(LoEdge,j,t)*dx;
    PresTermY2(t)=PresTermY2(t)-Pressure2(LoEdge,j,t)*dx;
    UnsteadyTermX2(t)=UnsteadyTermX2(t)-X(j)*V(LoEdge,j,t)*dx;
    UnsteadyTermY2(t)=UnsteadyTermY2(t)-Y(LoEdge)*V(LoEdge,j,t)*dx;
end
%   Right Edge---Lower to Upper
for i=LoEdge:UpEdge-1
    FluxTermX(t)=FluxTermX(t)+U(i,RiEdge,t)*U(i,RiEdge,t)*dy;
    FluxTermY(t)=FluxTermY(t)+V(i,RiEdge,t)*U(i,RiEdge,t)*dy;
    PresTermX(t)=PresTermX(t)+Pressure(i,RiEdge,t)*dy;
    PresTermX2(t)=PresTermX2(t)+Pressure2(i,RiEdge,t)*dy;
    UnsteadyTermX2(t)=UnsteadyTermX2(t)+X(RiEdge)*U(i,RiEdge,t)*dy;
    UnsteadyTermY2(t)=UnsteadyTermY2(t)+Y(i)*U(i,RiEdge,t)*dy;
end
%   Upper Edge---Right to Left
for j=RiEdge:-1:LeEdge-1
    FluxTermX(t)=FluxTermX(t)+U(UpEdge,j,t)*V(UpEdge,j,t)*dx;
    FluxTermY(t)=FluxTermY(t)+V(UpEdge,j,t)*V(UpEdge,j,t)*dx;
    PresTermY(t)=PresTermY(t)+Pressure(UpEdge,j,t)*dx;
    PresTermY2(t)=PresTermY2(t)+Pressure2(UpEdge,j,t)*dx;
    UnsteadyTermX2(t)=UnsteadyTermX2(t)+X(j)*V(UpEdge,j,t)*dx;
    UnsteadyTermY2(t)=UnsteadyTermY2(t)+Y(UpEdge)*V(UpEdge,j,t)*dx;
end
end

UnsteadyTermX2=gradient(UnsteadyTermX2,dt);
UnsteadyTermY2=gradient(UnsteadyTermY2,dt);

cL=zeros(t_count,1);
cD=zeros(t_count,1);
cL2=zeros(t_count,1);
cD2=zeros(t_count,1);
cL3=zeros(t_count,1);
cD3=zeros(t_count,1);
cL4=zeros(t_count,1);
cD4=zeros(t_count,1);
for t=1:t_count
    cD(t)=2*(-UnsteadyTermX(t)-FluxTermX(t)-PresTermX(t))/(chord*FreeS^2);
    cL(t)=2*(-UnsteadyTermY(t)-FluxTermY(t)-PresTermY(t))/(chord*FreeS^2);

    cD2(t)=2*(-UnsteadyTermX2(t)-FluxTermX(t)-PresTermX(t))/(chord*FreeS^2);
    cL2(t)=2*(-UnsteadyTermY2(t)-FluxTermY(t)-PresTermY(t))/(chord*FreeS^2);

    cD3(t)=2*(-UnsteadyTermX(t)-FluxTermX(t)-PresTermX2(t))/(chord*FreeS^2);
    cL3(t)=2*(-UnsteadyTermY(t)-FluxTermY(t)-PresTermY2(t))/(chord*FreeS^2);

    cD4(t)=2*(-UnsteadyTermX2(t)-FluxTermX(t)-PresTermX2(t))/(chord*FreeS^2);
    cL4(t)=2*(-UnsteadyTermY2(t)-FluxTermY(t)-PresTermY2(t))/(chord*FreeS^2);
end
tStart=toc(tStart)

[cL_r, ~]=size(cL);
if cL_r~=t_count
    cL=cL.'; cD=cD.';
end
clear cL_r

save(wrtename);
figure(1);plot(cL,'r');hold on;plot(cL3);
figure(2);plot(cL3,'r');hold on;plot(cL4);

```

```

%LagrangeAcceleration Function-----
function [LagrangeAccX,LagrangeAccY,errx,erry,iterLag] =
LagrangeAcceleration(EulerAccX,EulerAccY,U,V,airpos,X,Y,dt,tol,max_iter)

[y_count,x_count,t_count]=size(U);

LagrangeAccX=zeros(y_count,x_count,t_count);
LagrangeAccY=zeros(y_count,x_count,t_count);

x_final=zeros(y_count,x_count,t_count);
y_final=zeros(y_count,x_count,t_count);
x_intal=zeros(y_count,x_count,t_count);
y_intal=zeros(y_count,x_count,t_count);

x_final2=zeros(y_count,x_count,t_count);
y_final2=zeros(y_count,x_count,t_count);
x_intal2=zeros(y_count,x_count,t_count);
y_intal2=zeros(y_count,x_count,t_count);

x_final3=zeros(y_count,x_count,t_count);
y_final3=zeros(y_count,x_count,t_count);
x_intal3=zeros(y_count,x_count,t_count);
y_intal3=zeros(y_count,x_count,t_count);

U_final=zeros(y_count,x_count,t_count);
V_final=zeros(y_count,x_count,t_count);
U_intal=zeros(y_count,x_count,t_count);
V_intal=zeros(y_count,x_count,t_count);

U_final2=zeros(y_count,x_count,t_count);
V_final2=zeros(y_count,x_count,t_count);
U_intal2=zeros(y_count,x_count,t_count);
V_intal2=zeros(y_count,x_count,t_count);

U_final3=zeros(y_count,x_count,t_count);
V_final3=zeros(y_count,x_count,t_count);
U_intal3=zeros(y_count,x_count,t_count);
V_intal3=zeros(y_count,x_count,t_count);

AccX=EulerAccX; AccY=EulerAccY;
errx=ones(y_count,x_count,t_count);
erry=ones(y_count,x_count,t_count);
iterLag=zeros(y_count,x_count,t_count);

for t=1:t_count
    for i=2:y_count-1
        for j=2:x_count-1
            if isnan(airpos(i,j,t))
                LagrangeAccX(i,j,t)=NaN;
                LagrangeAccY(i,j,t)=NaN;
                continue
            end
            while errx(i,j,t)>tol && erry(i,j,t)>tol && iterLag(i,j,t)<max_iter

                if t==1
                    x_final(i,j,t)=X(j)+U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
                    y_final(i,j,t)=Y(i)+V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);

                    [U_final(i,j,t),V_final(i,j,t)]=...

CloudInCell(x_final(i,j,t),y_final(i,j,t),X,Y,U(:, :,t+1),V(:, :,t+1));

                    LagrangeAccX(i,j,t)=(U_final(i,j,t)-U(i,j,t))/dt;
                    LagrangeAccY(i,j,t)=(V_final(i,j,t)-V(i,j,t))/dt;

                else if t==t_count
                    x_intal(i,j,t)=X(j)-U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
                    y_intal(i,j,t)=Y(i)-V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);

                    [U_intal(i,j,t),V_intal(i,j,t)]=...

CloudInCell(x_intal(i,j,t),y_intal(i,j,t),X,Y,U(:, :,t-1),V(:, :,t-1));
                    LagrangeAccX(i,j,t)=(U(i,j,t)-U_intal(i,j,t))/dt;
                    LagrangeAccY(i,j,t)=(V(i,j,t)-V_intal(i,j,t))/dt;

                else if t==2 || t==3 || t==t_count-2 || t==t_count-1

```



```

x_final(i,j,t)=X(j)+U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
y_final(i,j,t)=Y(i)+V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);
x_intal(i,j,t)=X(j)-
U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
y_intal(i,j,t)=Y(i)-
V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);

[U_final(i,j,t),V_final(i,j,t)]=...
CloudInCell(x_final(i,j,t),y_final(i,j,t),X,Y,U(:,t+1),V(:,t+1));
[U_intal(i,j,t),V_intal(i,j,t)]=...
CloudInCell(x_intal(i,j,t),y_intal(i,j,t),X,Y,U(:,t-1),V(:,t-1));

LagrangeAccX(i,j,t)=(U_final(i,j,t)-
U_intal(i,j,t))/(2*dt);
LagrangeAccY(i,j,t)=(V_final(i,j,t)-
V_intal(i,j,t))/(2*dt);
else
x_final(i,j,t)=X(j)+U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
y_final(i,j,t)=Y(i)+V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);
x_intal(i,j,t)=X(j)-
U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
y_intal(i,j,t)=Y(i)-
V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);

x_final2(i,j,t)=X(j)+U(i,j,t)*(2*dt)+2*AccX(i,j,t)*(dt^2);
y_final2(i,j,t)=Y(i)+V(i,j,t)*(2*dt)+2*AccY(i,j,t)*(dt^2);
x_intal2(i,j,t)=X(j)-
U(i,j,t)*(2*dt)+2*AccX(i,j,t)*(dt^2);
y_intal2(i,j,t)=Y(i)-
V(i,j,t)*(2*dt)+2*AccY(i,j,t)*(dt^2);

x_final3(i,j,t)=X(j)+U(i,j,t)*(3*dt)+4*AccX(i,j,t)*(dt^2);
y_final3(i,j,t)=Y(i)+V(i,j,t)*(3*dt)+4*AccY(i,j,t)*(dt^2);
x_intal3(i,j,t)=X(j)-
U(i,j,t)*(3*dt)+4*AccX(i,j,t)*(dt^2);
y_intal3(i,j,t)=Y(i)-
V(i,j,t)*(3*dt)+4*AccY(i,j,t)*(dt^2);

[U_final(i,j,t),V_final(i,j,t)]=...
CloudInCell(x_final(i,j,t),y_final(i,j,t),X,Y,U(:,t+1),V(:,t+1));
[U_intal(i,j,t),V_intal(i,j,t)]=...
CloudInCell(x_intal(i,j,t),y_intal(i,j,t),X,Y,U(:,t-1),V(:,t-1));

[U_final2(i,j,t),V_final2(i,j,t)]=...
CloudInCell(x_final2(i,j,t),y_final2(i,j,t),X,Y,U(:,t+2),V(:,t+2));
[U_intal2(i,j,t),V_intal2(i,j,t)]=...
CloudInCell(x_intal2(i,j,t),y_intal2(i,j,t),X,Y,U(:,t-2),V(:,t-2));

[U_final3(i,j,t),V_final3(i,j,t)]=...
CloudInCell(x_final3(i,j,t),y_final3(i,j,t),X,Y,U(:,t+3),V(:,t+3));
[U_intal3(i,j,t),V_intal3(i,j,t)]=...
CloudInCell(x_intal3(i,j,t),y_intal3(i,j,t),X,Y,U(:,t-3),V(:,t-3));

LagrangeAccX(i,j,t)=(U_final(i,j,t)-
U_intal(i,j,t))/(2*dt);
LagrangeAccY(i,j,t)=(V_final(i,j,t)-
V_intal(i,j,t))/(2*dt);

```

```

        if all(~isnan([U_intal3(i,j,t) U_intal2(i,j,t)
U_intal(i,j,t) U_final(i,j,t) U_final2(i,j,t) U_final3(i,j,t)]))
            LagrangeAccX(i,j,t)=(-
U_intal3(i,j,t)+9*U_intal2(i,j,t)-45*U_intal(i,j,t)...
+45*U_final(i,j,t)-
9*U_final2(i,j,t)+U_final3(i,j,t))/(60*dt);
        else if all(~isnan([U_intal2(i,j,t) U_intal(i,j,t)
U_final(i,j,t) U_final2(i,j,t)]))
            LagrangeAccX(i,j,t)=(U_intal2(i,j,t)-
8*U_intal(i,j,t)+8*U_final(i,j,t)-U_final2(i,j,t))/(12*dt);
        else if all(~isnan([U_intal(i,j,t)
U_final(i,j,t)]))
            LagrangeAccX(i,j,t)=(U_final(i,j,t)-
U_intal(i,j,t))/(2*dt);
        else if all(~isnan([U_intal3(i,j,t)
U_intal2(i,j,t) U_intal(i,j,t)]))
            LagrangeAccX(i,j,t)=(-
2*U_intal3(i,j,t)+9*U_intal2(i,j,t)-18*U_intal(i,j,t)+11*U(i,j,t))/(6*dt);
        else if all(~isnan([U_intal2(i,j,t)
U_intal(i,j,t)]))
            LagrangeAccX(i,j,t)=(U_intal2(i,j,t)-4*U_intal(i,j,t)+3*U(i,j,t))/(2*dt);
        else if all(~isnan([U_final3(i,j,t)
U_final2(i,j,t) U_final(i,j,t)]))
            LagrangeAccX(i,j,t)=(2*U_final3(i,j,t)-9*U_final2(i,j,t)+18*U_final(i,j,t)-
11*U(i,j,t))/(6*dt);
        else if
all(~isnan([U_final2(i,j,t) U_final(i,j,t)]))
            LagrangeAccX(i,j,t)=(-
U_final2(i,j,t)+4*U_final(i,j,t)-3*U(i,j,t))/(2*dt);
        end
    end
end
end
end
end
end
end
end

        if all(~isnan([V_intal3(i,j,t) V_intal2(i,j,t)
V_intal(i,j,t) V_final(i,j,t) V_final2(i,j,t) V_final3(i,j,t)]))
            LagrangeAccY(i,j,t)=(-
V_intal3(i,j,t)+9*V_intal2(i,j,t)-45*V_intal(i,j,t)...
+45*V_final(i,j,t)-
9*V_final2(i,j,t)+V_final3(i,j,t))/(60*dt);
        else if all(~isnan([V_intal2(i,j,t) V_intal(i,j,t)
V_final(i,j,t) V_final2(i,j,t)]))
            LagrangeAccY(i,j,t)=(V_intal2(i,j,t)-
8*V_intal(i,j,t)+8*V_final(i,j,t)-V_final2(i,j,t))/(12*dt);
        else if all(~isnan([V_intal(i,j,t)
V_final(i,j,t)]))
            LagrangeAccY(i,j,t)=(V_final(i,j,t)-
V_intal(i,j,t))/(2*dt);
        else if all(~isnan([V_intal3(i,j,t)
V_intal2(i,j,t) V_intal(i,j,t)]))
            LagrangeAccY(i,j,t)=(-
2*V_intal3(i,j,t)+9*V_intal2(i,j,t)-18*V_intal(i,j,t)+11*V(i,j,t))/(6*dt);
        else if all(~isnan([V_intal2(i,j,t)
V_intal(i,j,t)]))
            LagrangeAccY(i,j,t)=(V_intal2(i,j,t)-4*V_intal(i,j,t)+3*V(i,j,t))/(2*dt);
        else if all(~isnan([V_final3(i,j,t)
V_final2(i,j,t) V_final(i,j,t)]))
            LagrangeAccY(i,j,t)=(2*V_final3(i,j,t)-9*V_final2(i,j,t)+18*V_final(i,j,t)-
11*V(i,j,t))/(6*dt);
        else if
all(~isnan([V_final2(i,j,t) V_final(i,j,t)]))
            LagrangeAccY(i,j,t)=(-
V_final2(i,j,t)+4*V_final(i,j,t)-3*V(i,j,t))/(2*dt);
        end
    end
end
end
end
end
end
end
end

```



```

%LagrangeAcceleration2 Function-----
function [LagrangeAccX,LagrangeAccY,errx,erry,iterLag] =
LagrangeAcceleration2(EulerAccX,EulerAccY,U,V,airpos,X,Y,dt,tol,max_iter)

[y_count,x_count,t_count]=size(U);

LagrangeAccX=zeros(y_count,x_count,t_count);
LagrangeAccY=zeros(y_count,x_count,t_count);

x_final=zeros(y_count,x_count,t_count);
y_final=zeros(y_count,x_count,t_count);
x_intal=zeros(y_count,x_count,t_count);
y_intal=zeros(y_count,x_count,t_count);

U_final=zeros(y_count,x_count,t_count);
V_final=zeros(y_count,x_count,t_count);
U_intal=zeros(y_count,x_count,t_count);
V_intal=zeros(y_count,x_count,t_count);

AccX=EulerAccX; AccY=EulerAccY;
errx=ones(y_count,x_count,t_count);
erry=ones(y_count,x_count,t_count);
iterLag=zeros(y_count,x_count,t_count);

for t=1:t_count
    for i=2:y_count-1
        for j=2:x_count-1
            if isnan(airpos(i,j,t))
                LagrangeAccX(i,j,t)=NaN;
                LagrangeAccY(i,j,t)=NaN;
                continue
            end
            while errx(i,j,t)>tol && erry(i,j,t)>tol && iterLag(i,j,t)<max_iter
                if t==1
                    x_final(i,j,t)=X(j)+U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
                    y_final(i,j,t)=Y(i)+V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);

                    [U_final(i,j,t),V_final(i,j,t)]=...
CloudInCell(x_final(i,j,t),y_final(i,j,t),X,Y,U(:, :,t+1),V(:, :,t+1));

                    LagrangeAccX(i,j,t)=(U_final(i,j,t)-U(i,j,t))/dt;
                    LagrangeAccY(i,j,t)=(V_final(i,j,t)-V(i,j,t))/dt;
                else if t==t_count
                    x_intal(i,j,t)=X(j)-U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
                    y_intal(i,j,t)=Y(i)-V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);

                    [U_intal(i,j,t),V_intal(i,j,t)]=...
                    CloudInCell(x_intal(i,j,t),y_intal(i,j,t),X,Y,U(:, :,t-
1),V(:, :,t-1));

                    LagrangeAccX(i,j,t)=(U(i,j,t)-U_intal(i,j,t))/dt;
                    LagrangeAccY(i,j,t)=(V(i,j,t)-V_intal(i,j,t))/dt;
                else
                    x_final(i,j,t)=X(j)+U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
                    y_final(i,j,t)=Y(i)+V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);
                    x_intal(i,j,t)=X(j)-U(i,j,t)*dt+0.5*AccX(i,j,t)*(dt^2);
                    y_intal(i,j,t)=Y(i)-V(i,j,t)*dt+0.5*AccY(i,j,t)*(dt^2);

                    [U_final(i,j,t),V_final(i,j,t)]=...
CloudInCell(x_final(i,j,t),y_final(i,j,t),X,Y,U(:, :,t+1),V(:, :,t+1));
                    [U_intal(i,j,t),V_intal(i,j,t)]=...
                    CloudInCell(x_intal(i,j,t),y_intal(i,j,t),X,Y,U(:, :,t-
1),V(:, :,t-1));

                    if ~isnan(U_final(i,j,t)) && ~isnan(U_intal(i,j,t))
                        LagrangeAccX(i,j,t)=(U_final(i,j,t)-
U_intal(i,j,t))/(2*dt);
                    else if isnan(U_final(i,j,t))
                        LagrangeAccX(i,j,t)=(U(i,j,t)-U_intal(i,j,t))/dt;
                    else
                        LagrangeAccX(i,j,t)=(U_final(i,j,t)-U(i,j,t))/dt;
                    end
                end
            end
            if ~isnan(V_final(i,j,t)) && ~isnan(V_intal(i,j,t))

```

```

LagrangeAccY(i,j,t)=(V_final(i,j,t)-
V_intal(i,j,t))/(2*dt);
else if isnan(V_final(i,j,t))
LagrangeAccY(i,j,t)=(V(i,j,t)-V_intal(i,j,t))/dt;
else
LagrangeAccY(i,j,t)=(V_final(i,j,t)-V(i,j,t))/dt;
end
end
end
errx(i,j,t)=abs(LagrangeAccX(i,j,t)/AccX(i,j,t)-1);
erry(i,j,t)=abs(LagrangeAccY(i,j,t)/AccY(i,j,t)-1);
AccX(i,j,t)=LagrangeAccX(i,j,t);
AccY(i,j,t)=LagrangeAccY(i,j,t);
iterLag(i,j,t)=iterLag(i,j,t)+1;
end
end
end
sprintf('Lagr2 %d',t)
end
end

```

```

%grads Function-----
function [dudx, dvdx, dudy, dvdy, dudt, dvdt] = grads (U,V, airpos, dx, dy, dt)

[y_count, x_count, t_count]=size(U);

dudx=zeros(y_count,x_count,t_count);
dudy=zeros(y_count,x_count,t_count);
dvdx=zeros(y_count,x_count,t_count);
dvdy=zeros(y_count,x_count,t_count);
dudt=zeros(y_count,x_count,t_count);
dvdt=zeros(y_count,x_count,t_count);

for t=1:t_count
    for i=3:y_count-2
        for j=3:x_count-2

            if isnan(airpos(i,j,t))
                dudx(i,j,t)=NaN;
                dvdx(i,j,t)=NaN;
                dudy(i,j,t)=NaN;
                dvdy(i,j,t)=NaN;
                continue
            else
                n=isnan([airpos(i,j-2,t) airpos(i,j-1,t) airpos(i,j+1,t)
airpos(i,j+2,t)]);
                if all(~n)
                    dudx(i,j,t)=(U(i,j-2,t)-8*U(i,j-1,t)+8*U(i,j+1,t)-
U(i,j+2,t))/(12*dx);
                    dvdx(i,j,t)=(V(i,j-2,t)-8*V(i,j-1,t)+8*V(i,j+1,t)-
V(i,j+2,t))/(12*dx);
                else if sum(~n)==3
                    dudx(i,j,t)=(U(i,j+1,t)-U(i,j-1,t))/(2*dx);
                    dvdx(i,j,t)=(V(i,j+1,t)-V(i,j-1,t))/(2*dx);
                else if all(n==[0 0 1 1])
                    dudx(i,j,t)=(U(i,j-2,t)-2*U(i,j-1,t)+3*U(i,j,t))/(2*dx);
                    dvdx(i,j,t)=(V(i,j-2,t)-2*V(i,j-1,t)+3*V(i,j,t))/(2*dx);
                else if all(n==[1 1 0 0])
                    dudx(i,j,t)=(-U(i,j+2,t)+2*U(i,j+1,t)-
3*U(i,j,t))/(2*dx);
                    dvdx(i,j,t)=(-V(i,j+2,t)+2*V(i,j+1,t)-
3*V(i,j,t))/(2*dx);
                end
            end
            end
            n=isnan([airpos(i-2,j,t) airpos(i-1,j,t) airpos(i+1,j,t)
airpos(i+2,j,t)]);
            if all(~n)
                dudy(i,j,t)=(U(i-2,j,t)-8*U(i-1,j,t)+8*U(i+1,j,t)-
U(i+2,j,t))/(12*dy);
                dvdy(i,j,t)=(V(i-2,j,t)-8*V(i-1,j,t)+8*V(i+1,j,t)-
V(i+2,j,t))/(12*dy);
            else if sum(~n)
                dudy(i,j,t)=(U(i+1,j,t)-U(i-1,j,t))/(2*dy);
                dvdy(i,j,t)=(V(i+1,j,t)-V(i-1,j,t))/(2*dy);
            else if all(n==[0 0 1 1])
                dudy(i,j,t)=(U(i-2,j,t)-2*U(i-1,j,t)+3*U(i,j,t))/(2*dy);
                dvdy(i,j,t)=(V(i-2,j,t)-2*V(i-1,j,t)+3*V(i,j,t))/(2*dy);
            else if all(n==[1 1 0 0])
                dudy(i,j,t)=(-U(i+2,j,t)+2*U(i+1,j,t)-
3*U(i,j,t))/(2*dy);
                dvdy(i,j,t)=(-V(i+2,j,t)+2*V(i+1,j,t)-
3*V(i,j,t))/(2*dy);
            end
        end
    end
end
end
end
end

for t=1:t_count

```

```

for i=3:y_count-2
  for j=3:x_count-2
    if isnan(airpos(i,j,t))
      dudt(i,j,t)=NaN;
      dvdt(i,j,t)=NaN;
      continue
    else
      if t==1
        if ~isnan(airpos(i,j,t+1)) && ~isnan(airpos(i,j,t+2))
          dudt(i,j,t)=(-U(i,j,t+2)+2*U(i,j,t+1)-3*U(i,j,t))/(2*dt);
          dvdt(i,j,t)=(-V(i,j,t+2)+2*V(i,j,t+1)-3*V(i,j,t))/(2*dt);
        else if ~isnan(airpos(i,j,t+1))
          dudt(i,j,t)=(U(i,j,t+1)-U(i,j,t))/dt;
          dvdt(i,j,t)=(V(i,j,t+1)-V(i,j,t))/dt;
        else
          dudt(i,j,t)=NaN;
          dvdt(i,j,t)=NaN;
          continue
        end
      end
      else if t==t_count
        if ~isnan(airpos(i,j,t-1)) && ~isnan(airpos(i,j,t-2))
          dudt(i,j,t)=(U(i,j,t-2)-2*U(i,j,t-1)+3*U(i,j,t))/(2*dt);
          dvdt(i,j,t)=(V(i,j,t-2)-2*V(i,j,t-1)+3*V(i,j,t))/(2*dt);
        else if ~isnan(airpos(i,j,t-1))
          dudt(i,j,t)=(U(i,j,t-1)-U(i,j,t))/dt;
          dvdt(i,j,t)=(V(i,j,t-1)-V(i,j,t))/dt;
        else
          dudt(i,j,t)=NaN;
          dvdt(i,j,t)=NaN;
          continue
        end
      end
      else if t==2 || t==t_count-1
        if all(~isnan([U(i,j,t-1) U(i,j,t+1)]))
          dudt(i,j,t)=(U(i,j,t+1)-U(i,j,t-1))/(2*dt);
          dvdt(i,j,t)=(V(i,j,t+1)-V(i,j,t-1))/(2*dt);
        else if isnan(U(i,j,t-1)) && ~isnan(U(i,j,t+1))
          dudt(i,j,t)=(U(i,j,t+1)-U(i,j,t))/dt;
          dvdt(i,j,t)=(V(i,j,t+1)-V(i,j,t))/dt;
        else if isnan(U(i,j,t+1)) && ~isnan(U(i,j,t-1))
          dudt(i,j,t)=(U(i,j,t)-U(i,j,t-1))/dt;
          dvdt(i,j,t)=(V(i,j,t)-V(i,j,t-1))/dt;
        else
          dudt(i,j,t)=NaN;
          dvdt(i,j,t)=NaN;
          continue
        end
      end
    end
  else
    n=isnan([airpos(i,j,t-2) airpos(i,j,t-1) airpos(i,j,t+1)
airpos(i,j,t+2)]);
    if all(~n)
      dudt(i,j,t)=(U(i,j,t-2)-8*U(i,j,t-1)+8*U(i,j,t+1)-
U(i,j,t+2))/(12*dt);
      dvdt(i,j,t)=(V(i,j,t-2)-8*V(i,j,t-1)+8*V(i,j,t+1)-
V(i,j,t+2))/(12*dt);
    end
    if n(1)==1 || n(4)==1
      if n(2)==0 && n(3)==0
        dudt(i,j,t)=(U(i,j,t+1)-U(i,j,t-1))/(2*dt);
        dvdt(i,j,t)=(V(i,j,t+1)-V(i,j,t-1))/(2*dt);
      end
    end
    if all(n==[0 0 1 1]);
      dudt(i,j,t)=(U(i,j,t-2)-2*U(i,j,t-
1)+3*U(i,j,t))/(2*dt);
      dvdt(i,j,t)=(V(i,j,t-2)-2*V(i,j,t-
1)+3*V(i,j,t))/(2*dt);
    end
    if all(n==[1 1 0 0]);
      dudt(i,j,t)=(-U(i,j,t+2)+2*U(i,j,t+1)-
3*U(i,j,t))/(2*dt);
      dvdt(i,j,t)=(-V(i,j,t+2)+2*V(i,j,t+1)-
3*V(i,j,t))/(2*dt);
    end
  end
end

```

```
if sum(~n)==1
    if n(2)==1
        dudt(i,j,t)=(U(i,j,t)-U(i,j,t-1))/dt;
        dvdt(i,j,t)=(V(i,j,t)-V(i,j,t-1))/dt;
    else if n(3)==1
        dudt(i,j,t)=(U(i,j,t+1)-U(i,j,t))/dt;
        dvdt(i,j,t)=(V(i,j,t+1)-V(i,j,t))/dt;
    end
end
end
end
end
end
end
end
end
end
end
```



```

%grads2 Function-----
function [dudx,dvdx,dudy,dvdy,dudt,dvdt] = grads2(U,V,dx,dy,dt)

[y_count,x_count,t_count]=size(U);

dudx=zeros(y_count,x_count,t_count);
dudy=zeros(y_count,x_count,t_count);
dvdx=zeros(y_count,x_count,t_count);
dvdy=zeros(y_count,x_count,t_count);
dudt=zeros(y_count,x_count,t_count);
dvdt=zeros(y_count,x_count,t_count);

for t=3:t_count-2
    for i=3:y_count-2
        for j=3:x_count-2
            dudx(i,j,t)=(U(i,j+1,t)-U(i,j-1,t))/(2*dx);
            dvdx(i,j,t)=(V(i,j+1,t)-V(i,j-1,t))/(2*dx);
            dudy(i,j,t)=(U(i+1,j,t)-U(i-1,j,t))/(2*dy);
            dvdy(i,j,t)=(V(i+1,j,t)-V(i-1,j,t))/(2*dy);
        end
    end
end

for t=3:t_count-2
    for i=3:y_count-2
        for j=3:x_count-2
            dudt(i,j,t)=(U(i,j,t+1)-U(i,j,t-1))/(2*dt);
            dvdt(i,j,t)=(V(i,j,t+1)-V(i,j,t-1))/(2*dt);
        end
    end
end
end

```

```

%PIVdata_readin Function-----
function [X,Y,U,V,Vort,x_count,y_count,airpos,origin_x,origin_y,geom] ...
    = PIVdata_readin(FileName,grid_length,vel_length,des_length,LE_coordline)
%Reads in PIV data set

%Akış Alanı Verisinin Alınması-----
fid=fopen(FileName,'r');
data=textscan(fid,'%n %n %n','headerlines',2);
col1=data{1};
col2=data{2};
col3=data{3};
fclose(fid);
j=1;
for i=1:length(col3);
    if isnan(col3(i))==0
        u(j,1)=col3(i);
        j=j+1;
    end
end
for i=0:length(col1)/2-1
    x(i+1,1)=col1(2*i+1);
    y(i+1,1)=col2(2*i+1);
end
vort=zeros(length(col1)/2,1);
v=zeros(length(col1)/2,1);
for i=1:length(col1)/2
    v(i,1)=col1(2*i);
    vort(i,1)=col2(2*i);
end
X=unique(x); X=X.';
Y=unique(y); Y=Y.';
U=reshape(u,length(X),length(Y)); U=U.';
V=reshape(v,length(X),length(Y)); V=V.';
Vort=reshape(vort,length(X),length(Y)); Vort=Vort.';
x_count=length(X); col_size=x_count;
y_count=length(Y); row_size=y_count;
%-----
%Plaka Geometrisi Bilgisinin Alınması-----
fid=fopen(FileName,'r');
data=textscan(fid,'%n %n','headerlines',x_count*y_count*2+3);
fclose(fid);
col4=data{1};
col5=data{2};

geom(:,1)=col4(3:end,:);
geom(:,2)=col5(3:end,:);

if isfloat(LE_coordline) && LE_coordline <= length(col4) && LE_coordline > 0
    origin_x=col4(LE_coordline);
    origin_y=col5(LE_coordline);
else
    origin_x=NaN;
    origin_y=NaN;
end

airpos=zeros(y_count,x_count);
for j=1:x_count
    for i=1:y_count
        if U(i,j)==0 && V(i,j)==0
            airpos(i,j)=NaN;
        else
            airpos(i,j)=1;
        end
    end
end

if ~strcmp(des_length,grid_length)
    if strcmp(des_length,'m')
        X=X*10^-3;
        Y=Y*10^-3;
        origin_x=origin_x*10^-3;
        origin_y=origin_y*10^-3;
        geom=geom*10^-3;
        Vort=Vort*10^3;
    else if strcmp(des_length,'mm')
        X=X*10^3;

```

```

        Y=Y*10^3;
        origin_x=origin_x*10^3;
        origin_y=origin_y*10^3;
        geom=geom*10^3;
        Vort=Vort*10^-3;
    end
end
end

if ~strcmp(des_length,vel_length)
    if strcmp(des_length,'m')
        U=U*10^-3;
        V=V*10^-3;
        Vort=Vort*10^-3;
    else if strcmp(des_length,'mm')
        U=U*10^3;
        V=V*10^3;
        Vort=Vort*10^3;
    end
end
end

end

```

```

%PressGradInteg Function-----
function [Pressure] = PressGradInteg(Pressure, dpdx, dpdy, dx, dy, airpos, CtrlVolume)

LoEdge=CtrlVolume(1);
UpEdge=CtrlVolume(2);
LeEdge=CtrlVolume(3);
RiEdge=CtrlVolume(4);
flag=0;
for i=LoEdge:UpEdge
    for j=LeEdge:RiEdge

        if i==LoEdge && j==LeEdge
            continue
        end

        if isnan(airpos(i,j))
            continue
        end

        delp=zeros(1,4); press=0; contr=0;
        adjc=[i-1,j-1;i-1,j;i-1,j+1;i,j-1];

        for k=1:4
            delp(k)=deltaP(dpdx,dpdy,CtrlVolume,airpos,Pressure,adjc(k,:),i,j,dx,dy);
        end

        press=nansum(delp);
        contr=sum(~isnan(delp));

        if contr~=0
            Pressure(i,j)=press/contr;
        else

            i_idx=i; flag=1;
            for i=i_idx:UpEdge-1
                Pressure(i+1,1)=dpdy(i,1)*dy+Pressure(i,1);
            end
            for j=1:RiEdge-1
                Pressure(UpEdge,j+1)=dpdx(UpEdge,j)*dx+Pressure(UpEdge,j);
            end

            for i=UpEdge:-1:i_idx
                for j=RiEdge:-1:LeEdge

                    if i==UpEdge && j==RiEdge
                        continue
                    end
                    if isnan(airpos(i,j))
                        continue
                    end

                    delp=zeros(1,4); press=0; contr=0;
                    adjc=[i+1,j-1;i+1,j;i+1,j+1;i,j+1];

                    for k=1:4

delp(k)=deltaP(dpdx,dpdy,CtrlVolume,airpos,Pressure,adjc(k,:),i,j,dx,dy);
                        end

                    press=nansum(delp);
                    contr=sum(~isnan(delp));
                    Pressure(i,j)=press/contr;

                end
            end
        end
    end
end

if flag==1

Pressure=integforward(i_idx,UpEdge,LeEdge,RiEdge,CtrlVolume,dpdx,dpdy,airpos,Pressure,
dx,dy);

Pressure=integbackward(i_idx,UpEdge,LeEdge,RiEdge,CtrlVolume,dpdx,dpdy,airpos,Pressure,
dx,dy);

```

```
else
Pressure=integbackward(LoEdge,UpEdge,LeEdge,RiEdge,CtrlVolume,dpdx,dpdy,airpos,Pressure,dx,dy);
end

end
```

```

%integbackward Function-----
function [Pressure, iidx, jidx, flag] =
integbackward(i1, i2, j1, j2, CtrlVolume, dpdx, dpdy, airpos, Pressure, dx, dy)
flag=0; iidx=0; jidx=0;
LoEdge=CtrlVolume(1);
UpEdge=CtrlVolume(2);
LeEdge=CtrlVolume(3);
RiEdge=CtrlVolume(4);
for i=i2:-1:i1
    for j=j2:-1:j1

        if i==i2 && j==j2
            continue
        end

        if isnan(airpos(i,j))
            continue
        end

        delp=zeros(1,4);
        adjc=[i+1,j-1;i+1,j;i+1,j+1;i,j+1];

        for k=1:4
            delp(k)=deltaP(dpdx, dpdy, CtrlVolume, airpos, Pressure, adjc(k,:), i, j, dx, dy);
        end

        press=nansum(delp);
        contr=sum(~isnan(delp));

        if contr~=0
            Pressure(i,j)=press/contr;
        else
            iidx=i; jidx=j; flag=1;
            break
        end
    end

    if flag==1
        break
    end
end

end

%integforward Function-----
function [Pressure, iidx, jidx, flag] =
integforward(i1, i2, j1, j2, CtrlVolume, dpdx, dpdy, airpos, Pressure, dx, dy)
flag=0; iidx=0; jidx=0;
LoEdge=CtrlVolume(1);
UpEdge=CtrlVolume(2);
LeEdge=CtrlVolume(3);
RiEdge=CtrlVolume(4);
for i=i1:i2
    for j=j1:j2

        if i==i1 && j==j1
            continue
        end

        if isnan(airpos(i,j))
            continue
        end

        delp=zeros(1,4);
        adjc=[i-1,j-1;i-1,j;i-1,j+1;i,j-1];

        for k=1:4
            delp(k)=deltaP(dpdx, dpdy, CtrlVolume, airpos, Pressure, adjc(k,:), i, j, dx, dy);
        end

        press=nansum(delp);
        contr=sum(~isnan(delp));

        if contr~=0
            Pressure(i,j)=press/contr;
        else

```

```
        iidx=i; jidx=j; flag=1;
        break
    end
end
if flag==1
    break
end
end
end
```

CURRICULUM VITAE

Onur Paça

e-mail: paca@itu.edu.tr; onurpaca@gmail.com

DOB: 22/05/1991, İstanbul

Education:

- **MSc:** İstanbul Technical University, Aerospace Engineering Graduate Program
Load Estimation Using Digital Particle Image Velocimetry Data for Flapping Wing Aerodynamics
- **BSc:** İstanbul Technical University, Astronautical Engineering
Vortex Identification In Data Processing of Digital Partice Image Velocimetry