ENERGY SAVINGS FOR UAV FLIGHT IN GUSTING CONDITIONS THROUGH TRAJECTORY OPTIMIZATION AND ACTIVE FLOW CONTROL

BY

LOU GRIMAUD

Submitted in partial fulfillment of the
requirements for the degree of
Master of Science in Mechanical, Materials, and Aerospace Engineering
in the Graduate College of the
Illinois Institute of Technology

Approved ____________________________
Advisor

Chicago, Illinois
May 2014

ACKNOWLEDGMENT

This dissertation could not have been written without Dr. X who not only served as my supervisor but also encouraged and challenged me throughout my academic program. He and the other faculty members, Dr. Y and Dr. Z, guided me through the dissertation process, never accepting less than my best efforts. I thank them all.

(Don't copy this sample text. Write your own acknowledgement.)

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

LIST OF SYMBOLS

Symbol	Definition
β	probability of non-detecting bad data
δ	Transition Coefficient Constant for the Design of Linear-Phase FIR Filters
ζ	Reflection Coefficient Parameter

ABSTRACT

The purpose of this thesis is to show how micro unmanned aerial vehicles can extract energy from natural wind gusts and how this energy extraction can be greatly improved by the use of an active flow control system.

The trajectory of a small UAV flying through wind gusts is simulated with a two degrees of freedom model. The non-dimensional model is set to include vertical and horizontal gusts of varying amplitude and duration. From this model an optimization routine is performed in order to obtain the minimum gust amplitude needed to get a neutral energy trajectory. With these results, it is discovered that neutral energy flight is possible through gusts only 10 to 20% of the flying speed of the aircraft . However the lift coefficient has to be changed very rapidly in order to perform these maneuvers in short duration gusts.

To achieve this performances an active flow control system is investigated. With the help of a wing fitted with piezoelectric actuators different ways of driving a set of leading edge synthetic jets are investigated. After defining the most suitable way to drive the actuator a controller is designed. This controller aims to control the lift variations under unsteady pitching conditions. Since this controller has to work at high angle of attack during rapid pitch changes, a model based on the Goman-Khrabrov paper [4] is used to predict the unsteady lift coefficient.

The resulting controller is shown to be able to greatly reduce lift oscillation and improve the available lift for partially separated flow under unsteady pitching conditions.

CHAPTER 1
INTRODUCTION

1.1 Motivation

1.1.1 Dynamic soaring. The main challenge for electric small size unmanned aerial vehicle is the autonomy. Battery energy density is limited and can rapidly become a important part of the weight of vehicle. Since most of the energy is used by the electric engine for propulsion, optimizing the control laws and trajectory could have a dramatic effect on endurance. With the progress in autonomous control software successful attempt have been made by Allen [2] and Edwards [3] to extract energy from natural updraft. These experiment has shown that a UAV can take advantage of localized vertical winds naturally produced by thermal convection effects.

However, within an urban environment, such as the one mini and micro-UAV are dedicated to, the gust's profile is vastly different. [INSERT REF FROM NEXT DOORS LAB !!!]. Wind blowing through an group of building produce turbulent conditions with both vertical and horizontal vortexes. These turbulences can reach speed representing a significant portion of micro-UAV's glide speed [GET REFERENCE FOR THATH].

Figure from, PIV of the building array, ask Bruno for it

In flow fields such as this the gusts encountered are both faster and more arguably more complex than the ones due to thermal convection. The resulting effect is that the optimal trajectory through such gust profile is widely different from the one shown developed previously by XXXXXXXXXXXXXX.

1.1.2 Active flow control system. The dynamic soaring maneuvers required for energy extraction in environments described in 1.1.1 necessity rapid variations of the lift coefficient. If these variations are performed by pitch change a number of unsteady effect are introduced and can produce some unexpected lift behaviors. For this reason a controller capable of dealing with unsteady aerodynamics for a fast pitching wing is needed.

Moreover the lift range needs to be as wide as possible, as it is shown in chapter 2. This means that the angle of attack will be close to the one when partial flow separation occurs. In the partially separated domain, it has been shown [REF NEEDED] that disturbing the shear layer leads to a sporadic increase in the lift coefficient. These lift changes can be significant enough to be able to improve the maximum attainable Cl by as much as 40%. The disturbances can be created by a wide array of means. The goal is to change the path of the shape of the shear layer by applying a force on the flow. Mechanical retractable vortex generator [8], combustion actuator [REF COMBUSTION ACTUATOR] , compressed air [5] and even Lorentz force actuator [1] have been used to control the flow.

Active flow control (AFC) provides a convenient mean to achieve the performances in both speed and range for the lift coefficient.

There is no mention in the literature of real control system for the lift under varying pitch angle. Some success have been reported in manually synchronizing a sinusoidal pitching motion with a AFC input to keep the lift coefficient under control, however these system only works for specific frequencies and waveforms. The objective for this project was to investigate the possible performances of such a controller for arbitrary pitch input and to assess its limitations in bandwidth, range and precision.

The controller has to handle the interactions between unsteady pitching effect and active flow control.

1.2 Previous investigations/literature review

As explained in the previous part 1.1.1, the bulk part of the research on trajectory optimization for small flying vehicle has been focused on either natural convection such as the one glider pilots and some birds of prey take advantage of in plains, or wind gradients such the ones found close to the surface of the ocean. The later are often exploited by seabird such as albatrosses.

Lissaman [7] has conducted a study for 3D trajectories in differently shaped wind gradients close to the ground. His optimization is performed on a non-dimensional set of equation that has been reused in this study. He also uses different kind of profiles for the wind gradient in order to represent more accurately real wind gradients.

In

CHAPTER 2

ENERGY EXTRACTION OPTIMIZATION

2.1 2 DoF model

2.1.1 Non-dimensional equations of motion. The model chosen for this simulation is a simple two degree of freedom, two dimension, point mass model. The aircraft is assumed to be a glider to simplify the optimization routine. With such assumption the equations of motion in the ground reference frame is :

$$\begin{aligned} \ddot{x} &= -L' \cdot sin(\gamma) + D' \cdot cos(\gamma) \\ \ddot{z} &= L' \cdot cos(\gamma) - D' \cdot sin(\gamma) - m \cdot g \end{aligned} \tag{2.1}$$

The lift and drag are defined are:

$$\begin{aligned} L' &= \tfrac{1}{2}\rho v^2 C_l \\ D' &= \tfrac{1}{2}\rho v^2 C_d \end{aligned} \tag{2.2}$$

With v being the relative wind for the vehicle.

Since this simulation is mainly concerned with Newtonian physics (rather than fluid phenomenons) the usual fluid dynamics non-dimensional variables make little sense. Here the equations are normalized by the optimal glide speed and g, the gravitational acceleration. This is more representative of the performances of the aircraft.

Following Lissaman's [7] implementation of the equation of motion we define V^* the optimal glide speed for the aircraft. This speed is achieved at the optimal lift to drag ratio of the aircraft. With C_l^* and C_d^* the angle of attack for the maximum lift to drag ratio and γ the pitch angle with respect to the horizon the optimal glide speed is:

$$\begin{aligned} \gamma^* &= -atan(\frac{C_l^*}{C_d^*}) \\ V^* &= \sqrt{\frac{2mg}{\rho S(C_l^* cos(\gamma^*) - C_d^* sin(\gamma^*)}} \end{aligned} \tag{2.3}$$

From we define U and W the non dimensional horizontal and vertical speed in the inertial reference frame.

$$\begin{aligned} U &= \frac{\dot{x}}{V^*} \\ V &= \frac{\dot{z}}{V^*} \end{aligned} \tag{2.4}$$

The time is normalized by g/V^*.

Since the speed is seen as a fraction of the optimal glide speed it makes sens to also normalize the lift and drag coefficients by their corresponding values at the optimal lift to drag ratio.

$$\begin{aligned} L &= \frac{C_l}{C_l^*} \\ D &= \frac{C_d}{C_d^*} \end{aligned} \tag{2.5}$$

Finally we introduce Q the dynamic pressure as:

$$Q = \frac{L'}{MgL} = \frac{\frac{1}{2}\rho V^2 C_l C_l^*}{Mg} \tag{2.6}$$

From there the equation of motion 2.1 can be expressed as:

$$\frac{dU}{dT} = -LQ \cdot sin(\gamma) + DQ \cdot cos(\gamma)$$
$$\frac{dW}{dT} = LQ \cdot cos(\gamma) - DQ \cdot sin(\gamma) - 1 \tag{2.7}$$

With

$$\gamma = -atan(\frac{W - W_g}{U - U_g}) \tag{2.8}$$

W_g and U_g are the vertical and horizontal wind speeds in the inertial reference frame.

Finally the remaining thing to consider is Q the dynamic pressure. If we define the speed of the wind gust as W_g and U_g we can express:

$$Q = V^2 = (W - W_g)^2 + (U - U_g)^2 \tag{2.9}$$

With these definitions we have the basic formulation of our non-dimensional equation of motions, normalized by the performances at the optimal glide trajectory in a calm environment.

2.1.2 Lift and drag models. The normalized equation of motion 2.7 are not accounting for the fluid dynamic part of the flight. The most important factor for glide performance is the lift to drag ratio. In his paper, Lissaman [6] is using a relatively simple quadratic model for the relationship between lift and drag:

$$D = \frac{Q}{2G}(1 + L^2) \tag{2.10}$$

This simple model work relatively well for simple airfoil but is inadequate for more complex shapes. For various reasons, small UAVs tend to have non classical designs such as blended wing body or flying wing shapes. The advantages of these designs reside in bigger space available for the payload while keeping the drag low. However the flying wing airfoil profiles have very different lift to drag characteristics compared to more classical airfoils.

To get the lift to drag curve of a typical flying wing a typical flying wing shape is tested in the XFLR5 software.

Figure 2.1. Flying wing UAV shape used in XLFR5

This software perform flow simulation over the aircraft with a panel based method. The simulation is performed on a range of angle of attack from negative 5 degrees to 13 degree.

Due to the presence of sharp edges at the tip of the wings the Reynolds num-

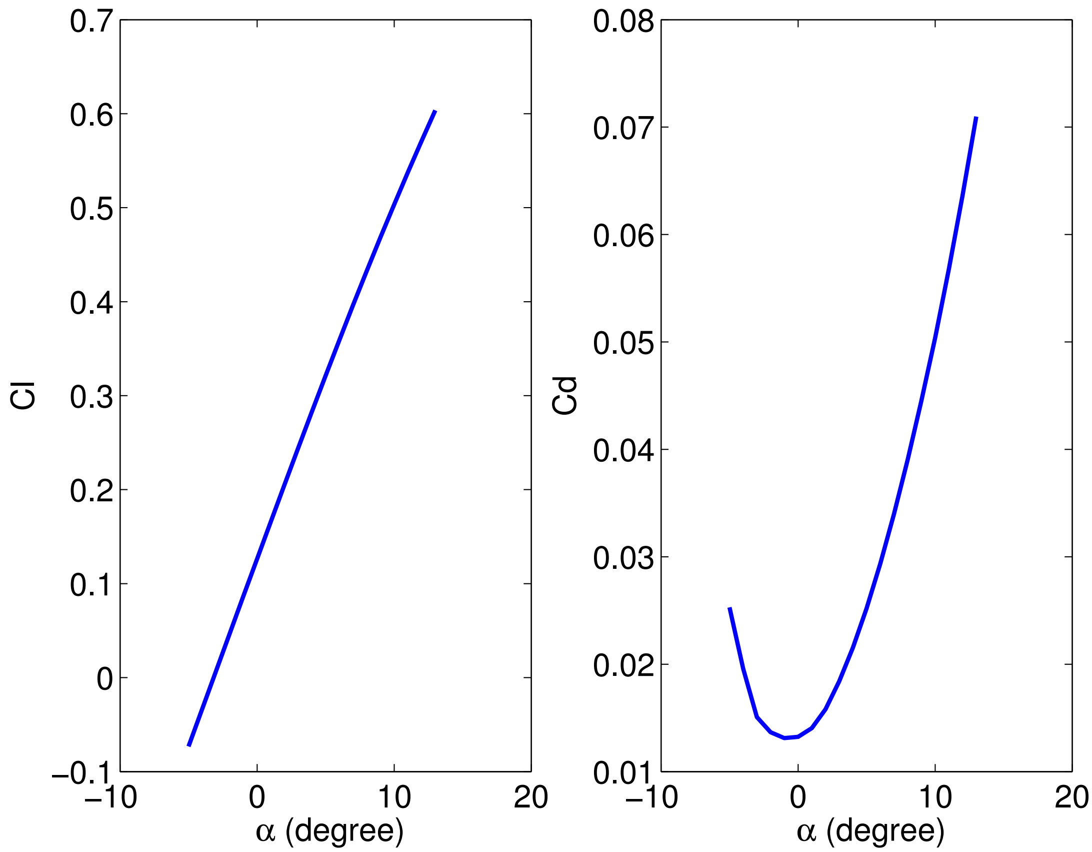

Figure 2.2. Lift and drag characteristics of the notional UAV

ber had to be limited to 3000. At each angle of attack the software automatically computed the steady lift and drag coefficients.

This results, while being arguably more realistic than a simple quadratic approach, are still only considering quasi-steady change in the angle of attack. This limitation will be discussed more in depth in the result discussion section 2.3.

2.1.3 Wind profiles. Most of the studies done on dynamic soaring has either been done with vertical wind gusts or thermal updraft, or horizontal wind gradient foxed in time. In this optimization procedure we chose to consider three different wind profiles made out of first order sinusoidal gusts.

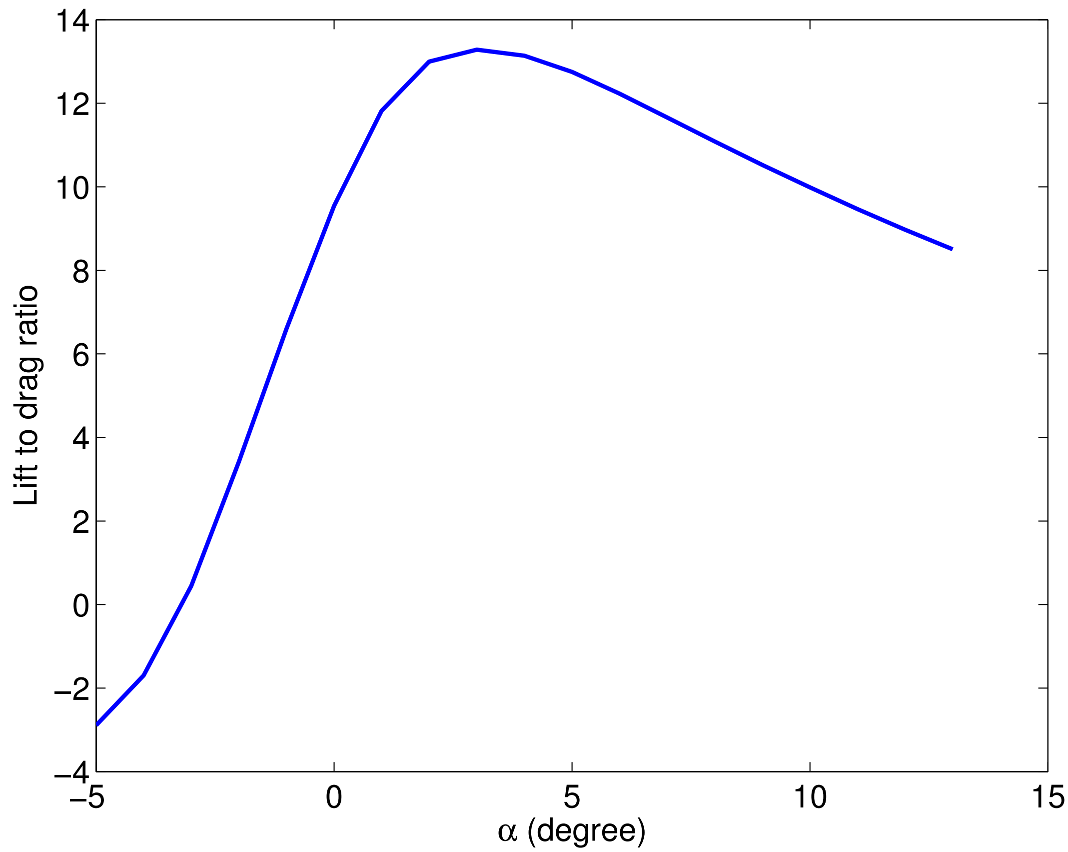

Figure 2.3. Lift to drag ratio for the notional UAV

Our first gust profile is a simple vertical gust.

$$\begin{aligned} W_g &= W_a \cdot sin(2\pi T) \\ U_g &= 0 \end{aligned} \tag{2.11}$$

Similarly the horizontal gust is defined as:

$$\begin{aligned} W_g &= 0 \\ U_g &= W_a \cdot cos(2\pi T) \end{aligned} \tag{2.12}$$

Finally a more complex combined gust is defined. This gust profile is the sum of the two previously defined gusts. Moreover we introduce ϕ, a phase difference between the two component of the gust.

$$\begin{aligned} W_g &= W_a \cdot sin(2\pi T) \\ U_g &= W_a \cdot cos(2\pi T + \phi) \end{aligned} \tag{2.13}$$

2.2 Optimization process, cost function and constraints

2.2.1 General consideration on optimization. The general principle for the optimization routines resides in defining a so called "cost function" that will represent a quantity we want to minimize. While the algorithm tries to minimize this scalar, a set of constraints have to be respected. These constraints can represent physical limitations or specific requirements related to the system at hand. The cost and constraints are expressed as functions of a set of system state variables. The state variables can represent temporal or spatial values. The optimization is performed in a sequential fashion where different algorithm are used to step from one set of values for the state variables to another.

Optimization routines are divided into two families.

The first method is called the gradient method, it requires a good knowledge of the physics behind the problem. The cost function as well as the constraints have to be explicitly defined. In this method the gradient of the cost function and the constraints is used to determine the direction of the next step in the optimization. Different algorithm are used to chose the step size, and sometime the direction of the previous step can influence the current step. The gradients for either the cost function or the constraints do not have to be explicitly defined as modern optimization routines, such as the one included in Matlab, can perform numerical gradient estimation. However inputting an user defined gradient into the routine will significantly speed up the overall process.

The second method is using the so called “evolutionary algorithms”. This method relies a lot less on knowing the underlying physical phenomenon. Its basic principle is a “try and see” process. Random changes are performed on the state variables and their effects on the cost function are assessed. The best steps are selected as a starting point for the next generation. While with this method each step is a less computation intensive than with the previous method, the number of steps is a lot higher.

The first method has been used in this optimization as it provides more insight on the physics behind the problem. However it should be noted that the resulting “optimal” point is usually only assured to be *a local* minimum of the cost function. Several different starting states should be tested to ensure that the optimization converges toward a reasonable minimum.

2.2.2 Cost function. Our problem here consists in optimizing the trajectory in a gusting environment to minimize energy lose. The most obvious cost function would be something like

$$-\frac{1}{2}mV(T_f)^2 - gX(T_f) \tag{2.14}$$

Which would be equivalent to maximizing the total energy at the end of the gust. However after testing this has shown to leave to much freedom to the algorithm. As a result the local minimum found are the result of trajectories such as very steep dives, clearly far from the optimum.

Once again we refer to the Lissaman paper [6] and chose, instead of minimizing energy loss for a given gust condition, to find the minimum gust amplitude to satisfy an energy neutral trajectory over the gust period. This means that the cost function is the wind gust amplitude, which will have to be added to the state vector in order to be explicit, and that the neutral energy trajectory will have to be added to the constraints

2.2.3 State vector and constraints formulation. In our case a gust cycle of duration T_f is divided into N discrete instants T_i (usually between 31 and 101). At each of these points we need to know the state of the vehicle. Since we are considering a two degree of freedom model the two positions X, Z and speed U, W variables are the most simple choices. However this is not enough to describe the system completely, we also need to know what our input is going to be, in this case the lift available and where on the lift vs drag curve we are. Theres is two possible choices for this. If you consider only the quasi steady part, the angle of attack α seems obvious. However since the drag is a function of the lift (the inverse is not true), it is possible to use only the L to define our point on the lift to drag curve. This allows us to use one less variable.

With this five variables defined at each considered time points the state and input vector look like:

$$x = \begin{bmatrix} \dots \\ X_i \\ Z_i \\ U_i \\ W_i \\ L_i \\ \dots \\ W_a \end{bmatrix} \quad i \in [1, N] \tag{2.15}$$

All this variables have to be constrained to achieve a realistic trajectory. The first and most obvious constrain is done with the equation of motion 2.7. This equation has to be changed from a continuous differential equation to a discrete equation. This is done by using the Simpson's 1/3rd rule as derived by Zhao [9].

In order to satisfy the equations of motion we need to define the state variable at the time T_i:

$$y_i = \begin{bmatrix} X_i \\ Z_i \\ U_i \\ W_i \end{bmatrix} \tag{2.16}$$

Then with $\dot{y}_i$ the derivative of the state variables, given by the equation of motion 2.7 and:

$$
\begin{gathered}
y_m = \tfrac{1}{2}(y_k + Y_{k+1}) - \tfrac{1}{8}(\dot{y_{k+1}} - \dot{y_k})\delta t \\
L_m = \tfrac{1}{2}(L_i + L_{i+1})
\end{gathered}
\tag{2.17}
$$

The condition to satisfy the equation of motion becomes

$$
0 = y_{k+1} - y_k - \frac{1}{6}(\dot{y_k} + 4\dot{y_m} + \dot{y_{k+1}})\delta t \quad \forall i \in [1, N-1] \tag{2.18}
$$

Another constraint is on the neutral energy loop condition. To account for that the initial and final Z values are fixed at zero and the initial and final vertical and horizontal speeds are set to be equal.

Since we are looking at only one cycle, in order for it to be repeatable, we need to have a smooth transition from one to another. This means setting the derivative of the speed to be equal at the start and at the end of the cycle.

$$
\begin{gathered}
W_2 - W_1 = W_N - W_{N-1} \\
U_2 - U_1 = U_N - U_{N-1}
\end{gathered}
\tag{2.19}
$$

Finally the last set of constraint is on the physical limits of the aircraft. Typically an aircraft flight envelope is limited by its maximum speed (depends on the dynamic pressure), its maximum load and its maximum lift (which determines the stalling speed). Since our aircraft is will be flying around its optimal glide speed, over speeding isn't going to be an issue. Moreover the drag increasing proportionally to the square of speed, high speeds will be avoided as much as possible by the optimization routine. The limit on the load can conveniently be expressed as:

$$
L_i Q_i \leq g_m ax \quad \forall i \in [1, N] \tag{2.20}
$$

With g_max the maximum load in Gs.

Finally the maximum lift condition can be expressed

$$L_i \leq \frac{C_l^{max}}{C_l^*} \quad \forall i \in [1, N] \tag{2.21}$$

As it will be seen in section 2.3 the value of C_l^{max} has a profound impact on the performances of the UAV.

It is also sometime advisable to limit γ in the $\pm 90°$ range to prevent loops and backtracking.

2.2.4 Matlab optimization function. Matlab offers several ways of doing optimization. Since this scripting language allows for easy parallelization, it is relatively painless to implement your own optimization code. However in most cases, "classical" optimization problems, such as weight reduction, topology optimization or mechanism design are reducible to a set of linear equations and constraints. In our case the equations of motions as well as the lift and drag properties are not linear at all, and trying to linearize this problem would make any solution meaningless. For this reason a already existing optimization function has been chosen.

Since non linear optimizations like that are a computation intensive process dedicated tools have been developed to tackle the problem. SNOP is one of these software and seems to be widely use. Another tool appearing in the literature is a Fortran library called NPSOL. Since our laboratory's language of predilection is Matlab, the optimization toolbox from MathWorks will be used.

The optimization toolbox provides with a helpful function for non-linear optimization called *fmincon()*. This function needs an initial guest for the x vector. When using this function the initial guest has quite a big influence on the converging speed and on the local optimal solution found. To account for that several educated guesses were made and tested for the different types of the wind profiles and gusts duration. These guesses are refined as new results are obtained.

2.3 Results for quasi-steady aerodynamics

2.3.1 Typical trajectories. The first step is to validate the code implemented here against previous results. Even if the lift and drag profiles are different from Lissaman's assumptions a similar case is optimized. The gust duration is set to be $T_g = 4T+$, for a purely vertical gust. Since it is only a validation test the same lift and drag characteristics are used. Equation 2.10 plugged into the equation of motion part of the code. An optimal lift to drag ratio of $G_{max} = 20$ is chosen, as seen in the original Lissaman paper [6].

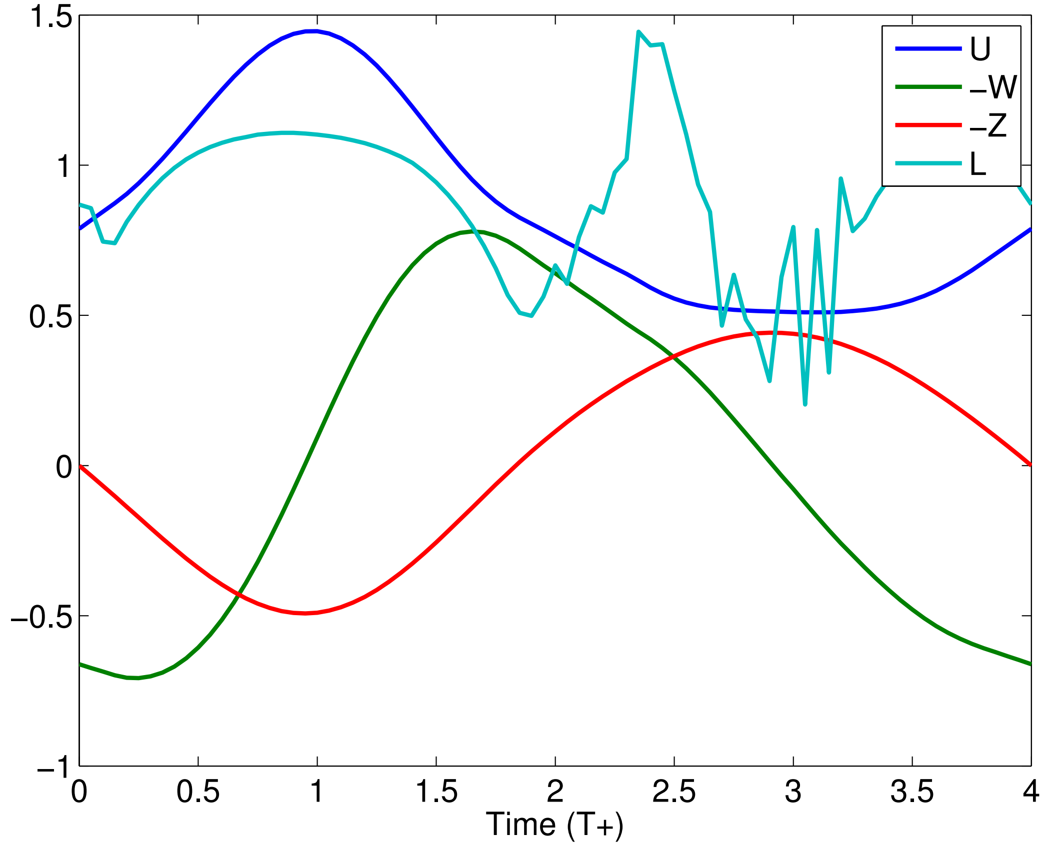

Figure 2.4. Optimization results for a $4T+$ long vertical gust

Lissaman, with is optimal lift to drag ratio of 20 found a wind gust amplitude of 0.129. Here the minimum required for neutral energy loop is 0.128 (see figure 2.4). The shape of the state and control parameters curves are also consistent with the Lissaman results.

Similarly an optimization is performed for a purely horizontal wind gust.

The resulting minimum wind amplitude is higher than for the vertical gust. However this shows that it is possible to take advantage of horizontal wind gusts to save energy if the performances are high enough.

Finally a combined horizontal and vertical gust is simulated.

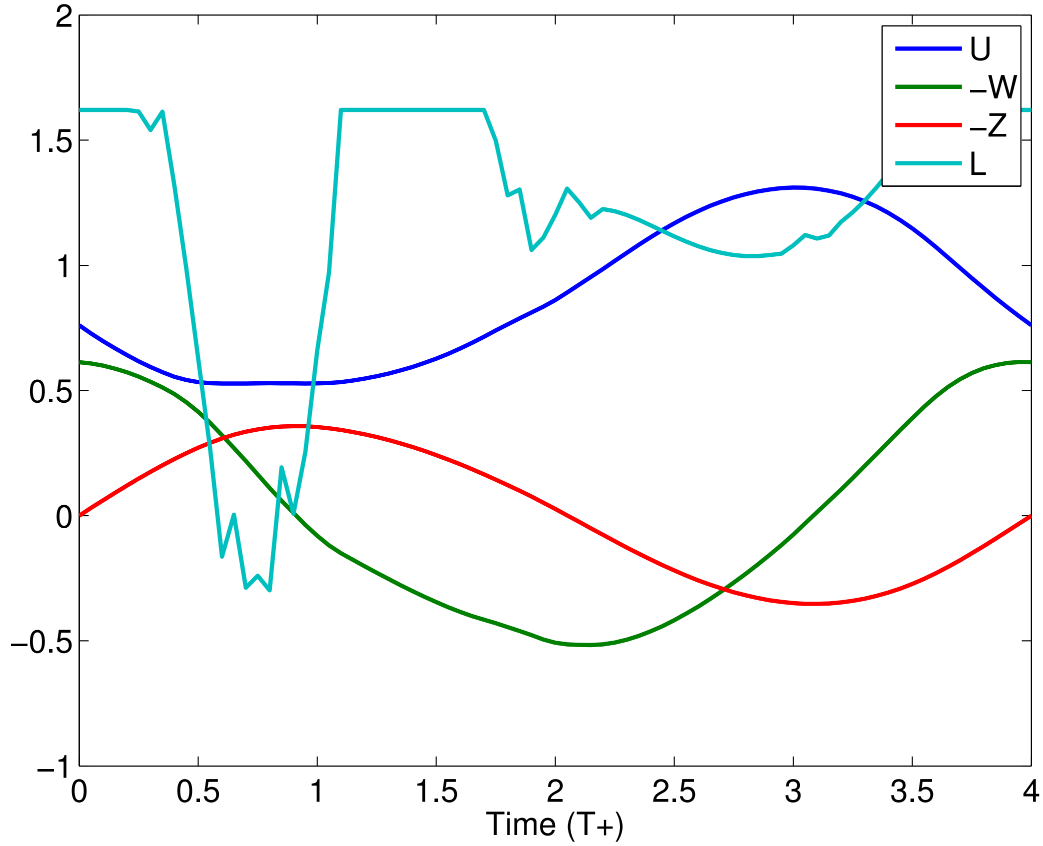

Figure 2.5. $4T+$ long horizontal gust for $G = 20$, $W_a = 0.246$

Unsurprisingly the neutral energy loop trajectory exists also for this case.

The general idea for flying when there is a wind gust is the “belly to the wind” technique described by Zhao [9].

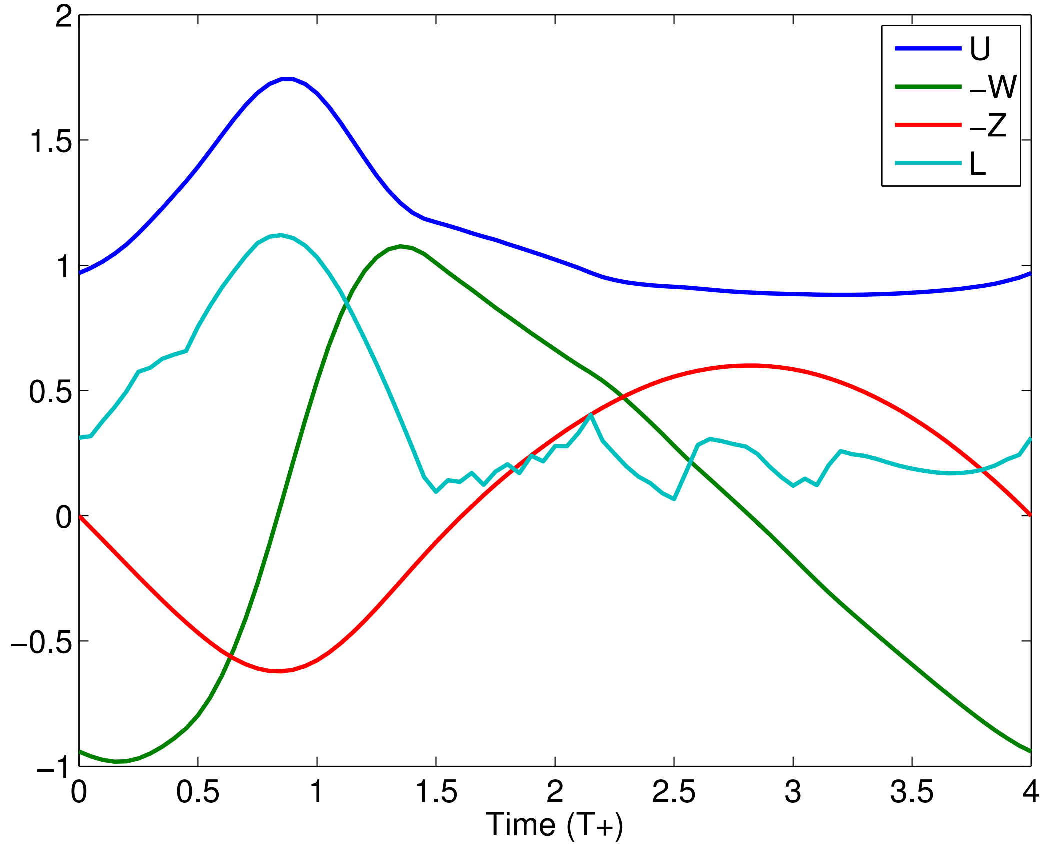

Figure 2.6. $4T+$ long combined gust for $G = 20$, $W_a = 0.232$

A similar batch of optimizations is done with the more realistic lift and drag profiles.

As it can be seen on the figures 2.7 and 2.8 the trajectories are similar in shape. However the gust amplitude needed to achieve neutral energy flight are a lot higher. Such differences can be explained by looking at the maximum lift to drag ratio for both conceptual aircraft. The quadratic drag profile used by Lissaman has a G_{max} of 20. The notional UAV used is closer to 14.

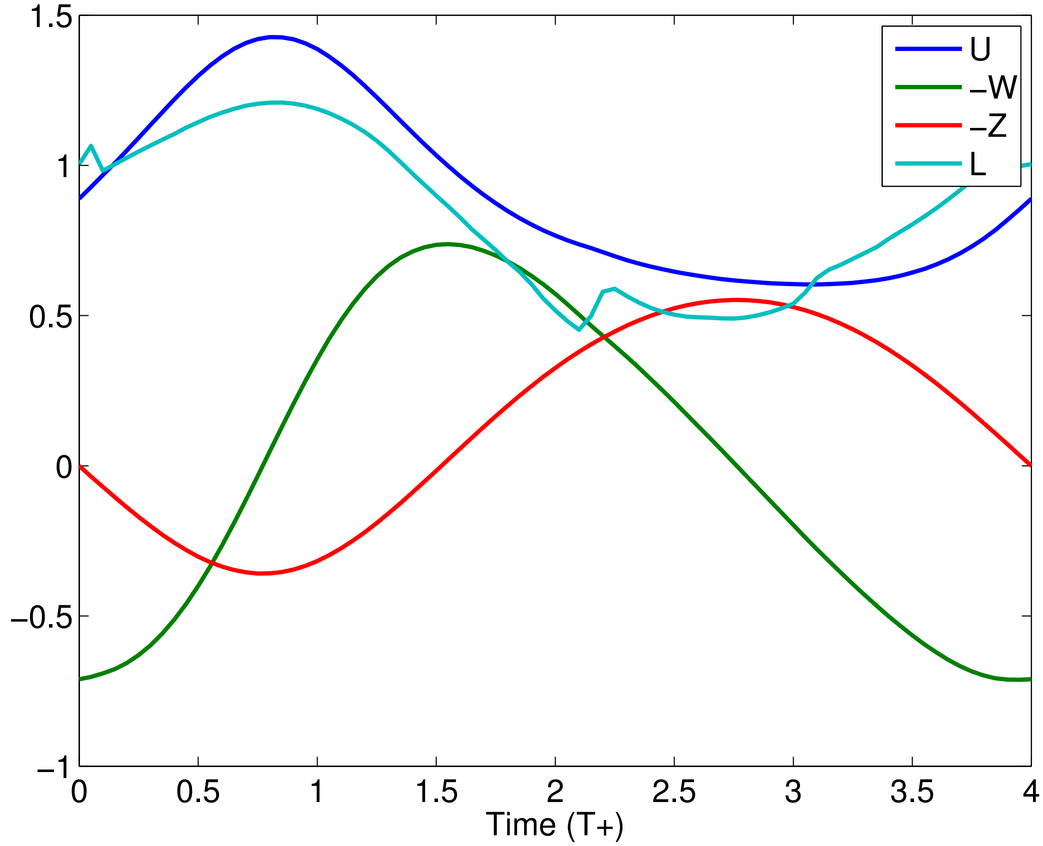

Figure 2.7. $4T+$ long vertical gust for UAV, $W_a = 0.205$

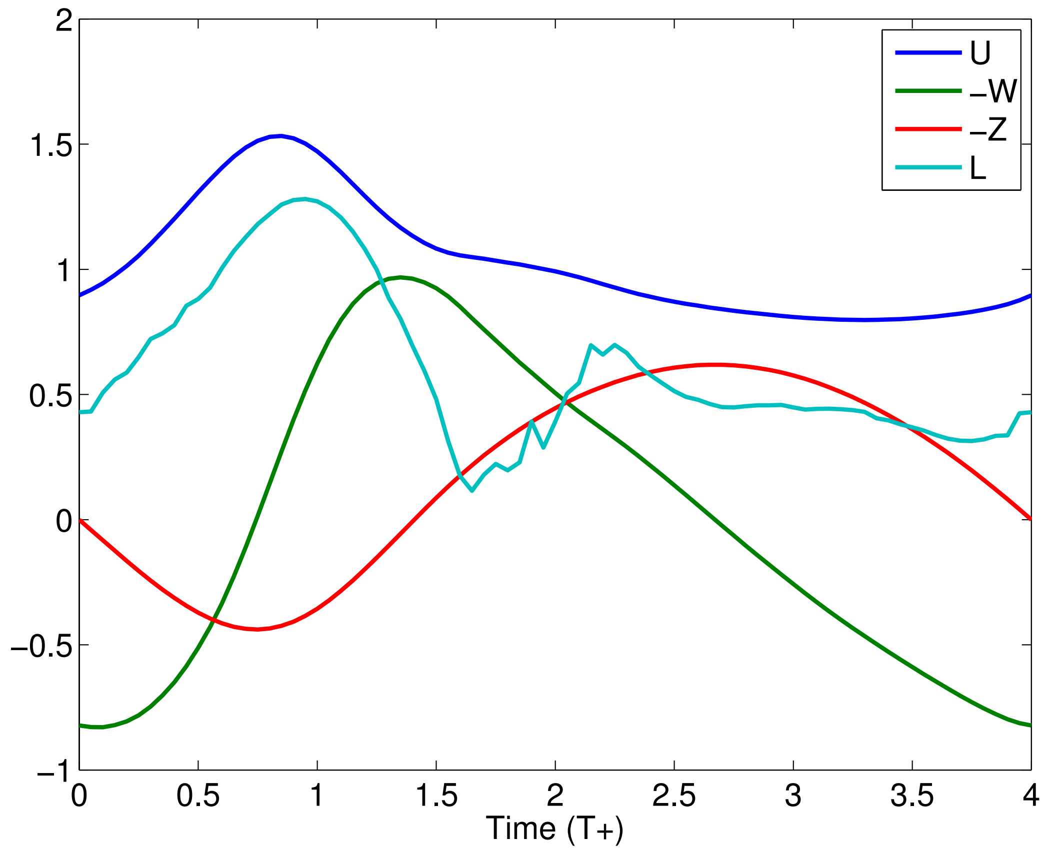

Figure 2.8. $4T+$ long combined gust for UAV, $W_a = 0.387$

With this kind of performances, a purely horizontal gust can't sustain a neutral energy loop.

To account for these differences a third batch of simulation is performed with by setting the G_{max} parameter in the quadratic drag profile to 13.3.

The results are similar. Since the quadratic lift to drag profile isn't that different to the UAV one, the optimized gust amplitude is relatively close for both case.

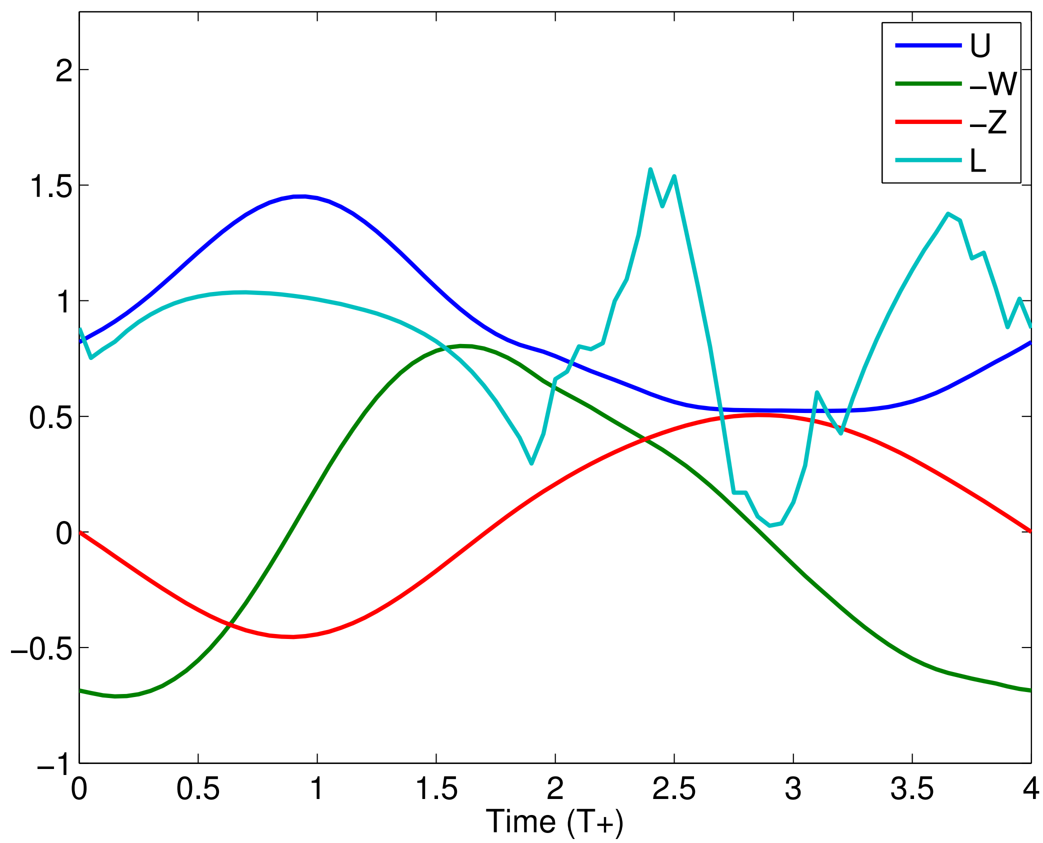

Figure 2.9. $4T+$ long vertical gust for $G = 14$, $W_a = 0.194$

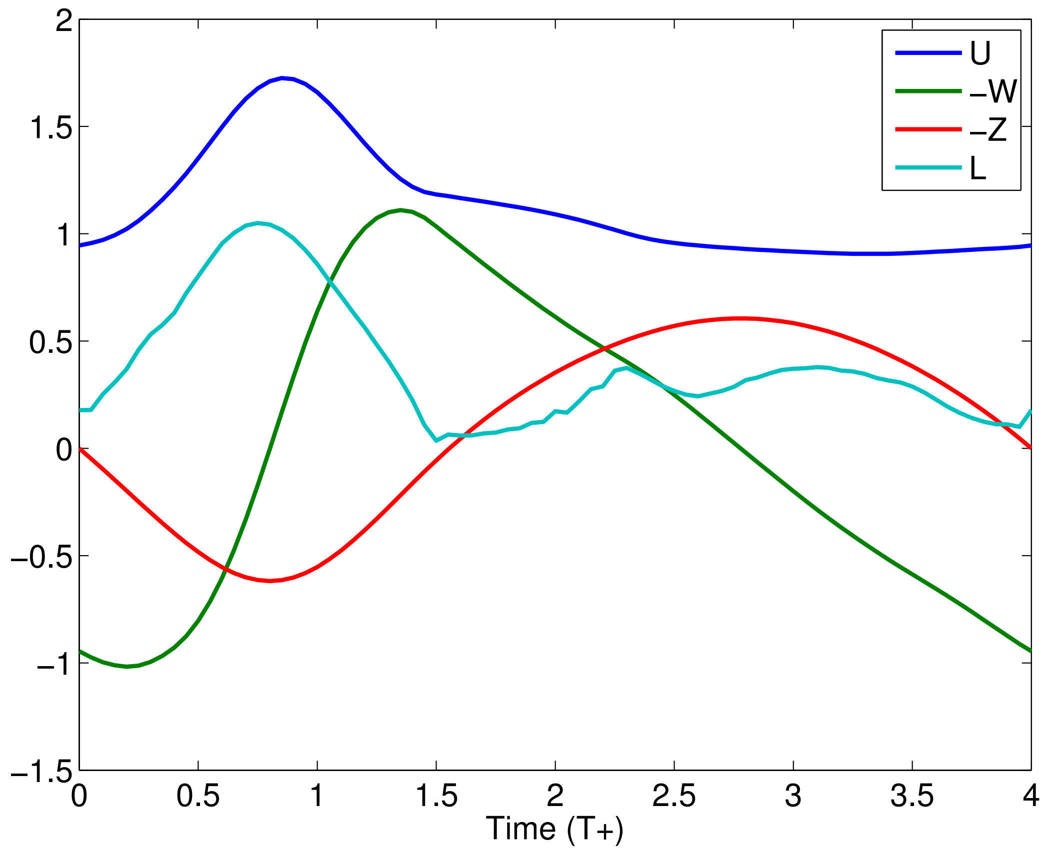

Figure 2.10. $4T+$ long combined gust for $G = 14$, $W_a = 0.360$

2.3.2 Influence of the gust duration. From our literature review it seems like most of the studies done on gusting winds has been conducted on gusts duration greater than $2T+$. Considering shorter gusts seems unreasonable of you only consider quasi steady aerodynamics, a $2T+$ gust is about 2 seconds for vehicle flying 10 m/s. However since the purpose this research is to extend the energy extraction envelope, such cases should be considered.

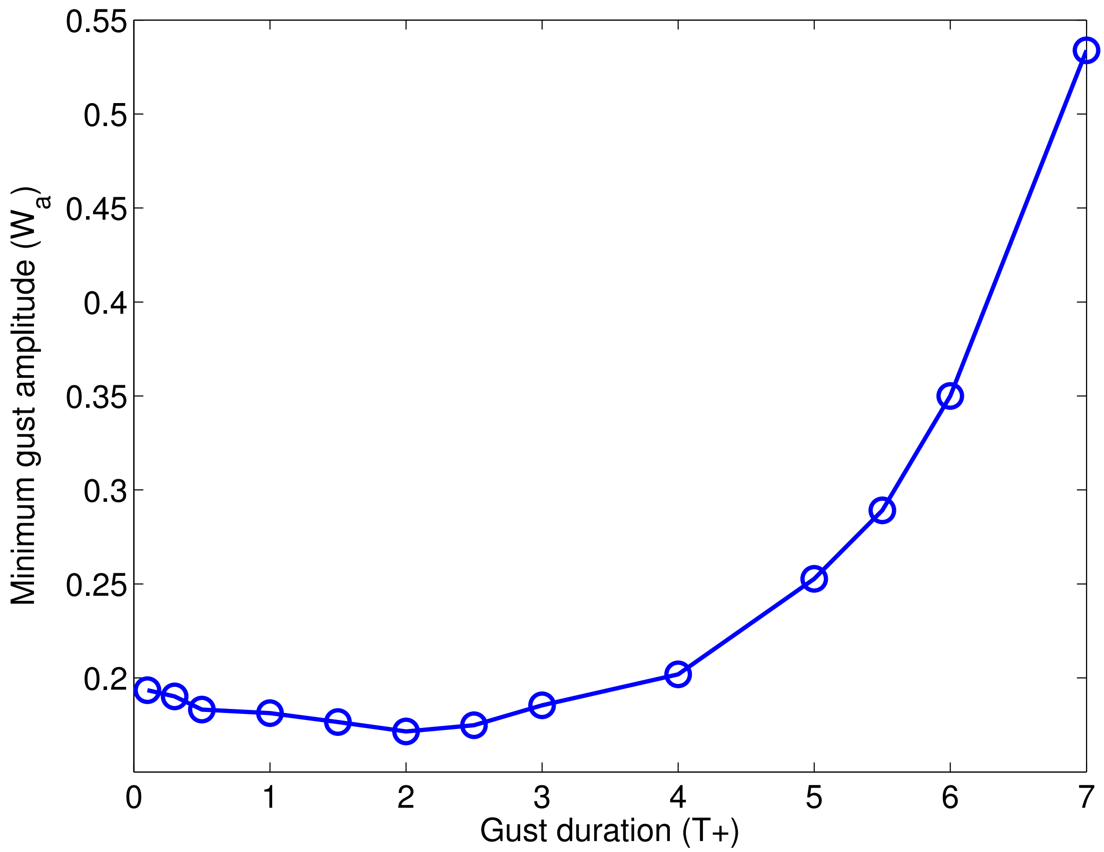

Figure 2.11. Influence of gust duration on the minimum gust amplitude for vertical gusts

Interestingly shorter gusts require less wind amplitude than the long ones. This seems to indicate that most of the lost energy is due to the non-conservative drag force and not due to the downwind effects. However the actual minimum gust amplitude required for neutral energy flight has a minimum for $2T+$ long vertical gusts.

A similar graph can be made for combined gusts.

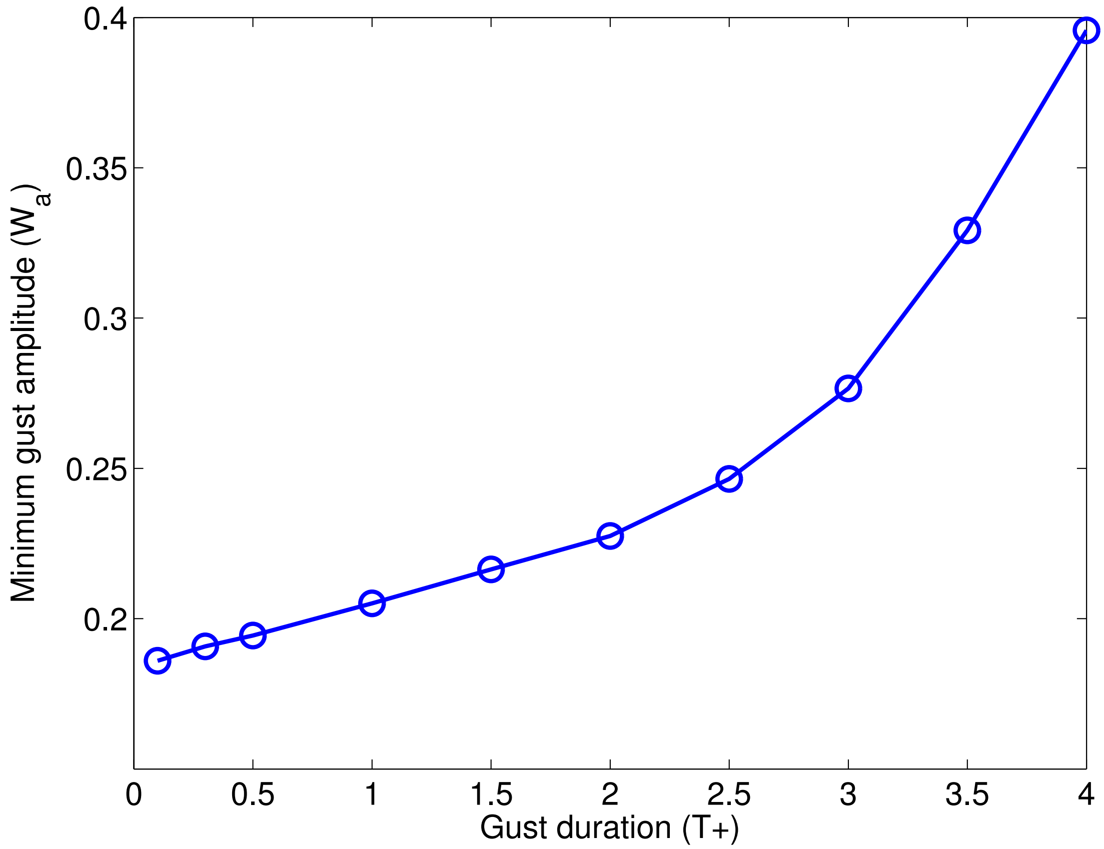

Figure 2.12. Influence of gust duration on the minimum gust amplitude for combined gusts

This time no minimum is found. This seems to reinforce the idea that the losses are mainly due to the energy dissipated by the drag since in this case the drag influence is mitigated by the horizontal component of the combined gust.

2.3.3 Influence of phase variation in the combined gust case. For combined vertical and horizontal gusts another parameter can be changed. So far the phase between the two components of the gust has been constant.

For this we define the phase ϕ as:

$$W_g = W_a cos(2\pi T)$$
$$U_g = W_a sin(2\pi T + \phi) \tag{2.22}$$

After Simulations are performed by 10 degrees steps with the following results.

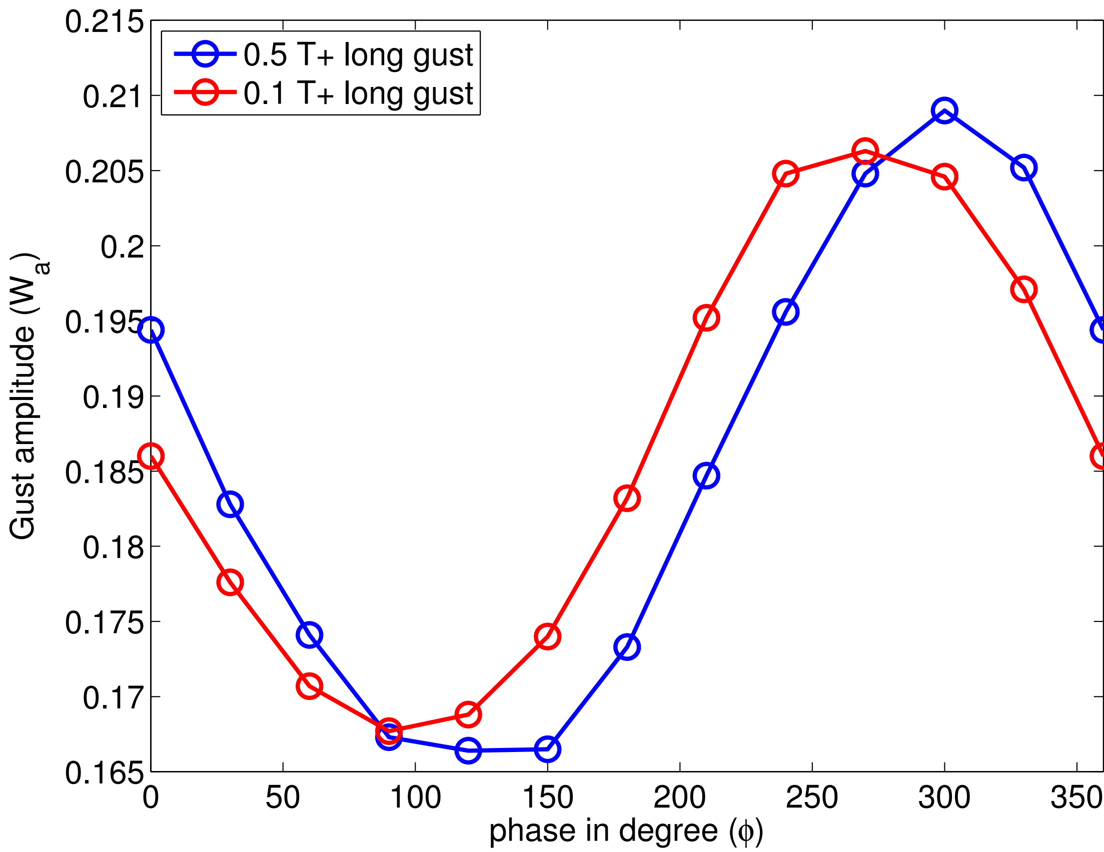

Figure 2.13. Influence of the phase between the component of the combined gust

This clearly shows that our minimum gust amplitude for a 0 phase was actually close to the worst case possible. The best case scenario is when the phase is around 90 to 120 degrees and the worst is around 270 to 300.

We can see that the results are different for different gust durations. One possible explanation for this is that at some point the inertia is too great and start to act as a low pass filter, introducing some phase shift in the trajectory.

2.3.4 Maximum lift available effects.

2.3.5 Additional remarks. As noticed by Lissaman the exact shape of the lift input isn't really important. In his paper he approximated the input with a simple sinusoidal with amplitude and phase control, similar inputs in this simulation have produced minimum gust amplitude similar to the previous results. This raises the hope that even very basic controllers should be able to improve UAV endurance.

While these results are supposed to be the optimal solution, it does not mean a controller can achieve such performances. Even if the trajectory and lift curves are physically possible, the optimization algorithm assume a known gust shape and optimize all of the time points at once. This means that contrary to a real controller the optimized trajectory can anticipate the wind change and preventively react. Even if the wind gusts were perfectly sinusoidal, since a controller is casual it would not be able to anticipate. Finally in a real life scenario the wind would of course not be a simple sinusoidal gust.

The simulation is also limited by its inability to account for the moment of inertia along the pitch axis. Even is part of the lift change can be handled by the active flow control system, at low angle of attack, most of the lift comes from the changes in α.

Even with all this limitations these results provide good insight into what would be needed implement energy extraction trajectories in UAVs.

CHAPTER 3

MODELIZATION OF THE LIFT COEFFICIENT UNDER UNSTEADY PITCHING MOTION

3.1 The Goman and Khrabrov model

3.1.1 Motivation. In their 1994 paper entitled "State-Space Representation of Aerodynamic Characteristics of an Aircraft at High Angle of Attack" [4] Goman and Khrabrov introduce a new model for characterizing the lift and moment coefficients for slender delta wings. Their goal was to study the stability of delta wing fighter jets where maneuverability is important, and to link it to physical fluid dynamic phenomenons such as vortex breakdown or flow separation.

The classical stability analysis method relies on a Taylor series expansion of the aerodynamic coefficients. This linear representation is relatively accurate for fully attached flow but the model breaks down at higher angle of attack when separation occurs. In the semi separated region the aerodynamic effects are mainly driven by the degree of flow separation happening on the wing. For this reason they chose to define C_l as a function of α, the angle of attack, and a state variable x representing the degree of separation. This degree of separation can be defined as the position of the vortex breakdown point if you are looking at delta wings, or the position of the reattachment point in the case of 2D airfoils. This allows for a model tightly defined by the physics of the flow.

3.1.2 Flow physics and state variables. Since this study was performed with a 2D NACA0009 airfoil, we define the state variable x as the position of the reattachment point. Its value linearly change from 1 when it is situated at the leading edge to 0 when it gets to the trailing edge and beyond. For quasi-steady cases separation point is a function of the angle of attack. If we define x_0 as the separation point position in a quasi-steady situation then

$$C_l^{qs} = f(\alpha, x_0(\alpha)) \tag{3.1}$$

The unsteady part of the flow physics can be divided into two groups of phenomenons.

The firsts are the effects of the angle of attack variation speed on the position of the separation point. Goman and Khrabrov argue that this is roughly proportional to the pitch rate $\dot{\alpha}$ and as such they can be included by modifying the quasi-steady state value by using $x_0(\alpha - \tau_2\dot{\alpha})$

The second phenomenon is due to the dynamics of the separated flow. The flow has a certain relaxation characteristic under a disturbance input. This can be modeled using a first order differential equation.

$$\tau_1 \frac{dx}{dt} + x = x_0(\alpha - \tau_2\dot{\alpha}) \tag{3.2}$$

3.2 Experimental Setup

3.2.1 Equipment and facilities.

Figure 3.1. Airfoil model inside the wind tunnel

All of the experimental part of this research was performed into the Andrew Fejer Unsteady Wind Tunnel at the Illinois Institute of Technology, Chicago. This is a low velocity wind tunnel with a 60cm by 60cm test section. The wind tunnel is mainly used for unsteady aerodynamic studies. Airfoils are mounted on a motorized sting outfitted with two linear electric servo-motors. These servos are powered by an amplifier with a integrated PID system and driven by an analog voltage input signal proportional to the desired position.

Figure 3.2. Pitching and plunging mechanism

As seen on figure 3.2 combining the motion of the front and back servo allows for the wing to be plunged as well as pitched around a range of axis. The tunnel is also equipped with a system of shutters that can be used to create wind gusts. However this feature will not be used in this project.

The input signal for the servos is made with Simulink® and fed through D-Space® as an analog voltage.

Several sensors are used for data acquisition. A pair of linear potentiometers measures the position of the servos in order to get the airfoil pitch angle. The flow speed is measure via a Pitot tube and pressure transducer plugged into a acquisition box. In parallel to this acquisition box the forces exerted on the airfoil can be measured. A piezoelectric ATI Nano17 force balance seats between the sting and the airfoil. This sensor measures both absolute forces and moments along 3 different axis.

The wing is made out of balsa wood with a 3D printed leading edge housing the active flow control system. This system will be described in more detail in the appropriate chapter. The structure is wrapped in mono-coat, a heat-shrunk plastic film. Its chord length is 245mm its width 560mm with a NACA0009 profile. It connects to the force balance at a point at 25 percent of the chord. The maximum was made to keep the weight and moment of inertia as small as possible to minimize the inertial effects when the wing is moving.

3.2.2 Experimental procedure and data processing. Different pitch input have been tried. There was some fears at first that if the pitching axis wasn't on the axis symmetry, at the quarter chord of the airfoil, additional aerodynamic phenomenon would affect the data. After testing different pitching input that placed the rotation axis either at the top of the front servo, at the top of the force balance or at the top of the back servo, it was determined that the optimal way to drive the pitching mechanism was to move only the back servo. Other input method induced to much mechanical vibrations and did not seems to make any difference aerodynamically.

The amplifier driving the electric servos has its own PID control system, however even after careful tunning some error exists between the commanded angle of attack and the actual angle of attack. To negate that effect the actual servo position, as given by the potentiometers, is used for our measurements. This data is used to transform the normal and tangent force into lift and drag (via a simple rotation matrix). They are then normalized to get the aerodynamics coefficients.

Unless specified otherwise, all the acquisitions have been done at a flow speed of 3m/s which correspond to a Reynolds number of 50000.

For each experimental case the force balance as well as the servo position and a synchronization signal are simultaneously acquired. A first offset with the tunnel off and the wing pitching is taken to let us get the force balance offset as well as record the inertial effects. Even tho the wing is only weighing around 300 grammes, these inertial effects represent the majority of the forces measured by the force balance. Moreover some of the force measured come from the springiness of the cables used for the active flow control part. After the first offset the real case is taken, followed by a second offset to account for the drift in the force balance measurement sometime seen over the course of several minutes.

During each acquisitions at least 50 cycles are recorded. This allows us to perform what we call phase averaging. This is done by slicing the files into individual cycles (thank to the synchronization signal) and then making an average of these cycles. With this technique the signal to noise ratio of greatly improved. Once this has been done with the 2 offsets and the proper acquisition itself, aerodynamic forces are obtained by subtracting the offsets. All this processing is done with Matlab®.

The servo actuation system has a small but noticeable dead band as well as a delay between the input and output. This makes the actual pitching motion slightly different from the input. To account for that the actual measured pitch angle is used as an input of the GK model when we want to compare its prediction with the experimental data.

Finally the GK model itself is also implemented in Matlab. The code can be seen in annex A.

3.3 Adapting the GK model to the NACA0009

3.3.1 Steady lift and stalling behavior. With the basics of the GK model defined, the goal is now to adapt it to our objectives. If this model is to be used for optimization purposes the drag also needs to be calculated. The original model defined by Goman and Khrabrov was included the lift and pitching moment coefficients. Similarly to their model the assumption is made that the lift and drag coefficients share the same state variable. As such we define f and g as

$$\begin{aligned} C_l &= f(\alpha, x) \\ C_d &= g(\alpha, x) \end{aligned} \tag{3.3}$$

The other difference with their case study is that we are considering a 2D airfoil whereas they modeled a 3D delta wing. This means that we can't reuse the same lift function f as the original paper.

In order to get an accurate equation for the lift and drag a quasi-steady map of the lift and drag coefficients is made. This map is done by very slowly (0.1 degree per seconds) pitching the wing between -5 and 25 degrees. The free stream speed has to be corrected to account for the flow slowing down during the higher blockage ratio at high angle of attack.

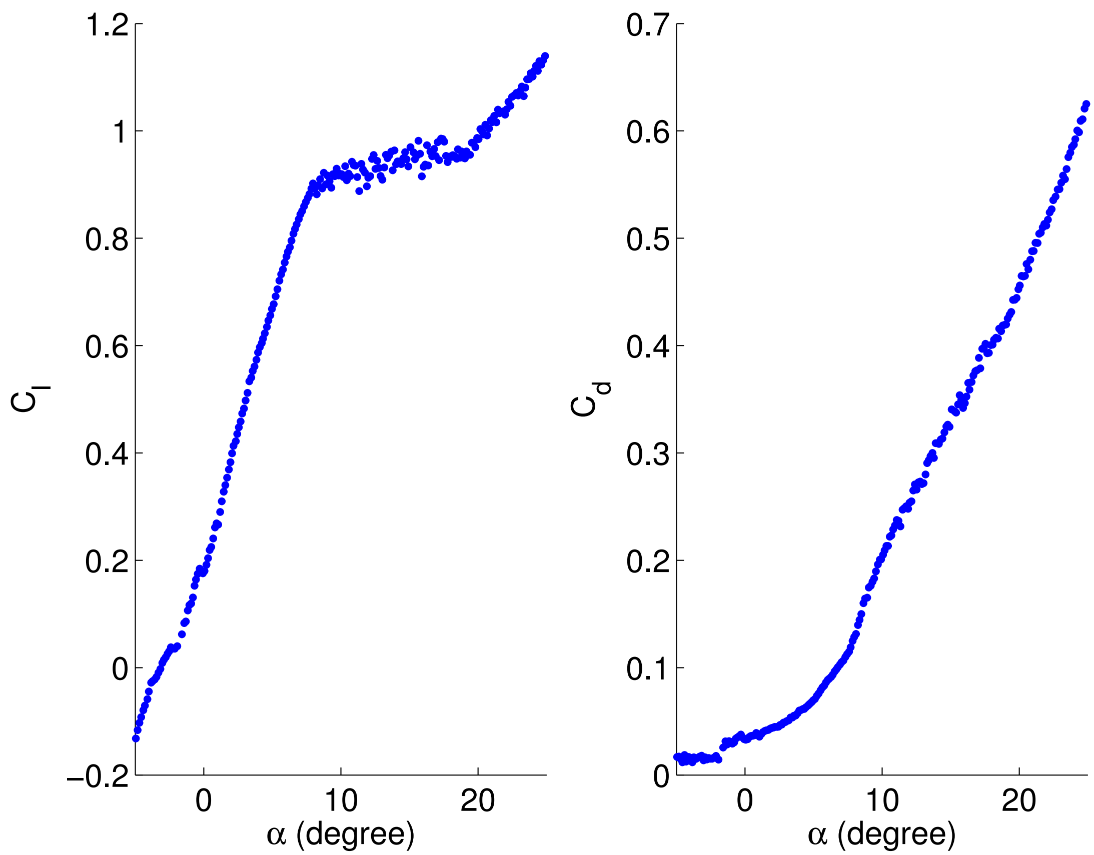

Figure 3.3. Lift and drag coefficient in the quasi-steady case

Figure 3.3 shows how the aerodynamics behave for our NACA0009 airfoil. The lift coefficient is close to a clean linear function when the flow is attached. The separation happens around 8 degrees and the lift coefficient remains constant in the 10 to 20 degrees zone when the flow is partially separated. At higher angle of attack the flow is totally separated and C_l is once again proportional to α but with a different slope this time. Even though the NACA0009 has a symmetric profile the measured lift coefficient for a angle of attack of zero is not null. It is suspected that the sting onto which the airfoil is fixed may disturb the flow and cause this asymmetry. Moreover this curve differs slightly from the ones found in the literature. Once again this can be attributed to the experimental setup; other than the sting effects the couple of millimeters of clearance between the wall of the wind tunnel and the edge of the airfoil are probably to blame as they induce some 3D effects. These gaps are necessary for to allow for the both pitching and plunging of the wing.

From this static map we can approximate the part where the flow is still attached (¡8 degrees) by

$$\begin{aligned} C_l &= 2\pi \cdot \alpha + C_l 0 \\ C_d &= \tfrac{C_l^2}{2G_{max}} + C_d 0 \end{aligned} \tag{3.4}$$

Which is remarkably close to the classical theoretical result for a 2D airfoils in a ideal inviscid attached flow.

3.3.2 State variable approximation. When the flow is still attached the value of x is 1. This means that we are considering the separation point to be at the trailing edge. Similarly when the flow is totally separated the separation point is at the leading edge and $x = 0$. Since for totally separated flow the slop of the lift coefficient as a function of α can be approximated to about 0.4 of the slope for the attached flow, we choose to use the following equation for the lift over the whole range of angle of attack.

$$C_l(\alpha, x) = 2\pi \cdot \alpha(0.6x + 0.4) + C_l 0 \tag{3.5}$$

By inverting this equation the value of x_0 can then be adjusted so that the output of this function matches the experimental data. The resulting profile for $x_0(\alpha)$ can be seen in figure 3.4

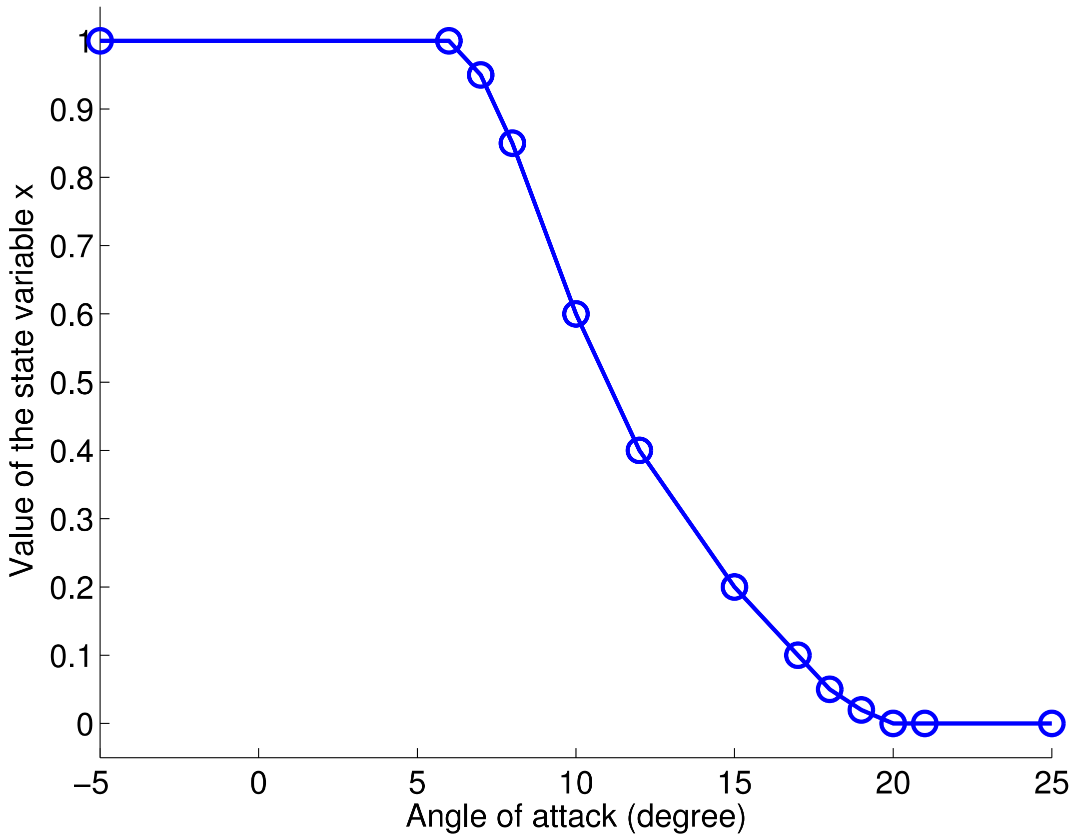

Figure 3.4. Quasi-steady profile for the state variable x

With this profile we get a good approximation of the experimental $C_l(\alpha)$ (cf figure 3.5) for quasi steady cases.

The assumption that the drag share the same state variable as the lift is confirmed when the following equation produces similarly accurate results compared to experimental data, as seen on figure 3.6.

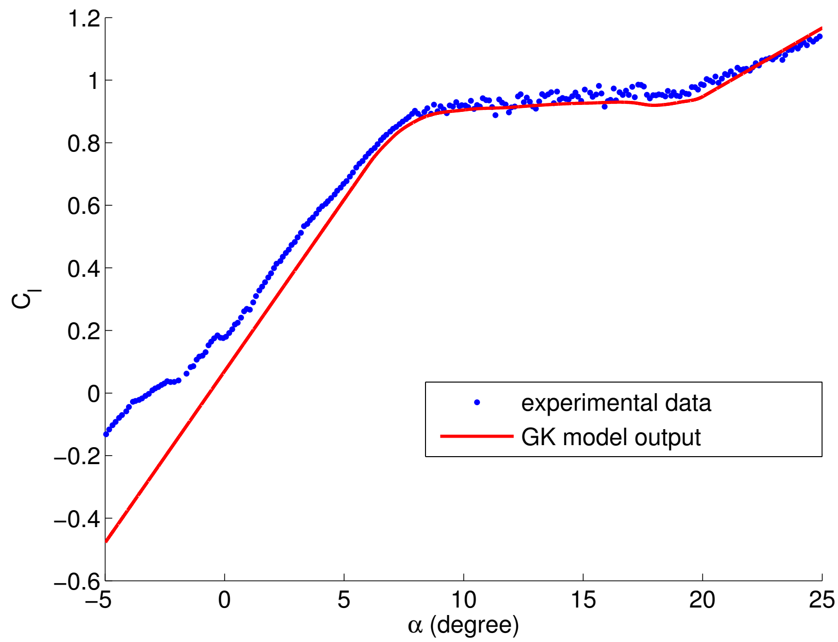

Figure 3.5. Comparison between the experimental and model quasi-steady lift

$$C_d(\alpha, x) = \frac{((2-x)\,C_l)^2}{2G_{max}} + C_d0 \tag{3.6}$$

These two relatively simple equations shows that a physics based GK model can be implemented for both lift and drag and that they indeed depend on the same state variable. The two time constants τ_1 and τ_2 will be determined in the next section when the wing undergo unsteady pithing.

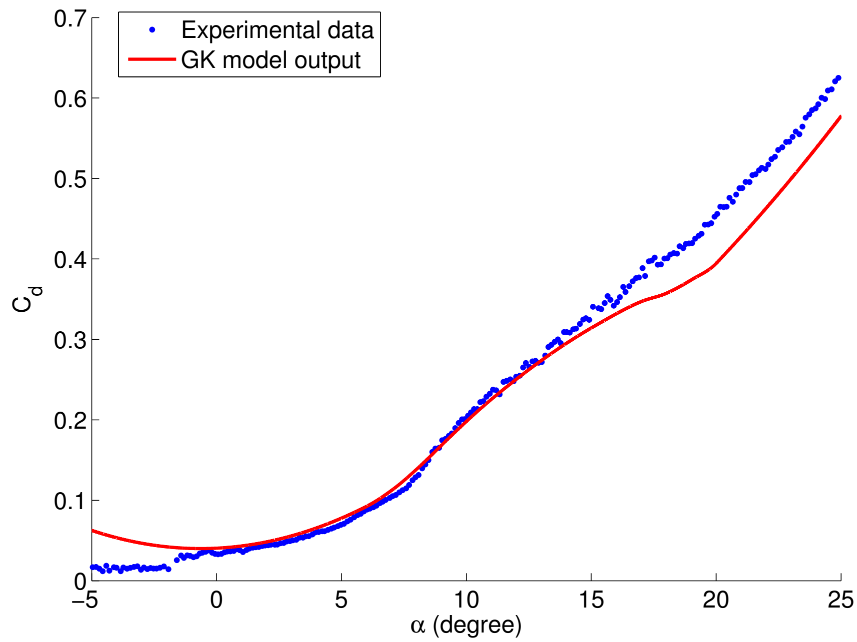

Figure 3.6. Comparison between the experimental and model quasi-steady drag

3.4 Model validation

While the ability to predict lift and drag based on separation can be useful, the real strength of the GK model resides in its ability to work on unsteady cases. The first step is to determine the 2 τ time constants. To do that a series of pitching cases are performed. The pitching inputs are the following

$$\alpha(t) = A \sin\left(2\pi \frac{t}{f}\right) + \alpha_0 \tag{3.7}$$

With $A = 2°$ and $\alpha_0 = 12°$. The frequency f is set to 0.25, 0.5, 1 and 2 Hz (respectively K of 0.064, 0.128, 0.257 and 0.513)

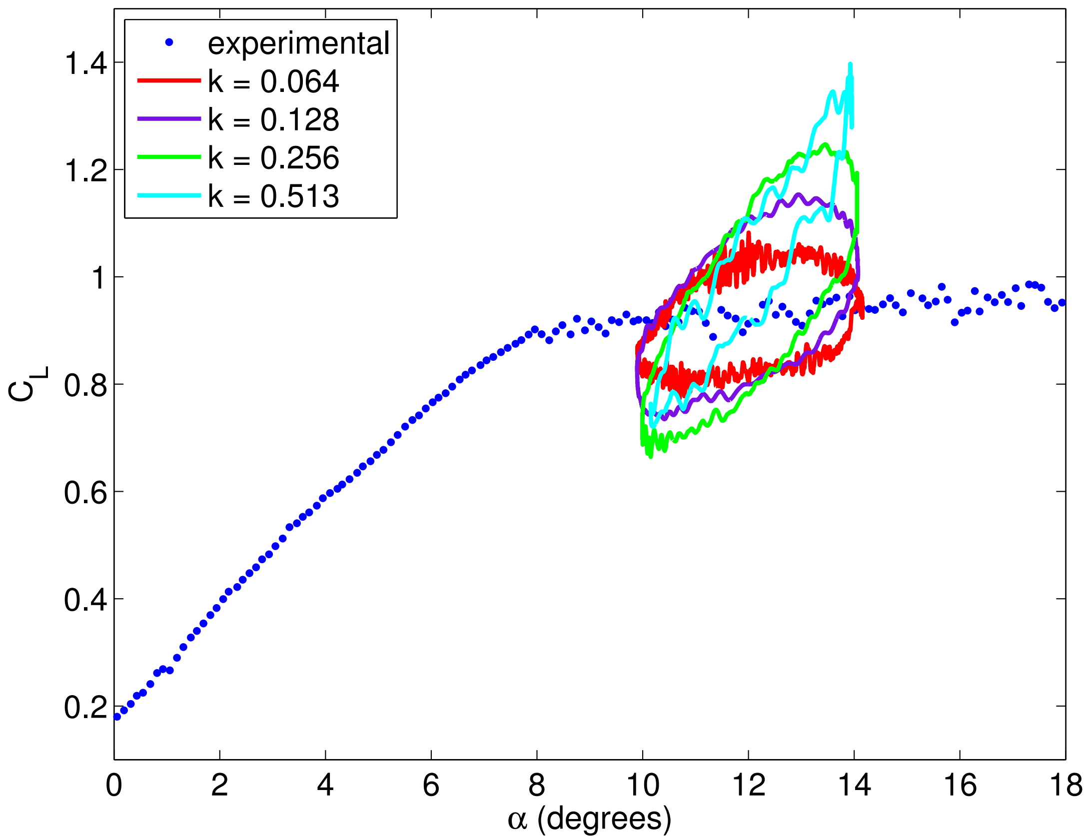

Figure 3.7. Unsteady effects on the lift of sinusoidal pitching around 12 degree

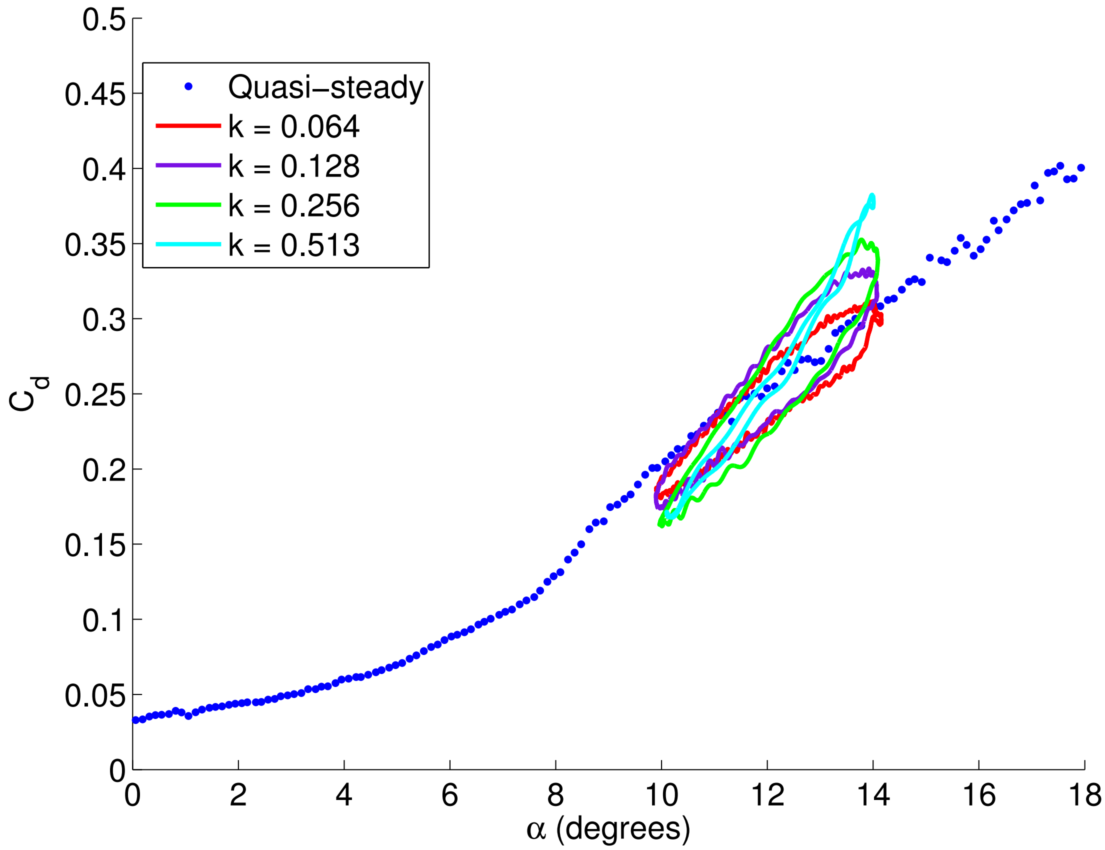

Figure 3.8. Unsteady effects on the drag of sinusoidal pitching around 12 degree

On these figures it is easy to notice the influence of the time delays on the aerodynamic coefficients. At the lower frequencies the loops are quite open and a significant difference exists between the lift obtained during the pitch up and the pitch down phase. These lift value circulate on these loops rotating in an anti-clockwise direction. This means that the lift is higher during pitch down maneuver. Contrastingly this behavior disappears at higher frequencies. For K values of 0.257 and 0.513 the difference between the pitch up and pitch down is lower. However the lift variation amplitude is more pronounced in those cases.

Before using the Gk model as a predictive tool the time constants need to be found. This is done by trial and error. The two time constants are determined manually and are tuned to produce the best results at the different frequencies tested. τ_1 is found to be equal to 3.06 t+ (0.25s) and τ_2 is 4.29 t+ (0.35s).

Theoretically the value of τ_1 could be found by analysing the output of small step input for the angle of attack. In this situation $\dot{\alpha}$ at the time the step is taken but it doesn't last long enough affect the value of the state variable since the first order differential equation for x acts as a low pass filter. This means that for a small step C_l from 12 to 13 degrees the output lift looks like the following figure.

Figure 3.9. Cl behavior for a instantaneous step from 12 to 13 degrees as simulated by the GK model

From this classical methods used to find the time constant for first order system can be used. As you can see in the figure 3.10, the τ_1 constant found from the GK model output is really close to the one used in the model.

Figure 3.10. Identification of τ_1 from the step response

While this method is fine in theory, it is impossible to implement experimentally. The force balance used to measure lift and drag is very fragile so a fast step could be enough to break it. Furthermore any slower of even smoothed step input modifies the lift response enough to make the time constant identification impossible. While this method is unpractical for experimental cases it should be applicable in the case of CFD simulations.

Now that the model is complete its accuracy can be checked. The most obvious result is that the shape of the lift and drag versus angle of attack curves are similar to the experimental results.

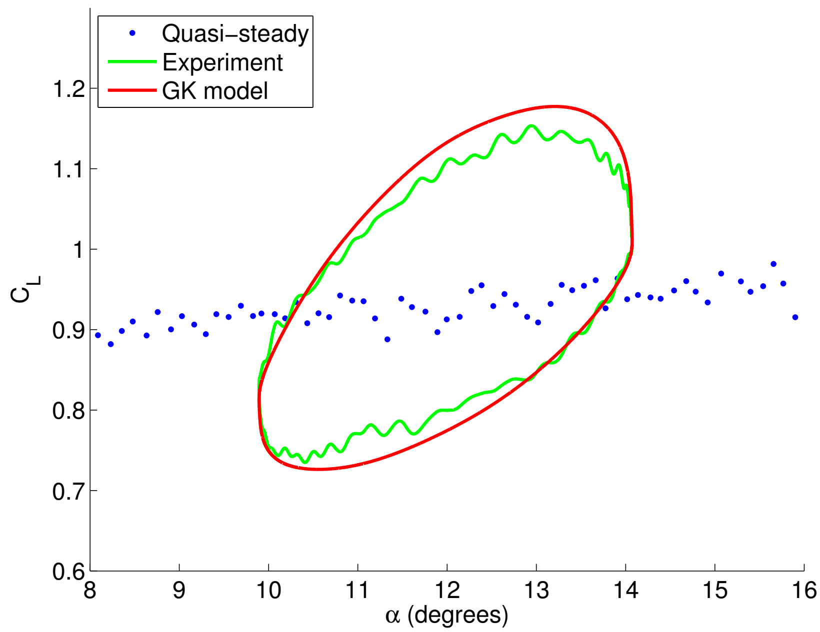

Figure 3.11. Comparison of experimental lift coefficient and model prediction after tunning of the time constant at k =0.128

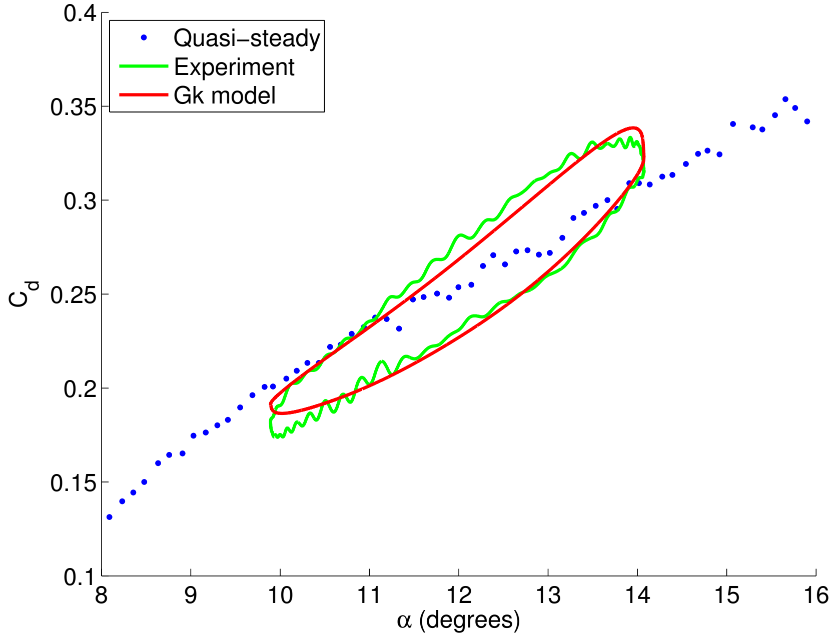

Figure 3.12. Comparison of experimental drag coefficient and model prediction after tunning of the time constant at k =0.128

This behavior can be checked for other frequencies as well.

Similarly another set of acquisitions is made at a mean angle of 10 degrees (see figures 3.15 and 3.16). The behavior is comparable at K of 0.257 and 0.513 but the drag has a noticeably different shape at K of 0.128. For some unknown reason the model does not seems to account for the hysteresis at this frequency.

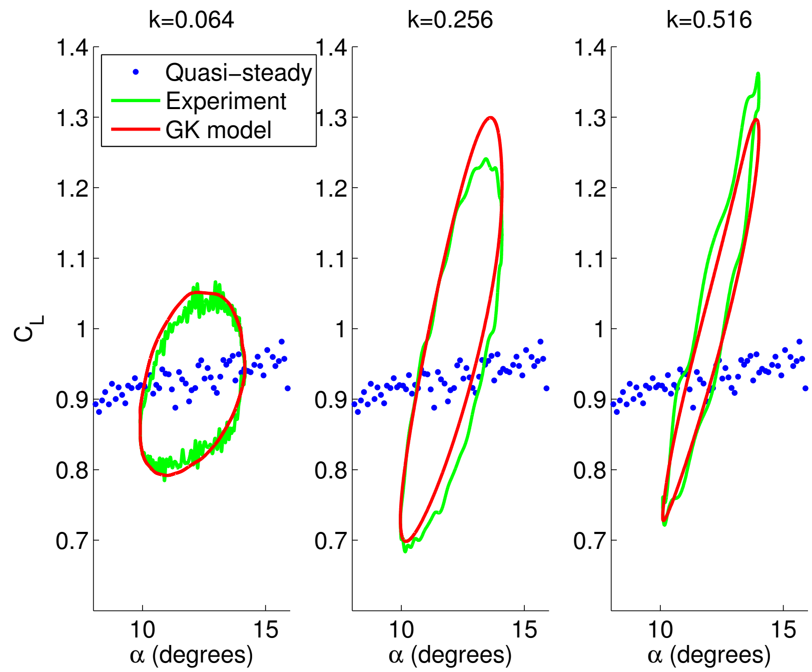

Figure 3.13. Lift measurement and prediction during sinusoidal pitching around 12 degree

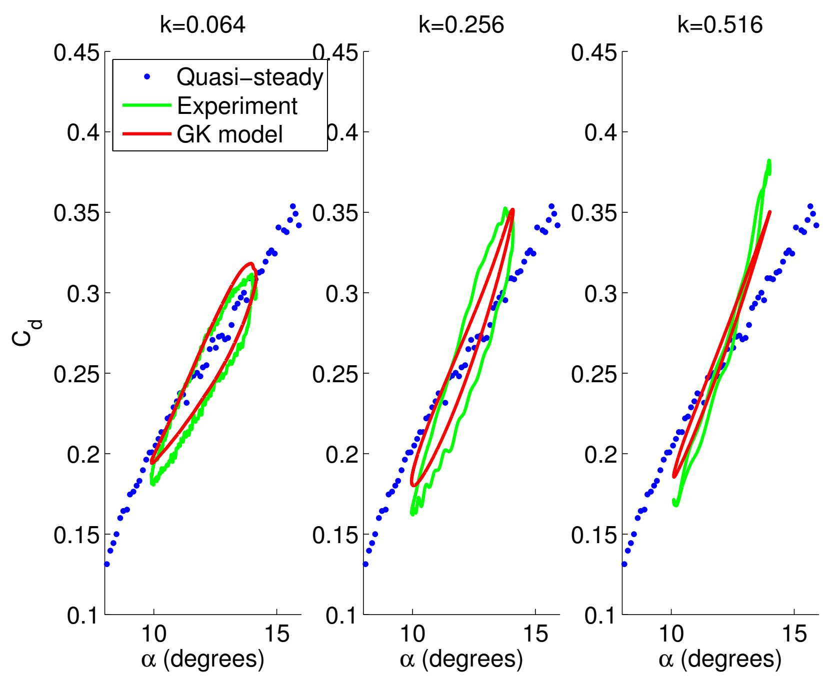

Figure 3.14. Drag measurement and prediction during sinusoidal pitching around 12 degree

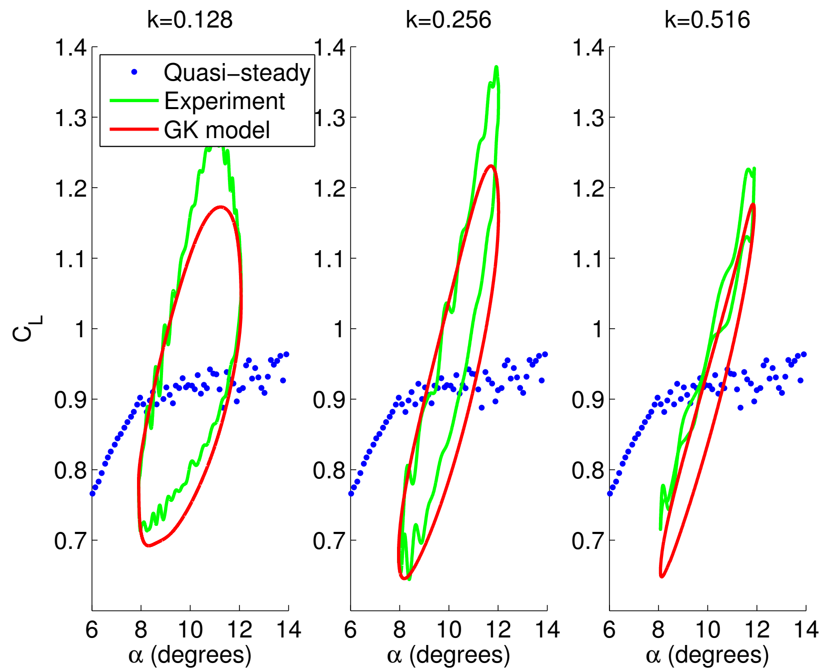

Figure 3.15. Lift measurement and prediction during sinusoidal pitching around 10 degree

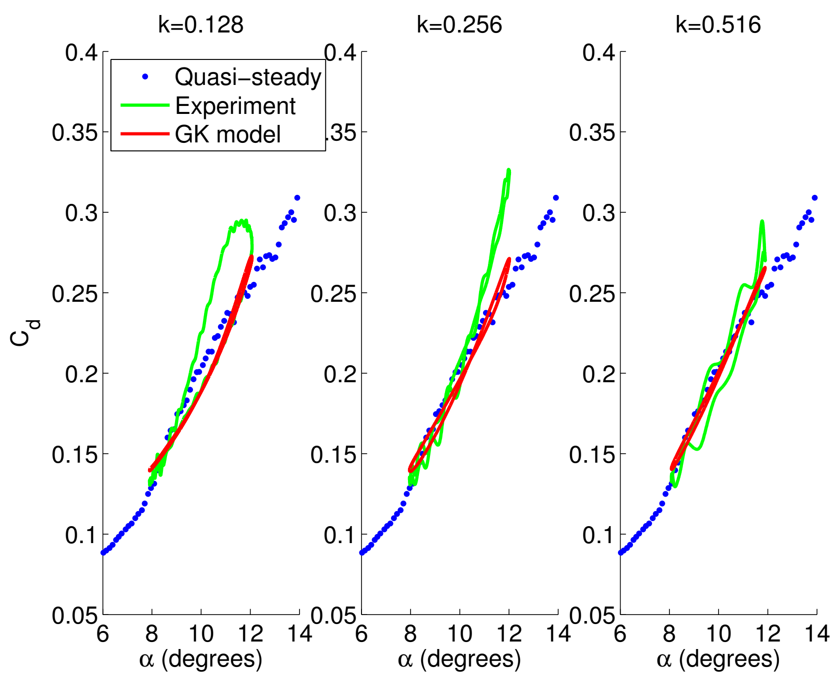

Figure 3.16. Drag measurement and prediction during sinusoidal pitching around 10 degree

Another obvious parameter to check for our model is the amplitude of the oscillations. The amplitude is set to a range from 1 to 4 degrees at different mean angle of attack. The predictions still reasonably match the experimental results.

To simulate a more realistic pitch profile, a pseudo-random pitch profile is designed. The input is constructed as seen in equation 3.8 with a randomized phase difference φ_i between each harmonic components.

$$\alpha_{random} = \frac{\sum_{i=1}^{10} \sin(\frac{2\pi t}{f_i} + \varphi_i)}{B} + \alpha_0 \tag{3.8}$$

The frequency are regularly spaced between 0.25 and 2Hz and the constant B is chosen to make sure the maximum deviation from the α_0 value is no more than 2 degrees. This is done so that the bandwidth of the input signal is limited.

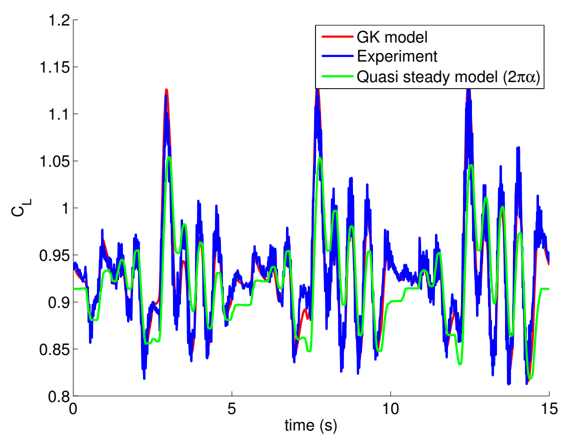

Figure 3.17. Unsteady effects of random pitching on the lift

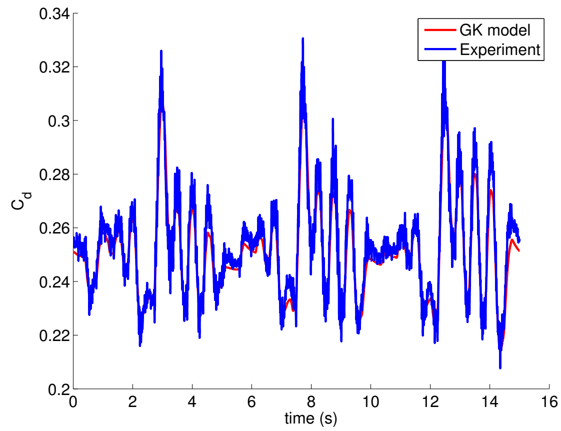

Figure 3.18. Unsteady effects of random pitching on the drag

This model is producing accurate results that account for both the dynamic effects and the flow separation. Moreover the procedure is light enough to be implemented into the optimization algorithm without increasing too much the computation cost

3.5 Implementation in the energy extraction algorithm

As seen in the previous chapter the optimization process for the energy extraction trajectory only requires a way to calculate the relationship between the lift and drag coefficient. Since in our case these 2 variables depend only on the angle of attack and its change rate over time, it is fairly easy to implement it into the algorithm. However the non dimensional time constant are different, the energy extraction one consider the optimal glide speed and the gravity acceleration whereas the one used for the GK model uses the flight speed and the chord length.

3.5.1 Relation between the different time scales.

As said before the time scale used in the two models are different. To solve this issue the ratio of the two time constants are plotted (see figure 3.19) for a wide variety of flying objects.

$$\frac{T}{t+} = \frac{V_{opt}}{g} \cdot \frac{U}{c} \tag{3.9}$$

Or if the aircraft flies near its optimal glide speed

$$\frac{T}{t+} = \frac{V_{opt}^2}{g \cdot c} \tag{3.10}$$

This ratio happens to be the Froude number.

Figure 3.19. T to t+ ratio for divers flying objects

It is interesting to notice that this ratio is in the same order of magnitude for all these objects. The value of 90 is chosen as a default for this ratio as it represent a good average of the data compiled.

Another issue is that the GK model is dependent on the initial value of the state variable x. The initial value of x is taken as the quasi-steady value. To minimize the effect of the transition from quasi-steady to unsteady flow at the beginning of the maneuver, the cycle is simulated twice and then only the second cycle is considered. This is possible to do since the conditions applied on the trajectory constrain the initial and final angle of attack to be the same.

3.5.2 Effects of considering unsteady aerodynamics on the optimal trajectory.

The unsteady model is expected to produce two effects on the lift and drag characteristics that can influence the optimization process. The first is the that the unsteady lift can reach values that are impossible to achieve in the quasi-steady case. If the airfoil is pitched up fast enough the flow doesn't have the time to separate and high values of Cl can be attained. The second effect is the delay between the angle of attack variations and the flow response.

If the exact same constraints are used for optimization the problem illustrated by figure 3.20.

¡++¿

Figure 3.20. Optimization for vertical wind gusts with the same constraints as the previous cases

The issue is that the in order to reach the favorable high lift regions the algorithm produces a

Figure 3.21. Cl versus alpha for different cases (shows the unsteadyness)

CHAPTER 4

CONTROLLER DESIGN AND VALIDATION

4.1 Active flow control

4.1.1 Basic principle.

4.1.2 Input signal.

4.1.3 Experimental setup.

4.2 Proof of concept for combining AFC with pitching

4.2.1 Open loop experimental results.

As seen in the previous chapter the lift resulting from unsteady pitching present some significant hysteresis.

4.2.2 Comparison with models output.

4.3 Controller structure

4.3.1 AFC transfer function inverse.

4.3.2 Controller input and objective discussion.

4.3.3 Controller block choice.

4.4 Results

4.4.1 Periodic pitching motion.

4.4.2 Limited bandwidth random pitching motion.

4.5 Discussion of the results

4.5.1 Bandwidth limitation.

4.5.2 Precision issues.

4.6 Possible improvements and feedback implementation

CHAPTER 5

CONCLUSION

5.1 Summary

This was just to create a sample section...

APPENDIX A

GOMAN KHRABROV MODEL MATLAB ®IMPLEMENTATION

Your Appendix will go here !

APPENDIX B

NAME OF YOUR SECOND APPENDIX

Your second appendix text....

BIBLIOGRAPHY

[1] Thomas Albrecht, Tom Weier, G Gerbeth, Bruno Monnier, and David R Williams. Separated flow response to single pulse actuation. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2013.

[2] Michael J Allen and Victor Lin. Guidance and control of an autonomous soaring vehicle with flight test results. 867, 2007.

[3] Daniel J Edwards. Implementation details and flight test results of an autonomous soaring controller. *North Carolina State University*, 2008.

[4] M Goman and A Khrabrov. State-space representation of aerodynamic characteristics of an aircraft at high angles of attack. *Journal of Aircraft*, 31(5):1109–1115, 1994.

[5] Wesley Kerstens, Jens Pfeiffer, David Williams, Rudibert King, and Tim Colonius. Closed-loop control of lift for longitudinal gust suppression at low reynolds numbers. *AIAA journal*, 49(8):1721–1728, 2011.

[6] PBS Lissaman and CK Patel. Neutral energy cycles for a vehicle in sinusoidal and turbulent vertical gusts. *AIAA paper*, 2007.

[7] Peter Lissaman. Wind energy extraction by birds and flight vehicles. *AIAA paper*, 241, 2005.

[8] A Le Pape, M Costes, G Joubert, F David, and J-M Deluc. Dynamic stall control using deployable leading-edge vortex generators. *AIAA journal*, 50(10):2135–2145, 2012.

[9] Yiyuan J Zhao. Optimal patterns of glider dynamic soaring. *Optimal control applications and methods*, 25(2):67–89, 2004.