Time Domain Modeling of Unsteady Aerodynamic Forces on a Flapping Airfoil

by

Eric Durand

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 1998

Author .....
Department of Aeronautics and Astronautics
May 8, 1998

Certified by .....
James Paduano
Associate Professor
Thesis Supervisor

Accepted by .....
Jaime Peraire
Chair, Graduate Office

Time Domain Modeling of Unsteady Aerodynamic Forces on a Flapping Airfoil

by

Eric Durand

Submitted to the Department of Aeronautics and Astronautics
on May 8, 1998, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

The goal of this research is to develop a versatile and fast code to compute the unsteady lift and thrust forces generated by a flapping airfoil and apply it to engineering problems. We consider both plunging and pitching types of motion and develop a time marching simulation based on early work in unsteady aerodynamics. The ability of the code to compute lift and thrust forces is validated against published results. We then study ways to maximize thrust generation by considering sinusoidal and square motions and optimizing parameters such as flapping frequency, amplitude, and phase difference between pitch and plunge. Finally, the code is applied to the take-off problem of a micro-UAV to illustrate the ability of the code to compute transient forces. The novelty of this research resides in the consideration of square motions and the optimization of parameters to maximize thrust. Also, the ability of the code to show transient forces and yet run faster than real time makes it a valuable tool for a wide variety of applications.

Thesis Supervisor: James Paduano
Title: Associate Professor

Acknowledgments

As I type the last sentences of my thesis, my first thoughts go to my advisor Professor James Paduano. Thank you for your patience and guidance throughout this project. I would also like to recognize Professor Ken Hall from Duke University for his expertise and advice on the field of unsteady aerodynamics while on sabbatical at M.IT.

My undergraduate years were spent at the University of Kansas were I greatly enjoyed the education I received from the Department of Aerospace Engineering. I would like to mention three professors who have had an influence on my career choices and whom I consider as mentors: Professors Jan Roskam, Saeed Farokhi and Ray Taghavi.

In my two years spent at M.I.T. I had the privilege to meet some very special people who in a way or another gravitated around the mailing list cercle@mit.edu. Manu, Benoit and Sandrine, Ralph and Claire, Seb, Nico, Greg, Martin, Stephan, Renaud and the others reminded me that once you get out of the lab there is a lot more to life. Also, in the I.C.E. lab I enjoyed the times spent with my labmates: Alex, Arkadiy, Emilio, Jae (Dr. Ho), Jerry, Dr. Li, Sean and Vlad. Whether it was discussing world politics over coffee or providing help on a research related issue, they were always available.

My family has and still plays a most important part in my life, so at this moment, my thoughts go to my parents Alain and Josiane in Gemenos near Marseilles, France for whom this thesis is in many ways. Also, to my brother Alain-Philippe, his wife Sherri and my niece/goddaughter Chloé in North Carolina for being my only family in the US and being greatly supportive.

Finally, there is a person, who has had to cope with my times of doubt, my changes of mood, my odd hours and my lack of time and yet agreed to become my wife. Melissa, thank you for being such a wonderful person, you pushed me to finish this and kept me on the right track.

Contents

List of Figures

List of Tables

Nomenclature

b	wing span [m]
C	force coefficient
$\overline{C}$	force coefficient averaged over one period
c	chord length [m]
f	frequency of oscillation [rad/s]
k	pitching reduced frequency
L	lift force perpendicular to the airflow and per unit length [N/m in Chapter 2 and N in Chapter 4]
l	plunging reduced frequency
P_x	suction force parallel to the airflow and per unit length [N/m in Chapter 2 and N in Chapter 4]
s	Laplace transform [1/s]
S	suction coefficient [$m^{3/2}/s$]
T	thrust force parallel to the airflow and per unit length [N/m in Chapter 2 and N in Chapter 4]
T	one period of oscillation [s]
t	time [s]
V_∞	airflow velocity [m/s]
w	downwash [m/s]
x	horizontal position [m]
z	vertical position [m]
$\dot{z}$	airfoil plunge rate [m/s]
$\ddot{z}$	airfoil plunge acceleration [m/s^2]

Greek

α	angle of attack measured between the airflow and the chord [rad]
$\dot{\alpha}$	angle of attack rate of change, also pitch rate [rad/s]
Γ	circulation [m^2/s]
γ	flight path angle [deg]
γ	vorticity [m/s]
ϕ	phase difference between pitch and plunge [deg]
ρ	airflow density [sl/m^3]
τ	non-dimensional time
ω_n	natural frequency [rad/s]
ξ	damping ratio
ξ	non-dimensional distance

subscripts

0	steady bound
1	unsteady bound
a	airfoil
i	inertial
L	lift
max	maximum
n	normal to the chord
s	steady
T	thrust
w	wake

Chapter 1

Introduction

1.1 Motivation and Perspective

Propulsion through wing flapping has long been a compelling subject, as the primary mode of flight propulsion in the animal kingdom. For example, birds such as the hawk shown on Figure 1-1, have mastered the art of wing flapping.

Figure 1-1 The Hawk

Theories as to how a typical lifting surface can be oscillated to produce both lift and propulsion have their foundations in unsteady aerodynamics, whose development began in the 20's [1], [2], [3] and [4] , and whose maturation has occurred primarily for the purpose of understanding aeroelasticity [5], rather than propulsion through flapping. The latter subject has been researched both as a means of understanding how birds accomplish flight [6] and [7], and by those interested in the engineering prospects of ornithopters [8], [9], [10], [11], [12] and [13].

While much progress has been made in understanding the basic mechanisms involved in propulsive flapping, practical ornithopters have not been developed for

various reasons. The most obvious of these is the severe mechanical challenge associated with building a flapping wing. Even if this challenge could be overcome, the efficiency afforded by propellers (the obvious choice for low-speed propulsion) has not been improved upon by oscillating airfoils in any theoretical or experimental study.

Nevertheless, interest in ornithopters has piqued recently for several reasons. Foremost is the current interest in micro-UAVs, uninhabited aerial vehicles with dimensions similar to that of a small bird. Because the flight regime of these vehicles is exactly that of birds, revisiting the overall engineering question of ornithopters has become relevant.

The question of mechanization is also mitigated by various recent developments. Again, the issue of size stands out: flapping is more easily mechanized in small vehicles due to scaling laws; for instance it is much easier to make a structure that will support the forces associated with high-frequency flapping if the wing is small. In addition, micro-machined devices, smart materials, and advanced composites all represent new technologies that could make flapping flight more feasible mechanically on small systems.

Two of the remaining issues motivate the research presented here. The first is efficiency of propulsion. Several studies [8], [14], [11] and [13] indicate that the thrust producing capability of wings, if not their efficiency, can be increased dramatically by departing from the classical notion of a single airfoil undergoing sinusoidal motion. Our goal is to develop a framework in which design studies can be conducted for non-traditional types of flapping. Non-traditional flapping includes non-sinusoidal motions, and tandem airfoils flapping with the proper consonance.

The second issue that we would like to begin to address is stability and control. If one is to use flapping as the sole means of propulsion, methods to modulate thrust levels, coordinate thrust and lift, and use wing forces for vehicle

maneuvering are needed. To study stability and control issues one needs an engineering model of unsteady aerodynamics, which is consistent (in both complexity and versatility) with flight control design and analysis techniques. Such an engineering model is the subject of this thesis.

1.2 Background on Unsteady Aerodynamics

The evolution of the field of unsteady aerodynamics can be divided into three areas: the foundations, the advent of computers and application to engineering problems. We develop in this section the contributions we found predominant to each one of the three areas fore-defined. We also outline which aspects in each contribution are relevant to our research and which are not.

The foundations of unsteady aerodynamics were laid down in the 20's. The pioneers in the field of aerodynamics include Wagner, Von Kármán, Garrick and Theodorsen. Wagner [1] was the first to publish a way to calculate the distribution of vorticity in the wake of an airfoil undergoing unsteady motions. Later, Von Kármán used Kelvin circulation theory and a wake integral approach to express the lift and thrust developed by a flapping airfoil [2] and [3]. Theodorsen [5] approached the problem by potential flow theory and solved the wake integral with Bessel functions, hence adding great mathematical complexity. Also, his work was solely concerned with lift forces and applied to the problem of flutter. Garrick [4] gave a more detailed way to calculate the thrust forces given by Von Kármán by combining them with the work of Theodorsen. The theoretical background of the work presented here is inspired from these early publications, and more specifically Von Kármán and Garrick. Additional insight was obtained by studying Bisplinghoff [15] and Anderson [16]. However, these approaches assume sinusoidal motion, which does not allow one to study the possible benefits of non-sinusoidal flapping. Transient effects are also difficult to study in a model which does not allow time-resolved solution. Computers alleviate this problem.

The advent of computers sparked a renewed interest in the topic of flapping wings and unsteady aerodynamics. Among the computational approaches we find the work of Platzer [11] who applied a panel method to compute unsteady lift and thrust forces for pitch and plunge motion of one or two airfoils. Hall and Hall [12] address the issue of minimum power flapping flight and compute the optimal flapping frequency by using a vortex-lattice model of the wake. These approaches allow more detailed studies, but are not suitable for stability and control studies, because of the large overhead associated with finding solutions. After initial trade studies and control law designs are done using more tractable engineering models, computationally intensive models can be used for final validation and tuning.

Hugo [9] used an unsteady lifting line theory, based on Theodorsen results to compute lift forces and to control the unsteady lift occurring in gusty conditions. He also applied his method to compute wing loading [10]. One disadvantage of this method is that because it is based on Theodorsen's method it does not allow the computation of thrust forces. McCune [8] followed a similar lifting line theory approach based on Von Kármán's results but allowed the wake elements to freely interact and roll-up. Interestingly, he set up his code so that it was possible to follow the time history of the flow at each time step. However, this approach, aside from being computationally intensive, was not originally intended for computation of thrust forces.

1.3 Approach and Thesis Outline

In the light of the passed research discussed here and in accordance with our goals, we decided to adopt an approach leading to a versatile and non intensive computational tool. The work of Von Kármán and Garrick was the most appropriate to meet our goal since it presented equations to compute lift and thrust forces that could be implemented with minimal computational effort. Also, the equations they developed presented a natural way to modify the type of input motion to fit our needs. Finally, the equations could be implemented in a way

similar to the way that McCune presented, that is provide a time history of the forces at each time step. Such a time marching approach allows the study of transient force generation where speed may be varying significantly.

The thesis is organized as follows in accordance with the approach and goals stated earlier. We start in Chapter 2 by compiling the equations developed by Von Kármán and Garrick for lift and thrust and adapting them to our problem through proper non-dimensionalization and introduction of a direct relationship between input motion and output forces. Chapter 3 is devoted to the implementation of the computational approach and its validation. Finally, Chapter 4 of the thesis is applied to two engineering problems. 1) An initial engineering survey of oscillatory waveforms is conducted, to demonstrate the utility of the code and to suggest alternate flapping strategies. 2) To demonstrate the new model's capability to handle time-resolved transients with speed variations, a take-off time history is conducted for a simplified one degree of freedom flight model, which models only the forces in the thrust direction.

Chapter 2

Model Development

2.1 Overview and Organization

This chapter presents the derivation of the equations needed to calculate the aerodynamic forces experienced by a flapping airfoil. In steady aerodynamics these forces are defined as a lift term, perpendicular to the airflow and calculated by the Kutta-Joukowski formula, and a drag term, parallel to the airflow and due to viscous forces. In unsteady aerodynamics, because the motion of the airfoil is time-varying, vortex elements are shed into the flow and in accordance with Kelvin's circulation theorem [16], an equivalent and opposite bound circulation will develop over the airfoil (see Anderson, page 265). Kelvin's conservation of circulation theorem, illustrated in Figures 2-1 and 2-2, states that the circulation Γ within a bounded flow must remain constant.

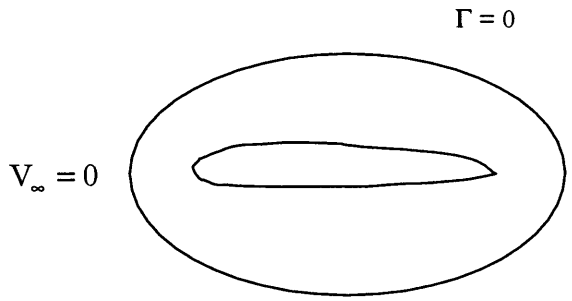

Figure 2-1 Fluid at Rest Relative to the Airfoil

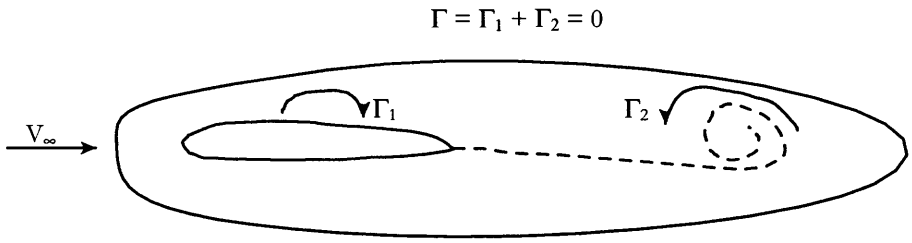

Figure 2-2 Fluid After Motion Has Started

After motion has started, a region of high vorticity will develop at the trailing edge of the airfoil and travel downstream at the velocity of the airflow. The circulation (Γ_2 on Figure 2-2) associated with this region of vorticity will be canceled by a bound circulation (Γ_1 on Figure 2-2) on the airfoil in accordance with Kelvin's conservation of circulation theorem. This bound circulation will, in turn, induce a component of velocity normal to the chord, or downwash, and hence an unsteady force.

These unsteady forces have been computed in the past and several different approaches exist. We decided to follow the work of Durand, Von Kármán and Burgers [2] and Von Kármán and Sears [3] also known as the vortex-sheet approach [15]. Their work fits our problem particularly well since they are among the only ones to have documented how to compute the thrust force. Also, another advantage is that their work is clear and well documented in spite of being half a century old. Von Kármán and Sears also divided the resulting lift and thrust forces developed over the airfoil into physically understandable forces.

The total instantaneous circulation on an airfoil is:

$$\Gamma = \int_{-c/2}^{c/2} \gamma_a(x,t)dx \qquad \text{(Eq. 2.1)}$$

where γ_a, the running bound circulation, is for convenience written as:

$$\gamma_a(x,t) = \gamma_0(x,t) + \gamma_1(x,t) \qquad \text{(Eq. 2.2)}$$

where γ_o is called the quasi-steady bound circulation and will be used in sections 2.5 and 2.6 to compute the quasi-steady lift L_s, and the inertial force, L_i. γ_1 is the additional circulation due to the unsteadiness of the flow according to Kelvin's circulation theorem. Each increment to γ_1 generates an opposite element in the wake; γ_w. γ_1 and γ_w will be used to compute the wake induced force L_w and the suction force P_x.

In the equations developed by Von Kármán and Sears, a lot of emphasis was put on the methods to derive the forces from fundamental principles such as Kelvin's circulation theorem. Note that their original work did not present in their paper a treatment of the thrust force. The information needed to compute thrust is found in an earlier work by Von Kármán [2] and also a paper by Garrick [4]. In this thesis, we start from the equations they developed and apply them to our problem. Our goal is to show the relationship between the input airfoil motion and the output forces. We also introduce various simplifying assumptions, such as assuming that the downwash is constant over the airfoil. Finally, we introduce non-dimensional parameters to further reduce the force equations and present a method to compute the average thrust generated.

A summary of the forces exerted on the airfoil is shown on the free-body diagram in Figure 2-3.

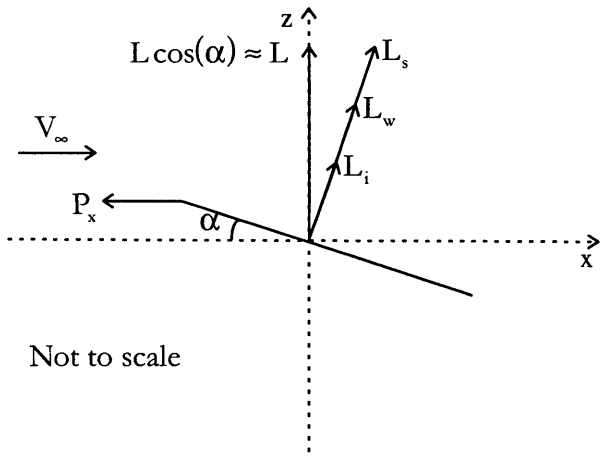

Figure 2-3 Free-Body Diagram Showing Forces Exerted on the Airfoil

The resulting lift and thrust forces, defined parallel and perpendicular to the airflow, can be found from Figure 2-3 by solving the free body diagram:

$$L = (L_s + L_i + L_w)\cos(\alpha) \approx L_s + L_i + L_w$$

$$T = P_x - L\sin(\alpha) \qquad \text{(Eq. 2.3)}$$

In terms of non-dimensional coefficients:

$$C_L = C_{L_s} + C_{L_i} + C_{L_w}$$

$$C_T = C_{P_x} - C_L \sin(\alpha) \qquad \text{(Eq. 2.4)}$$

where each force L (or T) is non-dimensionalized as follows: $L = C_L \bar{q} c$, where $\bar{q} = \frac{1}{2}\rho V_\infty^2$ is the dynamic pressure and c is the airfoil chord length.

Finally, if one is interested in the average thrust coefficient developed over one period of oscillation (both pitch and plunge), the following equation may be

used:

$$\overline{C_T} = \frac{1}{T}\int_{penod} C_T dt \qquad \text{(Eq. 2.5)}$$

2.2 Assumptions

While developing the force equations we are faced with the necessity to constrain our problem with simplifying assumptions about the flow and the airfoil.

The first assumption is to limit our study to a two-dimensional analysis. This implies neglecting induced forces and assuming constant airfoil shape in the spanwise direction. The wake flow is also dramatically simplified in a two-dimensional model.

The second assumption, also geometric, is to consider the airfoil to be a flat plate. Such an assumption allows us to apply the vortex sheet method to model the airfoil. Also, it simplifies the equations by making the downwash constant along the chord.

The third assumption is that the flow is inviscid. This means that no friction force is generated by the boundary layer. In the steady case, this assumption would imply that an airfoil can develop lift, but has zero drag. We will see that in unsteady flow, drag (or thrust) forces can be generated, even without viscous effects.

Finally, our fourth assumption is that the vortex elements shed in the flow do not interact with each other. This assumption is sometimes referred to as the flat wake assumption [3].

2.3 Non-Dimensionalization and Notation

The equations developed by Von Kármán and Sears can be simplified by introducing non-dimensional parameters such as non dimensional time and reduced frequency. A non dimensional time of one unit corresponds to the time it takes a particle at the freestream velocity to traverse half a chord length. A reduced frequency of one unit corresponds to a sinusoidal oscillation whose period is 2π non-dimensional time units.

The following terms will be used to express the forces in term of dimensionless coefficients:

$$\tau = \frac{2V_\infty t}{c} \text{ is the non-dimensional time} \qquad \text{(Eq. 2.6)}$$

$$k = \frac{f_{pitch} c}{2V_\infty} \text{ is the pitching reduced frequency} \qquad \text{(Eq. 2.7)}$$

$$l = \frac{f_{plunge} c}{2V_\infty} \text{ is the plunging reduced frequency} \qquad \text{(Eq. 2.8)}$$

2.4 Downwash

Section 2.1 introduced the notion that a downwash is induced over the airfoil in unsteady motions. The "no normal flow" condition helps to explain this effect and will lead to an expression to compute the downwash as follows: In thin airfoil theory, it is desirable to view the chord as a streamline of the flow. This condition requires that the component of velocity normal to the chord line must be zero at all points along the chord (Anderson, page 268). Figure 2-4 gives physical insight as to how one calculates w_a, the vertical downwash.

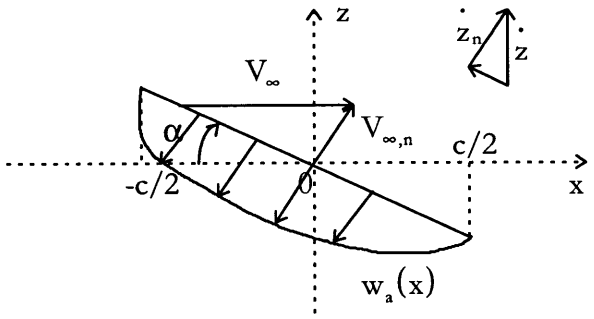

Figure 2-4 Definition of Downwash, w_a

It can be seen from Figure 2-4 that the condition of no flow through the airfoil is satisfied for:

$$w_a + V_{\infty,n} + \dot{z}_n = 0 \qquad \text{(Eq. 2.9)}$$

where $(\dot{\cdot})$ denotes the derivative with respect to the argument, which in this case is dimensional time.

Noting that $V_{\infty,n} = V_\infty \sin(\alpha)$ and $\dot{z}_n = \dot{z}\cos(\alpha)$, Equation 2.9 becomes:

$$\frac{w_a}{V_\infty} = -\sin(\alpha) - \frac{\dot{z}\cos(\alpha)}{V_\infty} \qquad \text{(Eq. 2.10)}$$

Note that in pitching motion ($\dot{z} = 0$), w_a will be negative for a positive angle of attack.

As will be seen in subsequent sections, it is useful to differentiate Equation 2.10 with respect to time:

$$\frac{\dot{w}_a(t)}{V_\infty} = -\dot{\alpha}\cos(\alpha) - \frac{\ddot{z}\cos(\alpha) - \dot{z}\dot{\alpha}\sin(\alpha)}{V_\infty} \qquad \text{(Eq. 2.11)}$$

Where we assume that V_∞ is slowly time-varying and therefore its time derivative is zero. To be perfectly clear about whether the 'dot' notation refers to differentiation with respect to dimensional or non-dimensional time, the argument of the function (t or τ) will be shown.

2.5 The Quasi-Steady Force

The term quasi-steady is used here because its value is directly derived from the expression for the steady lift. The force, however, is time-varying. The Kutta-Joukowski theorem states that the resultant force acting on an object in steady and inviscid flow is perpendicular to the airflow and its magnitude is defined according to (Anderson, page 216):

$$L_s(t) = \rho V_\infty \Gamma_0(t) \qquad \text{(Eq. 2.12)}$$

where Γ_0 is the steady airfoil circulation at time t.

Γ_0 may be computed from the following formula (see Bisplinghoff, page 289):

$$\Gamma_0(t) = \int_{-\frac{c}{2}}^{\frac{c}{2}} \gamma_0(x,t)dx = -c\int_{-1}^{1}\sqrt{\frac{1+\xi}{1-\xi}}w_a(\xi,t)d\xi \qquad \text{(Eq. 2.13)}$$

where w_a was defined in Section 2.4 and $\xi = \frac{2x}{c}$ is a non-dimensional position variable of integration here.

Equation 2.13 can be reduced to the following (see Appendix A.1):

$$\Gamma_0(t) = -\pi c w_a(t) \qquad \text{(Eq. 2.14)}$$

Note that Equation 2.14 is only valid when the airfoil downwash is not position dependent. This means that we assume the airfoil to be a flat plate. The integration would have to be solved numerically for the case where a cambered airfoil is considered.

Plugging Equation 2.14 into Equation 2.12, the quasi steady lift becomes:

$$L_s(t) = -\rho V_\infty \pi c w_a(t) \qquad \text{(Eq. 2.15)}$$

Equation 2.15 can be non-dimensionalized by introducing the steady lift coefficient:

$$C_{L_s} = \frac{2L_s}{\rho V_\infty^2 c} \qquad \text{(Eq. 2.16)}$$

Plugging Equation 2.15 into Equation 2.16 results, after simplification and time non-dimensionalization, in:

$$C_{L_s}(\tau) = -2\pi \frac{w_a(\tau)}{V_\infty} \qquad \text{(Eq. 2.17)}$$

2.6 The Inertial Force

As the airfoil oscillates in the flow, the fluid surrounding the airfoil exerts an inertial reaction on the airfoil due to the accelerated fluid masses. This force, known as the "apparent mass contribution" (see McCune, page 3 and Von Kármán and Sears, page 383), can be computed as follows (Bisplinghoff, page 290):

$$L_i(t) = -\rho \frac{d}{dt} \int_{-c/2}^{c/2} \gamma_0(x,t) x dx \qquad \text{(Eq. 2.18)}$$

The derivation in Appendix A.1 shows it can be inferred from Equation 2.14 that:

$$\gamma_0(x,t) = -2\sqrt{\frac{1+2x/c}{1-2x/c}} w_a(x,t) \qquad \text{(Eq. 2.19)}$$

Plugging Equation 2.19 into Equation 2.18 and performing the following change of variable: $\xi = 2x/c$ results in Equation 2.20. The change of variable is equivalent to non dimensionalizing distance with respect to chord length; the chord length changes from c to 2 and the limits of integration change from '-c/2 to c/2' to '-1 to 1'. Throughout the derivations, the variable x will be used for dimensional distance in meters and ξ will be used for non-dimensional distance in half chord lengths.

$$L_i(t) = \frac{\rho c^2}{2}\frac{d}{dt}w_a(t)\int_{-1}^{1}\sqrt{\frac{1+\xi}{1-\xi}}\xi d\xi \qquad \text{(Eq. 2.20)}$$

Again, the downwash is assumed to not be position dependent and is pulled out of the integral of Equation 2.20.

Equation 2.20 can be rewritten using the non-dimensional time parameter by noting that $dt = \frac{c}{2V_\infty}d\tau$ from Equation 2.6:

$$L_i(\tau) = \rho V_\infty c\,\dot{w}_a(\tau)\int_{-1}^{1}\sqrt{\frac{1+\xi}{1-\xi}}\xi d\xi \qquad \text{(Eq. 2.21)}$$

The integral of Equation 2.21 is solved with the MAPLE program to give:

$$L_i(\tau) = \frac{\rho V_\infty c\pi}{2}\dot{w}_a(\tau) \qquad \text{(Eq. 2.22)}$$

where $\dot{w}_a$, the rate of change of downwash, is given in Equation 2.11.
Equation 2.22 can be non-dimensionalized by introducing the inertial force coefficient:

$$C_{L_i} = \frac{2L_i}{\rho V_\infty^2 c} \qquad \text{(Eq. 2.23)}$$

Plugging Equation 2.22 into Equation 2.23 results, after simplifications, in:

$$C_{L_i}(\tau) = \frac{\pi\dot{w}_a(\tau)}{V_\infty} \qquad \text{(Eq. 2.24)}$$

2.7 The Wake Induced Force

Due to the oscillatory motion of the airfoil, a continuous vortex line is shed in the wake of the airfoil. If the flow is discretized, each time step will correspond to a new vortex point shed in the wake. Figure 2-5 presents how the airfoil wake is discretized. We have stated before that we assume the airfoil to be a flat plate.

However, on Figure 2-5, we decide to show a pitching NACA 0009 airfoil [17] for clarity purposes. Each vortex point will induce a vorticity distribution over the airfoil because of Kelvin's circulation conservation theorem which will in turn produce a force over the airfoil. A more detailed discussion of this phenomenon is given in Von Kármán and Sears [3], Bisplinghoff [15] and McCune [8].

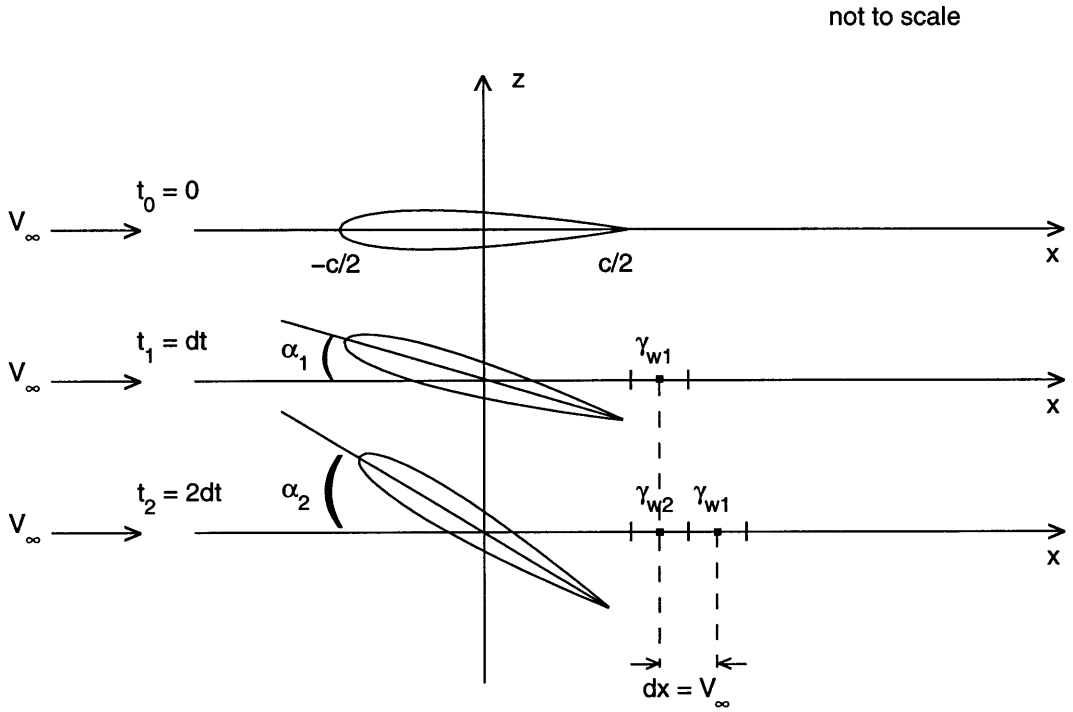

Figure 2-5 Airfoil Wake Discretization

The resulting wake induced force is found from (Bisplinghoff, page 290):

$$L_w(t) = \frac{\rho V_\infty c}{2} \int_{c/2}^{\infty} \frac{\gamma_w(x,t)}{\sqrt{x^2 - (c/2)^2}} dx \qquad \text{(Eq. 2.25)}$$

where L_w is defined perpendicular to the airfoil.

Note from the integral of Equation 2.25 that the effect of the γ_w's is taken into account and as the wake elements travel further downstream, their effect on the airfoil diminishes.

γ_w is determined by solving the wake integral equation, also known as the Wagner equation (Bisplinghoff, page 289):

$$\Gamma_0(t)+\int_{c/2}^{\infty}\sqrt{\frac{x+c/2}{x-c/2}}\gamma_w(x,t)dx=0 \qquad \text{(Eq. 2.26)}$$

One way to solve Equation 2.26 for the γ_w's as a function of time is to differentiate it with respect to time. The first step in doing so is to change the integral of equation 2.26 from a position integral to a time integral by the following change of variable: $x=\frac{c}{2}(1+\tau)$. This change of variables is a statement that vortex elements are convected downstream at a constant rate proportional to the free-stream velocity, so their position is completely determined by the time history of their generation. Time is set to start at the trailing edge, i.e. $\tau=0$ when $x=\frac{c}{2}$. The result is shown in Equation 2.27:

$$\Gamma_0(\tau)+\frac{c}{2}\int_0^{\infty}\sqrt{\frac{2+\tau}{\tau}}\gamma_w(\tau)d\tau=0 \qquad \text{(Eq. 2.27)}$$

Equation 2.27 can now be differentiated with respect to non-dimensional time to give:

$$\dot{\Gamma}_0(\tau)+\frac{c}{2}\sqrt{\frac{2+\tau}{\tau}}\gamma_w(\tau)=0 \qquad \text{(Eq. 2.28)}$$

It is now possible to solve for $\gamma_w(\tau)$ by using Equation 2.14:

$$\gamma_w(\tau)=2\pi\dot{w}_a(\tau)\sqrt{\frac{\tau}{2+\tau}} \qquad \text{(Eq. 2.29)}$$

Before solving for the wake induced force, we need to change the integral of Equation 2.25 from a position to a time integral. As before, through the following change of variable: $x=\frac{c}{2}(1+\tau)$, Equation 2.25 becomes:

$$L_w(\tau) = \frac{\rho V_\infty c}{2} \int_0^\infty \frac{\gamma_w(\tau)}{\sqrt{\tau(\tau+2)}} d\tau \qquad \text{(Eq. 2.30)}$$

L_w is now found by plugging Equation 2.29 into Equation 2.30. After simplifications:

$$L_w(\tau) = \pi \rho V_\infty c \int_0^\infty \frac{\dot{w}_a(\tau)}{2+\tau} d\tau \qquad \text{(Eq. 2.31)}$$

Equation 2.31 can be non-dimensionalized by introducing the wake induced force coefficient:

$$C_{L_w} = \frac{2L_w}{\rho V_\infty^2 c} \qquad \text{(Eq. 2.32)}$$

Plugging Equation 2.31 into Equation 2.32 results, after simplifications, in:

$$C_{L_w}(\tau) = \frac{2\pi}{V_\infty} \int_0^\infty \frac{\dot{w}_a(\tau)}{2+\tau} d\tau \qquad \text{(Eq. 2.33)}$$

2.8 The Suction Force

As mentioned in section 2.1, in steady flow, the resultant steady force is perpendicular to the airflow and no drag is generated. However, in terms of flat plate theory (and that is what the code will assume) the only way to generate a force is by a pressure differential between the upper and lower surface of the plate. The resulting force will therefore be perpendicular to the plate and not to the flow (unless the plate is parallel to the flow, i.e. zero angle of attack). The resulting force will therefore contribute to both a vertical (lift) and horizontal component (drag) and therefore violate the zero drag condition in steady state. This effect is known as the Kutta-Joukowski paradox. To compensate for this unexpected horizontal component, a suction force must be introduced at the leading edge. This force is theoretically justified by the fact that the leading edge is a point of infinite vorticity. (See Von Kármán and Burgers, page 51 and Garrick, page 422). Garrick defines a force, the suction force to be parallel to the airflow and with a magnitude that

exactly cancels the contribution of the lift term in the horizontal direction when the flow is steady. In addition, the contribution of the lift term in the vertical direction and in steady flow must equal the value given by the Kutta-Joukowski theorem.

The mathematical equation of the suction force is (Garrick, page 422):

$$P_x(t) = \pi\rho S(t)^2 \qquad \text{(Eq. 2.34)}$$

where:
$$S = \lim_{x=-c/2} \frac{1}{2}\gamma_a\sqrt{x+c/2} \qquad \text{(Eq. 2.35)}$$

where γ_a, the running bound circulation was defined in section 2.1 to be:

$$\gamma_a = \gamma_0 + \gamma_1. \qquad \text{(Eq. 2.36)}$$

As previously stated, γ_o is called the quasi steady bound circulation and γ_1 is the additional circulation due to the unsteadiness of the flow.

Equation 2.36 is now rewritten after the following change of variable: $x = \frac{c}{2}\xi$.

$$S = \lim_{\xi=-1} \frac{1}{2}\gamma_a\sqrt{\frac{c}{2}}\sqrt{\xi+1} \qquad \text{(Eq. 2.37)}$$

Note that γ_0 as expressed in Equation 2.19 cannot be used to solve Equation 2.35 because the limit as $x \to -c/2$ (i.e. $\xi = -1$) is not properly accounted for. Instead, a more general expression for γ_0 must be used (Bisplinghoff, page 289):

$$\gamma_0(\xi,\tau) = \frac{2}{\pi}\sqrt{\frac{1-\xi}{1+\xi}}\int_{-1}^{1}\sqrt{\frac{1+\xi'}{1-\xi'}}\frac{w_a(\xi',\tau)}{(\xi-\xi')}d\xi' \qquad \text{(Eq. 2.38)}$$

where ξ' denotes a dummy variable of integration representing the non-dimensional position along the chord length. After simplifications, Equation 2.38 becomes:

$$\gamma_0(\xi,\tau) = \frac{2w_a(\tau)}{\pi}\sqrt{\frac{1-\xi}{1+\xi}}\int_{-1}^{1}\sqrt{\frac{1+\xi'}{1-\xi'}}\frac{1}{(\xi-\xi')}d\xi' \qquad \text{(Eq. 2.39)}$$

An expression for γ_1 is given in Von Kármán and Sears, page 381 for the wake circulation, Γ', of a single point vortex:

$$\gamma_1(\xi) = \frac{1}{\pi}\frac{\Gamma'}{\xi' - \xi}\sqrt{\frac{1-\xi}{1+\xi}}\sqrt{\frac{\xi'+1}{\xi'-1}} \qquad \text{(Eq. 2.40)}$$

where ξ is the non-dimensional position along the airfoil and ξ' is the non-dimensional position along the wake. To account for all the wake elements, which for our discretized formulation have a circulation of $\gamma_w d\xi$, we define: $\Gamma' = \int_1^\infty \gamma_w(\xi', \tau) d\xi'$ and the expression for γ_1 becomes:

$$\gamma_1(\xi, \tau) = \int_1^\infty \frac{\gamma_w(\xi', \tau)}{\pi(\xi' - \xi)}\sqrt{\frac{\xi'+1}{\xi'-1}} d\xi' \sqrt{\frac{1-\xi}{1+\xi}} \qquad \text{(Eq. 2.41)}$$

Plugging Equations 2.41 and 2.40 into Equations 2.36 and 2.35 results in (see derivations in Appendix A.3):

$$S(\tau) = -\sqrt{c} w_a(\tau) + \frac{\sqrt{c}}{2\pi}\int_0^\infty \frac{\gamma_w(\tau)}{\sqrt{\tau(\tau+2)}} d\tau \qquad \text{(Eq. 2.42)}$$

The suction force can be obtained by plugging Equation 2.42 into Equation 2.34:

$$P_x(\tau) = \pi\rho c\left(w_a(\tau) - \frac{1}{2\pi}\int_0^\infty \frac{\gamma_w(\tau)}{\sqrt{\tau(\tau+2)}} d\tau \right)^2 \qquad \text{(Eq. 2.43)}$$

Finally, Equation 2.43 can be non-dimensionalized by introducing the suction force coefficient:

$$C_{P_x} = \frac{2P_x}{\rho V_\infty^2 c} \qquad \text{(Eq. 2.44)}$$

Plugging Equation 2.43 into Equation 2.44 results, after simplifications, in:

$$C_{P_x}(\tau) = 2\pi\left(\frac{w_a(\tau)}{V_\infty} - \frac{1}{2\pi V_\infty}\int_0^\infty \frac{\gamma_w(\tau)}{\sqrt{\tau(\tau+2)}}d\tau\right)^2 \qquad \text{(Eq. 2.45)}$$

By plugging Equation 2.29 into Equation 2.45, one obtains:

$$C_{P_x}(\tau) = 2\pi\left(\frac{w_a(\tau)}{V_\infty} - \frac{1}{V_\infty}\int_0^\infty \frac{\dot{w}_a(\tau)}{2+\tau}d\tau\right)^2 \qquad \text{(Eq. 2.46)}$$

Which can be further simplified by using 2.33 to:

$$C_{P_x}(\tau) = 2\pi\left(\frac{w_a(\tau)}{V_\infty} - \frac{C_{L_w}(\tau)}{2\pi}\right)^2 \qquad \text{(Eq. 2.47)}$$

This concludes the presentation of the unsteady forces generated by a flapping airfoil. To summarize, the unsteady lift and thrust forces can be obtained by plugging Equations 2.17, 2.24, 2.33 and 2.47 into Equation 2.5.

Chapter 3

Code Development and Validation

In this chapter, we present the way we implemented the equations developed in Chapter 2 into a computer code. The validity of the code is then tested against published results.

3.1 Matlab Implementation

The supporting computational tool to implement the equations developed in Chapter 2 is Matlab. The code, included in Appendix B, is discussed here in some detail. The code allows the user to calculate the unsteady aerodynamic forces developed over a flapping airfoil for various types of motion. The user commands the reduced frequencies, maximum amplitudes, duty cycles and phase difference for both pitch and plunge as well as the type of motion: sinusoidal or square wave. The program produces the resulting thrust coefficient averaged over one period of motion. Because it is interesting to visualize the motion of the airfoil, the user can also ask the code to create an animation of the airfoil motion. The choice of inputs and outputs is done by executing the main program called code.m. The program code.m will call Simulink routines and Matlab functions to produce the desired output. The Matlab function trailing.m will plot the airfoil motion. Most quantities defined in Chapter 2 are produced by the program forces.m and can be retrieved by using the mapping of Table 3-1. Note that the symbols not included on Table 3-1 are the same in Chapter 2 and the Matlab code.

Table 3-1 Mapping of Symbols from the Notation of Chapter 2 to Matlab

Symbol in Chapter 2	Symbol in Matlab code
α	alpha
$\dot{\alpha}$	alphadot
γ_w	gammaw
C_{L_i}	CFi
C_{L_w}	CFw
C_L	CL
C_{L_s}	CLs
C_{P_x}	CPx
C_T	CT
$C_{T_{ave}}$	CTave
V_∞	V
$\frac{w_a}{V_\infty}$	wav
$\frac{w_a}{V_\infty}$	wadotv
$\dot{z}$	zdot
$\ddot{z}$	zddot

Even though most equations in Chapter 2 are derived to allow a direct implementation in Matlab, a few of them require special attention. In particular the input parameters for periodic flapping motion, α and z, have to be calculated for given reduced frequencies, maximum amplitudes, duty cycles (square motion only) and phase difference between plunge and pitch for both square and sinusoidal motion. Simulink was found to be the most convenient way to produce α and z for the given input. Subsection 3.1.1 describes how the discontinuity introduced by the square wave motion is treated and subsection 3.1.2 describes how the average thrust coefficient calculation is implemented in Matlab.

3.1.1 Treatment of Discontinuities

This subsection describes how the discontinuity introduced by square wave motion is treated. As seen in Equations 2-10 and 2-11, the first derivative of the pitch motion as well as the first and second derivatives of the plunge motion must be computed to calculate the forces experienced over the airfoil. In the case of sinusoidal motion, this does not represent a problem as the derivative of a sinusoidal function is also sinusoidal. However, square functions are non differentiable at the points where the function switches sign. Experience shows that because of the infinite derivatives, the thrust coefficients would themselves tend to infinity with finer step size. Such discontinuities can be avoided by filtering the square wave with a second order transfer function:

$$\frac{\text{output(s)}}{\text{input(s)}} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \qquad \text{(Eq. 3.1)}$$

Figure 3.1 shows a square wave and corresponding filtered motion. For our application it is found acceptable to use a natural reduced frequency of 6 and a damping ratio of 0.707. Acceptability, in this case, is defined as convergence of the thrust coefficient for finer step sizes.

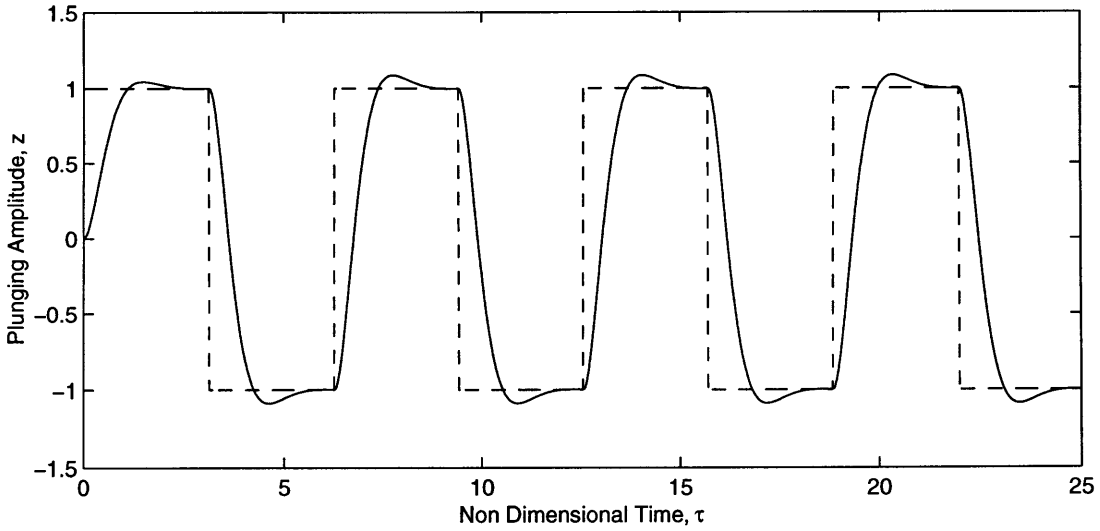

Figure 3-1: Filtered Square Motion Command

3.1.2 Average Thrust Coefficient Computation

This subsection describes how the average thrust coefficient calculation is implemented in Matlab. Equation 2.5 gives a mathematical expression to compute the average thrust coefficient. In this expression, the period T is defined as the time it takes a signal to reproduce itself. If the two input signals are at the same frequency, the period of the airfoil motion is equal to the period of the plunge or pitch motion since they are equal. However, if the two motions have different frequencies, a period of the resulting motion is the product of the two periods. Special cases occur when the input periods are multiples of one another. Consider the general case first, and take y_1 and y_2 to be the pitch and plunge time histories. Then the problem can be stated as follows:

Given: $$y_1 = \sin\left(\frac{2\pi}{T_1}t\right)$$

$$y_2 = \sin\left(\frac{2\pi}{T_2}t\right)$$

$$y = y_1 + y_2$$

Find: T, the period of y

If T_1 and T_2 are integers, the answer is the least common multiple.

If T_1 and T_2 are not integers, we need to find the value of T that will satisfy the sufficient condition for a period:

$$\frac{T}{T_1} \in N \text{ and } \frac{T}{T_2} \in N.$$

The smallest answer is the desirable one and is found to be:

$$T = nT_1 = mT_2, \text{ where } \frac{T_1}{T_2} = \frac{m}{n} \text{ and } (m,n) \in N.$$

Based on this result we see that T_1/T_2 must be rational. Note that if $T_1 = \pi$ (irrational) and $T_2 = 2\pi$ (irrational) you are still fine.

Finally, one can check that the answer satisfies the sufficient condition:

$$\frac{T}{T_1} = \frac{nT_1}{T_1} = n \in N$$

$$\frac{T}{T_2} = \frac{mT_2}{T_2} = m \in N$$

Once the period has been determined, the average thrust coefficient can be calculated by measuring the area under the curve (for example by using the MATLAB function trapz) and dividing by the period.

3.1.3 Airfoil Motion Animation

This subsection discusses the feature implemented in the code that creates an animation of the airfoil motion. The basic steps of how the program works are listed below:

1. Run program code.m and input airfoil motion.
2. code.m calls the animation subroutine trailing.m.
3. Load NACA 0009 data points [17].
4. Calculate and store one period of trailing edge and leading edge motion for the given input.
5. Define and store the chord as the line connecting the leading edge to the trailing edge at each time step.
6. Add and store the airfoil data points to the chord line at each time step. This step requires resampling of the airfoil data points. The spline function of Matlab gives satisfactory results.
7. Use the moviein, getframe and movie functions of Matlab and plot the leading and trailing edge trajectories, and the airfoil.

We show on Figure 3-2 the case where the commanded airfoil motion is shown in Table 3-2.

Table 3-2 Commanded Inputs for the Trajectories Shown on Figure 3-2

Parameter	Value
Motion	Sinusoidal
z_{max}	1 chord length
α_{max}	22 degrees
ϕ	206 degrees
k	0.5
l	0.5

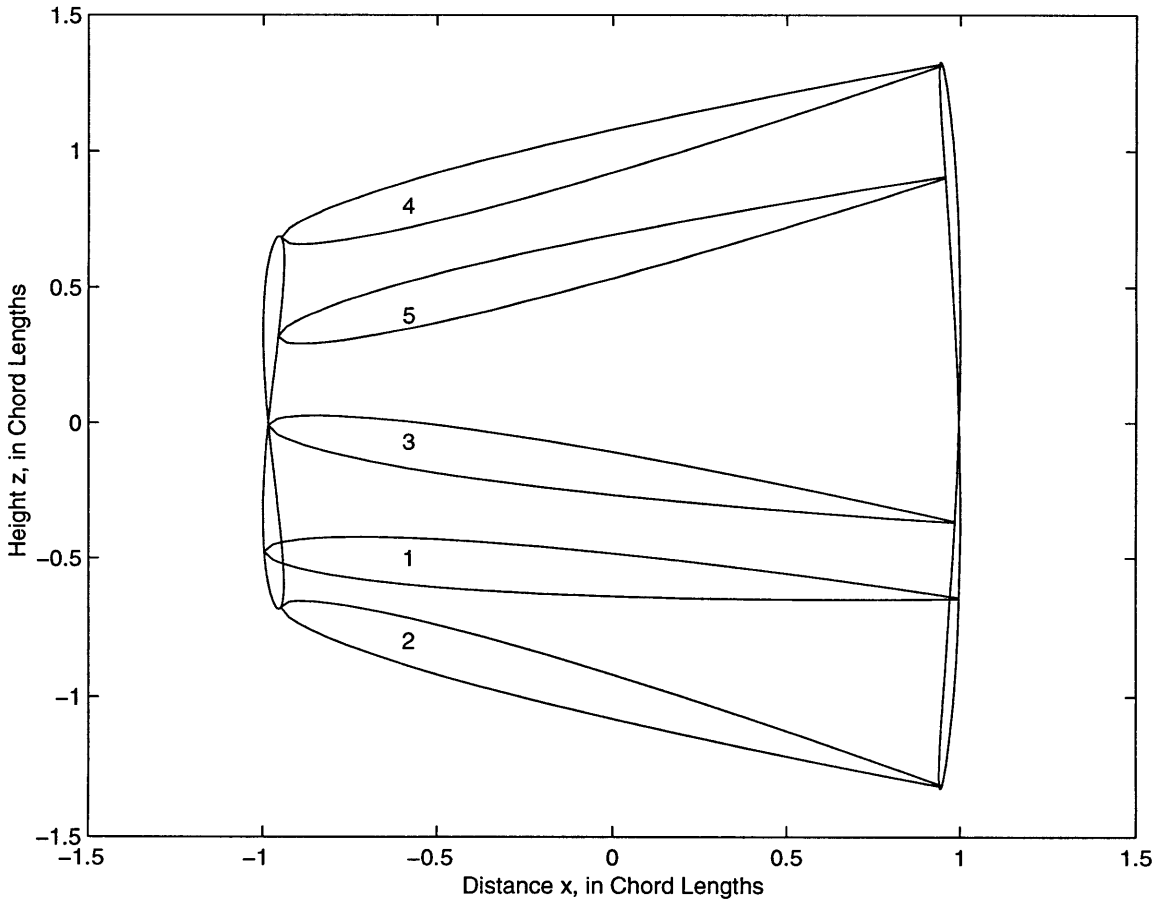

Figure 3-2 Leading Edge and Trailing Edge Trajectories of NACA 0009 Airfoil

3.2 Validation of Unsteady Lift Calculation

The purpose of this section is to validate the code developed against published results. McCune [8] used the equations of Von Kármán [3] and added the non-linear effects due to point vortices in the wake affecting each other. He applied his code to a variety of pitching motions computed lift.

In Figure 3-3, our code is compared with the results of McCune for a step input in pitch of 1 degree applied for 30 non-dimensional time increments. Figure 3-3a shows the individual lift contributions along with the total lift that we obtained. The total lift is recognizable by the triangles denoting each time step. Figure 3-3b compares the total lift of McCune (dashed line) with ours (solid line). Figure 3-3c compares the wake induced force contribution and Figure 3-3d compares the quasi-steady lift contribution. The inertial force contribution is not shown here because it is equal to zero. Recall from Equations 2.24 and 2.11 that the inertial force is a function of the rate of change in angle attack. In the case of a step input, the angle of attack is constant except for a discontinuous jump when the function steps up and down. The quasi steady lift contribution is found to be in perfect agreement with McCune's results. There is a minor discrepancy in the wake induced force, which is also reflected in the total lift plot. Note that McCune's paper includes the non-linear wake force contribution. This contribution is removed from Figure 3-3b to give a more fair comparison.

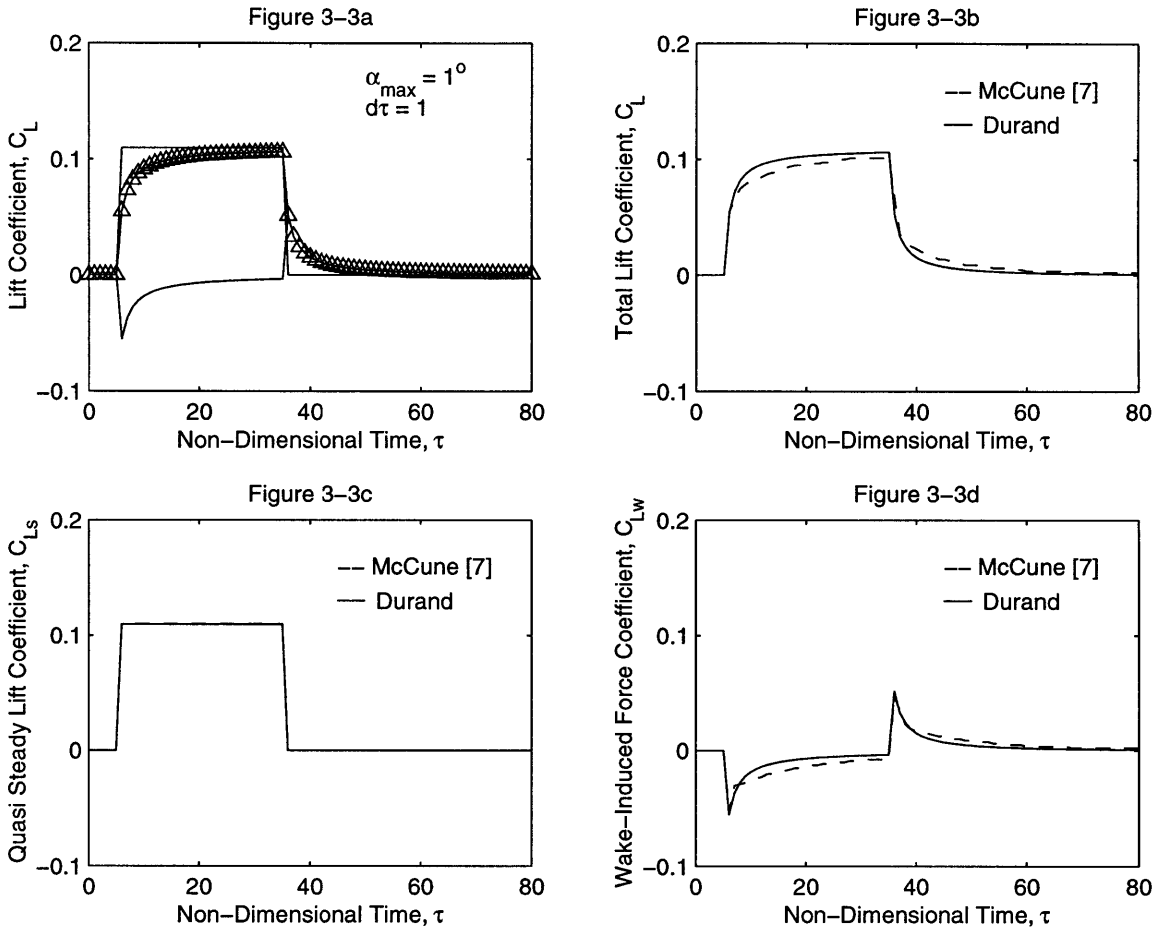

Figure 3-3 Lift Response to a Step Input Angle of Attack of 1 Degree

In Figure 3-4, the code is compared with the results of McCune [8] for one period of oscillation in plunging motion of 20 non-dimensional time steps with maximum amplitude of two degrees. On Figure 3-4a, we plot the individual lift contributions along with the total lift that we obtained. The total lift is recognizable by the triangles denoting each time step. Figure 3-4b compares the total lift of McCune without the non-linear wake force contribution (dashed line) to our total lift (solid line). Also plotted is the total lift of McCune if the contribution of the inertial force (dashed-dot line) is considered to be in the opposite direction. See the discussion of Figure 3-4d to understand why the inertial force was inverted.

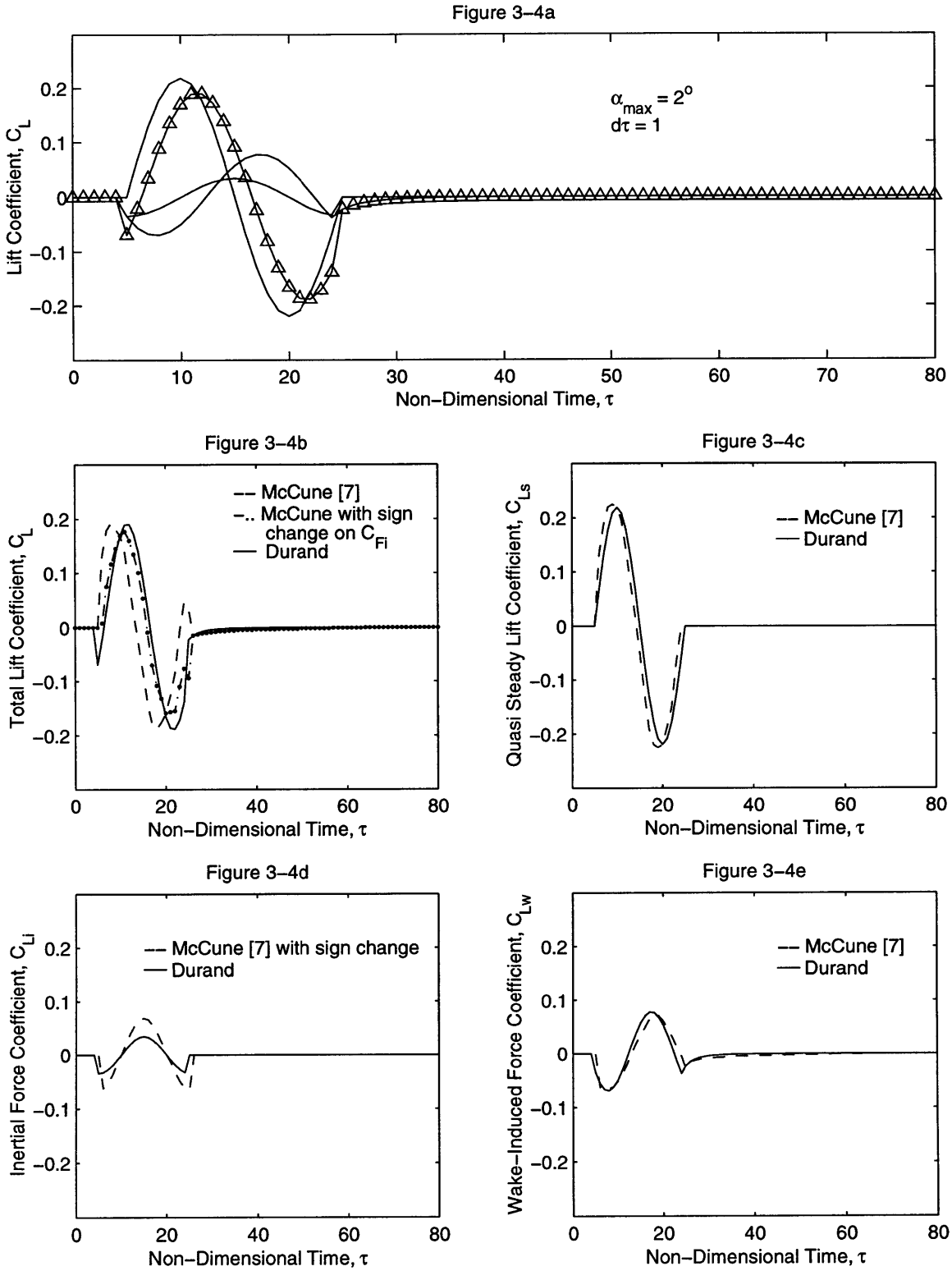

Figure 3-4 Lift Response to a One Period Oscillation in Angle of Attack

Figure 3-4c compares the quasi steady lift contribution, Figure 3-4d the wake induced force contribution and Figure 3-4e compares the inertial force contribution. The quasi steady lift contribution is found to be similar to McCune's in spite of a shift of exactly one time step. This could be due to a difference in the way the equations were implemented in the codes. The inertial force plot shows comparable shapes in spite of lower amplitudes in our result. Note however that the original results showed an opposite sign for the inertial force component. The physical meaning of the inertial force term as worded by McCune himself was: "the contribution to the lift which would occur in unsteady motion even if the airfoil failed to develop any circulation ... is understood as being due to the inertial reaction of the fluid surrounding the airfoil." (McCune, page 3) Based on this definition, and in agreement with the mathematical derivation, the inertial force component is expected to be opposed to the time derivative of downwash, which has units of acceleration (in the case of plunging motion, $\dot{w}_a = \ddot{z}$). If we assume that the air mass is accelerating with the airfoil, then the inertial force must oppose this motion, and therefore be opposite to both $\ddot{z}$ and $\dot{\alpha}$ (see equations 2.24 and 2.11). Based on both the mathematical derivation and this physical insight, we trust our own results to be correct and invert McCune's original curve for comparison purposes.

There is a minor discrepancy in the wake induced force, similar to the one observed in Figure 3.3, which is also reflected in the total lift plot. Also reflected on the total lift plot is the time shift from the quasi steady lift.

Based on the results presented on Figure 3.3 and 3.4 we claim that our code correctly calculates the lift developed by unsteady changes in angle of attack. This includes the quasi-steady lift, the wake induced force and the inertial force contributions.

In Figure 3-5, the code is compared with the results obtained by Hugo [9] for the case of an airfoil oscillating in angle of attack about the mid-chord point with a reduced frequency of 1.5 and maximum amplitude of two degrees. In his paper, Hugo followed the methods of Theodorsen which consisted of expressing the pitching motions in complex variables and resolving the wake integral with Bessel functions. Figure 3-5 shows the plot of the lift coefficient as a function of angle of attack as obtained by Hugo (dashed) compared with our result (solid). Despite a difference in amplitude not in excess of 15%, the two curves show the same trends. This result shows that Theodorsen's [5] approach is also matched.

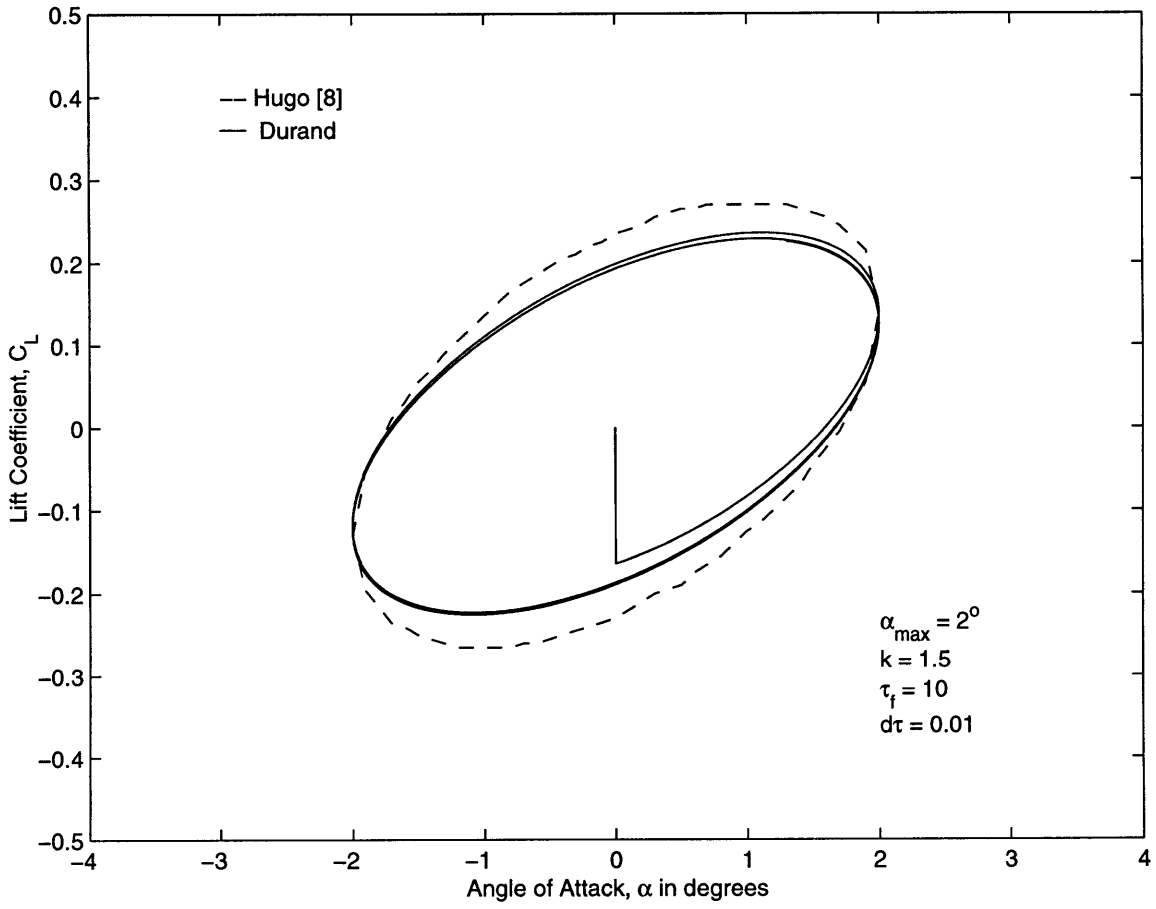

Figure 3-5 Lift Response to Sinusoidal Pitch Motion about Half Chord

3.3 Validation of Unsteady Thrust Calculation

In this section, the ability of the code to compute thrust is tested. Both cases of pure plunge and pure pitch motion are considered and compared with the results of Platzer [11]. In his paper, Platzer developed a panel code to calculate lift and thrust forces. He also considers the wake self-interaction effects, so that the wake is not flat as originally assumed in Section 2.2. Figure 3-6 shows the net thrust (or drag) averaged over one period at steady state for an airfoil sinusoidally oscillating in plunge and pitch over a range of reduced frequencies. The top plot shows very good agreement between our code (solid) and Platzer's (dashed) for plunge motion; this is especially true at lower reduced frequencies. The bottom plot compares the results in pitch motion. Again, a similar trend is obtained. The results shown on Figure 3-6 validate our code for thrust calculations and plunge motions.

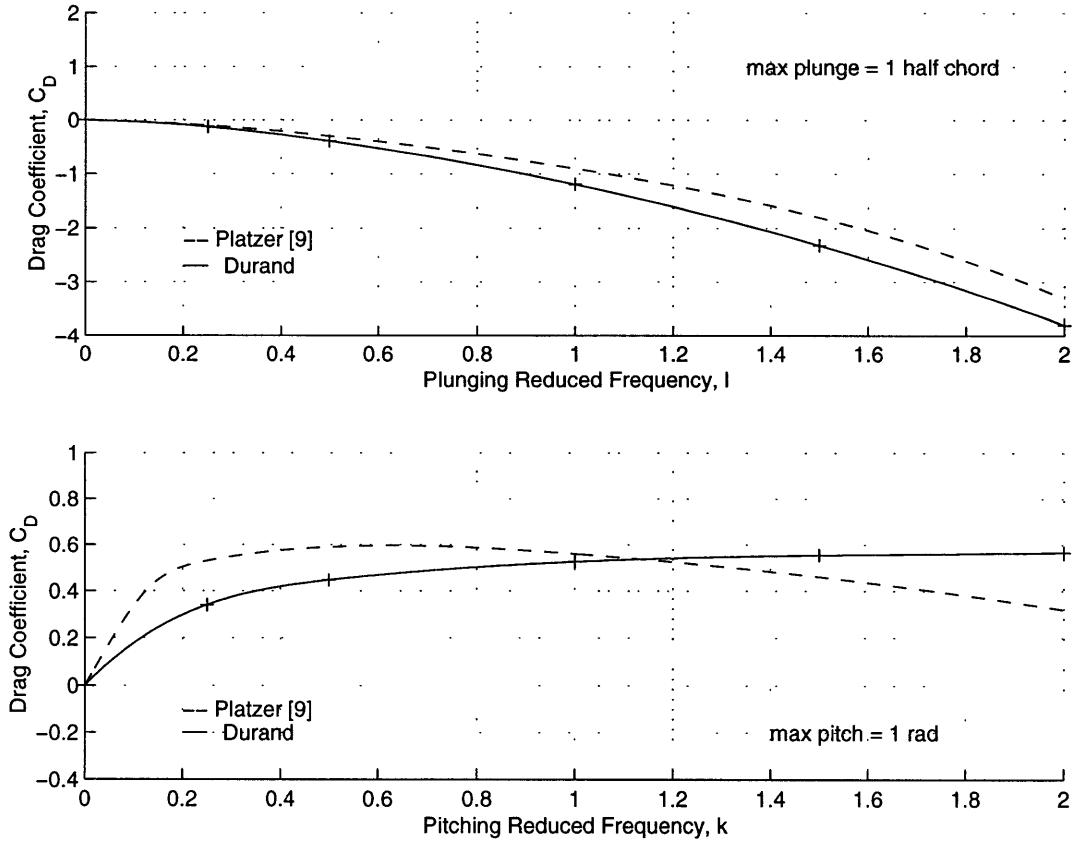

Figure 3-6 Thrust Coefficient Averaged over One Period for Various Pitch and Plunge Flapping Frequencies

Chapter 4

Applications: Optimal Flapping And Transient Thrust

4.1 Introduction

Having validated the code in Chapter 3, in this chapter we demonstrate its capabilities in two applications. First, we use the possibility to input different types of flapping motions to determine which are optimal in Section 4.2. Secondly, we visit the take-off problem and prove that the code is suited to compute transient thrust forces in Section 4.3.

4.2 Optimal Flapping

In this section, the code is used to determine optimal flapping combinations. The criterion for optimality is to maximize thrust. Possible variables to optimize over are as follows: plunging frequency, pitching frequency, plunging amplitude, pitching amplitude, phase delay between pitch and plunge and type of motion (sinusoidal or square). To start, the plunging and pitching frequencies will be set to be equal:

$$l = k. \qquad \text{(Eq. 4.1)}$$

This may seem to be an arbitrary decision, but it makes sense from a mechanical feasibility point of view. The maximum allowable plunge and pitch amplitude will be set to:

$$z_{max} = 1c/2$$
$$\alpha_{max} = 1\text{rad} \qquad \text{(Eq. 4.2)}$$

The choice of setting the maximum plunge amplitude equal to one half chord length results from physical intuition whereas the maximum angle of attack amplitude was dictated by stall conditions. A more realistic value could be obtained by conducting a dynamic stall analysis.

As seen in Figure 3-6, the thrust developed by an airfoil increases monotonically with plunging frequency. Thus no optimal frequency exists and we must choose a frequency based on other considerations. This effect also requires us to fix the plunging frequency; frequency selection is discussed in sub-section 4.2.1. Also, it is found that thrust increases monotonically with plunging amplitude. As a result, the plunging amplitude will be set fixed to its maximum value of one half chord length.

The optimal value of angle of attack is determined in sub-section 4.2.1. The optimal phase difference between the plunging and pitching motions is also computed in sub-section 4.2.1. Finally, a similar study will be conducted for square flapping in sub-section 4.2.2.

4.2.1 Sinusoidal Flapping

In this section, the case of sinusoidal flapping is studied. The optimal flapping frequency is determined first. Then, the optimal phase difference between the pitching and plunging motion, along with the optimal pitching amplitude for maximum plunge is determined.

Optimal Flapping Frequency

Figure 3-5 shows that the thrust generated in plunging motion increases with the flapping frequency. This means that an optimization with no constraint on the reduced frequency will not have a solution. Hall and Hall [12] determined the optimal reduced frequency by considering power requirements and efficiency, and taking into account viscous effects. They found that the optimal frequency is given by:

$$l'_{opt} = \frac{fz_{max}}{\pi V_\infty} \cong 0.3 \qquad \text{(Eq. 4.3)}$$

Note the prime notation to stress that Hall and Hall used different parameters to non-dimensionalized the frequency. In terms of our notation we get:

$$l_{opt} = \frac{fc}{2V_\infty} = \frac{fz_{max}}{\pi V_\infty}\frac{\pi c}{2z_{max}} = l'_{opt}\frac{\pi c}{2c} \cong 0.5 \qquad \text{(Eq. 4.4)}$$

Optimal Plunging to Pitching Phase Difference and Plunging to Pitching Amplitude Ratio

The following simulation is run to determine the optimal plunging to pitching phase difference and plunging to pitching amplitude ratio:

$l = 0.5$

$k = 0.5$

$z_{max} = 1c$

$0° \le \alpha_{max} \le 60°$

$0° \le \phi \le 360^0$

where: $\phi = \left(\dfrac{\text{time } \alpha \text{ crosses zero going up} \; - \; \text{time z crosses zero going up}}{\text{one period}} \right) 360°$

Figure 4-1 shows the contour plot for varying angles of attack and phase differences. The optimal combination occurs when $\alpha_{max} = 28°$ and $\phi = 210°$. Also shown in Figure 4-1 is one period of the optimal flapping motion. Except for a small $(\sim 30°)$ change in phase, we note from Figure 4-1 that optimality is reached when the pitch and plunge motions are opposite.

We note that the thrust generated is sensitive to changes in angle of attack. This sensitivity comes from the presence of the rate of change of downwash ($\dot{w}_a$) in the expression used to calculate the suction force (Eq. 2.11 and 2.46). Such an observation is a hint that motions generating higher rates of change of downwash might produce more thrust. Therefore, square motions will be studied in section 4.3.

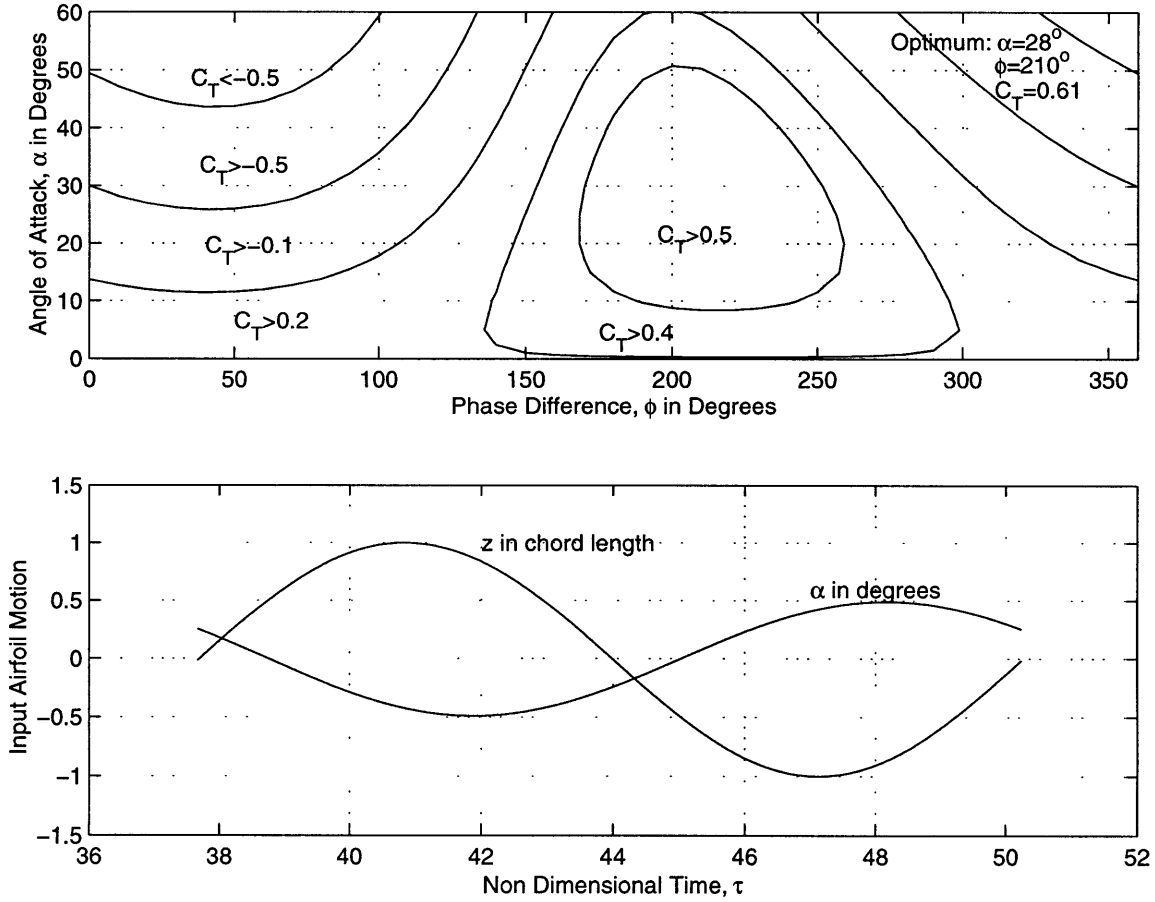

Figure 4-1 Optimal Flapping Phase and Angle of Attack for Sinusoidal Motion

4.2.2 Square Motion

In this sub-section, we determine the optimal input flapping combination when the motion is square. The optimal flapping frequency derived in sub-section 4.2.1 is still applicable, as are the simulation conditions laid out in sub-section 4.2.1.

Figure 4-2 shows the contour plot for varying angles of attack and phase differences. The optimal combination occurs when $\alpha_{max} = 46^{\circ}$ and $\phi = 180^{\circ}$. Also shown in Figure 4-1 is one period of the optimal flapping motion where the notion of 50% duty cycle can be easily understood. As in the sinusoidal case, the optimal solution occurs when the pitch and plunge motions are opposite.

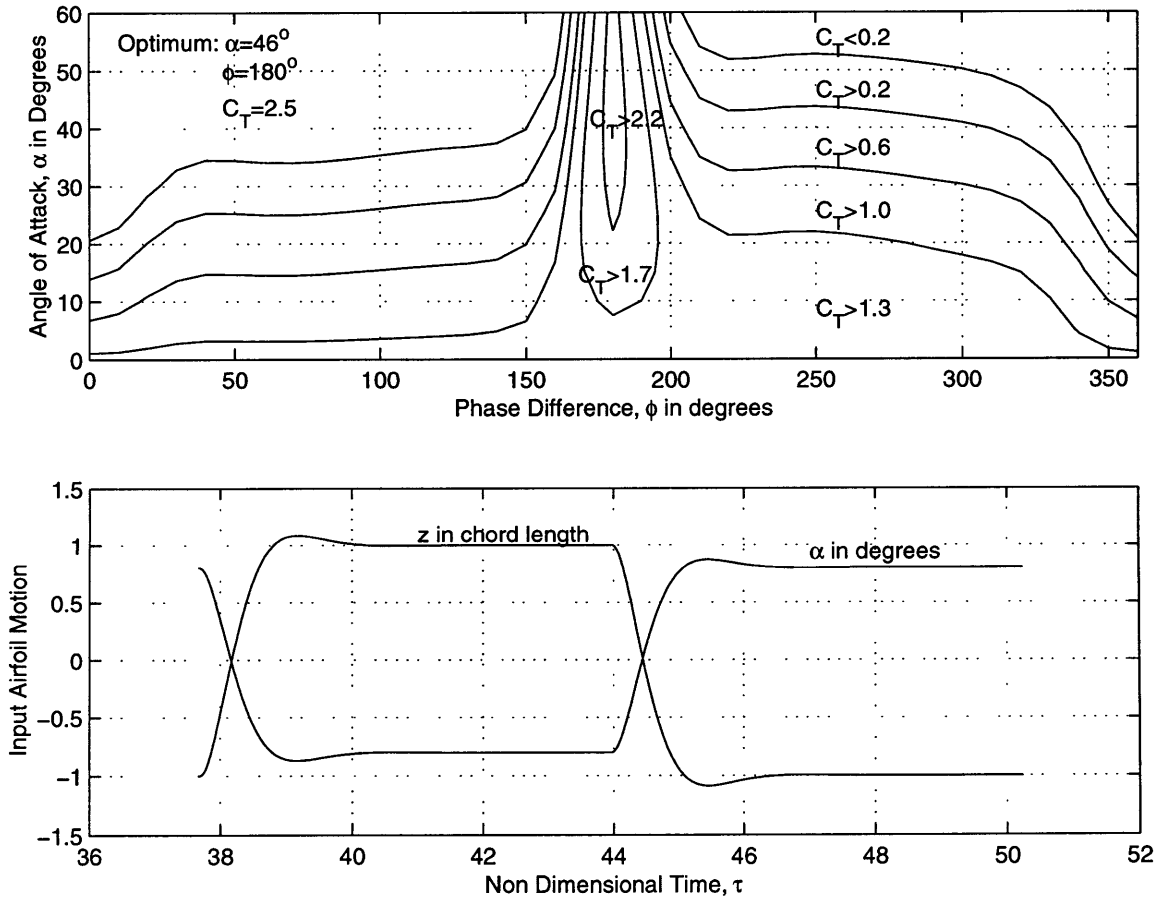

Figure 4-2 Optimal Flapping Phase and Angle of Attack for 50 % Duty Cycle Square Motion

It is also of interest to note from Figure 4-2 that the maximum thrust generated is larger than for the sinusoidal case. This is due to the higher rates of change of pitch and plunge producing higher rate of change of downwash ($\dot{w}_a$) as expected from section 4.2.2. Finally, Figure 4-2 says that square motions are not as 'forgiving' as sinusoidal motion when it comes to deviation from the optimum.

Another question of interest is whether duty cycle has an effect on the amount of thrust generated. For example, would plunging and pitching up and down quickly, and then stopping for the remainder of the oscillation (hence low duty cycle) be more effective than the case studied in Figure 4-2, where you spend as much time up as down? The answer, shown on Figure 4-3, is that the duty cycle has a very weak effect on the amount of thrust generated. A variety of cases are presented in Figure 4-3. The first case determines the optimal flapping combination for pitching and plunging square motions with 10 % duty cycle. The result, as for the sinusoidal and 50 % duty cycle square case, shows that the most thrust is generated when the changes in pitch and plunge occur simultaneously in opposite direction. In the 50 % duty cycle and sinusoidal case, the periods during which thrust is generated are when combinations of plunge up/pitch down and plunge down/pitch up are occurring. However, because the duty cycle is 10% in the first case shown on Figure 4-3, the plunge down/pitch up sequence is the only period of thrust generation. Hence the resulting maximum thrust is reduced by about half in the 10 % duty cycle case when compared to the 50 % case.

The second plot on Figure 4-3 shows what happens when the pitch and plunge motions have 10 % and 90 % duty cycles, respectively. This case presents the same amount of rest time (i.e. time between motion) as the first case shown on Figure 4-3. However, there are twice as many thrust generating periods: plunge down/pitch up and plunge up/pitch down. Not surprisingly, the maximum amount of thrust generated is similar to the one obtain in the 50 % duty cycle case. Note however that the optimum amplitude for angle of attack is higher.

Finally, the third plot on Figure 4-3 shows that the same results are obtained whether the plunging duty cycle is 90 % and pitching duty cycle in 10 % or vice versa. Physically, this means that the same amount of thrust will be generated whether the wings are in the plunge up and pitch down position for 90 % or 10 % of the time . The lift generated, on the other hand, may be significantly higher in one case than the other!

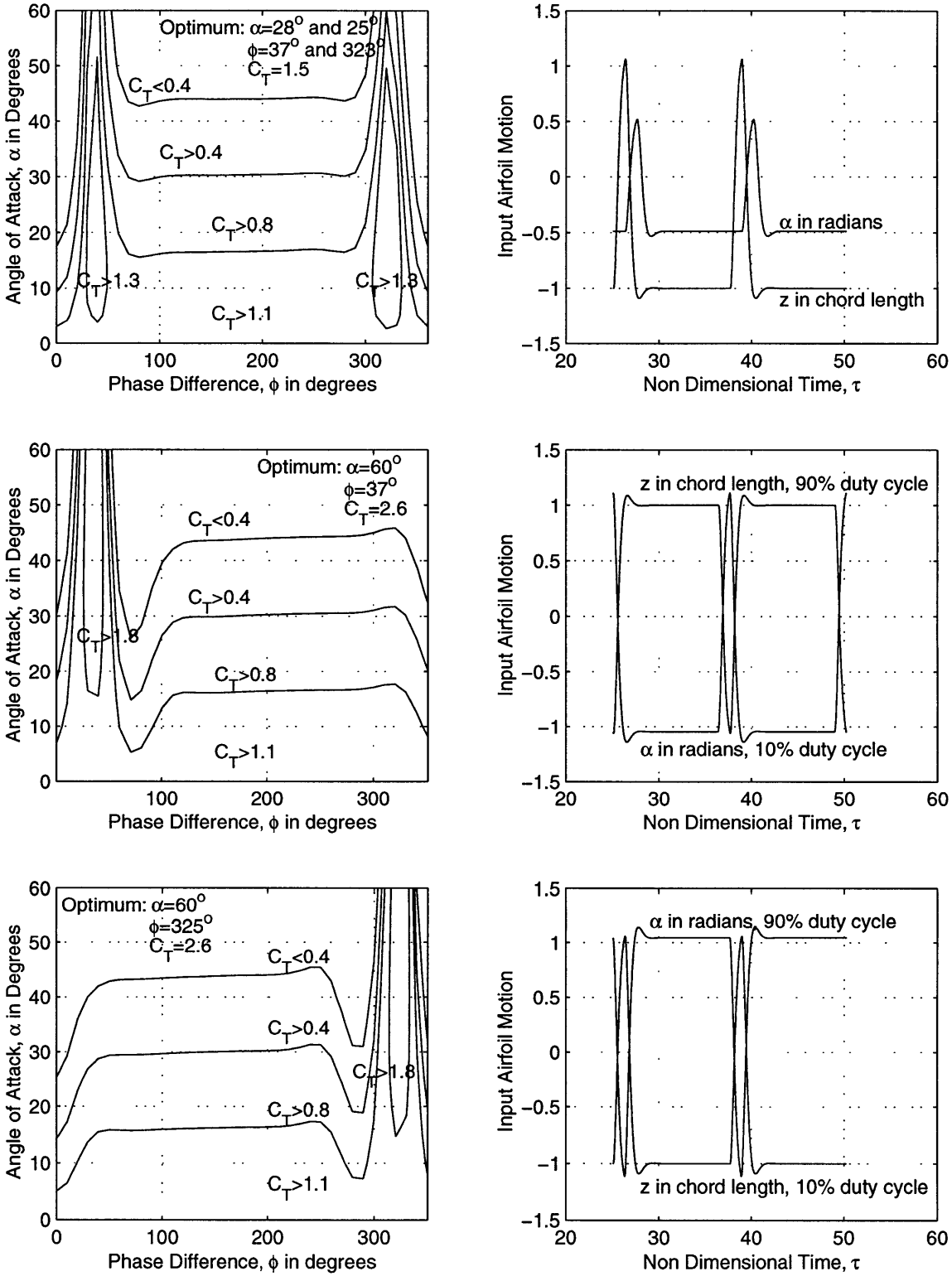

Figure 4-3 Optimal Flapping Phase and Angle of Attack
for 10 % and 90 % Duty Cycles Square Motion

4.4 Take-Off Analysis

In this section, we use the capability of the code to compute transient forces to apply it to the take-off problem. Our goal is to demonstrate that the code can be used in simulations of aircraft dynamics, even if those dynamics involve speed transients. We start, in sub-section 4.4.1, by transforming the non-dimensional equations developed in Chapter 2 to dimensional equations. We then pose, in sub-section 4.4.2, the problem we are considering, and in sub-section 4.4.3 we present our results.

4.4.1 Dimensional Force Equations

The force equations developed in Chapter 2 are non-dimensionalized with respect to time and force. In this application, because we want to study the behavior of an aircraft, we dimensionalize the equations to include the atmospheric effects of velocity, air density and earth gravity, and the physical effects of chord length, wing span and mass. Also, because we allow the velocity to change at each time step, the position at which the wake vortices are shed in the flow changes at each time step. Finally, the force equations of Chapter 2 assumed wing spans of unity and must be modified accordingly.

Equations 2.15 is an expression for the quasi steady force per unit span:

$$L_s(t) = -\rho V_\infty \pi c w_a(t) \qquad \text{(Eq. 4.5)}$$

Multiplying Equation 4.5 by the wing span gives the quasi steady force in Newtons:

$$L_s(t) = -\rho V_\infty(t) \pi c b w_a(t) \qquad \text{(Eq. 4.6)}$$

Equation 2.22 is the inertial force for non-dimensional time and unit span:

$$L_i(\tau) = \frac{\rho V_\infty c\pi}{2} \dot{w}_a(\tau) \qquad \text{(Eq. 4.7)}$$

Equation 4.7 can be time dimensionalized by noting that:

$$\dot{w}_a(\tau) = \frac{dw_a(\tau)}{d\tau} = \frac{dw_a(\tau)}{d\left(\frac{2V_\infty}{c}t\right)} = \frac{c}{2V_\infty}\dot{w}_a(t) \qquad \text{(Eq. 4.8)}$$

Plugging Equation 4.8 into equation 4.7 and correcting for span gives:

$$L_i(t) = \frac{\rho c^2 b\pi}{4} \dot{w}_a(t) \qquad \text{(Eq. 4.9)}$$

Equation 2.31 gives an expression for the wake induced force as a function of non-dimensional time and per unit span:

$$L_w(\tau) = \pi\rho V_\infty c \int_0^\infty \frac{\dot{w}_a(\tau)}{2+\tau} d\tau \qquad \text{(Eq. 4.10)}$$

Dimensionalizing Equation 4.10 with respect to time by using equations 4.8 and 2.6 and correcting for wing span gives:

$$L_w(t) = \frac{\pi\rho V_\infty(t) c^2 b}{2} \int_0^\infty \frac{\dot{w}_a(t)}{c + V_\infty(t)t} dt \qquad \text{(Eq. 4.11)}$$

Equation 4.11 assumes that the wake vortices are shed at constant distances. This assumption is only valid in the non-dimensional case when the force coefficients are not velocity dependents. In the dimensional case, the freestream velocity changes at each time step and therefore the wake vortices are shed at non-constant distance increments. To account for this effect we replace t by $\frac{x(t)}{V_\infty(t)}$ and Equation 4.11 becomes:

$$L_w(t) = \frac{\pi\rho c^2 b}{2} \int_0^\infty \frac{\dot{w}_a\{x(t)/V_\infty\}}{c + x(t)} dx(t) \qquad \text{(Eq. 4.12)}$$

Finally, Equations 2.43 gives an expression for the suction force as a function of non-dimensional time and per unit span:

$$P_x(\tau) = \pi\rho c\left(w_a(\tau) - \frac{1}{2\pi}\int_0^{\infty} \frac{\gamma_w(\tau)}{\sqrt{\tau(\tau+2)}} d\tau \right)^2 \qquad \text{(Eq. 4.12)}$$

Plugging Equation 2.29 into Equation 4.12 reduces to:

$$P_x(\tau) = \pi\rho c\left(w_a(\tau) - \int_0^{\infty} \frac{\dot{w}_a(\tau)}{\tau+2} d\tau \right)^2 \qquad \text{(Eq. 4.13)}$$

As before, Equation 4.13 is dimensionalized with respect to time by using Equations 4.8 and 2.6 and corrected for span length to give:

$$P_x(t) = \pi\rho cb\left(w_a(t) - \frac{c}{2}\int_0^{\infty} \frac{\dot{w}_a(t)}{c+V_\infty t} dt \right)^2 \qquad \text{(Eq. 4.14)}$$

To account for the time varying velocity t is replaced by $\frac{x(t)}{V_\infty(t)}$ and Equation 4.14 becomes:

$$P_x(t) = \pi\rho cb\left(w_a(t) - \frac{c}{2V_\infty(t)}\int_0^{\infty} \frac{\dot{w}_a\{x(t)/V_\infty\}}{c+x(t)} dx(t) \right)^2 \qquad \text{(Eq. 4.15)}$$

We now recall from Equation 2.3 how the forces presented in this sub-section add to the lift and thrust forces:

$$T = P_x - L\sin(\alpha) \qquad \text{(Eq. 4.16)}$$

4.4.2 Problem Statement

The following take-off situation is considered. Consider a vehicle to be launched with an initial velocity, $V_\infty(0)$. We want to look at the average thrust generated by the optimal sinusoidal flapping case determined in Section 4.2.1 for an aircraft climbing at a fixed flight path angle, γ. Our goal is to show that the amount of thrust generated per period goes through a transient phase before reaching a steady value. We wish only to show that the code is capable of calculating thrust due to flapping during this transient period. Based on this consideration, we simplify the problem by not including a drag term in our analysis. Figure 4-4 shows a free body diagram of the problem under consideration.

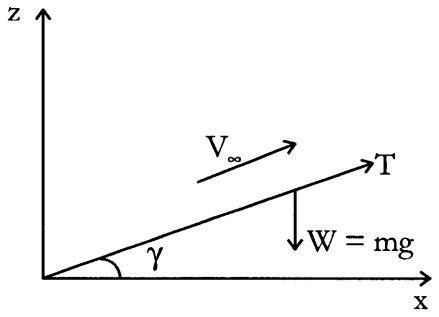

Figure 4-4 Free Body Diagram of the Take-Off Problem

4.4.3 Results and Discussion

We start this section by assigning values to the atmospheric and physical parameters that will be needed to solve the take-off problem. The parameters and their values are shown in Table 4-1.

Table 4-1 Parameters Needed to Solve the Take-Off Problem

Parameter	Value	Rationale
$V_\infty(0)$	4 m/s	Initial velocity for hand launched aircraft.
ρ	1.23 kg/m^3	Air density at sea level.
g	9.8 m/s^2	Gravitational constant.
γ	5 deg	Flight path angle for climbing regime.
c	0.1 m	Typical chord length for a micro-UAV.
b	0.5 m	Typical wing span for a micro-UAV.
C_L	1	Typical lift coefficient for take-off.
m	0.05 kg	$m = \frac{1}{2}\rho V_\infty^2 cbC_L \frac{1}{g}$ for steady-state rectilinear flight.

The following steps outline the procedure used to solve the take-off problem.

Step 1: Initialize the parameters given in Table 4-1 for initial conditions.

Step 2: Compute the thrust force from Equation 4.16.

Step 3: From the free body diagram of Figure 4-4, compute the acceleration in the direction of flight. It can be obtained from Newton's second law: $\sum F = m\dot{v}$. Applied to our problem, we get: $\dot{v} = \frac{T}{m} - g\sin\gamma$.

Step 4: Update the free stream velocity, V_∞ by using a refined Euler method [18].

Step 5: Increment the time by one time step. If time = final time go to step 6; else go to step 2.

Step 6: Compute the average thrust generated for each period. Stop.

The result of the steps outlined for the optimal sinusoidal flapping combination described in Section 4.2.1 and for 11 periods is shown on Figure 4-5.

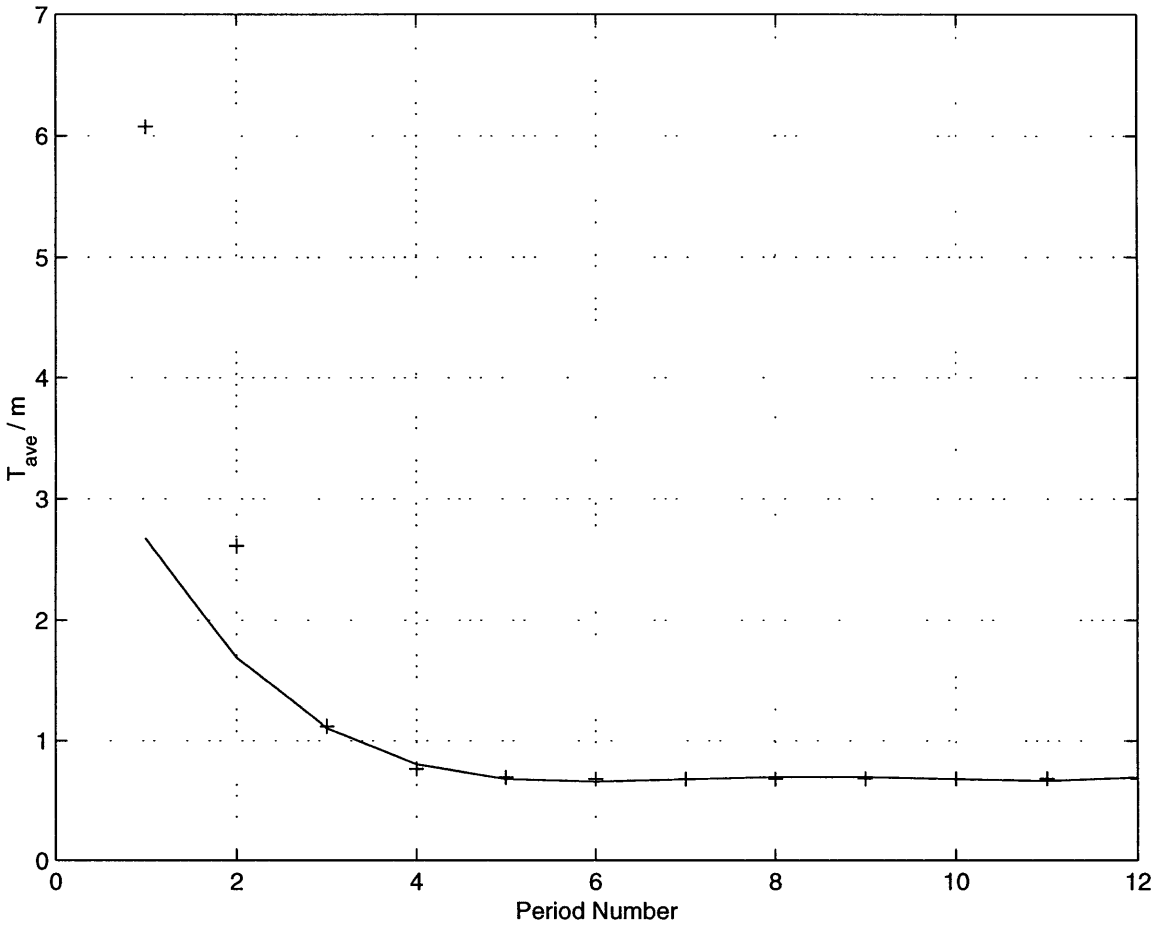

Figure 4-5 Average Acceleration Due to Thrust Showing Transient Effects

As expected, Figure 4-5 shows that the thrust averaged over one period exhibits transient behavior. The transient behavior is due to a combination of the unsteady aerodynamics effects and the aircraft dynamics. The transient phase due to aerodynamics effects does not die out but is not observed on Figure 4-5 because Figure 4-5 shows a time average force. The transient due to the aircraft dynamics is observed in Figure 4-6 where the time history of the velocity is plotted for 12 periods. Also, it is shown in Figure 4-5 that a polynomial of order 4 fits the average acceleration due to thrust after the third period of oscillation. This shows that the transient behavior is stronger in the time immediately following the start of motion.

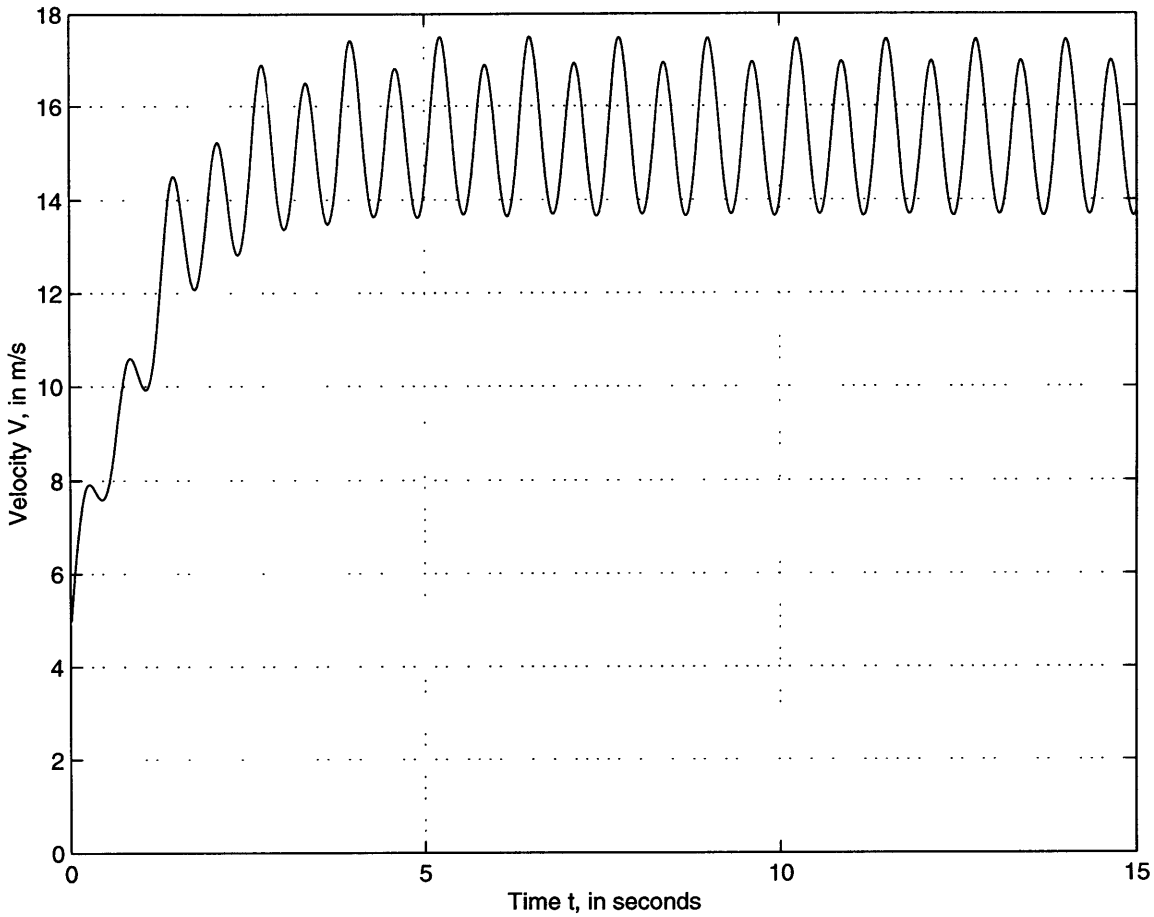

Figure 4-6 Velocity Time History for 12 periods

The convergence of the average acceleration due to thrust can be explained by looking back at the steps listed earlier. At each time step a new value of the thrust force is computed in step 2. If the thrust generated overcomes the effect of gravity, the aircraft will experience a positive acceleration, as seen in step 3. A positive acceleration will produce an increase in velocity (step 5). This increase in velocity will have two effects on the next thrust calculation. The first effect will be to increase the thrust in accordance with the relationship: $T = \frac{1}{2}\rho V_\infty^2 \, b \, c \, C_T$. The second effect is that an increase in velocity is equivalent to a decrease in reduced frequency, as shown by the relationship: $k = \frac{fc}{2V_\infty}$. C_T is, in turn, exponentially dependent on reduced frequency. Hence, the two effects oppose each other until

an equilibrium is reached. This phenomenon explains the convergence of the average acceleration due to thrust.

Chapter 5

Conclusions and Recommendations

5.1 Conclusions

We present in this section concluding remarks regarding the results obtained in the previous chapters. This research has presented how a fast and versatile computer code could be implemented to calculate the unsteady forces generated by a flapping airfoil.

The derivation was based on early work in unsteady aerodynamics and the validity of the code to compute these forces was successfully checked against published results for the lift resulting from step and sinusoidal pitch motions and the thrust resulting from pure pitch and pure plunge motion.

The code was then applied to maximize the thrust produced by a combination of pitch and plunge motions for both sinusoidal and square waves. Table 5-1 compares the optimal cases for sinusoidal and square waves for pitching and plunging reduced frequencies of 0.5, plunge amplitude of one half a cord length and pitch amplitude ranging from 0 to 60 degrees. In Section 4.2 we explain why these parameters were fixed.

Table 5-1 Optimal Flapping Configurations

Type of Motion	Duty Cycles		Optimized Parameters		$C_{T_{max}}$
	Pitch	Plunge	Pitch Amplitude	Phase Difference	
Sinusoidal	N/A	N/A	28 deg.	210 deg.	0.6
Square	50 %	50 %	46 deg.	180 deg.	2.5
Square	10 %	10 %	28 deg. and 25 deg.	37 deg. and - 37 deg.	1.5
Square	10 %	90 %	37 deg.	60 deg.	2.6
Square	90 %	10 %	- 35 deg.	60 deg.	2.6

We find that all optimal cases correspond to cases where the pitch and plunge motion are simultaneous and opposite; this can be seen by looking at the phase differences between pitch and plunge. We also find in Table 5-1 that square motions produce more thrust than sinusoidal ones. This is due to the higher pitch and plunge rates. However, in the square case, departures from the optimum phase degrade more rapidly than in the sinusoidal case. Also, the optimal sinusoidal case requires lower amplitudes of motion. Another interesting fact shown in Table 5-1 is that duty cycles do not have to be equal to 50 %. Indeed, similar thrust coefficients to the 50 % duty cycle case are obtained with duty cycles of 90 % and 10 %. Again, the important feature is that the pitch and plunge motions are simultaneous and opposite.

The second application for the code was to study the take-off problem faced by a micro-UAV to show the ability of the code to compute transient forces. We find that for a given set of initial conditions, a hand launched micro-UAV will exhibit transient behavior due to a combination of the unsteady aerodynamics effects and the aircraft dynamics.

5.2 Recommendations

Research is an open ended subject. It is always a researcher's desire to see his or her work expanded and improved. This is the reason why in this section we propose areas of the work presented subject to improvement as well as possible future applications made possible or easier by the development of the code included in Appendix B. We also mention directions that we have explored to a certain extent and did not seem to lead to improvements to our work.

The first application we studied in Section 4.2 dealt with optimal flapping. In addition to the results presented, we investigated a variety of other flapping combinations with no improvement over the square case. They included triangular wave, combinations of sinusoidal and square waves and step waves. Higher thrust coefficients may be obtained by rotating the airfoil about a different point than the mid-chord. There is also evidence that placing a second airfoil in the wake of the first one dramatically increases the efficiency of the thrust produced. As far as improving the accuracy of the results obtained for optimal flapping, one may develop a more physically realistic model for the filter used for the square wave as discussed in sub-section 3.1.1. Also, the flat wake assumption can be lifted by tracking the points where the wake vortices are shed.

Our second application showed the ability of the code to compute transient forces for a micro-UAV in the climbing regime. A more rigorous analysis can be conducted by using more realistic geometric data. Among other applications of interest is the landing regime or how to flap the wings to bring the velocity of the aircraft to nearly zero.

Finally, experimental data would be a perfect way to study how our analysis expands to three-dimensional viscous flow. There is no doubt that the physical implementation of the optimal flapping combinations that we proposed opens

another vast area of research. This is required to make the flight of a flapping aircraft a reality.

References

[1] Wagner, H., "Dynamischer Auftrieb von Tragflügeln", *Zeitschrift für Angewandte Mathematik und Mechanik*, Vol. 5, 1925, p.17.

[2] Durand, W. F., Von Kármán, T. and Burgers, J. M., *Aerodynamic Theory*, Vol. II, Division E, Julius Springer, Berlin, 1935.

[3] Von Kármán, T. and Sears, W. R., "Airfoil Theory for Non-Uniform Motion", *Journal of Aeronautical Science*, Vol. 5, No. 10, pp. 379-390, August 1938.

[4] Garrick, I. E., "Propulsion of a Flapping and Oscillating Airfoil", NACA Report 567, 1936.

[5] Theodorsen, T., "General Theory of Aerodynamic Instability and the Mechanism of Flutter", NACA Report No. 496, 1935.

[6] Knoller, R., "Die Gesetze des Luftwiderstandes", Flug und Motortechnik (Vienna), Vol. 3, No. 21, pp. 1-7, 1909.

[7] Betz, A., "Ein Beitrag zur Erklaerung des Segelfluges", Z. f. Flugtechnik und Motorluftschiffahrt, Vol. 3, pp. 269-272, 1912.

[8] McCune J. E., Lam, C.-M. G. and Scott M. T., "Nonlinear Aerodynamics of Two-Dimensional Airfoils in Severe Maneuver", *AIAA Journal*, Vol. 28, No. 3, pp. 385-393, March 1990.

[9] Hugo, R. and Jumper, E., "Controlling Unsteady Lift Using Unsteady Trailing-Edge Flap Motions", AIAA Paper 92-0275, January 1992.

[10] Jumper, E. and Hugo, R., "The Loading Characteristics of Finite Wings Undergoing Rapid Unsteady Motions: A Theoretical Treatment", AIAA Paper 91-3263, September 1991.

[11] Platzer, M. F., Neace, K. S. and Pang, C.-K., "Aerodynamic Analysis of Flapping Wing Propulsion", AIAA Paper 93-0484, January 1993.

[12] Hall, K. C., Pigott, S. A. and Hall, S. R., "Power Requirements for Large-Amplitude Flapping Flight", AIAA Paper 97-0827, January 1997.

[13] Kraemer, C. C. and Hanaki, S. Y., "Improving Thrust Characteristics in a Heaving Airfoil System Using Active Vortex Control", Statement of Progress, Spring 1995.

[14] Bosch, H., "Interfering Airfoils in Two-dimensional Unsteady Incompressible Flow", AGARD CP-227, Paper No. 7, September 1977.

[15] Bisplinghoff, R. L., Ashley, H. and Halfman, R. L., A*roelasticity*, Addison-Wesley Publishing Co., 1955.

[16] Anderson, John D., Jr., *Fundamentals of Aerodynamics*, 2d ed., McGraw-Hill Book Company, New York, 1991.

[17] Abbott, I. H., and A. E. von Doenhoff, *Theory of Wing Sections*, McGraw-Hill Book Company, 1949; also Dover Publications, Inc., New York, 1959.

[18] Braun, M., *Differential Equations and Their Applications*, Vol. 15, 2d ed., Springer-Verlag, New York, 1975.

Appendix A

Mathematical Derivations

A.1 Analytic Solution of the Airfoil Bound Circulation, Γ_0 and Vorticity Distribution, $\gamma_0(x,t)$:

Bisplinghoff, page 289 gives (also Equation 2.13):

$$\Gamma_0(t) = \int_{-\frac{c}{2}}^{\frac{c}{2}} \gamma_0(x,t)dx = -c\int_{-1}^{1}\sqrt{\frac{1+\xi^*}{1-\xi^*}}w_a(\xi^*,t)d\xi^* \qquad \text{(Eq. A.1)}$$

where, $\xi^* = \frac{2\xi}{c}$ and ξ is a position dummy variable of integration in Bisplinghoff.

If we now return to our notation and for w_a constant for all x (i.e. along the chord):

$$\Gamma_0(t) = -c\, w_a(t)\int_{-1}^{1}\sqrt{\frac{1+x}{1-x}}dx \qquad \text{(Eq. A.2)}$$

And finally using Equation A.12:

$$\Gamma_0(t) = -w_a(t)c\pi \qquad \text{(Eq. A.3)}$$

After the change of variable: $\xi^* = 2x/c$; Equation A.1 may be rewritten as:

$$\Gamma_0(t) = -c\int_{-c/2}^{c/2}\sqrt{\frac{1+2x/c}{1-2x/c}}w_a(x,t)\frac{2dx}{c} = \int_{-c/2}^{c/2} -2\sqrt{\frac{1+2x/c}{1-2x/c}}w_a(x,t)dx \qquad \text{(Eq. A.4)}$$

Now, comparing Equations A.1 and A.4 yields the airfoil bound vorticity distribution:

$$\gamma_0(x,t) = -2\sqrt{\frac{1+2x/c}{1-2x/c}}w_a(x,t) \qquad \text{(Eq. A.5)}$$

A.2 Computation of $\int_{-1}^{1}\sqrt{\frac{1+x}{1-x}}dx = I$

From Schaum's Mathematical Handbook, page 64:

$$\int\sqrt{\frac{px+q}{ax+b}}dx = \frac{\sqrt{(ax+b)(px+q)}}{a} + \frac{aq-bp}{2a}\int\frac{dx}{\sqrt{(ax+b)(px+q)}} \qquad \text{(Eq. A.6)}$$

With a=-1, b=1, p=1 and q=1,

$$I = \left.\frac{\sqrt{(-x+1)(x+1)}}{-1}\right|_{-1}^{1} + \frac{-1-1}{2(-1)}\int_{-1}^{1}\frac{dx}{\sqrt{(-x+1)(x+1)}} \qquad \text{(Eq. A.7)}$$

After simplifications:

$$I = \int_{-1}^{1}\frac{dx}{\sqrt{(x+1)(-x+1)}} \qquad \text{(Eq. A.8)}$$

From Schaum's Mathematical Handbook, page 63:

$$\int\frac{dx}{\sqrt{(ax+b)(px+q)}} = \frac{2}{\sqrt{-ap}}\tan^{-1}\sqrt{\frac{-p(ax+b)}{a(px+q)}} \quad \text{for } ap<0 \qquad \text{(Eq. A.9)}$$

With a=1, b=1, p=-1 and q=1:

$$I = \frac{2}{\sqrt{1}} \tan^{-1} \sqrt{\frac{1(x+1)}{1(-x+1)}}\Bigg|_{-1}^{1} \qquad \text{(Eq. A.10)}$$

After simplifications:

$$I = \tan^{-1} \sqrt{\frac{x+1}{-x+1}}\Bigg|_{-1}^{1} = 2\tan^{-1}(\infty) - 2\tan^{-1}(0) \qquad \text{(Eq. A.11)}$$

Conclusion:

$$I = \int_{-1}^{1} \sqrt{\frac{1+x}{1-x}} dx = \pi \qquad \text{(Eq. A.12)}$$

A.3 Computation of S the suction coefficient

Plugging Equations 2.35 and 2.33 into Equation 2.31 and 2.30 results in:

$$S = \lim_{\xi=-1} \frac{1}{2} \sqrt{\frac{c}{2}} \sqrt{\xi+1} \left[\begin{array}{l} \frac{2w_a}{\pi} \sqrt{\frac{1-\xi}{1+\xi}} \int_{-1}^{1} \sqrt{\frac{1+\xi'}{1-\xi'}} \frac{1}{(\xi-\xi')} d\xi' \\ + \frac{1}{\pi} \sqrt{\frac{1-\xi}{1+\xi}} \int_{1}^{\infty} \frac{\gamma_w(\xi',t)}{(\xi'-\xi)} \sqrt{\frac{\xi'+1}{\xi'-1}} d\xi' \end{array} \right] \qquad \text{(Eq. A.13)}$$

Canceling the $\sqrt{\xi+1}$ term and substituting -1 for ξ gives:

$$S = \frac{-\sqrt{c} w_a}{\pi} \int_{-1}^{1} \sqrt{\frac{1+\xi'}{1-\xi'}} \frac{1}{(1+\xi')} d\xi' + \frac{\sqrt{c}}{2\pi} \int_{1}^{\infty} \frac{\gamma_w(\xi',t)}{(\xi'+1)} \sqrt{\frac{\xi'+1}{\xi'-1}} d\xi' \qquad \text{(Eq. A.14)}$$

After simplifying the radical expressions:

$$S = \frac{-\sqrt{c} w_a}{\pi} \int_{-1}^{1} \frac{1}{\sqrt{1-\xi'^2}} d\xi' + \frac{\sqrt{c}}{2\pi} \int_{1}^{\infty} \frac{\gamma_w(\xi',t)}{\sqrt{\xi'^2-1}} d\xi' \qquad \text{(Eq. A.15)}$$

The first integral has a closed form solution and is evaluated with MAPLE. Equation A.15 becomes:

$$S = -\sqrt{c}w_a + \frac{\sqrt{c}}{2\pi}\int_1^{\infty} \frac{\gamma_w(\xi',t)}{\sqrt{\xi'^2 - 1}} d\xi' \qquad \text{(Eq. A.16)}$$

The wake integral can be changed from a non-dimensional position integral to a non-dimensional time integral with the change of variables: $\xi = 1 + \tau$

$$S = -\sqrt{c}w_a + \frac{\sqrt{c}}{2\pi}\int_0^{\infty} \frac{\gamma_w(\tau)}{\sqrt{\tau(\tau+2)}} d\tau \qquad \text{(Eq. A.17)}$$

Appendix B

Matlab Code

The files included in this appendix are matlab and simulink functions. The files are:

code.m	This is the first file to run, to enter inputs and choose outputs.
simsin.mdl	This is a simulink block diagram that will be executed if the input sinusoidal motion is selected in code.m. It computes alpha, alphadot, z, zdot and zddot.
simsquare.mdl	This is like simsin.mdl when the input square motion is selected.
forces.m	This is a matlab function that will compute the forces experienced by the airfoil as described in Chapter 2.

```
%%%%%%%%%%%%%%%%%%%% CODE.M %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  This is the first file to run to enter inputs and choose outputs       %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all

%% select sinusoidal or square motion

motion2=input('Choose type of motion: 1=sinusoidal, 2=square: ');

if      motion2==1

        %% sinusoidal selected
        %% select frequencies, max amplitudes and phase difference

        k=input('Enter pitching reduced frequency: ');
        l=input('Enter plunging reduced frequency: ');
        alphamax=input('Enter maximum pitching amplitude in degrees: ');
```

```
zmax=input('Enter maximum plunging amplitude in half chord lengths: ');

phi=input('Enter phase between plunge and pitch in degrees: ');
ds=0.01*pi/min(k,l);
sf=8*pi/min(k,l)
sft=6*pi/min(k,l)

%% select desired output

out=input('Enter desired output: 1=average thrust coefficient only,
2=average thrust coefficient and trailing edge motion, 3=trailing edge
motion only: ');

if      out==1

        %% sinusoidal, CTave only: start simulink program simsin and matlab
        %% program forces

        sim('simsin')
        forces
elseif  out==2

        %% sinusoidal, CTave and airfoil motion: start simulink program
        %% simsin, matlab program forces, simulink program trailsin and
        %% matlab program trailing

        sim('simsin')
        forces
        sim('trailsin')
        trailing
elseif  out==3

        %% sinusoidal, airfoil motion only: start simulink program trailsin
        %% and matlab program trailing

        sim('simsin')
        trailing
end

elseif  motion2==2

        %% square selected
        %% select frequencies, max amplitudes, duty cycles and phase difference

        k=input('Enter pitching reduced frequency: ');
        l=input('Enter plunging reduced frequency: ');
        alphamax=input('Enter maximum pitching amplitude in degrees: ');
        zmax=input('Enter maximum plunging amplitude in half chord lengths: ');
```

```
        dutyalpha=input('Enter pitching duty cycle in %: ');
        dutyz=input('Enter plunging duty cycle in %: ');
        phi=input('Enter phase between plunge and pitch in degrees: ');
        wn=3; ksi=0.707;
        ds=0.002*pi/min(k,l);
        sf=8*pi/min(k,l);
        sft=6*pi/min(k,l);

        sim('simsquare')
        forces

end
```

```
%%%%%%%%%%%%%%%%  SIMSIN.MDL  %%%%%%%%%%%%%%%%%%%%%%%%
%% This simulink block diagram is executed if the sinusoidal input motion is  %%
%% selected in code.m.  It computes alpha, alphadot, z, zdot and zddot.        %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

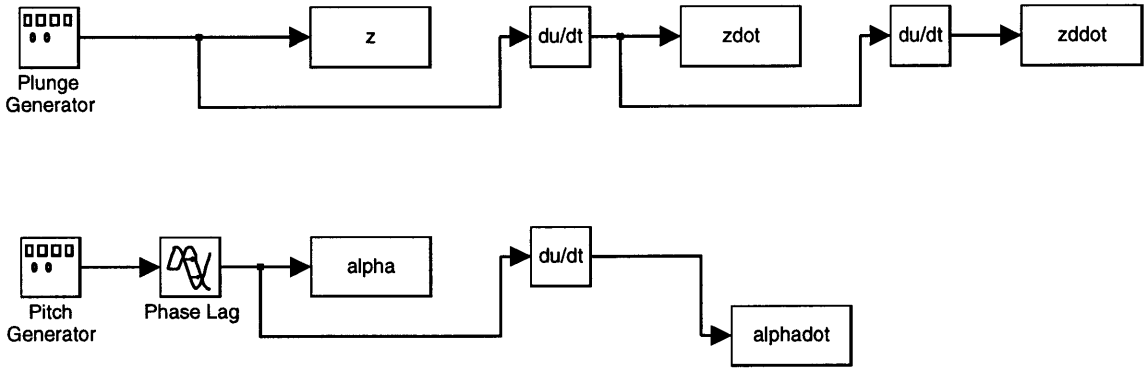

Figure F-1: Simulink Block Diagram for Sinusoidal Motion

Table F-1: Definition of the Blocks shown in Figure F-1

Block Diagram Name	Characteristic	Value
Parameters	Start time Stop time Type Fixed step size	0 sf Fixed-step Discrete ds
Plunge Generator	Name Wave form Amplitude Frequency Units	Signal Generator Sine zmax+0.0001 1 rad/sec
Pitch Generator	Name Wave form Amplitude Frequency Units	Signal Generator Sine alphamax*pi/180+0.0001 k rad/sec
Phase Lag	Name Time delay	Transport Delay pi*phi/180/k

```
%%%%%%%%%%%%%%%%%%%%%%%%%%  SIMSQUARE.MDL  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% This is like simsin.mdl when the input square motion is selected.      %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

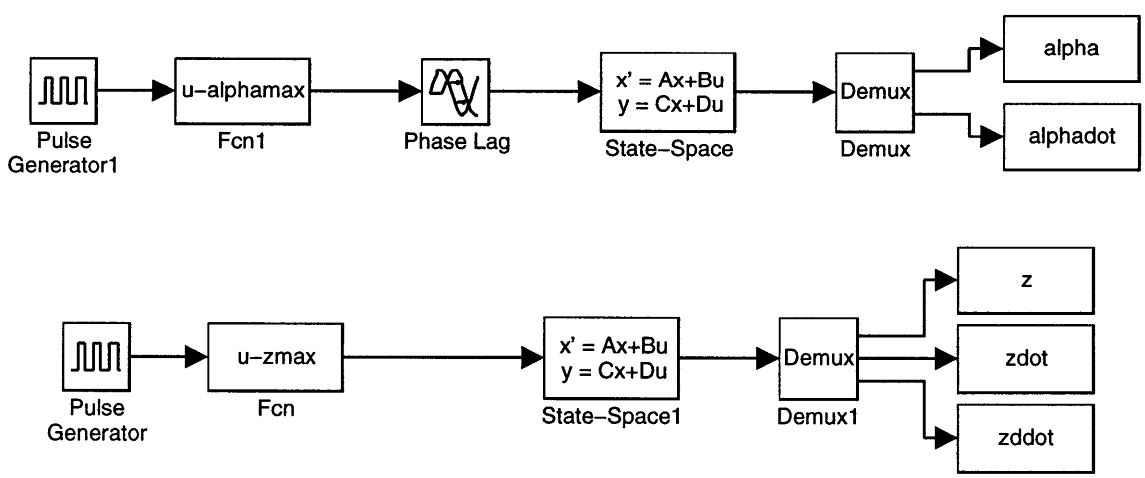

Figure F-2: Simulink Block Diagram for Square Motion

Table F-2: Definition of the Blocks shown in Figure F-2

Block Diagram Name	Characteristic	Value
Parameters	Start time Stop time Type Fixed step size	0 sf Fixed step ode5 ds
Plunge Generator	Name Period (secs) Duty cycle (% of period) Amplitude	Pulse Generator 2*pi/l dutyz zmax*2+0.0001
Pitch Generator	Name Period (secs) Duty cycle (% of period) Amplitude	Pulse Generator 2*pi/k dutyalpha alphamax*pi/180*2+0.0001
Fcn1	Name Expression	Function u-alphamax*pi/180
Fcn	Name Expression	Function u-zmax
State Space	Name A B C D	State Space [0 1; -wn^2 -2*wn*ksi] [0; wn^2] [1 0; 0 1] [0; 0]
State Space1	Name A B C D	State Space [0 1; -wn^2-2*wn*ksi [0; wn^2] [1 0; 0 1; -wn^2 -2*wn*ksi] [0; 0; wn^2]
Phase Lag	Name Time delay	Transport Delay pi*phi/180/k

```
%%%%%%%%%%%%%%%% FORCES.M %%%%%%%%%%%%%%%%%%%%%%%%
%% This is a matlab function that will compute the forces experienced by the  %%
%% airfoil as described in Chapter 2.                                          %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

start=cputime;

V=1;

s=(0:ds:sf)';

% wa (Eq. 2.7);
wav=-sin(alpha) - zdot.*cos(alpha)/V;

% wadot (Eq. 2.8)
wadotv=-alphadot.*cos(alpha) - ...
(zddot.*cos(alpha)-zdot.*sin(alpha).*alphadot)/V;

% CLs (Eq. 2.14)
CLs=-2*pi*wav;

% gammaw (Eq. 2.24)
gammaw=2*pi*sqrt(s./(2+s)).*wadotv;

% CFw (Eq. 2.27)
X=1./(s+2);
for     n=1:sf/ds+1
        CFw(n)=2*pi*wadotv(1:n)'*X(n:-1:1)*ds;
end
CFw=CFw';

% CFi (Eq. 2.21)
CFi=pi*wadotv;

% CPx (Eq. 2.41)
CPx=2*pi*(wav-CFw/(2*pi)).^2;

% CL (Eq. 2.2)
CL=CLs+CFw+CFi;

% CT (Eq. 2.2)
CT=CPx-(CLs+CFw+CFi).*sin(alpha);

CTperiod=CT(round(length(s)-2*T/ds):length(s));
CLperiod=CL(round(length(s)-2*T/ds):length(s));
period=[s(length(s))-2*T:ds:s(length(s))];
ctint=1/2/T*trapz(period, CTperiod);
clint=1/2/T*trapz(period, CLperiod);
```

```
stop=cputime;
run=stop-start;

info=['CT = ', num2str(ctint), '; sf = ', num2str(sf), ';  ds = ', num2str(ds), ';
cputime =  ', num2str(run), ' seconds']



%%%%%%%%%%%%%% TRAILING.M %%%%%%%%%%%%%%%%%%%%%%
%% This is a matlab function that will plot the airfoil motion in a movie style  %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

start=cputime;

load naca009

s=0:ds:sf;

x=naca009(:,1)/50-1;
xtr=cos(alpha);
ztr=sin(-alpha)+z;
xl=-cos(alpha);
zl=sin(alpha)+z;

zu=naca009(:,2)/50;
xc=x*cos(alpha');

for     n=1:length(s)
        for     m=1:18
                zc(m,n)=z(n)-xc(m)*sin(alpha(n));
        end
end

M=moviein(21);
for     n=13:10:143
        N=(n-12+9)/10;
        hold off
        plot(xc(:,n), zc(:,n)+zu)
        hold on
        plot(xc(:,n), zc(:,n)-zu)
        axis([-2 2 -2 2]); grid
        set(gca, 'Visible', 'on')
        M(:,N)=getframe;
end
movie(M);
```