

Low Order Modelling of Flapping Wing Aerodynamics for Real-Time Model Based Animation of Flapping Flight

A dissertation submitted to The University of Manchester for the degree of M.Sc. in
Theoretical and Applied Fluid Dynamics in the Faculty of Engineering and Physical
Sciences

2005

Ben Parslew

School of Mathematics

CONTENTS

Contents.....	1
List of Figures.....	3
List of Tables.....	4
Nomenclature.....	5
Subscripts.....	6
Abbreviations.....	6
Abstract.....	7
Declaration.....	8
Copyright Statement.....	9
Acknowledgements.....	10
1. Introduction.....	11
1.1 Aims and Objectives.....	11
1.2 Success Criteria.....	12
1.3 Project structure.....	12
2. Background Information.....	14
2.1 Historical Background.....	14
2.2 Modelling for Animation.....	15
2.3 Recent Developments.....	17
2.4 Aerodynamic Modelling.....	19
2.5 Summary.....	23
3. Theory.....	24
3.1 Research Program.....	24
3.2 Definitions and Assumptions.....	25
3.2.1 Axes and Angles.....	25
3.2.2 The Order of Rotations.....	28
3.2.3 Blade-Element Theory.....	31
3.2.4 Sinusoidal Motions.....	32
3.2.5 The Gaits of Motion – Flight Regimes.....	33
3.2.6 Power Consumption.....	35
3.3 Base-Model Design.....	37
3.3.1 Numerical Method.....	37
3.3.2 Base-Model Functionality.....	37
3.4 Preliminary Analysis - The Advance Ratio.....	39
3.4.1 Flapping Modes.....	40
3.4.2 Translating Wing.....	40
3.4.3 Translating, Rotating Wing.....	42
3.4.4 Cruise and Hover Regimes.....	45
3.5 Flapping Kinematics.....	48
3.5.1 Velocity Distribution and Control Point Location.....	48
3.5.2 Base Kinematic Model.....	52
3.5.3 Kinematic Model 1.....	56
3.5.4 Kinematic Model 2.....	58
3.5.5 Kinematic Model 3.....	61
3.5.6 Kinematic Model 4.....	64
3.5.7 Kinematic Model 4 – Parameter Range.....	69
3.6 Optimisation Methods.....	72

3.6.1	Complete Search Algorithm.....	72
3.6.2	Gradient Based Methods.....	74
3.6.3	Genetic Algorithm.....	76
3.6.4	Method comparison.....	79
3.6.5	Method Selection.....	80
4.	Results.....	82
4.1	Numerical and Physical Results.....	82
4.1.1	Base Kinematic Model Solution.....	82
4.1.2	Kinematic Model 1 Solution.....	100
4.1.3	Kinematic Model 3 Solution.....	102
4.1.4	Kinematic Model 4 Solution.....	106
4.2	Visualisation and Performance.....	115
4.2.1	Animations.....	115
4.2.2	Pathlines.....	119
4.2.3	Looking Ahead.....	120
4.2.4	Performance.....	121
5.	Conclusions and Further Work.....	122
5.1	Conclusions.....	122
5.2	Further Work.....	125
	Appendices.....	128
	Appendix 1a - Base-Model.....	128
	Appendix 1b - Source Code, SIMULINK Model and Solution.....	136
	Source Code.....	136
	SIMULINK Model.....	139
	Solution Contour Plots From Supplied Source Codes.....	140
	Appendix 2 - Genetic Algorithm Source Code.....	141
	Appendix 3 - Solutions for Varying Distance-Cost Weighting.....	146
	Appendix 4 - Inappropriate Initial Guess Values for Varying Velocity.....	147
	Appendix 5 - Solutions for varying F_x and F_z	148
	Varying F_x	148
	Varying F_z	149
	Appendix 6 - Fully Populated Results for Kinematic Model 1.....	150
	Appendix 7 - Fully Populated Results for Kinematic Model 3.....	151
	Appendix 8 - Fully Populated for Kinematic Model 4.....	152
	Bibliography.....	153

Final word count, including footnotes and endnotes: 30,590

LIST OF FIGURES

Figure 1: Model-based animation research program	24
Figure 2: Axis Systems	26
Figure 3: Bird geometry. Motion angles	27
Figure 4: Effect of applying rotation matrices in different orders.....	30
Figure 5: Flight regimes	34
Figure 6: Pseudo-code for the function <code>flapcost</code>	38
Figure 7: Uncambered wing undergoing harmonic vertical translation	41
Figure 8: Uncambered wing undergoing harmonic translation, and twisting	42
Figure 9: Variation in thrust force F_x , with advance angle and twist angle	44
Figure 10: Wing motion, velocity and force distribution.....	49
Figure 11: Base-model translating wing motion	53
Figure 12: Force and power results from base kinematic model ('hovering').....	54
Figure 13: Force and power results from base kinematic model ('cruising')	55
Figure 14: Kinematic model 1 flapping wing motion.....	56
Figure 15: Force and power results from kinematic model 1 ('hovering').....	57
Figure 16: Force and power results from kinematic model 1 ('cruising')	58
Figure 17: Kinematic model 2 flapping wing motion.....	59
Figure 18: Force and power results from kinematic model 2 ('hovering').....	60
Figure 19: Force and power results from kinematic model 2 ('cruising')	60
Figure 20: Kinematic model 3 flapping wing motion.....	62
Figure 21: Force and power results for kinematic model 3 ('hovering').....	63
Figure 22: Force and power results from model 3 ('cruising')	63
Figure 23: Kinematic model 4 flapping wing and wrist motions	67
Figure 24: Force and power results from kinematic model 4 ('hovering').....	67
Figure 25: Force and power results from model 4 ('cruising')	68
Figure 26: Variation of $F_z(\max)$ with Δ and Ψ	70
Figure 27: Lag and bias values to provide maximum $F_{z(\max)}$	70
Figure 28: CSA solution for optimisation.....	73
Figure 29: GBM solution for optimisation.....	75
Figure 30: GA solution for optimisation.....	78
Figure 31: Variation of predicted solution minima with no. iterations.....	80
Figure 32: Base-model solution for varying velocity	83
Figure 33: Optimisation spaces for $V=0.2, 0.7$ and 3.0	84
Figure 34: Optimisation spaces with distance cost weighting	86
Figure 35: Base-model solution for varying velocity with updated initial guess.....	88
Figure 36: Cost fractions for varying velocity solution of base-kinematic model	89
Figure 37: Base-model solution for Φ_f using underrelaxation	91
Figure 38: Successful base-model solution for forward and backward passes	92
Figure 39: Pseudo code and depiction of the process of populating the solutions.....	94
Figure 40: Populated lookup tables for base-model solution.....	95
Figure 41: Control point paths and chord lines of base-model during hovering.....	97
Figure 42: Control point paths and chord lines of base-model during take-off	98
Figure 43: Control point paths and chord lines of base-model during cruise	99
Figure 44: Comparison of base kinematic model and model 1 solutions	100
Figure 45: Kinematic model 3 solutions for varying V	103

Figure 46: Control point paths and chord lines of model 3 during hover	104
Figure 47: Control point paths during takeoff.....	104
Figure 48: Control point paths of model 3 during cruise	105
Figure 49: Variation in force F_z with wrist activity parameter and body pitch.....	107
Figure 50: Optimisation spaces for kinematic model 4	108
Figure 51: Optimisation spaces for kinematic model 4	109
Figure 52: Kinematic model 4 solutions for varying V	110
Figure 53: Control point paths during hover	112
Figure 54: Control point paths during takeoff.....	112
Figure 55: Control point paths during cruise.....	113
Figure 56: Hovering animation sequence for base kinematic model.....	117
Figure 57: Hovering animation sequence for kinematic model 4.	117
Figure 58: Cruising animation sequence for base kinematic model.	118
Figure 59: Cruising animation sequence for kinematic model 4.....	118
Figure 60: Pathlines for control points in the base-kinematic model during cruise .	119
Figure 61: Pathlines for control points in model 4 model during cruise	119
Figure 62: Simulated landing manoeuvre for kinematic model 4	120
Figure 63: Computation times to populate results arrays	121

LIST OF TABLES

Table 1: Aerodynamic method evaluation	22
Table 2: Rotation matrices	29
Table 3: Comparison of errors and iterations for different optimisation methods	79

NOMENCLATURE

a	acceleration vector
A	advance ratio
c	wing chord
C_d	wing drag coefficient
C_{d0}	wing profile drag coefficient
C_{dmax}	maximum drag coefficient
C_{dbody}	body drag coefficient
C_F	force coefficient
C_l	wing lift coefficient
C_{lmax}	maximum lift coefficient
C_{Vn}	distance-cost function
D	drag force
f	quadratic test function
F	force vector
g	harmonic test function
i	F_z iteration counter
j	F_x iteration counter
k	velocity counter
K	wrist activity coefficient
L	lift force
m_w	wing mass
N	no chromosomes
O	Order of magnitude
q	dynamic pressure
r	radial direction
R	rotation matrix
s	wing semispan
t	time
T	time period
V	velocity vector
V_∞	freestream flow velocity
w_f	wing flapping velocity
x,y,z	Cartesian coordinate system in body fixed axes
X,Y,Z	Cartesian coordinate system in Earth fixed axes
α	angle of attack of local wind
γ_f	wing sweep angle
Γ_f	wing sweep amplitude
δ	underrelaxation parameter
δV	velocity increment
Δ	phase lag
Δ^*	phase lag for maximum force
ε	small perturbation
ζ	advance angle
θ	body pitch attitude
θ_f	wing twist angle

Θ_f	wing twist amplitude
θ_{sp}	sweep plane angle
θ_w	hand twist angle
Θ_w	hand twist amplitude
v	distance cost function weighting
ρ	air density
φ_f	wing flap angle
Φ_f	wing flap amplitude
φ_w	hand flap angle
Φ_f	hand flap amplitude
Ψ	bias
Ψ^*	bias for maximum force
ω	frequency
Ω	angular velocity

SUBSCRIPTS

1	in body axis system
2	in wing axis system
3	in local axis system
e	in earth axis system
f	relating to motion of the shoulder
w	relating to motion of the wrist
max	maximum parameter value
min	minimum parameter value

ABBREVIATIONS

BAS	Body Axis System
EAS	Earth Axis System
WAS	Wing Axis system
CP	Control Point
CSA	Complete Search Algorithm
GBM	Gradient-Base Methods
GA	Genetic Algorithm
SQP	Sequential Quadratic Programming
CFD	Computational Fluid Dynamics

ABSTRACT

This dissertation describes the development of physics-based models to be used in the analysis of flapping flight. A method of producing realistic animations from numerical solutions is given for generic bird models with various levels of complexity.

Computational methods are employed to determine aerodynamic forces arising from the flapping motion of bird wings. Input parameters that dictate a desired flight path are supplied at the user level, and optimisation methods are invoked to determine the bird movement that is most suited to the specified input.

This work balances scientific analysis and model-based animation to give a thorough view of flapping flight. Results are presented in the form of numerical data and visual simulations, and show that a physics-based model can be developed to accurately simulate avian motion. The project concludes that strategic implementation of numerical methods can produce realistic animations with strong physical foundations.

DECLARATION

No portion of the work referred to in the dissertation has been submitted in support of an application for another degree or qualification of this or any university or other institute of learning.

COPYRIGHT STATEMENT

Copyright in text of this dissertation rests with the author. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the author. Details may be obtained from the appropriate Graduate Office. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without permissions (in writing) of the author.

The ownership of any intellectual property rights which may be described in this dissertation is vested in the University of Manchester, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the University, which will prescribe the terms and conditions of any such agreement.

Further information on the conditions under which disclosures and exploitation may take place is available from the Head of the School of Mathematics

ACKNOWLEDGEMENTS

I would like to thank Dr. William J. Crowther for his supervision and ongoing support throughout the course of this project. Gratitude must also go to my friends and family for their patience and advice during my postgraduate studies.

1. INTRODUCTION

Over the past 20 years, computer animation has advanced significantly in modelling animal locomotion. Studies have investigated the motion of terrestrial, aquatic and avian creatures in order to recreate natural behaviour. Traditional methods focussed on scripting motion paths and key framing to achieve realistic animations; however, recent studies have aimed to incorporate a more scientific approach in producing simulations.

The motion of animals in flight is perhaps one of the most intriguing phenomena observable in nature. Underlying the grace and elegance of flying creatures, are a variety of complicated scientific principles that are still undergoing research. However, strategic simplifications can lead to the development of physics-based methods of modelling flight. This project will aim to produce a technique to simulate flapping flight in order to produce realistic animations, and also further the understanding of avian motion.

1.1 AIMS AND OBJECTIVES

This project will focus on the development of physical models to be incorporated into model-based animation software. A supplied *base-model*¹ will be used as a starting point for the analysis and will provide a shell, in which newly constructed physical models can be tested and evaluated. The analysis will aim to validate, or otherwise, the solution from the base-model, and extend this system to consider other models with increasing degrees of complexity. Models will be constructed, tested and used to obtain solutions with the objective of enhancing the quality of simulations of animated bird flight.

¹ The 'base-model' refers to unpublished work produced by Dr. William J. Crowther at the University of Manchester (2003). Information can be found in Appendix 1, and will be described as necessary throughout the project.

1.2 SUCCESS CRITERIA

It is important to consider the ways in which this dissertation may be deemed successful. Success criteria for this project will relate largely to the quality of results produced. However it is also useful to consider the success of more general aspects of the project such as the way in which it is implemented, the analysis of results, efficient use of time etc. The following criteria will be used to measure the success of this project:

- Results accuracy: results are considered accurate, in that they match expectations, similar works, or intuitive reasoning
- Analysis methods: results are assessed objectively using well founded techniques
- Time management: an appropriate amount of time is spent on each area of the project
- Meeting aims and objectives: the solution and analysis match the project aims and objectives
- Design methodology: the solution is obtained in a structured, coherent fashion

1.3 PROJECT STRUCTURE

A brief outline of the project structure will now be given:

BACKGROUND INFORMATION

The background information section will develop understanding of the problem domain. Discussions of historical background to the project, and similar work in this field will be made.

THEORY

The current research into model-based animation will be described. This chapter will describe in detail, the definitions and assumptions that will be made throughout this project. The design of the base-model will be given, and some preliminary analysis

will be used to establish the problem more thoroughly. Details of kinematic models and optimisation methods will be discussed at length.

RESULTS

The results chapter will focus firstly on presenting the numerical solutions to the problem, with detailed analysis of the resulting data. This will lead to production of visual results in the form of computer animations, which will also be discussed, along with the system performance details.

CONCLUSION AND FURTHER WORK

Finally, the findings of the project will be summarised and conclusions will be made. The project will be analysed in terms of the chosen success criteria. Suggestions of suitable continuations to the project will be given.

2. BACKGROUND INFORMATION

The aim of this chapter is to highlight important studies in the field of flapping flight and therefore establish the context of this particular project. The strengths and weaknesses of previous analyses will be used as a means of determining a suitable project strategy. The reader will be provided with a firm understanding of the problem domain and the reasons for the author's choice of design methodology. This chapter will give a critical analysis of the chosen literature with regards to the key factors affecting this project; a significant area for discussion will be the selection of a suitable aerodynamic model. A summary of the key points obtained from the literature will be given at the end of the chapter.

2.1 HISTORICAL BACKGROUND

Some of the earliest attempts to analyse the physical phenomenon of flapping flight date back to the beginning of the 20th century. Aerodynamicists such as von Karman & Burgers (1934) attempted to explain the mechanism of flapping wing propulsion, while quantitative analysis from Weis-Fogh (1956) studied the methods by which insects are able to perform sustained flight. Applied mathematicians such as Lighthill (1987) produced a substantial quantity of work on the theoretical reasoning behind flapping flight, and comparisons with results obtained from observation.

More recently, Ellington (1984) explained many of the key principles involved in flapping wing aerodynamics, with a well documented comparison between analytical techniques and experimental data. The series of papers presented by Ellington (1984) highlights important physical characteristics in the study of hovering insects, which can be readily extended to avian flight. These results were used as the founding to the base-model work (see Appendix 1), from which this project was derived.

Ellington illustrates some crucial factors regarding the nature of the aerodynamics of flapping wings. In addition to this, useful accounts of flapping kinematics and power

expenditure contribute to the general understanding of the subject. Many of these principles will be referred to throughout the course of this project.

2.2 MODELLING FOR ANIMATION

The first example of physics-based models of bird flight is seen by Ringham & House (1997) in an attempt to simulate behavioural flocking. This work considered flapping motions of individual birds to be incorporated into standard flocking algorithms. The goal of the investigation was to enhance the quality of flocking animations, with less interest in rigorous scientific analysis.

The overall approach taken by Ringham & House (1997) utilises a three-level control system to define the motion of an individual bird. High level decision making is used to determine desired directions, speeds and altitudes for each bird. The method then uses predetermined values of various kinematic parameters to derive forces that match the desired input.

The kinematics used by Ringham & House (1997) are somewhat limited in that motions are described purely from field observations of wing paths. It appears that flapping frequency, angle of attack and lift efficiency factor are the only independent variables assumed to define the wing motion. While some discussion is made on the ability to sweep wings, and also to reduce wingspan, it is unclear how these factors are determined.

The simplification of aerodynamics and the method of obtaining motion parameters from precomputed lookup tables offers a suitable level of model detail without impeding the overall performance. The original approach used by Ringham & House (1997) has inspired others to investigate further the incorporation of physical principles into computer animation of birds.

The next major advance in animating flapping flight is seen in the paper of Ramakrishnananda & Wong (1999) in modelling forward flight of birds. This work adopts a more sophisticated bird geometry, with multi-jointed wings offering degrees

of freedom that mimic those found in nature; a useful physiological description to support the mechanical wing model is included.

As seen in the work by Ringham & House (1997), the physical methods included by Ramakrishnananda & Wong (1999) are used in determining the dynamics of the bird as a complete system, with clear descriptions of inertial effects. While this may be a key factor in producing realistic animations, it leaves less focus on determining accurate wing motions: all of the governing wing beat parameters are user-defined and have been given values deemed appropriate by the authors.

Ramakrishnananda & Wong (1999) calculate aerodynamic forces by deriving local wind incidence angles, in conjunction with an undisclosed lift-curve slope to determine normal and tangential force coefficients. The wing geometry is such that the cross-section varies along the span and hence solutions must be numerically integrated to establish the net force on the wing, which appears to offer suitable accuracy for a problem of this nature.

One of the key points to arise from the work by Ramakrishnananda & Wong (1999) is the method of targeting parameters to achieve desired solutions. Given a series of target locations, the system attempts to derive the appropriate model parameters that will produce the desired flight path. The body pitch angle is the fundamental parameter used in controlling the model behaviour; it is adjusted so that aerodynamic forces will drive the body along the desired path. Similar methods could be employed with a greater number of free parameters, to target a more general bird motion to a desired solution.

There does appear to be some imbalance between the high complexity of the wing mechanics, and the simplicity of targeting only the body pitch (or wing sweep) by Ramakrishnananda & Wong (1999). However, the work should be commended for offering some useful guidelines in the design methodology: obtaining solutions to simplified problems, followed by a gradual increase in the level of model sophistication.

Finally, with regards to performance, the solution times found in Ramakrishnananda & Wong (1999) are substantially longer than what would be desired in this project. The efficient method of using predetermined parameter values wherever possible, as mentioned in Ringham & House. (1997), is still favoured.

2.3 RECENT DEVELOPMENTS

The most recent developments in the field of study are seen in the work of Wu & Popović (2003), in a comprehensive paper on realistic modelling of bird flight animation. This work can be regarded as the closest example to the base-model and also the current project. Wu & Popović (2003) describe the numerical method of calculating a series of wingbeats to enable birds to follow specified trajectories. The overall model aims to avoid mimicking observed avian motion, by concentrating on the development of an accurate physical system.

The bird is modelled as a highly sophisticated articulated skeleton with 21 DOFs; with the wings accounting for 12 of these (6 DOFs per wing). Wu & Popović (2003) state the importance of correctly modelling the wings, and describe their model as a “full representation”; although the ability of the outer-wing section to perform a sweeping motion as seen in nature (and included in Ramakrishnananda & Wong, 1999) is not accounted for. While the overall mechanical skeleton has a large number of DOFs, it is important to note that within the model some parameters, such as the “forearm twist” and “feather spread”, are algebraically linked and are not represented as completely independent.

This is the first study found, where individual feathers have been included as part of the physical makeup of the bird. Wu & Popović (2003) justify the use of feathers as critical for determining a bird’s ability to fly; the validity of this approach when developing low-level physical modes is questionable. The modelling of feathers is closely linked to observed physiology; however it yields a massive increase in the complexity of the system, which seems unfit for rigorous scientific analysis. A more desirable model would be designed for general flapping flight so that un-feathered species such as insects and bats could also be considered.

Wu & Popović (2003) employ composite functions, based on observations from literature, to describe the motion of the various wingbeat parameters. A novel method of adjusting the duration of upstroke and downstroke is used to allow more freedom to the mechanical system².

Solutions to the problem are obtained by Wu & Popović (2003) using a simulated annealing process³ to determine an optimal set of wingbeats that will allow a bird to follow a specified path. Wu & Popović (2003) clearly appreciate that while a low level of user intervention (in defining motion parameters) is desirable, using many free DOFs increases the dimensionality of the search space and hence inhibits the optimisation performance. However, the choice of free and prescribed parameters seems unusual: the wrist and elbow bend mechanisms (which are believed crucial in analysing flapping wings, Baumel, 1993) are prescribed, while the forearms twist (that generally occurs in conjunction with shoulder twist, Baumel, 1993) is a free parameter.

The optimisation process of Wu & Popović (2003) is performed by associating a cost with a particular wingbeat, and minimising this value by adjusting the various free parameters using simulated annealing. The cost function evaluates how closely the bird flies to the specified trajectory, to produce the desired graphical output. However, Wu & Popović (2003) also apply costs to measure the “gracefulness of motion”; this constraint oversimplifies the optimisation and infringes the underlying physics of the problem. This project would be more inclined towards producing *true* physical representation, with detailed analysis of motions deemed ‘ungraceful’.

The optimisation procedure of Wu & Popović (2003) requires computation time of 3-5 hours per simulation, in order to produce solutions for flight trajectories lasting around 10 seconds (using a 2.8GHz Pentium 4 PC). While the animations produced are of high quality, this amount of time would be impractical for purposes of analysis.

² Wu & Popović (2003) include free solution parameters to represent the fractions of time spent on the upstroke and downstroke during a single oscillation.

³ Simulated annealing is a solution technique used in combinatorial optimisation, that is based on the Monte Carlo search method (Mitchell, 1998)

The final point to be considered in the work of Wu & Popović (2003) is the aerodynamic method. While Ringham & House (1997) and Ramakrishnananda & Wong (1999) both immediately assumed force-incidence relationships for wings, Wu & Popović (2003) consider the possibility of other methods. Computational Fluid Dynamics (CFD) is mentioned as the “most accurate method”, but is dismissed due to computational expense. While numerous other techniques could be applied, no others are considered. This project will aim to use a more careful selection process for the aerodynamic model.

Wu & Popović (2003) determine the aerodynamic forces by applying experimental data to define the lift and drag coefficients. The method derives the angle of attack for each individual feather and applies experimental data obtained for complete wings; this method does not seem sufficiently accurate. Furthermore, where experimental data was unavailable, Wu & Popović (2003) employ “synthesised” functions for force coefficients that prescribe unrealistic lift coefficients of approximately 1, in the angle of attack range 10° to 90° .

The work of Wu & Popović (2003) should be credited for the quality of animations produced. However, the intent of creating a physics-based model certainly lacks balance between the complexity of the physical model and aerodynamic method. A relatively high degree of understanding of bird physiology was required to achieve feasible results, and the overall design methodology is less focussed on rigorous scientific analysis. This project will aim to avoid the problems highlighted in the work of Wu & Popović (2003).

2.4 AERODYNAMIC MODELLING

In order to establish a suitable design methodology for this project, it is important to evaluate the possible techniques that could be employed to solve the particular problem. While the aerodynamic principles are clearly recognised as a crucial physical aspect of modelling flight, previous research has neglected to consider many possible aerodynamic methods that could be used for animation purposes. This

section will aim to give a brief discussion of four common aerodynamic methods that could be used to model flapping flight.

Methods will be assessed in terms of their applicability to this project. A general selection process will be made based on relevant factors: accuracy, performance, implementation time, stability/user intervention requirements and the level of physical information obtained from the solution. A summary of these factors will be given towards the end of the section (Table 1).

BLADE-ELEMENT THEORY

Blade-element theory provides a method of determining forces and moments on a blade/wing by considering the aerodynamic properties of the cross-section (Filippone, 2003). Aerofoil data can be applied to predict force coefficients along the wing for given angles of attack.

The blade-element theory relies on a simple predefined lookup-function/table, and is therefore highly efficient. Given suitable aerofoil data, results can be reasonably accurate, though the standalone method does not account for the flow wake (downwash effects, induced velocity etc.). It imposes a quasi-steady⁴ assumption on the flow field and is therefore somewhat limited in terms of physical accuracy. However, this technique is simple to implement and has been used successfully for many years in the aerodynamic analysis of propellers (Filippone, 2003). It also offers high stability and a low level of user-intervention in computation.

The blade-element theory has been used in most previous works focussed on producing bird flight animation (Ringham & House, 1997, Ramakrishnananda & Wong, 1999, Wu & Popović, 2003). For a full description of the blade-element theory the reader should refer to the work of Filippone (2003).

⁴ The quasi-steady assumption assumes that the instantaneous forces on an aerofoil in unsteady motion are assumed to be those corresponding to steady motion at equal instantaneous velocity and attitude (Ellington, 1984).

VORTEX METHODS

Vortex methods describe an aerofoil and its wake by a series of vortex lines or surfaces. Common vortex theory involves manipulation of the inviscid Biot-Savart⁵ equations in order to track the main patterns of the vortex system. This provides information of the flow around the aerofoil and into the wake, and hence allows the induced drag of the system to be calculated.

A good example of the application of vortex methods to flapping wings is seen by Hall & Hall (2000), which contains a somewhat involved analytical process to consider the unsteady flow behaviour. Results are obtainable to a high level of accuracy, largely due to the resolution of the flow wake. The required time-marching approach is generally successful with relatively large discretization steps, and hence low computation time. Unfortunately this method requires a substantial amount of pre-processing in order to prescribe the geometric configuration and numerical method.

Previous work using vortex methods in a model-based animation can be seen by Rayner & Gordon (1998).

PANEL METHODS

Panel methods are a subset of the general class of potential flow methods⁶ that have been developed to analyse flow past arbitrary bodies in 2 and 3 dimensions. The ‘simple’⁷ panel method discretizes a body into a series of panels which are used to define vorticity and velocity variations around the body surface in order to calculate forces and moments accordingly.

Panel methods are relatively straightforward to implement, and the solution obtained from purely geometric properties is generally stable. The level of accuracy depends largely on the class of flow being modelled, with unsteady flows past wing

⁵ The Biot Savart law relates velocity induced by a vortex filament to its strength and orientation; it is commonly used when analysing 3d potential flows in order to track vortex systems (Filippone, 2003).

⁶ Potential flow methods use Laplace’s equation to define mass continuity

⁷ The ‘simple’ method refers to the lowest order methods: placing discrete vortices at panel quarter-chord points to determine forces, with no extension to flow field resolution

geometries posing as potentially difficult; examples of successful implementation of panel methods into flapping wing analysis can be found by Smith *et al.* (1996).

CFD

‘CFD’ is generally used to describe the numerical solutions to full potential, Euler and Navier-Stokes equations governing fluid motion (Filippone, 2003). CFD methods discretize a flow domain in order to numerically solve the governing equations for fluid properties such as pressure, velocity and vorticity.

CFD methods are highly complicated to design and implement; off-the-shelf software still requires substantial background knowledge to achieve solutions. For well-modelled, mesh-independent⁸ solutions a very high level of accuracy can be achieved. The level of detail gained with CFD methods can often surpass that achievable in experiments. Using even simplified CFD codes such as RANS⁹ methods, computation times are high and the solution procedure requires a significant level of user-intervention to ensure converged solutions.

CFD has been used to analyse flapping wings using Euler solutions (Neef & Hummel, 2001) and Navier-Stokes solvers (Liu & Kawachi, 2000), and complete flapping insect geometries using RANS solvers (Kroeger, 2003).

Table 1 gives a summary of the different methods and suggested scores for each category:

	Blade-element theory	Panel method	Vortex Method	CFD
Accuracy	**	**	**	***
Performance	***	**	**	*
Implementation¹⁰	***	**	**	*
Stability/user intervention	***	***	**	*
Results-information	*	**	**	***
Total *	12	11	10	9

Table 1: Aerodynamic method evaluation, ***=high, **=medium, *=low.

⁸ ‘Mesh independent’ solutions are those in which the solution is proven to be independent of the discretization step size

⁹ Reynolds Averaged Navier-Stokes (RANS) methods incorporate additional techniques to model the effects of turbulence

¹⁰ The category ‘Implementation’ refers to how quickly the various methods can be implemented

While the method of evaluation is somewhat limited, it does suggest that for the purpose of computer animation, suitable methods tend to be performance driven. This project does wish to perform a scientific analysis, and the results in Table 1 capture this, with more sophisticated methods scoring higher in the categories of ‘accuracy’ and ‘results-information’. However, overall, the blade-element theory best satisfies the design criteria and is therefore chosen for the analysis in this project.

An important point to be noted with the blade-element analysis is that it is commonly extended using axial momentum theory in order to take into consideration the effects of downwash. This would offer some consideration of the flow field effects on calculated forces, and therefore increase accuracy. While the extended theory will not be used in this particular project, success has been achieved in applying it to the base-model solution (Appendix 1).

2.5 SUMMARY

To clarify the use of the background information, a brief summary of the key points that will be applied to this project follows:

- Reference to well-founded theoretical and experimental works wherever possible (Ellington, 1984)
- Kinematic modelling with appropriate use of physiological data, to develop generic flapping-animal models; unfeathered wings to represent general flapping flight
- Careful balance between complexity of physical and aerodynamic modelling: gradual increase in model complexity, blade-element theory of force prediction
- Optimisation methods of parameter prediction, involving strategic selection of fixed and free solution parameters; results ‘costed’ by purely physical means
- Performance driven design, possible benefits of precomputed results tables for simulations

Combined with the project Aims and Objectives (1.1), these points will act as guidelines to the model construction.

3. THEORY

This chapter will be used to establish the theory behind model-based animation of flapping flight. The current research program¹¹ will be described with reference to previous analysis in order to found the work of this project. Definitions and assumptions of this investigation will be stated. The design of the base-model will be given, before some preliminary analysis to develop the project context. Details of the kinematic models and the selection of suitable optimisation methods will be discussed.

3.1 RESEARCH PROGRAM

This section will give a brief overview of the current research program for model-based animation. Information will be given to provide a broad view of the method of simulating flapping flight. Figure 1 depicts the program layout:

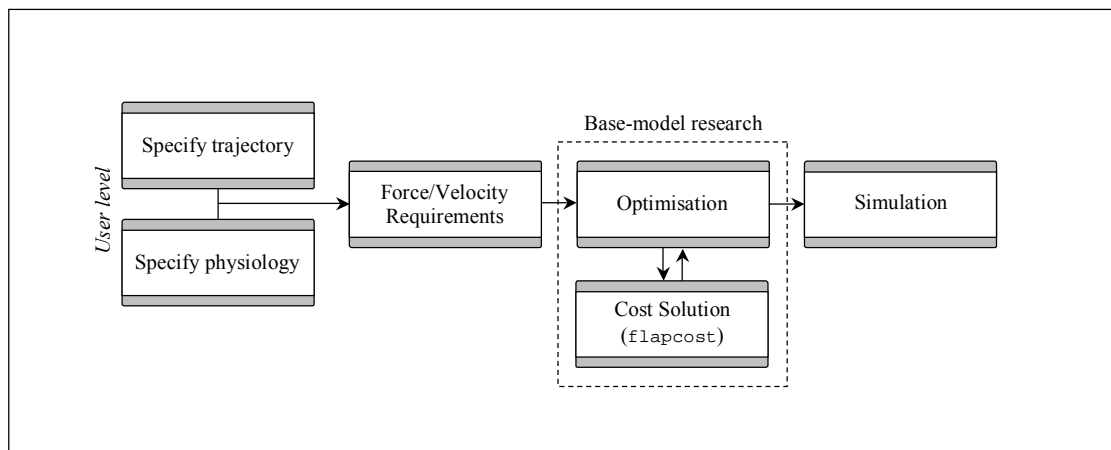


Figure 1: Model-based animation research program

The general program is designed in a similar fashion to the work by Wu & Popović (2003). By specifying a desired trajectory, force and velocity requirements can be determined as functions of time for a particular bird physiology. The next stage of the program involves optimisation of the physical model, in order to meet the

¹¹ The “research program” relates to ongoing research by Dr. William J. Crowther at the University of Manchester, in the field of model-based animation.

requirements; this was performed in the base-model and will be the focal point of analysis for this project. Having gained an optimised solution, a simulation can be produced.

Figure 1 highlights the area of research of the base-model in the overall program context; this project will also focus largely on these key areas of optimisation and costing solutions. Base-Model Design (3.3) will detail the functionality of the base-model, and therefore develop understanding of the technicalities of this project.

3.2 DEFINITIONS AND ASSUMPTIONS

This section will introduce some of the characteristics methods and parameters that will be used in the subsequent analysis. Several key assumptions and approximations will be described and justified. The reader will be gain information regarding key technical aspects of the problem. Specifically this section will explain the coordinate systems to be employed and the method of applying coordinate transformations. Details of the aerodynamic model will be provided, followed by the approximations to be made with regards to the basic flapping motion. Finally, some information on the concepts of motion gaits and power expenditure will be given.

3.2.1 AXES AND ANGLES

Throughout this project a standard set of reference frames and notation will be used. This section describes the definitions of key aspects of the bird geometry and flapping parameters.

AXIS SYSTEMS

The earth axis system (EAS) is shown in Figure 2 whereby the Z -axis is aligned with the earth gravity vector. Reference to forces and velocities acting *vertically* are equivalent to those acting parallel to the Z -axis in the earth reference frame. Equivalently, *horizontal* descriptions are those defined in the X -axis in the earth

reference frame. Force and velocity components acting in the earth reference frame are denoted by the subscript e .

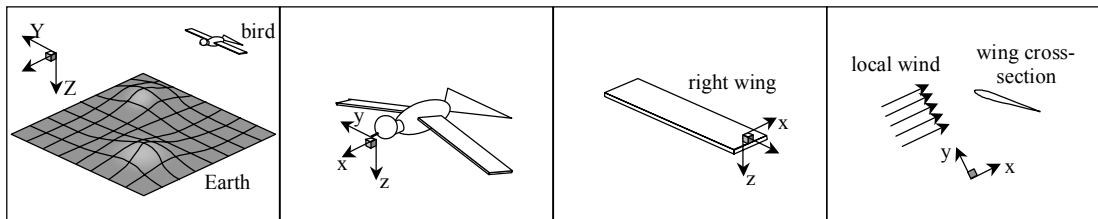


Figure 2: Axis Systems; from left to right, earth axis system, body axis system, wing axis system and wind axis system

The body axis system (BAS) is shown in Figure 2, whereby the x -axis is aligned with the bird major axis and y -axis directed to the right hand side of the body. The *neutral plane* of the wing is given by the x - y plane in the body axis system. Force and velocity components acting in the body reference frame are denoted by the subscript 1 .

The wing axis system (WAS) is shown in Figure 2 with the x -axis along the wing chord directed from trailing to leading edge, and y -axis along the span from wing root to tip. Force and velocity components in this reference frame are given the subscript 2 .

The wind axis system is given in alignment with the local wind vector, whereby the local wind is comprised of both freestream and wing flapping velocities. Force and velocity components in the wind reference frame are given the subscript 3 . Any further additions to the base-model geometry will follow a similar notation system in a logical order.

BIRD GEOMETRY/PHYSIOLOGY

The geometry of the base-model is shown in Figure 3. The geometric properties of all parts of the body are unspecified deliberately, as they are not required as part of the current analysis. The defining quantities are therefore the wing semi-span, s , and chord, c .

Throughout the course of the project, the main physiological features will be described in a logical fashion, with the terms *body* and *torso* used interchangeably. For simplicity, the structure of the wings will be defined in the sense of a human arm, using the terms *upper arm*, *forearm* and *hand* accordingly; the corresponding wing joints will be termed the *shoulder*, *elbow* and *wrist*. For a detailed description of bird physiology the reader is referred to Baumel (1993).

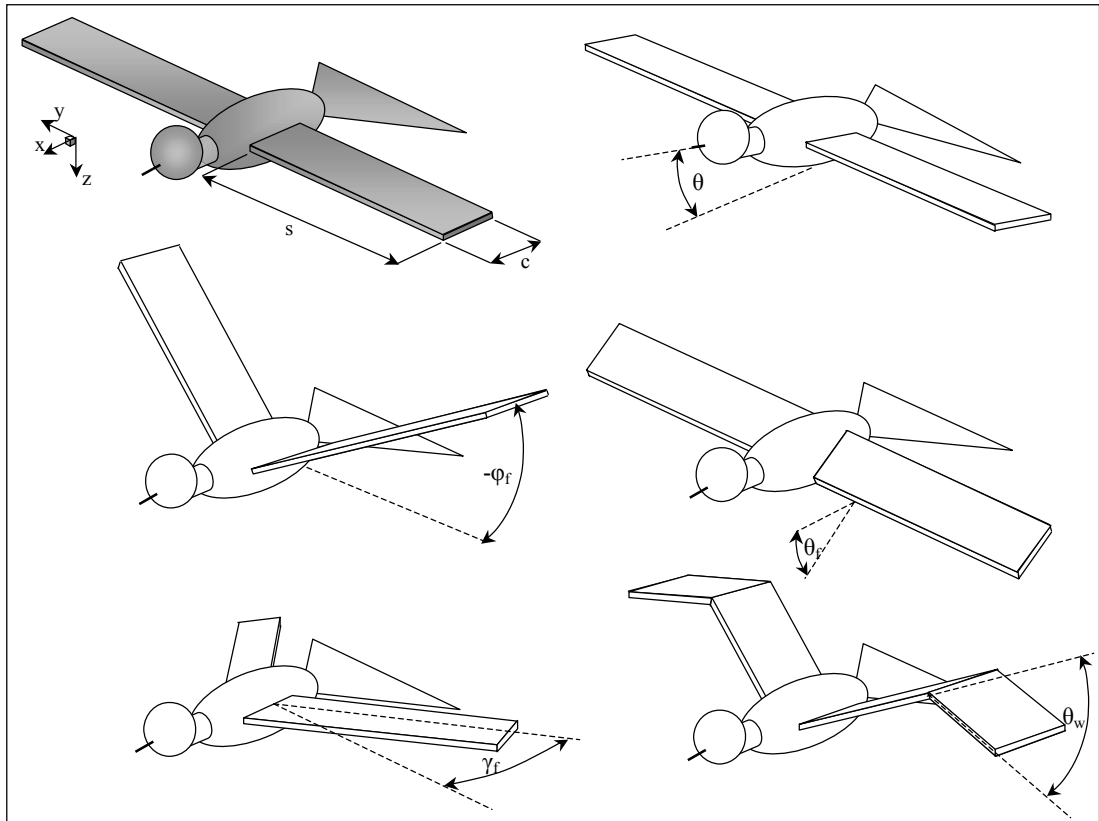


Figure 3: Bird geometry (top left). Motion angles for body pitch (above right), 'flapping' (centre left), 'twisting' (centre right), 'sweeping' (bottom left) and wrist flapping (bottom right).

FLAPPING ANGLES AND VELOCITIES

The flapping angles and velocities defined in the base-model using the BAS are shown in Figure 3. The term *flapping* defines the elevation and depression of the wing around the shoulder and wrist joints; the flapping angles are given as ϕ_f and ϕ_w respectively. Flapping motion is further defined using the *upstroke* and *downstroke* according to the wing motion relative to the body. *Twisting* refers to the rotation of the wing about its major axis to result in elevation of the trailing edge and depression of the leading edge; the twist angle is defined as θ_f . *Sweeping* is used to define the

(forward) protraction and (backward) retraction of the shoulder, and is given by the angle γ_f . The term *body pitch* refers to pitch attitude of the body, given by the angle θ . All angles are measured in radians.

It is important to distinguish between velocity components arising from the motion of the body (in the EAS), with velocity components arising from the wing motion (in the BAS). As with the corresponding angles, motions described by the wing are given the additional subscript f .

Through the course of this project as the model complexity and numbers of degrees of freedom increase, further information will be given to describe new angles and velocity terms. The maximum values of angles and velocities will be discussed in due course.

3.2.2 THE ORDER OF ROTATIONS

In order to calculate forces using the blade-element theory, the orientation of the local wind in relation to the wing must be deduced. This can be achieved by taking the net velocity in a chosen reference frame and applying a series of coordinate rotations to obtain the velocity vector in the wing reference frame. This section aims to give a brief summary of the use of rotation matrices as a means of applying coordinate transformations. The importance of the order in which the rotations are applied will be highlighted.

Table 2 contains the rotation matrices required to resolve the fundamental values of velocity, V , and force, F :

Body Pitch			
$\mathbf{V}_e = \begin{pmatrix} V_{e_x} \\ V_{e_y} \\ V_{e_z} \end{pmatrix}$	$V_{e_x} \rightarrow V_{e_x} \cos \theta - V_{e_z} \sin \theta$ $V_{e_y} \rightarrow V_{e_y}$ $V_{e_z} \rightarrow V_{e_x} \sin \theta + V_{e_z} \cos \theta$	$\therefore \mathbf{V}_1 =$	$\begin{pmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{pmatrix} \cdot \mathbf{V}_e = \mathbf{R}_{e_1} \cdot \mathbf{V}_e$
Wing Twist			
$\mathbf{V}_1 = \begin{pmatrix} V_{1_x} \\ V_{1_y} \\ V_{1_z} \end{pmatrix}$	$V_{1_x} \rightarrow V_{1_x} \cos \theta_f - V_{1_z} \sin \theta_f$ $V_{1_y} \rightarrow V_{1_y}$ $V_{1_z} \rightarrow V_{1_x} \sin \theta_f + V_{1_z} \cos \theta_f$	$\therefore \mathbf{V}_2 =$	$\begin{pmatrix} \cos \theta_f & 0 & -\sin \theta_f \\ 0 & 1 & 0 \\ \sin \theta_f & 0 & \cos \theta_f \end{pmatrix} \cdot \mathbf{V}_1 = \mathbf{R}_{12_{\text{twist}}} \cdot \mathbf{V}_1$
Wing Flap			
$\mathbf{V}_1 = \begin{pmatrix} V_{1_x} \\ V_{1_y} \\ V_{1_z} \end{pmatrix}$	$V_{1_x} \rightarrow V_{1_x}$ $V_{1_y} \rightarrow V_{1_y} \cos \phi_f + V_{1_z} \sin \phi_f$ $V_{1_z} \rightarrow -V_{1_y} \sin \phi_f + V_{1_z} \cos \phi_f$	$\therefore \mathbf{V}_2 =$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_f & \sin \phi_f \\ 0 & -\sin \phi_f & \cos \phi_f \end{pmatrix} \cdot \mathbf{V}_1 = \mathbf{R}_{12_{\text{flap}}} \cdot \mathbf{V}_1$
Wing Sweep			
$\mathbf{V}_1 = \begin{pmatrix} V_{1_x} \\ V_{1_y} \\ V_{1_z} \end{pmatrix}$	$V_{1_x} \rightarrow V_{1_x} \cos \gamma_f + V_{1_y} \sin \gamma_f$ $V_{1_y} \rightarrow -V_{1_x} \sin \gamma_f + V_{1_y} \cos \gamma_f$ $V_{1_z} \rightarrow V_{1_z}$	$\therefore \mathbf{V}_2 =$	$\begin{pmatrix} \cos \gamma_f & \sin \gamma_f & 0 \\ -\sin \gamma_f & \cos \gamma_f & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{V}_1 = \mathbf{R}_{12_{\text{sweep}}} \cdot \mathbf{V}_1$
Wind to wing axes rotation			
$\mathbf{F}_3 = \begin{pmatrix} F_{3_x} \\ F_{3_y} \\ F_{3_z} \end{pmatrix}$	$F_{3_x} \rightarrow F_{3_x} \cos \alpha - F_{3_z} \sin \alpha$ $F_{3_y} \rightarrow F_{3_y}$ $F_{3_z} \rightarrow F_{3_x} \sin \alpha + F_{3_z} \cos \alpha$	$\therefore \mathbf{F}_2 =$	$\begin{pmatrix} \cos \alpha & 0 & -\sin \alpha \\ 0 & 1 & 0 \\ \sin \alpha & 0 & \cos \alpha \end{pmatrix} \cdot \mathbf{F}_3 = \mathbf{R}_{32} \cdot \mathbf{F}_3$

Table 2: Rotation matrices

The inverse transformation matrices can obviously be used to perform a reverse transformation.

An important point that arises when applying coordinate transformations is the order in which the rotations are applied. In order to correctly establish the local wind velocity acting on a wing (and hence produce valid force calculations), the correct series of rotations must be applied. This problem will be demonstrated using the example of combined wing flapping and twisting: Consider a wing undergoing flapping and twisting motions, with no freestream velocity, thus define an instantaneous velocity vector in the body reference frame as

$$\mathbf{V}_1 = \begin{pmatrix} 0 \\ V_{fy} \\ V_{fz} \end{pmatrix}. \quad 1$$

So, applying the flap and then twist rotation matrices yields

$$\begin{aligned} \mathbf{V}_2 &= \mathbf{R}_{12\text{flap}} \cdot (\mathbf{V}_1 \cdot \mathbf{R}_{12\text{twist}}) = \begin{pmatrix} 0 \\ V_{fy} \cos \phi_f + V_{fz} \sin \phi_f \\ -V_{fy} \sin \phi_f + V_{fz} \cos \phi_f \end{pmatrix} \cdot \mathbf{R}_{12\text{twist}} \\ &= \begin{pmatrix} V_{fy} \sin \phi_f \sin \theta_f - V_{fz} \cos \phi_f \sin \theta_f \\ V_{fy} \cos \phi_f + V_{fz} \sin \phi_f \\ -V_{fy} \sin \phi_f \cos \theta_f + V_{fz} \cos \phi_f \cos \theta_f \end{pmatrix}. \end{aligned} \quad 2$$

However applying the twist and then flap rotation matrices yields

$$\begin{aligned} \mathbf{V}_2 &= \mathbf{R}_{12\text{twist}} \cdot (\mathbf{V}_1 \cdot \mathbf{R}_{12\text{flap}}) = \begin{pmatrix} 0 \\ V_{fy} \\ V_{fz} \cos \theta_f \end{pmatrix} \cdot \mathbf{R}_{12\text{flap}} \\ &= \begin{pmatrix} 0 \\ V_{fy} \cos \phi_f + V_{fz} \cos \theta_f \sin \phi_f \\ -V_{fy} \sin \phi_f + V_{fz} \cos \phi_f \cos \theta_f \end{pmatrix}. \end{aligned} \quad 3$$

Figure 4 depicts the two cases:

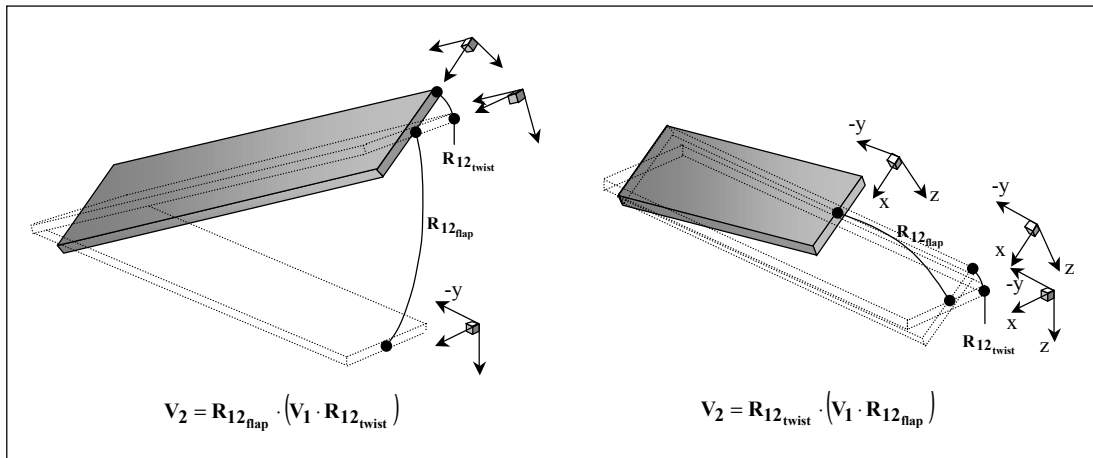


Figure 4: Effect of applying rotation matrices in different orders: (left) 'flap' and then 'twist' or (right) 'twist' and then 'flap' rotations

Without going into specific details of the result, the derivation using the second method gives a solution whereby the x -component of velocity in the wing reference frame is equal to zero. This would in turn lead to a local wind vector with zero chordwise component and hence would yield an incidence angle of $\pi/2$ rads. This is clearly not the case of the prescribed motion, and hence the series of rotations must be at fault; the first method is correctly selected. This type of analysis is required whenever a new degree of freedom is added to the model, in order to determine the appropriate point at which a new rotation should be applied.

3.2.3 BLADE-ELEMENT THEORY

The comparison of aerodynamic models performed in the Background Information (2.) led to the conclusion that blade-element theory would be the most suitable method in this program. This method requires an input of the angle of attack, along with the key aerodynamic properties of the wing in order to determine the lift and drag coefficient. The result is to obtain the aerodynamic lift and drag forces in the local wind reference frame.

The method of determining the local wind velocity vector in relation to the wing was given in The Order of Rotations (3.2.2) and hence gives the desired angle of attack of the wing. The required values of maximum lift and drag coefficients, and zero lift drag coefficient will be taken from the base-model; these values have proven to give satisfactory results in previous analyses, and closely represent typical values for an uncambered aerofoil section.

Using blade-element theory, the lift coefficient is given as a function of the angle of attack of the local wind:

$$C_l = C_{l_{\max}} \sin(2\alpha), \quad 4$$

and the drag coefficient as

$$C_d = C_{d_0} + C_{d_{\max}} \sin^2 \alpha. \quad 5$$

The wing aerodynamics properties of maximum lift and drag coefficients and zero lift drag coefficient (as found in the base-model) are as follows:

$$\begin{aligned} C_{l_{\max}} &= 2, \\ C_{d_0} &= 0.05, \\ C_{d_{\max}} &= 1. \end{aligned} \quad 6$$

It is important to note that blade-element theory predicts lift and drag coefficients based upon a 2d aerofoil section characteristics; it does not take into account any 3d flow effects. In the case of the flapping wing, any spanwise flow is assumed to have no influence on the force magnitudes or distributions on the wing.

The only other aerodynamic force included is the drag acting on the body. The base-model used the empirical formula

$$C_{d_{\text{body}}} = \sin \theta, \quad 7$$

which proved to give satisfactory results without overcomplicating the analysis. At this stage in the program the exact magnitude of the body drag force is not required, only an appropriate variation with the specified design parameters.

3.2.4 SINUSOIDAL MOTIONS

In previous research, harmonic oscillations have often been assumed in the analysis of flapping motion (Ellington, 1984). Sinusoidal motion is one of the most commonly studied fields in physical science and the properties of harmonic oscillating systems are well known. In addition to this, harmonic oscillations are easily extended to analyse more complicated behaviour by the summation of a series of harmonic components or varying amplitudes and/or frequencies in the form of a Fourier series.

Intuitive reasoning would also suggest that flapping motions governed by harmonic equations could closely represent those seen in nature. This confirms that the harmonic oscillations not only provide useful simplifications to the analysis, but also that the resulting behaviour would be assumed to provide visually realistic results.

By employing continuous variations in the key flapping parameters, flapping velocities and accelerations are easily derived from the derivatives of the displacement equations. This removes the need for more complicated gradient approximations that would be needed if the displacements were expressed as discretized functions.

It has been shown by Ellington (1984) that for many species, equal frequencies of flapping and twisting, combined with a phase difference of $\pi/2$ rads closely represents natural flapping behaviour. Ellington (1984) showed that most flapping animals aim to achieve maximum wing angle of attack at the middle of each half-stroke. The base-model uses this assumption to define variations for flap and twist angles:

$$\begin{aligned}\phi_f &= \Phi_f \sin(\omega t), \\ \theta_f &= \Theta_f \cos(\omega t).\end{aligned}\tag{8}$$

This project will continue to use these definitions, with additional motion parameters defined in a similar fashion. This will be discussed further in Flapping Kinematics (3.5).

3.2.5 THE GAITS OF MOTION – FLIGHT REGIMES

Many species have the ability to move from one location to another by more than one *style* of motion, for example humans are able to walk and also run. A more complicated system is seen in horses, with the multi-stage motion of trot, cantor and gallop. These so called *locomotive gaits* are an important feature of animal motion, and provide an interesting insight into the physical mechanisms employed.

Locomotive gaits are also observed in the world of flying organisms. Various studies have attempted to analyse flapping flight by considering the different motion gaits that can be deduced from the aerodynamic methods used. Rayner & Gordon (1998) and discussed the aerodynamic wake of birds in terms of *flight gaits*, which generally distinguishes low and high speed flight mechanisms.

In nature, flapping flight appears to be constrained within certain physical gaits that can be readily observed, without requiring any aerodynamic analysis. For example, it is easy to visualise the difference between a bird's motion during take-off, and during gliding flight. This project aims to study the governing factors behind a bird's choice of motion in these different cases. To avoid confusion between previous analyses of flight gaits, this project will use the terminology of a *flight regime* to characterise these different types of motions i.e. cruise, hover, take-off, landing, gliding etc.

From simple physical arguments, the forces that would be required during different flight regimes, and the velocity at which they occur can be estimated in terms of a species' capabilities. Figure 5 exemplifies the flight regimes for a generic bird model for given horizontal and vertical forces, and flight velocity:

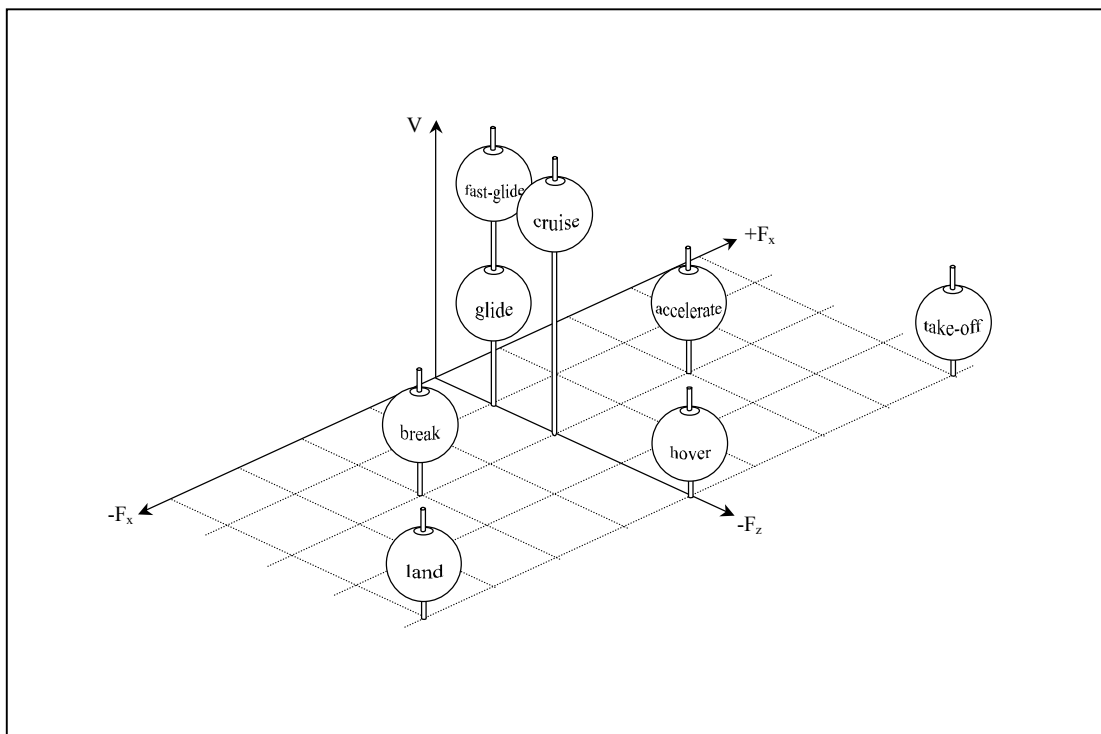


Figure 5: Flight regimes

Figure 5 shows, for example, that low velocity flight with large positive force F_x and large negative force F_z , represents the case of take-off. At high velocity with low force F_z , fast-gliding is observed. The reader should note that the flight regimes shown are a very general representation of flapping flight. However, this type of information will prove useful in the subsequent analysis. Not only does it provide a clear method of discussion of the results, but also a firm understanding of the broad context of the project.

The flight regimes will be used to establish suitable maximum values for the input values of force and velocity. This will be discussed further in Base-Model Design (3.3).

3.2.6 POWER CONSUMPTION

One of the key factors expressed by Ellington (1984) was the importance of power consumption considerations. The analysis performed on Advance Ratio (3.4) will show that within even a simple system (with few degrees of freedom), there may be more than one way to produce the same result; for example two different twist amplitudes that yield the same thrust force. To capture the most realistic motion, the solution that requires the least amount of power should be selected. This type of behaviour is found in all forms of locomotion in nature, whereby creatures strive for efficiency by minimising unnecessary work output.

There is also a strong coupling between the amount of power available to a particular animal, and its corresponding flight capabilities. While physical constraints on bird physiology may prevent some species from motions such as hovering, the power constraint is believed to help define the general flight regimes. Further discussion on this will be given in Base-Kinematic Model Solution, Cost Fraction (4.1.1).

The power consumption in flapping flight is traditionally divided into four components (Ellington, 1984), which are described as follows:

- parasite power: due to drag forces acting on the body
- profile power: due to forces acting on the wings
- inertial power: due to acceleration of the wings
- induced power: due to acceleration of the surrounding air

The base-model assumed the profile power to be the main source of power expenditure. Within this project the parasite power will be neglected: the cost due to power expenditure is used to determine to the motion of the wings rather than that of the body. This project has assumed a quasi-steady flapping analysis (by applying the blade-element theory), with no consideration of the properties of the surrounding air; hence the induced power is also ignored.

The derivation of the power consumption is determined by firstly considering Newton's second law of motion for a translating wing. The acceleration is obtained from the difference between the driving force (from the wing muscles) and aerodynamic force

$$\mathbf{F}_{1 \text{ driving}} - \mathbf{F}_{1 \text{ aerodynamic}} = m_w \cdot \mathbf{a}_1 . \quad 9$$

Within the base-model, the assumption is made that while the acceleration term varies in magnitude, the bird wing mass is always small in comparison to the forces expected to act on the wing. This therefore simplifies to

$$\mathbf{F}_{1 \text{ driving}} \approx \mathbf{F}_{1 \text{ aerodynamic}} , \quad 10$$

with the power used in driving the flapping motions given as

$$Power_1 \approx \mathbf{F}_{1 \text{ aerodynamic}} \cdot \mathbf{V}_{1f} . \quad 11$$

Both the aerodynamic forces and the velocities in the body reference frame can be derived as described in previous sections; hence the power consumption can be determined. It should be noted that power expenditure is assumed to occur purely in the BAS.

This method of determining the power expenditure will be used throughout this project.

3.3 BASE-MODEL DESIGN

As mentioned in Research Program (3.1), this project will contribute to a program for model-based animation by continuing work established in a base-model; details of which are given in Appendix 1. Before beginning the analysis, it is important to correctly understand how the base-model was constructed and how it will be adapted in this project. This section will give an overview of the base-model functionality and will serve as a reference point for the future analysis.

3.3.1 NUMERICAL METHOD

It was clear from the beginning of the research program that in order to successfully employ the desired method of optimisation, numerous individual calculations would have to be performed. In order to do this in an efficient manner, computational methods should be incorporated. It was recognised that a high performance development environment that integrates efficient computation with visualisation would be ideal for the purposes of analysis and animation. For these reasons, MATLAB¹² was cited as an appropriate software package.

3.3.2 BASE-MODEL FUNCTIONALITY

The method by which the base-model obtains optimal solutions for given input requirements will now be described. The reader should refer back to Figure 1 in

¹² The particular software used in this project is MATLAB version 6.5.0.180913a release 13

Research Program (3.1) for a depiction of the base-model, with full details provided in Appendix 1.

The optimisation module is designed to make appropriate choices of the various free solution parameters to fit the input requirements. These parameters represent the physiological degrees of freedom described in Axes and Angles (3.2.1). Optimisation calls upon the function `flapcost` to assess the fitness of a particular set of input variables. This can be done in ‘real time’ for a particular set of requirements, or alternatively, optimisation for a range of forces and velocities can be used to populate lookup tables for the various solution parameters; the base-model employs the latter method.

The function `flapcost` uses the input parameters from the optimiser to define a flapping motion. This motion defines the current state of the solution by associating a *cost* with the set of input parameters. Solutions are costed according to how closely they fit the desired force requirements; additional costs can also be included, such as the amount of power expenditure required for a particular solution. Figure 6 shows the general layout of the function `flapcost`:

```
obtain input parameters from optimiser (physiology, force requirements etc.)

define flapping parameter motion (sinusoidal);
define lift and drag coefficients for blade-element theory;

for time=0:timestep:timeperiod           %one flap cycle

    rotate freestream velocity, Ve, to body(Ve->Vle);
    define flapping velocity,Vlf;
    find total velocity,Vl=Vle+Vlf;
    rotate Vl to wing;
    calculate wind incidence, forces(F3);
    rotate F3 to wing and body (F3->F2->F1);
    find total force acting on body, F1=F1+Fbody, determine power consumption;
    rotate forces to earth (F1->Fe) for trajectory simulation;

end

sum forces over timeperiod;
combine flapping forces with body force;

calculate difference between desired force and current motion force
calculate power consumption
cost wingbeat;
```

Figure 6: Pseudo-code for the function `flapcost`

The velocity and force rotations are performed using the methods described in The Order of Rotations (3.2.2) and will be detailed for specific models in Flapping Kinematics (3.5). The exact method of determining power consumption is described in Power Consumption (3.2.6). Solutions are obtained over a single cycle to represent one wing-beat, as shown in Figure 6.

The base-model is capable of obtaining solutions for motion in the X - Z plane in the EAS and can therefore represent 2d flight cases; this project will also omit motion of the body¹³ in the Y -direction.

A range of force and velocity requirements were used in the base-model to study the extremes of motion. By comparing the resulting behaviour with the expected motion, maximum force and velocity values can be prescribed to retain realistic results. In this project similar force and velocities values will be used, and the flight regimes shown in Figure 5 (3.2.5) will be considered to validate the results. The method of populating lookup tables will be discussed further in Base Kinematic Model Solution, Populating Lookup Tables (4.1.1).

It should be noted that the base-model utilises non-dimensional values of force, velocity and power throughout; this approach will be continued in this project.

3.4 PRELIMINARY ANALYSIS - THE ADVANCE RATIO

As part of the design realisation, preliminary analysis is required to highlight some of the key factors that are likely to become important later in the project. This section will focus on important analytical results that can be derived from the base-model. Specifically, the ratio of flapping velocity to freestream velocity; know as the advance ratio (A), will be used to define limitations for general flapping motion.

¹³ The flapping motion of the wings in the Y -direction will be considered

3.4.1 FLAPPING MODES

In flapping wing systems, the flight regime can be further defined by specifying different *modes* of flight according to the energy transfer in the system. For example, in many flight regimes a flapping wing expels a finite amount of energy in order to provide propulsion. This is true in the case of hovering, where the wings are required to work on the surrounding fluid in order to provide a corresponding vertical force to balance the weight of the body. This type of flight mode will be referred to as a *positive* flight mode.

Conversely, a *negative* mode of flight is one in which the body motion draws energy from the surrounding air. The simplest example of this is seen in breaking flight: birds are able to rotate both wings and body in such a way that an increased drag force is used to decelerate the body in a desirable fashion. This type of motion requires no¹⁴ input of energy from the bird, and is therefore seen to be drawing energy from the surrounding air.

Ellington (1984) highlights the importance of the advance ratio when considering the aerodynamic mechanisms. It is believed that the advance ratio and the wing twist angle are the crucial factors for assessing the mode of flight.

3.4.2 TRANSLATING WING

In order to properly assess the influence of advance ratio and wing twist on the flight behavioural modes, it is necessary to look more closely at the forces that occur during flapping flight. Firstly, consider the simple case of an uncambered wing section undergoing a harmonic vertical translation, with no rotation, as shown in Figure 7:

¹⁴ It should be noted that some energy would be required from the wing muscles to retain a fixed twist angle against the local wind, though this is deemed to be small in comparison to the energy gained in the overall retardation.

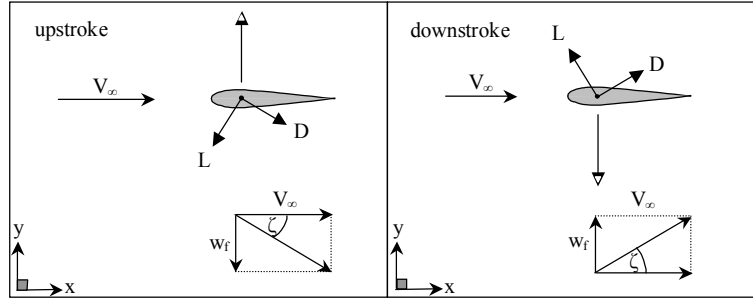


Figure 7: Uncambered wing undergoing harmonic vertical translation in horizontal freestream flow

If the wing is placed into a horizontal flow velocity, V_∞ , a thrust force will develop due to the horizontal component of the lift acting on both upstroke and downstroke. In order for the wing to accelerate, the thrust force must overcome the opposing horizontal force component that arises from the aerodynamic drag.

It can be easily shown that in this case the possibility of a net forward thrust is determined by the lift and drag characteristics of the wing. Given the lift force, L , and drag force D in the wind axis system, the force in the x -direction is given by

$$F_x = L \sin \zeta - D \cos \zeta , \quad 12$$

or in coefficient form:

$$C_{F_x} = C_L \sin \zeta - C_D \cos \zeta . \quad 13$$

Using blade-element theory:

$$C_{F_x} = C_{L_{\max}} \sin(2\zeta) \sin(\zeta) - (C_{D_0} + C_{D_{\max}} (\sin(\zeta))^2) \cos \zeta > 0 . \quad 14$$

Using double angle formula and cancelling for $\cos \zeta$ yields

$$\sin^2 \zeta (2C_{L_{\max}} - C_{D_{\max}}) - C_{D_0} > 0 . \quad 15$$

The angle ζ , known as the *advance angle*, is related to the advance ratio, A , by the equation

$$\zeta = a \tan\left(\frac{w_f}{V_{e\infty}}\right) = a \tan(A). \quad 16$$

At high freestream velocity, ζ (and A) decrease and hence in equation 15 the relative importance of C_{D0} increases. To provide net forward thrust at high freestream velocity, the maximum lift coefficient must be substantially higher than the maximum drag coefficient. Physically, this suggests that birds that travel at high velocities must utilise wings with such properties, in order provide sufficient acceleration.

3.4.3 TRANSLATING, ROTATING WING

Consider the case of the rotating-translating wing, as shown in Figure 8. It is now possible for the wing to be orientated at an angle greater than $\pi/2$ rads to the relative wind. Due to the lift coefficient variation with angle of attack given by the blade-element theory, it is possible for the wing to produce lift that gives a negative horizontal component. The drag coefficient will remain positive and always produce a negative horizontal force for $V_{\infty} > 0$.

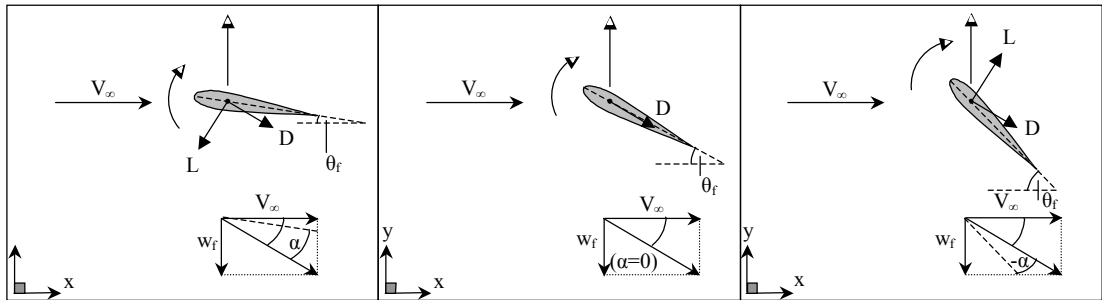


Figure 8: Uncambered wing undergoing harmonic vertical translation, and twisting, in freestream flow

To create of a positive flight mode, the wing angle of attack must be considered – or more specifically, the combination of wing twist angle and advance angle. To obtain a lift force that causes positive thrust over a cycle, it is clear that the advance angle must be greater than the wing twist angle, i.e.

$$\zeta > \theta_f, \quad 17$$

where,

$$\theta_f = \Theta_f \sin(\omega t). \quad 18$$

$$\text{Therefore } A > \tan \theta_f. \quad 19$$

From this simple relationship some interesting points may be noted. As the flapping angle increases to $\pi/2$ rads, the advance ratio must tend to infinity in order to maintain positive thrust. This suggests that as the freestream velocity reduces to zero, any amount of wing twist (up to $\pi/2$ rads) can be used to create a positive thrust. Physically speaking this suggest that hovering birds can employ twist amplitudes up to $\pi/2$ rads, which is what is seen in the case of a hummingbird (Ellington, 1984).

As the freestream velocity increases, the advance ratio, and hence the wing twist for positive thrust, decreases. This represents the case of cruising flight, whereby, maximum lift and drag coefficient values dictate the behaviour, thus verifying the discussion on the translating wing.

This analysis can be performed more rigorously by including the effects of drag forces in determining the net force F_x . The force coefficients are given by

$$C_{F_x} = C_L \sin \zeta - C_D \cos \zeta, \quad 20$$

$$C_{F_z} = C_L \cos \zeta + C_D \sin \zeta. \quad 21$$

As before, we aim to analyse the positive flight mode with $C_{F_x} > 0$, which yields

$$C_L \sin \zeta - C_D \cos \zeta > 0. \quad 22$$

Using blade-element theory:

$$C_{L_{\max}} \sin(2\zeta - 2\theta_f) - (C_{D_0} + C_{D_{\max}} (\sin(\zeta - \theta_f))^2) \cos \zeta > 0. \quad 23$$

This relationship is shown graphically in Figure 9 24

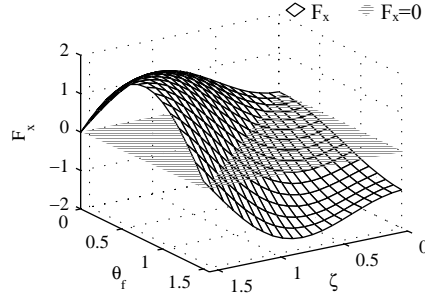


Figure 9: Variation in thrust force F_x with advance angle and twist angle

Figure 9 shows the values of F_x plotted over a feasible range of wing twist angles and the full range of advance angles. With maximum advance angle, i.e. in the case of hover, it is clear that any value of θ_f may be used in order to give a positive thrust force, as predicted previously. It is also clear that a single solution of $\theta_f = \pi/4$ rads gives the maximum F_x value, as expected due to the nature of the lift coefficient variation with incidence.

In the case of cruise, with small advance angle, increasing the twist from zero causes F_x to reduce; this represents the case of a negative flight mode. More specifically, it can be seen that using wing twist is only beneficial in generating thrust force in the range of advance angles $\pi/2 < \zeta < \pi/4$ rads, i.e. wing twist should be used when $w_f > V_\infty$. The limiting case between positive and negative flight modes is shown on the graph by the line $\theta_f = \zeta$, which confirms equation 19.

The final point to be noted here is that the maximum F_x that can be produced, decreases as the advance angle reduces to zero. This result is obtained without considering any factors such as the power/torque available to the wing; it shows that for any given flapping velocity the thrust available decreases with increasing freestream velocity.

3.4.4 CRUISE AND HOVER REGIMES

The final part of this analysis will consider the two most extreme cases, cruise and hover, in more detail.

CRUISE

It is known that during cruise the advance angle reduces, and can therefore be represented as

$$\zeta \sim O(\epsilon), \text{ where } \epsilon \text{ is small.} \quad 25$$

Equation 23 gave a relationship to represent the case of positive thrust. Using double angle formulae, equation 23 reduces to

$$C_{L_{\max}} (\sin 2\zeta \cos 2\theta_f - \cos 2\zeta \sin 2\theta_f) \sin \zeta - (C_{D_0} + C_{D_{\max}} (\sin \zeta \cos \theta_f - \cos \zeta \sin \theta_f)^2) \cos \zeta > 0. \quad 26$$

As ζ is small:

$$\begin{aligned} \sin \zeta &\sim \zeta, \\ \sin(2\zeta) &\sim 2\zeta, \end{aligned} \quad 27$$

and also,

$$\begin{aligned} \cos \zeta &\sim 1, \\ \cos(2\zeta) &\sim 1. \end{aligned} \quad 28$$

Therefore equation 27 can be reduced to

$$C_{L_{\max}} (2\zeta \cos 2\theta_f - \sin 2\theta_f) \zeta - (C_{D_0} + C_{D_{\max}} (\zeta \cos \theta_f - \sin \theta_f)^2) > 0. \quad 29$$

Considering the orders of magnitude of each term:

$$\begin{aligned}
& C_{L_{\max}} (2\zeta \cos 2\theta_f - \sin 2\theta_f) \zeta - \\
& \quad O(\varepsilon^2) \quad O(\varepsilon) \\
& (C_{D_0} + C_{D_{\max}} (\zeta^2 \cos^2 \theta_f - 2\zeta \cos \theta_f \sin \theta_f + \sin^2 \theta_f)) > 0. \\
& \quad O(\varepsilon) \quad O(\varepsilon^2) \quad O(\varepsilon) \quad O(1)
\end{aligned} \tag{30}$$

Considering the leading order terms:

$$O(1): \quad -\sin^2 \theta_f > 0 \tag{31}$$

The leading order term in equation 30 is negative. It can be seen that in cruise, a slight increase in the value of θ_f causes a large increase in drag force due to the leading order term, $\sin^2 \theta$. Hence, to retain positive thrust this term must be reduced by limiting θ_f . This confirms the results shown previously: wing twist must be limited in high velocity cases.

HOVER

In the case of hovering, the freestream velocity reduces to zero, which yields an advance angle of

$$\zeta \sim \pi/2. \tag{32}$$

Considering again, equation 26,

$$\text{As } \zeta \sim \pi/2, \tag{33}$$

$$\begin{aligned}
& \sin \zeta \sim 1, \\
& \sin(2\zeta) \sim \pi - 2\zeta,
\end{aligned} \tag{34}$$

and also,

$$\begin{aligned}
& \cos \zeta \sim \left(\frac{\pi}{2}\right) - \zeta, \\
& \cos(2\zeta) \sim -1.
\end{aligned} \tag{35}$$

Combing equations 34 and 35 with equation 26 yields

$$C_{L_{\max}} ((\pi - 2\zeta)\cos 2\theta_f + \sin 2\theta_f) - \left(C_{D_0} + C_{D_{\max}} \left(\cos \theta_f - \left(\frac{\pi}{2} - \zeta \right) \sin \theta_f \right)^2 \right) \left(\frac{\pi}{2} - \zeta \right) > 0. \quad 36$$

Now considering the orders of magnitude of each term:

$$C_{L_{\max}} \left(\begin{array}{c} (\pi - 2\zeta)\cos 2\theta_f + \sin 2\theta_f \\ O(\varepsilon) \quad O(1) \end{array} \right) - \left(\begin{array}{c} C_{D_0} + C_{D_{\max}} \left(\cos^2 \theta_f - 2 \cos \theta_f \left(\frac{\pi}{2} - \zeta \right) \sin \theta_f + \left(\frac{\pi}{2} - \zeta \right)^2 \sin^2 \theta_f \right) \\ O(\varepsilon) \quad O(\varepsilon) \quad O(\varepsilon^2) \quad O(\varepsilon^3) \end{array} \right) \left(\frac{\pi}{2} - \zeta \right) > 0. \quad 37$$

The leading order term is now seen to occur only due to the lift force acting:

$$O(1): \quad \sin(2\theta_f) > 0; \quad 38$$

This shows that in order to maximise the leading order term, the wing twist should tend towards a value of $\pi/4$ rads, which also confirms the results in Figure 3; positive thrust is best achieved using a wing twist angle $\theta_f = \pi/4$ rads.

3.5 FLAPPING KINEMATICS

The aim of this section is to detail the specific methods of describing flapping motion and deriving aerodynamic forces using a range of different kinematic models. The reader will gain knowledge of how each model transforms a specified flapping motion, combined with freestream conditions, to calculate forces. This section will firstly assess the method of simplifying force distributions into point forces. Following this, five different kinematic models will be analysed; details of the model degrees of freedom, with coordinate transformations will be shown. Example results, with performance details will be given for each model.

3.5.1 VELOCITY DISTRIBUTION AND CONTROL POINT LOCATION

It is desirable to group aerodynamic forces across the wing surface to simplify the analysis; this removes the need to consider velocity and force distributions, which would also vary with time. This section will aim to study the location of the control point, which will be used as a reference point for determining local wind velocities and the corresponding forces.

Figure 10 depicts the wing motion with approximate velocity and force distributions for four different wing motions.

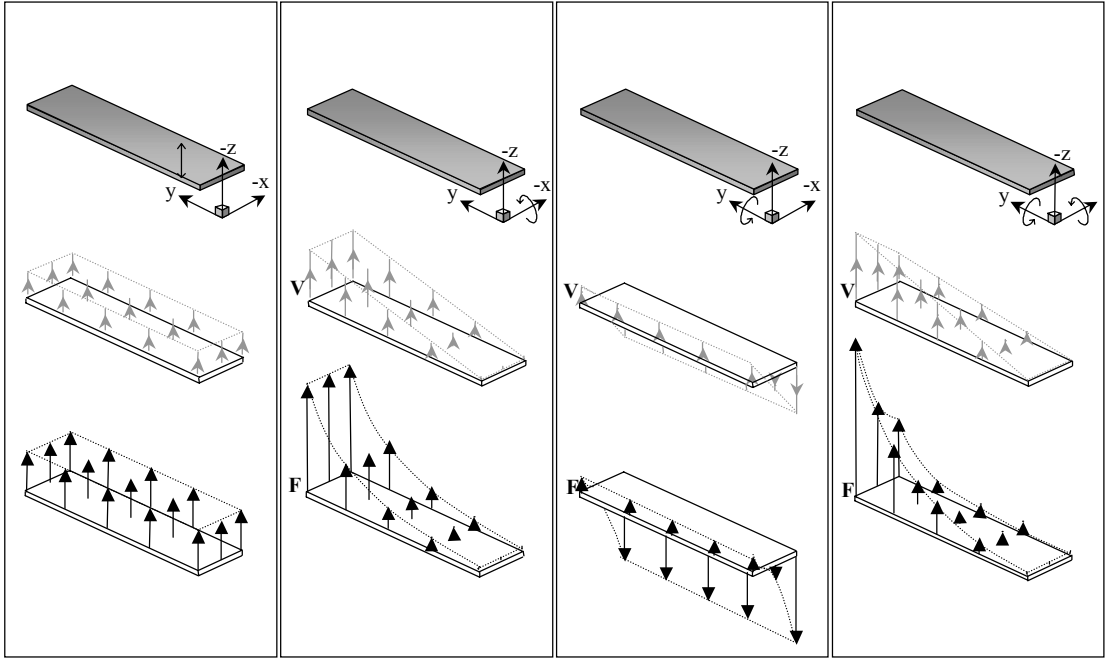


Figure 10: Wing motion, velocity and force distribution for (from left to right) vertically translating, flapping, twisting, and combined flapping and twisting motions

In the simplest case, a wing undergoing a harmonic vertical translation, with no freestream velocity, will be considered; this represents the kinematic model employed in the base-model solution. The vertical motion of the wing is given by

$$\phi_f = \Phi_f \sin(\omega t), \quad 39$$

which gives angular and linear velocity relationships

$$\Omega_f = \omega \Phi_f \cos(\omega t), \quad 40$$

$$V = \Omega r, \therefore V_3 = \frac{S}{2} \omega \Phi_f \cos(\omega t).^{15} \quad 41$$

The force acting on the wing defined as

¹⁵ The factor of 1/2 was included in the base-model as an empirical method of incorporating the effects of rotation due to the flapping angle.

$$\mathbf{F} = \begin{pmatrix} F_D \\ 0 \\ F_L \end{pmatrix} = \begin{pmatrix} C_D \\ 0 \\ C_L \end{pmatrix} \cdot qsc, \quad 42$$

where the dynamic pressure, q , is given by

$$q = \frac{1}{2} \rho (V_1^2 + V_2^2 + V_3^2). \quad 43$$

For the translating wing, $V_1 = V_2 = 0$ in the wing reference frame. Also, from blade-element theory $C_l = 0$, $C_d = C_{dmax}$. Equation 42 becomes

$$\mathbf{F} = F_D = \frac{1}{2} \rho V_3^2 sc C_D = BV_3^2, \quad 44$$

where B is a constant.

Substituting for the velocity found in equation 41 and multiplying by the wingspan the total force along the wing can then be found as

$$F_{D_{total}} = BV_3^2 s = B \frac{s^3}{4} \omega^2 \Phi_f^2 \cos^2(\omega t). \quad 45$$

Now considering the case of the ‘flapping’ wing as shown in Figure 10, the system now has an additional level of complexity due to the spanwise velocity distribution that arises. The angular displacement of the wing is given by

$$\phi_f = \Phi_f \sin(\omega t), \quad 46$$

which yields the corresponding angular and linear velocities

$$\Omega_f = \omega \Phi_f \cos(\omega t), \quad 47$$

$$V_3 = r\omega\Phi_f \cos(\omega t). \quad 48$$

We now have an equation describing the linear velocity variation along the wing span; this will lead to a corresponding force variation:

$$\mathbf{F} = F_D = BV_3^2, \text{ where } V_3 = V_3(r). \quad 49$$

In order to determine the net force acting on the wing, it is now necessary to integrate the force along the wingspan:

$$F_{D_{total}} = \int_0^s F_D dr = \int_0^s BV_3^2 dr = \int_0^s B(r\omega\Phi_f \cos(\omega t))^2 dr = B \frac{s^3}{3} \omega^2 \Phi^2 \cos^2(\omega t). \quad 50$$

Which is comparable to the previous ‘translating’ case (equation 45), and shows an increase in lift of 1/12.

For simplicity, it is preferable to define a control point at which a single representative velocity and force can be determined to describe the net force acting on the entire wing. It is clear that if we take the velocity at the control point as

$$V_{3_{CP}} = \frac{s}{\sqrt{3}} \omega\Phi_f \cos(\omega t), \quad 51$$

equation 49 can now be used to yield the total force given in equation 50 by simple multiplication rather than integration.

This suggests that the control point must be located at $r = s/\sqrt{3}$ in the case of the flapping wing, to yield the correct force. In comparison to the original control point location of $r = s/2$, only a slight change in location is necessary to compensate for the velocity variation.

It is believed that by using the original control point location at the wing semi-span, and hence generating a slight underprediction of force on the wing, the analysis

could be largely simplified without a substantial loss of accuracy. Before confirming this approximation, other possible motions of the wing should be considered.

Figure 10 also shows the affect of incorporating wing twisting motion. It is seen that the positive wing twist creates a chordwise velocity distribution; the velocity now varies from leading edge to trailing edge, with magnitudes depending on the exact location of the twist axis.

This type of motion would require the control point to be shifted in the chordwise direction, in order to use a single velocity value in determining forces. However from the geometry of the wing, assumed to be relatively high aspect ratio, it is clear that

$$s \gg c, \text{ therefore } V_{3\text{flap}} \gg V_{3\text{twist}}. \quad 52$$

This suggests that an even smaller variation in velocity occurs in the chordwise direction and hence a smaller relocation of the control point would be required to assume a constant velocity field. Similar arguments can be made for the sweeping, and combined sweeping and flapping motion.

It can therefore be concluded that the movement of the control point to account for velocity variations across the wing is deemed unnecessary in approximating the net forces applied. A control point located at the centre of the wing planform¹⁶, as seen in the base-model, will continue to be employed throughout the analysis. The construction of the kinematic models can now be considered.

3.5.2 BASE KINEMATIC MODEL: 3DOF, 1D FLAPPING

Base-Model Design (3.3) section described the structure and method used in the function `flapcost`. It was highlighted that for each kinematic model, the prescribed flapping velocity V_{lf} and the particular rotation matrices will vary. These factors relate directly to the dimensionality and degrees of freedom of the model.

¹⁶ Specifically this point lies at a distance equal to half of the wing semispan from the wing root, and half of the chord length from the leading (or trailing) edge

The base-model is a 3DOF model whereby the body pitch angle θ , flap angle ϕ_f and wing twist angle θ_f are used to describe the motion. A simplified flapping motion is used whereby the wing undergoes vertical translation, with no rotation due to flapping; this is shown in Figure 11. The following equations prescribe the wing motion:

$$\phi_f = \Phi_f \sin(\omega t), \quad 53$$

$$\theta_f = \Theta_f \cos(\omega t), \quad 54$$

with the control point velocity is defined by

$$\mathbf{V}_{1r} = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2} s \omega \Phi_f \cos(\omega t) \end{pmatrix}, \quad 55$$

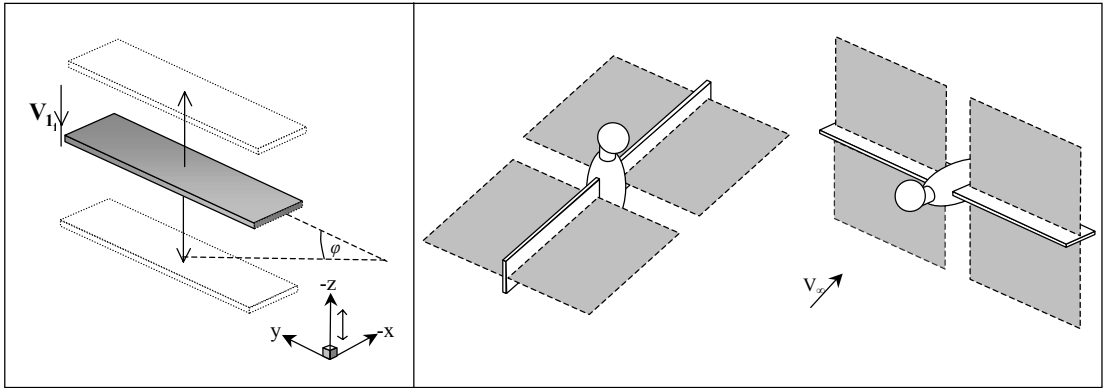


Figure 11: (left) Base-model translating wing motion to represent flapping; (right) suggested hovering and cruising regimes with sweep-planes depicted¹⁷

The wing therefore follows a 1-dimensional translation during flapping. Referring back to Table 2, the corresponding rotation required to derive the velocity in the wing reference frame, given the velocity in the body frame, is

¹⁷ The terms 'hovering' and 'cruising' are used in a general sense to prescribe motions that involved low and high freestream velocities respectively

$$\mathbf{V}_{2_f} = \mathbf{R}_{12_{\text{twist}}} \cdot \mathbf{V}_{1_f} = \begin{pmatrix} \cos \theta_f & 0 & -\sin \theta_f \\ 0 & 1 & 0 \\ \sin \theta_f & 0 & \cos \theta_f \end{pmatrix} \cdot \mathbf{V}_{1_f}, \quad 56$$

which involves resolving by the wing twist angle only. This can then be used to with the blade-element theory to determine the lift and drag coefficients.

In order to assess the validity of the prescribed motion, it is useful to test the current system with example values for the various model parameters. A solution is obtained by specifying the flap parameters as

$$\mathbf{V} = 0, \theta = \frac{\pi}{2}, \Phi_f = \frac{\pi}{2}, \Theta_f = \frac{\pi}{4}, \omega = 1: \quad 57$$

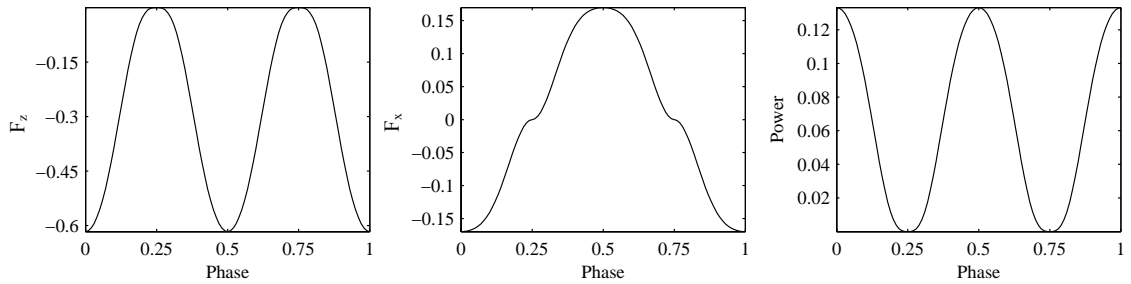


Figure 12: Force and power results from base kinematic model ('hovering')

Figure 12 shows the forces and power results in the EAS. The prescribed motion defines a fully-pitched bird body with large flapping and twisting amplitudes in zero freestream velocity. Physically, this case aims to represent hovering behaviour. The results show that over the flap-cycle an oscillating negative force F_z is produced: this shows that the wings produce a thrust force used to balance the weight of the bird. In conjunction with this, an oscillating horizontal force, F_x , arises as the wing passes back and forth in the horizontal plane: this yields no net force to drive the bird horizontally.

The power expenditure variation is synchronous with the forces. Peak force values are observed as the wing passes through the neutral plane, at which point maximum flapping velocities occur, which in turn yields large power output.

A second example solution is obtained using the flap parameters:

$$V = 0.5, \theta = \frac{\pi}{16}, \Phi_f = \frac{\pi}{4}, \Theta_f = \frac{\pi}{16}, \omega = 1 :$$

58

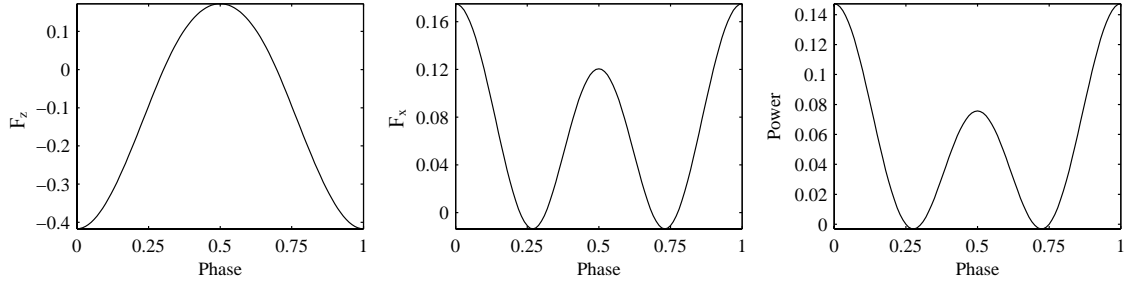


Figure 13: Force and power results from base kinematic model ('cruising')

Figure 13 shows the results for a bird with a slightly pitched body, undergoing small flapping and twist amplitude motions with an increased freestream velocity. This case aims to represent behaviour close to cruising.

The force F_z is now seen to remain negative over the majority of the cycle. A small positive force F_z is produced as the wing passes through the neutral plane on the upstroke ($Phase \sim 0.5$). Results from the Preliminary Analysis (3.4) suggest that the wing twist angle is greater than the advance angle at this point. A net negative F_z force is still clearly present over the complete cycle.

The force F_x also oscillates and remains positive over most of the cycle. The motion is now producing a mean thrust to provide a positive acceleration (therefore this is not strictly a cruising case). Negative F_x is seen as the wing reaches the bottom of the downstroke ($Phase \sim 0.25$) and again at the top of the upstroke ($Phase \sim 0.75$): in these cases the flapping velocity reduces to zero and the twist angle is also equal to zero. It is in fact the body pitch angle that defines a positive local incidence angle and the lift force is resolved to give negative F_x .

The power expenditure is again in phase with the force variation, though in this example more power is used on the downstroke than on the upstroke. This behaviour can again be explained by considering the body pitch: there is now a horizontal component of the downward flapping motion that acts in the positive X -direction i.e. into the freestream wind. This results in large relative velocity values and hence

greater power consumption than on the upstroke, which acts away from the freestream wind in the negative X -direction.

3.5.3 KINEMATIC MODEL 1: 3DOF, 2D FLAPPING

The first model to be developed specifically for this project is an extrapolation of the base-model. In kinematic model 1 the wing undergoes a true rotation as it flaps and therefore more closely resembles the physical behaviour of flapping. The equations defining the wing motion remain unchanged from equations 53 and 54, while the control point velocity is now defined as

$$\mathbf{V}_{1r} = \begin{pmatrix} 0 \\ -\frac{1}{2}s\omega\Phi_f \cos(\omega t)\sin\phi_f \\ \frac{1}{2}s\omega\Phi_f \cos(\omega t)\cos\phi_f \end{pmatrix}; \quad 59$$

Figure 14 depicts the flapping motion.

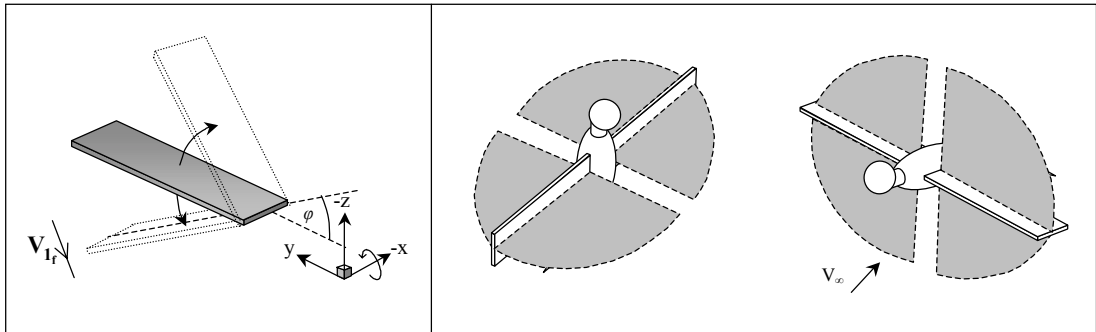


Figure 14: (left) Kinematic model 1 flapping wing motion; (right) suggested hovering and cruising regimes with sweep-planes depicted

The wing now traces a path that follows a semicircular arc with radius equal to the wing semispan. It is now necessary to apply two coordinate transformations to rotate by both flapping angle ϕ_f and twist angle θ_f . The rotation procedure involves firstly resolving by the flap and then by the twist angle; the reader should refer back to Table 1 (3.2.1) and also The Order of Rotations (3.2.2) for details:

$$\mathbf{V}_{2_f} = \mathbf{R}_{12_{\text{twist}}} \cdot (\mathbf{R}_{12_{\text{flap}}} \cdot \mathbf{V}_{1_f}) = \begin{pmatrix} \cos \theta_f & 0 & -\sin \theta_f \\ 0 & 1 & 0 \\ \sin \theta_f & 0 & \cos \theta_f \end{pmatrix} \cdot \left(\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_f & \sin \phi_f \\ 0 & -\sin \phi_f & \cos \phi_f \end{pmatrix} \cdot \mathbf{V}_{1_f} \right). \quad 60$$

An example solution will be obtained for hovering flight using the same flapping parameters as used in the previous (base-model) case:

$$\mathbf{V} = 0, \theta = \frac{\pi}{2}, \Phi_f = \frac{\pi}{2}, \Theta_f = \frac{\pi}{4} : \quad 61$$

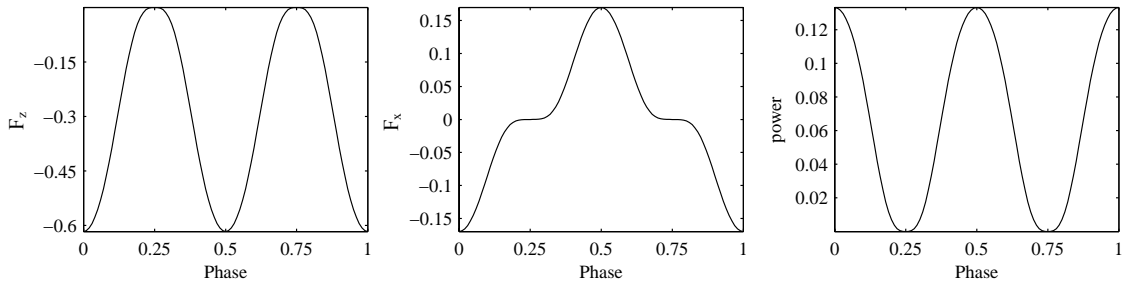


Figure 15: Force and power results from kinematic model 1 ('hovering')

Figure 15 shows that the F_z variation using model 1 appears similar to the base-model. This is what is expected, as the additional rotation is occurring around the body major axis (the x -axis) and therefore does not affect the vertical force component.

There is, however, an influence on the force, F_x . At large flapping angles ($Phase \sim 0.25, 0.75$), a fraction of the thrust force is now resolved sideways in the Y -direction and hence reduces the F_x component. The power consumption is the same as the base-model, as the local wind velocity and hence forces perpendicular to the wing are unchanged.

An example of cruising flight is given using

$$\mathbf{V} = 0.5, \theta = \frac{\pi}{16}, \Phi_f = \frac{\pi}{4}, \Theta_f = \frac{\pi}{16} : \quad 62$$

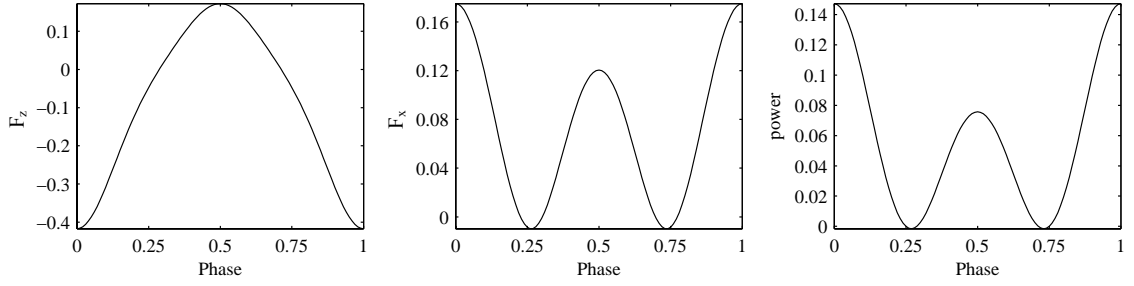


Figure 16: Force and power results from kinematic model 1 ('cruising')

Figure 16 shows results from the cruising flight from model 1, which again strongly resembles the base-model solution. While F_x and power results appear identical, there is some difference in the prediction of vertical force, F_z . This is again due to the resolution of forces in a direction normal to the wing, which now adds to the force in the Y -direction.

It should be noted that for both hovering and cruising, at small flapping angles the results from the base-model and model 1 would tend to the same values.

3.5.4 KINEMATIC MODEL 2: 4DOF, 3D FLAPPING

A more accurate representation of the wing requires a realistic 'ball and socket' shoulder joint that is capable of rotating around each local axis. Model 2 introduces a new angle that defines the wing sweep, which is the rotation of the wing about the z -axis. The wing is able to sweep back and forth by the angle γ_f . This motion is suggested to be important particular in cruising flight (Ramakrishnananda & Wong , 1999).

The equations of motion for the wing in this model are given as:

$$\phi_f = \Phi_f \sin(\omega t), \quad 63$$

$$\theta_f = \Theta_f \cos(\omega t), \quad 64$$

$$\gamma_f = \Gamma_f \cos(\omega t), \quad 65$$

with the velocity defined by

$$\mathbf{V}_{1_f} = \begin{pmatrix} \frac{s}{2} \omega \Phi_f \cos(\omega t) \sin \phi_f \sin \gamma_f - \frac{s}{2} \omega \Gamma_f \cos(\omega t - \Delta_{\text{sweep}}) \cos \phi_f \cos \gamma_f \\ -\frac{s}{2} \omega \Phi_f \cos(\omega t) \sin \phi_f \cos \gamma_f - \frac{s}{2} \omega \Gamma_f \cos(\omega t - \Delta_{\text{sweep}}) \cos \phi_f \sin \gamma_f \\ \frac{s}{2} \omega \Phi_f \cos(\omega t) \cos \phi_f \end{pmatrix}. \quad 66$$

The wing is now capable of tracing a path on a hemisphere with radius equal to the wing semispan; the motion is shown in Figure 17. Due to the wing moving in a spherical field, expressing the velocity in Cartesian coordinates becomes much lengthier.

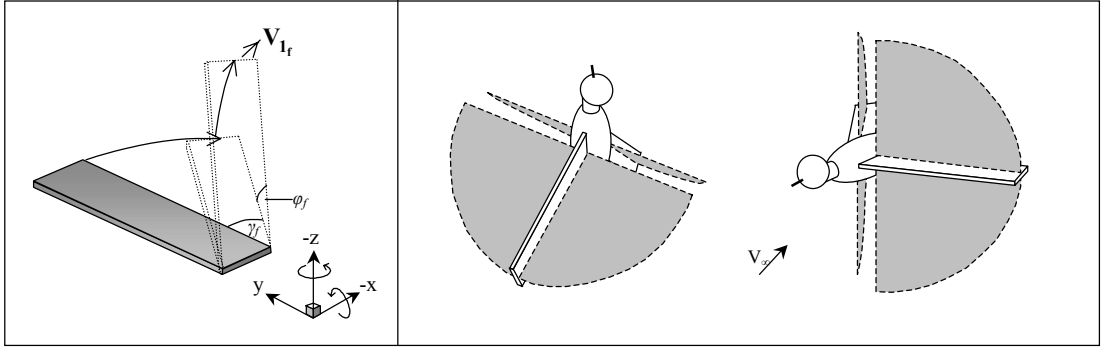


Figure 17: (left) Kinematic model 2 flapping wing motion; (right) suggested hovering and cruising regimes with sweep-planes depicted

The added degree of freedom results in an additional rotation requirement to account for the angle γ_f . Rotations are required in the order of sweep, flap and twist angles respectively:

$$\mathbf{V}_{2_f} = \mathbf{R}_{12_{\text{swist}}} \cdot (\mathbf{R}_{12_{\text{flap}}} \cdot (\mathbf{R}_{12_{\text{sweep}}} \cdot \mathbf{V}_{1_f})) = \begin{pmatrix} \cos \theta_f & 0 & -\sin \theta_f \\ 0 & 1 & 0 \\ \sin \theta_f & 0 & \cos \theta_f \end{pmatrix} \cdot \left(\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_f & \sin \phi_f \\ 0 & -\sin \phi_f & \cos \phi_f \end{pmatrix} \cdot \left(\begin{pmatrix} \cos \gamma_f & \sin \gamma_f & 0 \\ -\sin \gamma_f & \cos \gamma_f & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{V}_{1_f} \right) \right) \quad 67$$

An example of hovering flight will be shown to demonstrate the effects of the sweep angle. For simplicity a constant sweep angle will be applied to the motion:

$$V = 0, \theta = \frac{\pi}{2}, \Phi_f = \frac{\pi}{2}, \Theta_f = \frac{\pi}{4}, \gamma_f = \frac{\pi}{4}, \omega = 1:$$

68

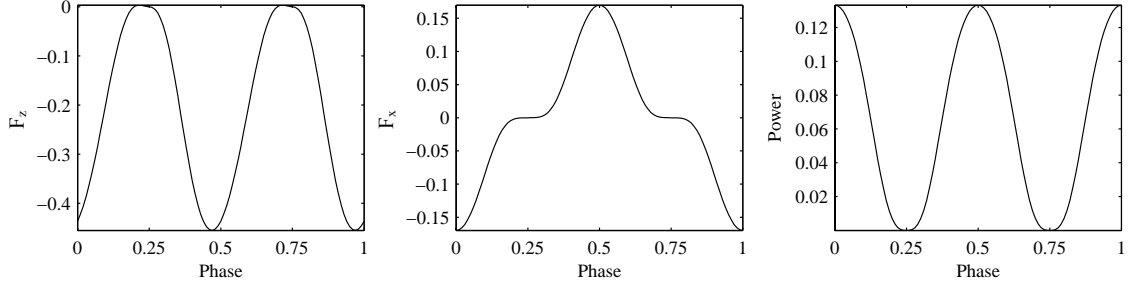


Figure 18: Force and power results from kinematic model 2 ('hovering')

Comparing Figure 15 and Figure 18, it can be seen the applied sweep angle produces a similar variation of force F_z over a cycle, though with reduced magnitudes. It is clear that resolving by the sweep angle causes a smaller component of force to act vertically in the Z -direction (and increased force in the Y -direction). The F_x force is unaffected by the sweep angle here, as the flapping velocity V_{lf} in X -direction is unchanged.

The power consumption now becomes negative at $Phase \sim 0.25$, $Phase \sim 0.75$: this corresponds to the upward vertical motion of the wing in the Z -axis as a result of the sweep angle.

Cruising flight is now defined with the addition of a constant sweep angle

$$V = 0.5, \theta = \frac{\pi}{16}, \Phi_f = \frac{\pi}{4}, \Theta_f = \frac{\pi}{16}, \gamma_f = \frac{\pi}{4}, \omega = 1:$$

69

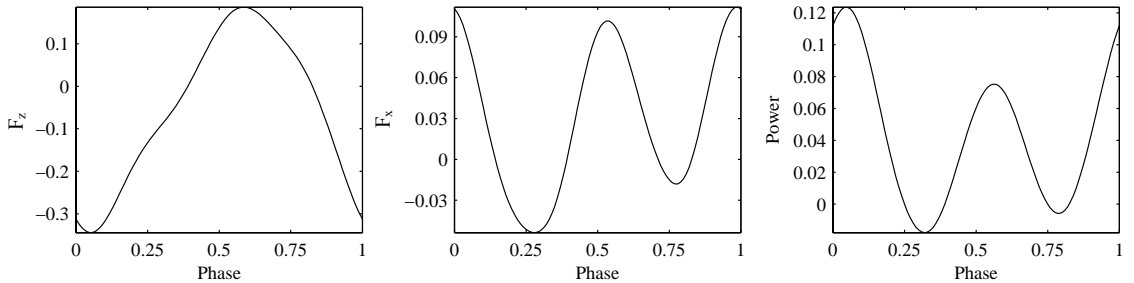


Figure 19: Force and power results from kinematic model 2 ('cruising')

The motion in this example is clearly more complicated than those observed previously, and the corresponding results are also more difficult to analyse. Figure

19 shows that the values of F_z now vary between upstroke and downstroke. With the applied sweep angle, the wing passes forwards and backwards into to the applied freestream wind. This results in increased local velocity on the downstroke ($0 < Phase < 0.25$) and reduced velocity on the upstroke ($0.25 < Phase < 0.5$).

The corresponding force F_x and power consumption are affected in a similar fashion. For $Phase \sim 0.25$ the downstroke acts directly into the freestream wind and produces a negative force in the X -direction (and negative power). However, for $0.75 < Phase < 1$ the downstroke acts in the same direction as the freestream flow (and with greater velocity) thus producing positive F_x force and positive power.

3.5.5 KINEMATIC MODEL 3 – 4DOF, 3D FLAPPING INCORPORATING SWEEP PLANE

Ellington (1984) highlighted that many flapping motions occur along a plane in 3d space, named the *sweep plane*. It is known that the sweep plane is a useful property in defining flapping motion in insects (Ellington, 1984) and is believed to also be important in avian motion. As an alternative to using the sweep angle γ_f as done previously, 3d flapping can be defined by incorporating a sweep plane angle θ_{sp} , which defines the inclination of the plane, relative to the BAS, in which flapping motion will occur. The equations of motion for the wing in this model are given as:

$$\phi_f = \Phi_f \sin(\omega t), \quad 70$$

$$\theta_f = \Theta_f \cos(\omega t), \quad 71$$

$$\theta_{sp} = \Theta_{sp} \cos(\omega t). \quad 72$$

The velocity in this case is defined by

$$\mathbf{V}_{1f} = \begin{pmatrix} \frac{s}{2} \omega \Phi_f \cos(\omega t) \cos \phi_f \sin \theta_{sp} \\ -\frac{s}{2} \omega \Phi_f \cos(\omega t) \sin \phi_f \\ \frac{s}{2} \omega \Phi_f \cos(\omega t) \cos \phi_f \cos \theta_{sp} \end{pmatrix}. \quad 73$$

The wing is still capable of tracing a path on a hemisphere, however as seen from equation 73 the linear velocity components are much more easily defined; Figure 20 shows the motion.

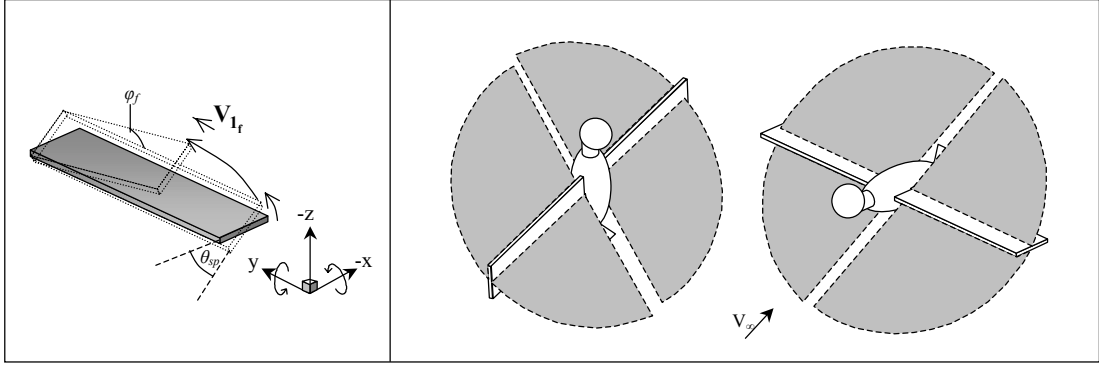


Figure 20: (left) Kinematic model 3 flapping wing motion; (right) suggested hovering and cruising regimes with sweep-planes depicted

Similar coordinate transformations are required as with model 2, however the initial rotation is now required about the y -axis to define the sweep plane:

$$\begin{aligned} \therefore \mathbf{V}_{2r} &= \mathbf{R}_{12_{\text{twist}}} \cdot (\mathbf{R}_{12_{\text{flap}}} \cdot (\mathbf{R}_{12_{\text{sweep-plane}}} \cdot \mathbf{V}_{1r})) = \\ &\begin{pmatrix} \cos \theta_f & 0 & -\sin \theta_f \\ 0 & 1 & 0 \\ \sin \theta_f & 0 & \cos \theta_f \end{pmatrix} \cdot \left(\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_f & \sin \phi_f \\ 0 & -\sin \phi_f & \cos \phi_f \end{pmatrix} \cdot \left(\begin{pmatrix} \cos \theta_{sp} & \sin \theta_{sp} & 0 \\ -\sin \theta_{sp} & \cos \theta_{sp} & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{V}_{1r} \right) \right) \end{aligned} \quad 74$$

To demonstrate the effects of incorporating a sweep plane, a hovering motion will be presented using a constant sweep plane angle:

$$\mathbf{V} = 0, \theta = \frac{\pi}{2}, \Phi_f = \frac{\pi}{2}, \Theta_f = \frac{\pi}{4}, \theta_{sp} = \frac{\pi}{4}, \omega = 1: \quad 75$$

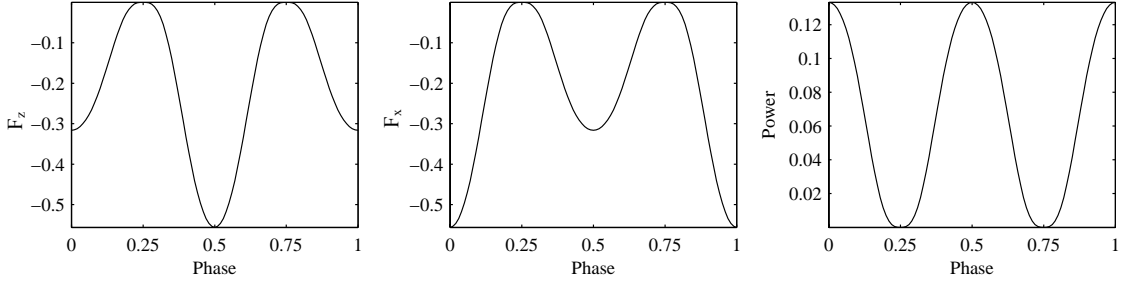


Figure 21: Force and power results for kinematic model 3 ('hovering')

The results in Figure 21 are derived from a flapping motion that is the same as that used in model 1, however, flapping now occurs along a plane inclined by an angle θ_{sp} to the xz -plane. The result of this inclination is that the force F_z now reduces as the wing passes vertically upwards on the downstroke ($0 < Phase < 0.25$, $0.75 < Phase < 1$), and increases as the wing passes vertically downwards on the upstroke ($0.25 < Phase < 0.75$).

With regards to the force F_x , it can now be seen that a net (negative) F_x force exists. The inclination of the sweep plane to the Z -axis results in the thrust force being resolved by the angle $(\theta - \theta_{sp})$ to give a net horizontal force. However the motion of the wing itself (relative to the local wind) is unchanged when compared to model 1, and hence the power consumption is identical between these two cases.

Cruising flight is defined as follows:

$$V = 0.5, \theta = \frac{\pi}{16}, \Phi_f = \frac{\pi}{4}, \Theta_f = \frac{\pi}{16}, \theta_{sp} = \frac{\pi}{4}, \omega = 1 : \quad 76$$

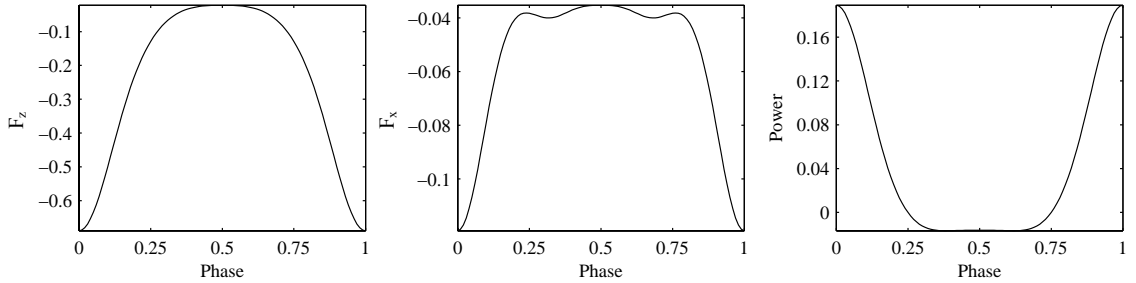


Figure 22: Force and power results from model 3 ('cruising')

The force F_z from model 3 now remains negative over the entire cycle. The steep inclination of the sweep plane angle is much more influential than the equivalent sweep angle used in model 2. Even on the upstroke, the wing produces a negative

vertical force that only reduces to zero as it reaches the neutral plane. The corresponding F_x force is now negative over the entire cycle, though with much smaller force magnitudes.

It appears as though the prescribed flapping parameters are actually producing forces that more closely resemble those seen in hovering flight than in cruise, even with the small body pitch angle. The inclination of the sweep plane acts in a similar way to the body pitch, whereby the thrust force produced by flapping is resolved into horizontal force for driving, and vertical force for balancing weight. The close coupling of these two parameters will prove important in the future analysis.

The power variation can again be related to the cases where the wing is passing into or away from the freestream wind. On the downstroke, the power consumption is seen to increase dramatically as it passes into the freestream wind. On the upstroke the opposite effect is seen, with almost no power expenditure as the wing moves along with the freestream.

As mentioned previously, the sweep plane is believed to be important in defining the flapping motion in avian flight, and therefore this model will be favoured over kinematic model 2 in the problem analysis. It will be interesting to compare the effect of using the sweep plane, over the more common application of the sweep angle.

3.5.6 KINEMATIC MODEL 4 – 5DOF, 3D FLAPPING INCORPORATING SWEEP PLANE, 2 PANEL WING

The final model to be introduced in this project involves developing a more sophisticated wing structure by incorporating an additional panel to the wing. The previous models have all aimed to represent a wing as a single panel with various levels of sophistication for modelling the shoulder joint. The second panel aims to represent the bird hand, with appropriate modelling for the wrist joint; the original panel will now represent the combined upper arm and forearm.

In this analysis the wrist joint is given a single degree of freedom whereby it can undergo a flapping motion similar to that seen previously. It is also believed that the twisting motion along the birds forearm is an important factor in defining the physical characteristics of flapping flight (Baumel, 1993). To incorporate this motion, a twist will also be applied to the wrist joint, which will be prescribed algebraically as a fraction of the shoulder twist angle¹⁸.

This model now contains two discrete panels, each requiring calculations of velocities, forces and power. The control point velocity of the original panel will be the same as in the previous case. Hence, forces and power acting on this panel are found in a similar way to the previous model. The equations of motion for this model are given by

$$\phi_f = \Phi_f \sin(\omega t), \quad 77$$

$$\theta_f = \Theta_f \cos(\omega t), \quad 78$$

$$\theta_{sp} = \Theta_{sp} \cos(\omega t), \quad 79$$

$$\phi_w = \Phi_w \sin(\omega t - \Delta) + \Psi. \quad 80$$

$$\theta_w = \Theta_w \cos(\omega t), \text{ where } \Theta_w = 0.25\Theta_f. \quad 81$$

The motion of hand is governed in a similar fashion to panel 1 using a flapping angle ϕ_w . In addition to this, the terms Δ and Ψ appear in equation 80 to represent the respective lag and bias of the panel 2 motion. The magnitudes of these factors and the influence on the overall system behaviour will be discussed further in Kinematic Model 4 – Parameter Range (3.5.7).

A new control point is defined on the second panel, and hence a new velocity has to be determined. It is now convenient to express the velocity relative to original wing panel:

¹⁸ From general observation, twisting of the outer wing section has been defined as approximately ¼ of that of the inner wing.

$$\mathbf{V}_{2_w} = \begin{pmatrix} 0 \\ -\left(\frac{s_2}{2}\right)\omega\Phi_w \cos(\omega t - \Delta)\sin\phi_w \\ \left(\frac{s_2}{2}\right)\omega\Phi_w \cos(\omega t - \Delta)\cos\phi_w \end{pmatrix}. \quad 82$$

There is also a velocity component at control point 2 in the body reference frame, arising from the motion of panel 1

$$\mathbf{V}_{1_f} = \begin{pmatrix} \left(s + \frac{s_2}{2}\right)\omega\Phi_f \cos(\omega t)\cos\phi_f \sin\theta_{sp} \\ -\left(s + \frac{s_2}{2}\right)\omega\Phi_f \cos(\omega t)\sin\phi_f \\ \left(s + \frac{s_2}{2}\right)\omega\Phi_f \cos(\omega t)\cos\phi_f \cos\theta_{sp} \end{pmatrix}. \quad 83$$

The second panel is therefore capable of moving in a semicircular arc, with radius equal to the length of the hand. In order to determine the local wind velocity acting on the second panel, a summation of the velocity derived from the panel 1 motion, and the velocity derived from panel 2 motion must be made (with appropriate rotations):

$$\mathbf{V}_{4_w} = \mathbf{R}_{24_{wtwist}} \cdot \left(\mathbf{R}_{24_{wflap}} \cdot \mathbf{V}_{2_w} \right) \quad 84$$

$$\mathbf{V}_{4_f} = \mathbf{R}_{24_{wtwist}} \cdot \left(\mathbf{R}_{24_{wflap}} \cdot \left(\mathbf{R}_{12_{twist}} \cdot \left(\mathbf{R}_{12_{flap}} \cdot \left(\mathbf{R}_{12_{sweep-plane}} \cdot \mathbf{V}_{1_w} \right) \right) \right) \right) \quad 85$$

$$\mathbf{V}_4 = \mathbf{V}_{4_w} + \mathbf{V}_{4_f} \quad 86$$

For clarity, the expansion of each rotation matrix has not been shown.

A demonstration of the effects of using an additional wing panel will now be shown for a hovering flight case; Figure 23 depicts the motions.

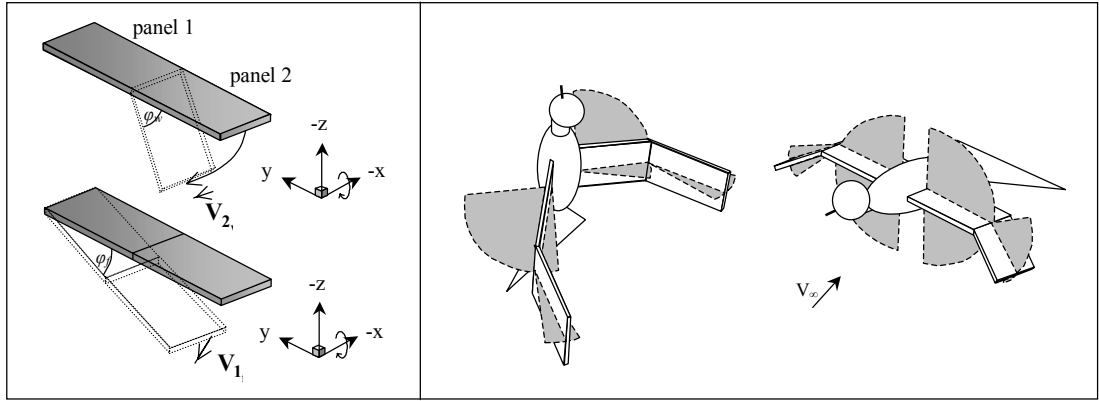


Figure 23: (left) Kinematic model 4 flapping wing and wrist motions; (right) suggested hovering and cruising regimes with sweep-planes depicted

$$V = 0, \theta = \frac{\pi}{2}, \Phi_f = \frac{\pi}{4}, \Theta_f = \frac{\pi}{4} \left(\because \Theta_w = \frac{\pi}{16} \right), \theta_{sp} = 0, \Phi_w = \frac{\pi}{8}, \Psi = 0, \Delta = 0: \quad 87$$

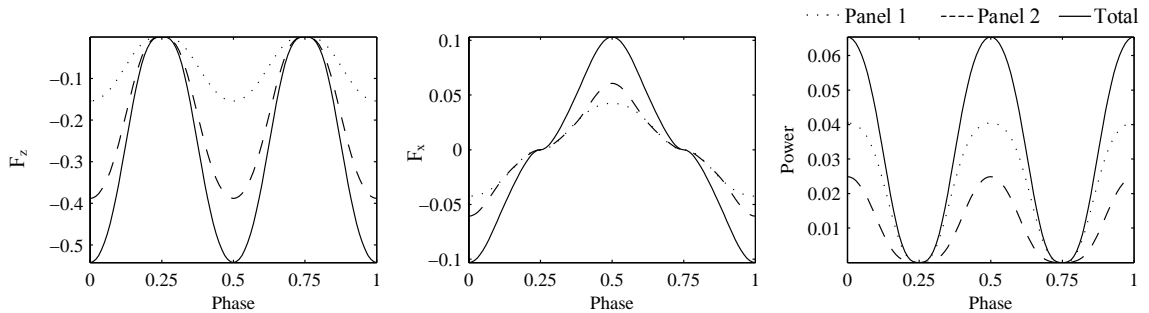


Figure 24: Force and power results from kinematic model 4 ('hovering')

This example models a body pitched vertically, with reduced flapping amplitude (compared to previous examples), to allow a flapping angle of the second panel to be introduced. The algebraic prescription of the panel 2 twist angle has also been included. The parameters have been chosen to loosely mimic those found in the previous models; therefore force magnitudes may differ between models.

Referring back to Figure 15, the variation in net force F_z is very similar between these two cases. It is useful to consider not only the net force acting on the wing, but also the 'individual' forces acting on each section of the wing. The results show that the force acting on panel 2 accounts for more than half of the total.

Considering now the force F_x , the overall variation is similar to the results from previous cases. Again, panel 2 accounts for the majority of the net force.

Now observing the power variation, while the majority of the force is acting on panel 2, most of the power expenditure is seen to occur on panel 1. This result may seem unusual, though intuitive reasoning would suggest that the shoulder is likely to be a more dominant driving component than the wrist; therefore panel 2 has more available power.

The small amplitude flapping motion occurring at the wrist is effective in increasing F_z and F_x forces without requiring large power expenditure. The location of the outer panel is such that the shoulder motion induces high velocity, and hence large forces, without requiring additional power at the wrist joint.

An example of cruising is given using the following parameters:

$$V = 0.5, \theta = 0, \Phi_f = \frac{\pi}{4}, \Theta_f = \frac{\pi}{16} \left(\because \Theta_w = \frac{\pi}{64} \right), \theta_{sp} = 0, \Phi_w = \frac{\pi}{8}, \Psi = \frac{\pi}{8}, \Delta = \frac{\pi}{2} : \quad 88$$

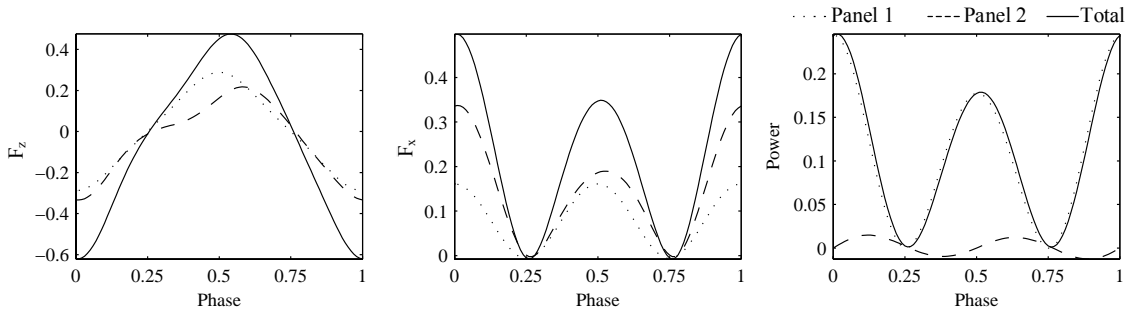


Figure 25: Force and power results from model 4 ('cruising')

Figure 25 shows a predicted cruising motion, in which the body pitch angle has now been set to equal zero. By applying a flapping amplitude in both panels, and offsetting the motion by prescribing a phase difference and bias to the motion of panel 2, a net F_z force can be produced. This is the first model in which the force F_z can be produced without requiring an inclined sweep plane (either from a positive body pitch, of sweep plane angle). This is achieved as the overall wing motion now differs between upstroke and downstroke. The result is that the magnitude of F_z force produced on the downstroke is greater than on the upstroke. More specifically, the force on panel 1 yields not net F_z force, while the force on panel 2 does.

The force F_x follows similar variations to those seen in the previous cases. It can be noted that the force F_x produced by panel 2 is again greater than panel 1 due to the additional twisting motion supplied by the wrist. Again, the power consumption results show that while the majority of forces (including a net F_z force in this case) are arising from panel 2; the shoulder joint that is dominant in producing power.

3.5.7 KINEMATIC MODEL 4 – PARAMETER RANGE

The previous section described the use of an additional panel to more closely resemble flapping wings found in nature. The example solutions were performed using feasible values for phase lag and bias, however for future analysis a less ambiguous approach is desirable. The constraints on the maximum values of wrist flapping amplitude, lag and bias will be determined by considering the optimal method of producing forces, without neglecting physiological limitations.

The previous section mentioned that the hand motion can be defined to produce a net F_z force without resolving forces using body pitch or sweep plane inclination. It has been suggested by Baumel (1993) that flapping motion in birds exploits the wing as a multi-panel structure by using the inner section to produce the majority of the thrust, and the outer section to produce the majority of the lift. This is done by a harmonic, thrust producing motion of the inner wing, and an offset lift producing motion of the outer wing. This type of behaviour is readily observable in nature as a bird's upstroke and downstroke are distinctly different.

The maximum flapping amplitude of both panels is defined by physiological constraints. For this project appropriate values must be chosen to allow for accurate modelling, without constraining the motion to that of any particular species. Observations from nature suggest that the values φ_f and φ_w should lie in the range $\pi/4$ to $-\pi/4$ rads.

Information regarding the lag and bias constraints is unavailable. Therefore in order to deduce suitable quantities, test solutions will be obtained using a range of values. By considering the cruising behaviour (as used previously), the maximum force $F_{z(max)}$ can be determined using a range of flapping amplitudes (for both φ_f and φ_w),

lag and bias values; the body pitch, sweep plane and twist amplitudes will be omitted for simplicity. The following parameters will be used:

$$V = 0.5, \theta = 0, \Phi_f = 0 : \frac{\pi}{2}, \Theta_f = 0 (\therefore \Theta_w = 0), \theta_{sp} = 0, \Phi_w = 0 : \frac{\pi}{2}, \Psi = 0 : \frac{\pi}{2}, \Delta = 0 : \frac{\pi}{2};$$

Figure 26 shows an example of the results produced:

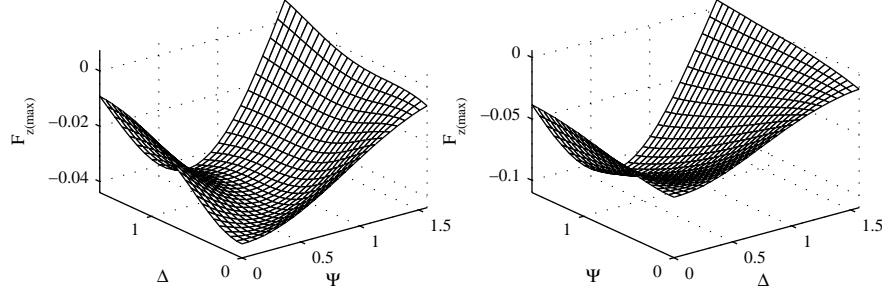


Figure 26: Variation of $F_{z(max)}$ with Δ and Ψ for $\Phi_f = \pi/5, \Phi_w = 2\pi/5$ (left) and $\Phi_f = 2\pi/5, \Phi_w = \pi/5$ (right)

Figure 26 shows the variation of $F_{z(max)}$ for two combinations of flapping amplitudes, over a range of lag and bias values. In both cases the maximum force occurs when using lag and bias values of $\pi/2$ rads and $\pi/4$ rads respectively. The results show that greater forces can be achieved when the larger amplitude is applied to panel 1, rather than panel 2, as would be expected. To assess the results more thoroughly, the bias and lag values that give the greatest $F_{z(max)}$ for each amplitude (Ψ^* and Δ^* respectively) are shown in Figure 27:

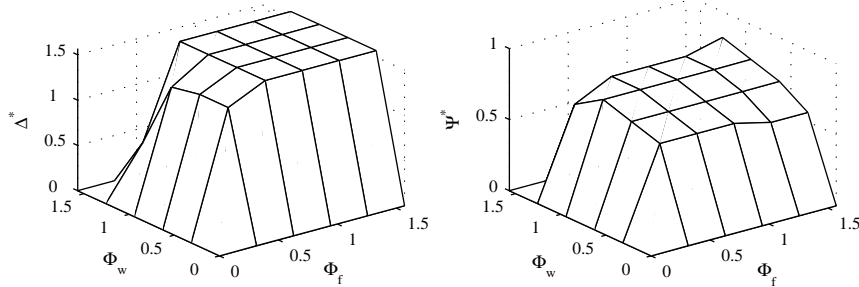


Figure 27: Lag and bias values to provide maximum $F_{z(max)}$

The results show that over the majority of the range of values of flapping amplitudes, Δ^* is equal to $\pi/2$ rads. At small Φ_f and large Φ_w , Δ^* reduces to zero, however these values represent a highly unrealistic case of excessive wrist flapping that is not

appropriate for this analysis. While Ψ^* shows slightly more variation, the results can be approximated as $\pi/4$ rads.

It was expected that for appropriate values of Φ_f and Φ_w , obtained from physiological data, suitable bias and lag values could be derived from the results in Figure 27. However a suitable simplification would be to take the values $\Delta_{(max)}=\pi/2$ rads and $\Psi_{(max)}=\pi/4$ rads for future constraints. This will provide closure to constraining the parameters that define the motion of panel 2, without suffering a noticeable loss of accuracy.

Having thoroughly tested the kinematic models, attention can now be turned to the other main functions of the program.

3.6 OPTIMISATION METHODS

Base-Model Design (3.3) discussed how the optimisation procedure is used to determine suitable motion parameters to meet input requirements. This chapter will aim to assess the possible optimisation methods that could be employed in this project. Information will be provided on three common methods of obtaining minimum values to functions of several variables.

A desirable optimisation method would offer reasonable accuracy, with low solution times and high stability, in accordance with the nature of the project. The chapter will detail the methods of Complete Search, Gradient Based methods and also Genetic Algorithms.

3.6.1 COMPLETE SEARCH ALGORITHM

The first method to be considered will be the Complete Search Algorithm (CSA). This is the simplest method of obtaining a numerical approximation to the minimum value of a function. By sweeping across the entire solution space and obtaining solutions at set intervals, an estimate of the function minimum can be made. This type of method is suitable for tackling problems in which function evaluation time is particularly small, and the function minimum is desired to within a given margin of error.

In this particular project the design parameters that are to be optimised may be found to given error defined from a scientific viewpoint, or alternatively from graphical tolerances. The error is used to determine the resolution with which the solutions must be obtained i.e. to define the spacing between nodes on the design space. The method operates by passing along each dimension in turn (in steps given by the desired resolution), calculating the solution value at each node, and updating the current solution minimum as necessary.

In order to test such a procedure, two functions will be used to model possible solution spaces for this project. The functions will aim to loosely represent typical

variations in cost function, with the variables acting as ‘dummy’ motion parameters. Also, by applying specific continuous functions, the solution minima can be easily derived analytically and compared with the numerical results.

The two functions are given as

$$f(x, y) = ((x - 0.5)^2 + 0.5)((y - 0.5)^2 + 0.5), \quad (0 \leq x, y \leq 1) \quad 89$$

$$g(x, y) = 0.25 \sin(2x) + \sin(2y), \quad 90$$

which yields

$$f_{\min} = f(0.5, 0.5) = 0.25, \quad 91$$

$$g_{\min} = g(0, 0) = 0. \quad 92$$

The CSA was invoked using 12 nodes along each dimension; the results are given in Figure 28:

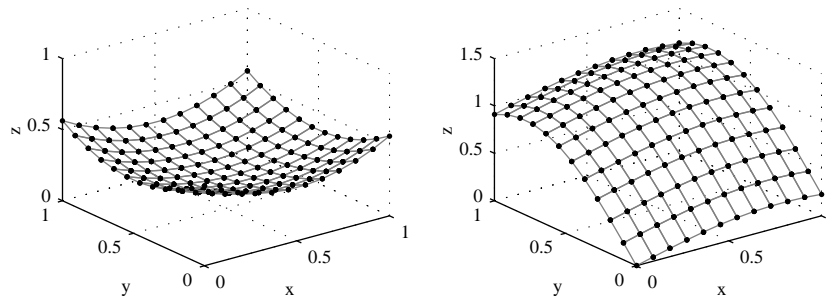


Figure 28: CSA solution for optimisation for $f(x,y)$ (left) and $g(x,y)$ (right) using 12 nodes in each dimension

Figure 28 shows how the CSA produces a numerical result at each node in the solution space. It is clear that this method is useful in visualising the design space for such a 2-dimensional problem.

The results for the minimum function values are given in Table 3. The CSA predicts a minimum value of the function $f(x,y)$ as $f_{\min}=f(5/11,5/11)=0.2521$ (4dp). Due to the nature of the dimension size and chosen node spacing, the method fails to capture the exact function minimum. It is clear from Figure 28 that the four nodes surrounding

the centre point on the design space each return the same minimum value, i.e. $f_{min}=f(5/11,5/11)=f(5/11,6/11)=f(6/11,5/11)=f(6/11,6/11)$. This result demonstrates the sensitivity of the solution to the node spacing.

The minimum value of the function $g(x,y)$ is predicted as $g_{min}g(0,0)=0$. While the node spacing remains unchanged, the solution is calculated with no error in this case. This is due to a node coinciding with the corner of the design space, which is where the global minimum lies. The CSA correctly selects the global minimum to the function, as opposed to the local minima found at other corners of the design space.

Providing that the node spacing is substantially smaller than spacing between local minima, the CSA will generally be successful in approximating the function global minimum.

It is clear from Figure 28 that the CSA requires a relatively large number of iterations in order to accurately predict the global minimum; this factor will be discussed further in the Method Comparison (3.6.4).

3.6.2 GRADIENT BASED METHODS

A more advanced approach to solving the problem is to employ Gradient Based Methods (GBM). This is an iterative class of methods whereby the function value and its derivatives at a point (in the solution space) are used to drive the solution towards the minimum value.

A common practise used when defining systems to be optimised using GBM is to apply boundaries and constraints to the problem wherever possible, in order to increase efficiency. A key characteristic of this problem is that the function minimum is to be derived within the given design space. The boundaries of the design space are defined by the minimum and maximum parameters values; in the context of this project, the parameter bounds will be set largely by physiological constraints.

Using MATLAB, the solution to a ‘constrained minimisation’ can be obtained using the function `fmincon`. This function is invoked using default settings which apply Sequential Quadratic Programming (SQP), Quasi-Newton and Line-Search methods to determine the solution.

SQP is a highly sophisticated method used in non-linear programming. This method is capable of “mimicking Newton’s method for unconstrained optimisation” (The MathWorks, Inc., 2002), which is known to be highly successful for a range of problems. The method works iteratively, whereby each iteration generates a quadratic sub-problem. These sub-problems are tackled using the Quasi-Newton Method to derive a suitable direction in which the search should continue in the solution space. Line search methods are used to determine suitable step sizes in the given direction. For precise details on the functionality of these methods, the reader is asked to refer to MATLAB product documentation.

This GBM will be tested using the functions $f(x,y)$ and $g(x,y)$ (3.6.1). The results are given in Figure 29:

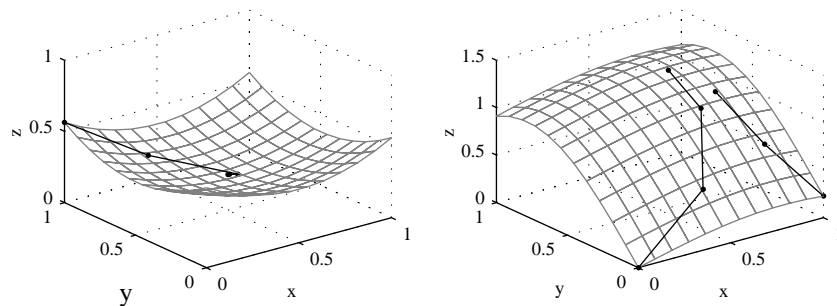


Figure 29: GBM solution for optimisation for $f(x,y)$ (left) and $g(x,y)$ (right)

The results for the minimum function values using the GBM are given in Table 3. The GGM predicts the solution minimum value of $f(x,y)$ as $f_{min}=f(0.5,0.5)=0.2500(4dp)$. Figure 29 shows how the initial guess value of $f(0,1)$ is directed towards the centre of the solution space to produce an accurate result in four iterations. The GBM is clearly highly effective for finding minima to such functions.

For the function $g(x,y)$, the optimisation process becomes somewhat more involved. While the optimiser is invoked in exactly the same manner, a starting value of $g(0,1)$

yields the result $g_{min}=g(0,1)=0.9093$ (4dp); the solution process begins and ends at the local minima at the boundary.

A second attempt is made by applying a starting point of $g(0.8,0.5)$ which lies outside of a local minimum. However, the optimiser again drives the solution towards a local minima $g_{min}=g(1,0)=0.2273$ (4dp), as shown in Figure 29. A third attempt starting at $g(0.7,0.7)$ obtains the correct solution, $g_{min}=g(0,0)=0$, in three iterations.

It is clear that the gradient based method can offer a vast improvement in performance compared to the CSA. This, more intelligent approach, exploits that fact that information over most of the design space can be made redundant by focussing solely on the path between a current solution and the function minimum.

However an immediate downfall of GBM has been cited with regards to local minima. If unsuitable starting points are supplied, the global minimum may not be attained. With the examples show in Figure 29, the function has been plotted for reference and hence suitable starting points can be predicted. In a more general case the location of local minima is unknown, hence the prescription of suitable starting points is much more difficult.

With the performance advantage comes a distinct reduction in the reliability of the approach: the optimiser can no longer be used ‘blindly’ and now requires much more careful consideration of the feasibility of the output.

3.6.3 GENETIC ALGORITHM

The final method to be tested is the Genetic Algorithm (GA). GAs are a particular class of evolutionary algorithms commonly applied to global optimisation. They are constructed by considering the application of biologically derived techniques to determine minimum values of linear and non-linear functions. For a full understanding of GAs the reader should refer to Mitchell (1998).

A characteristic feature of GAs is that they are highly problem specific and can be tuned to a particular study to achieve high accuracy, reliability and performance. For this reason, a unique algorithm is developed for this project, the source code for which can be found in Appendix 2.

In this particular study, the ‘chromosome’ is represented by an array, in which each element (or ‘gene’) is an independent parameter. Applying this to the kinematic model, each gene defines the value of chosen parameters (e.g. φ_f , θ_f etc.). Each gene is defined by a positive real number (b) that represents the fraction of the maximum possible value for that parameter, for example the flapping amplitude

$$\Phi_f = b \cdot \Phi_{f_{max}}, \text{ where } 0 < b < 1 \text{ and } \Phi_{max} \text{ is determined by physiological constraints.}$$

The original gene pool of N chromosomes is created using the `rand` function on MATLAB and is therefore unique for each simulation. Each chromosome is ranked according to the cost of the solution obtained using the given genes. The selection of the best chromosomes from the gene pool is done using the standard ‘roulette wheel’ method, whereby the probability of a chromosome being selected is inversely proportional to its cost. Selection occurs in the replacement method, meaning that the same chromosome can be selected more than once to become a parent.

The selection process chooses two chromosomes that will be used as parents. The parents undergo the biological process of ‘crossover’ according to a crossover probability, in order to form two offspring. The offspring are then ‘mutated’ at each gene with a given mutation probability. In addition, a ‘mutation severity’ is included whereby if mutation occurs, an additional random number prescribes the increase/decrease of the gene value. Selection occurs $N/2$ times to form a new generation.

To test this method, a gene pool containing ten chromosomes was formed. In order to apply the functions $f(x,y)$ and $g(x,y)$, each chromosome only contained two genes; while this may not be strictly applicable to the GA methodology, it is still useful for

comparative purposes. A solution was obtained using a mutation probability of 0.5 and a crossover probability of 0.5. The results are shown in Figure 30:

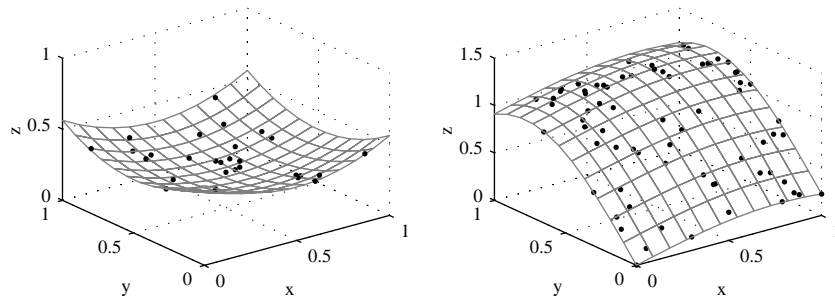


Figure 30: Genetic algorithm solution for optimisation for $f(x,y)$ (left) and $g(x,y)$ (right)

The results for the minimum function values using the GA are given in Table 3. This method predicts a minimum value $f(x,y)$ as $f_{min}=f(0.5306,0.5040)=0.2505(4dp)$. Figure 30 shows the seemingly random scattering of results around the solution domain. Crossovers and large mutations cause the solution to move rapidly around the solution space. Smaller mutations result in the gene pool being nudged towards a lower cost set of chromosomes; this behaviour is observable where results tend to cluster together, such as those around the function minimum. The overall solution is predicted to a high degree of accuracy.

The function $g(x,y)$ required an extensive search of the solution space before the global minimum was identified as $g_{min}=g(0,0)=0.000(4dp)$. As expected, the GA is successful in finding solutions at the edges of the design space, as any mutations that would force the solution out of the boundary are truncated to stop at the boundary. An important point to note is that the GA also identifies the local minimum $g_{min}=g(1,0)=0.2273$. Having reached this solution, the algorithm continues until a crossover or mutation moves the solution to another point. Again clustering of the results is visible, particularly around local minimum and regions of shallow gradients.

This final approach has offered a relatively reliable solution procedure that is capable of producing highly accurate results. It could be argued that the slight decrease in accuracy, when compared to the gradient based method, is balanced with an increase in reliability.

The GA method again relies on the fact that information across the entire design space is essential in deriving the global minima. However this method does take into account the effect of local minima on the solution, and offers multiple solutions to a given problem.

3.6.4 METHOD COMPARISON

Having considered each method individually, it is now useful to offer a direct comparison between the results. The complete solution data set has been summarised in Table 3:

	Complete Search	Gradient Based	Genetic Algorithm
$f(x,y)$			
Error in f_{min} (4dp)	0.0021	0.0000	0.0005
No. Iterations	144	4	27
$g(x,y)$			
Error in g_{min} (4dp)	0.0000	0.0000*	0.0000
No. Iterations	144	4*	44

Table 3: Comparison of errors and no. iterations for different optimisation methods

* Data from the successful trial using a starting point of $g(0.7,0.7)$

While the function $g(x,y)$ was useful in showing the limitations of some methods, the location of the global minimum is such that a successful solution is likely to give zero error for all methods. It is believed that the function $f(x,y)$ offers a more suitable test of the solution accuracy. With this in mind, the GBM proved to be most accurate, followed by the GA and lastly the CSA.

The other point that has not yet been considered is the solution time, or more specifically the number of iterations required to obtain a solution. Table 3 highlights that the number of iterations appears to correlate to the solution accuracy, with the GBM requiring fewer, followed by the GA and then the CSA. Figure 31 shows the solution behaviour during the iteration process:

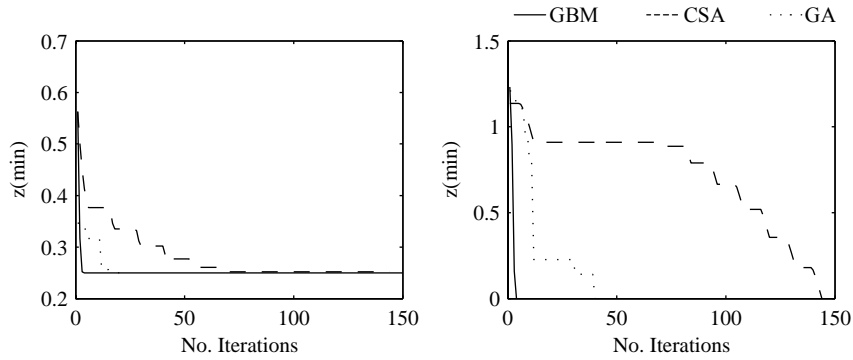


Figure 31: Variation of predicted solution minima with no. iterations for three optimisation methods for function $f(x,y)$ (left) and $g(x,y)$ (right)¹⁹

Solution using the CSA shows a stepping decrease on the predicted function minimum as the process sweeps across the solution space. For the function $f(x,y)$, this method obtains the minimum value after 66 iterations, however the full search is completed in 144 iterations.

The solution using Gradient Based Methods gives the expected characteristic behaviour of rapid convergence in 3 or 4 iterations. This occurs whether the derived minimum is global (as shown) or local.

The GA shows less predictable behaviour as the influence of random factors is unclear in the results. While solutions for these cases are shown to require less than 50 iterations, this is not always the case. This is an important point, as it is far less obvious when using evolutionary techniques, how to predict approximate solution times, as they are highly problem specific.

3.6.5 METHOD SELECTION

The analysis has shown that when moving from a CSA to GBM, an increase in accuracy and performance is coupled with an inherent loss in reliability. For this particular project, a desirable optimisation method would be more heavily weighted towards having lower solution times and high stability, with slightly less accuracy required. From these requirements and the results observed, it would seem that an evolutionary method such as the GA offers a strong compromise.

¹⁹ The solution obtained using the Complete Search method was initiated at $x=y=1$ for both function solutions

However it is believed that at this stage, in order to be use the GA confidently, a far greater degree of testing would be required. The GA was custom built for this project, and hence does not infer the same degree of reliability as the available MATLAB functions for example.

It is believed that the problems encountered using the GBM can be overcome relatively easily; one method of avoiding local minima would be to obtain solutions using a range of starting points, as done in the previous analysis: with such rapid solution, this would not inhibit the overall system performance.

In addition to this, physical information and intuition can be used to predict suitable starting points for the optimiser. The nature of this project is not only to produce correct results, but also to enhance understanding of the physical aspects of the problem; this type of approach is believed to fulfil both of these roles.

In cases where spurious results are seen, and the data is to be analysed more rigorously, the CSA will prove useful to identify the properties of the solution space. The GA may prove to be necessary as the research program develops and the number of solutions parameters (and hence the solution space dimensionality) increases.

4. RESULTS

4.1 NUMERICAL AND PHYSICAL RESULTS

Having defined the different kinematic models and identified a suitable optimisation method, solutions to the problem can now be obtained. This chapter will describe the optimisation of different kinematic models to obtain motions that meet specified input parameters. Details will be given of the solution attempts using the different models, problems that were encountered and the strategies employed to achieve feasible solutions. The reader will obtain a full understating of the intricacies of the solution procedure, with reference to the management of the optimisation tool and numerical methods.

Resulting solution spaces will be presented and compared to background knowledge of the problem domain. Discussions will be focussed on numerical results and their corresponding physical implication, while the visual results for simulation will be covered in the Visualisation and Performance (4.2). This section considers the solution procedure for four kinematic models with particular attention paid to the base-model.

4.1.1 BASE KINEMATIC MODEL SOLUTION

The first solution to be obtained uses the base-model to define the flapping kinematics; a solution to this problem given in Appendix 1b will serve as a useful reference point. The results and the solution procedure used in the base-model will be assessed in terms of accuracy and feasibility. Appropriate fixes to the method in Appendix 1 will be described in detail, in order to give the reader an insight into the sensitivity and complexity of the solution procedure. Following this, some example solutions will be explained and related to previous analysis in this, and other works.

SPURIOUS RESULTS

It was mentioned in Base-Model Design (3.3) that in order to apply the method of trajectory simulation, governing parameters that define the bird motion are required for a range of forces and velocities. As an initial step, a solution for constant forces F_x and F_z , can be obtained over a range of velocities. Figure 32 shows the resulting motion parameters for the optimised solution of the base-model²⁰:

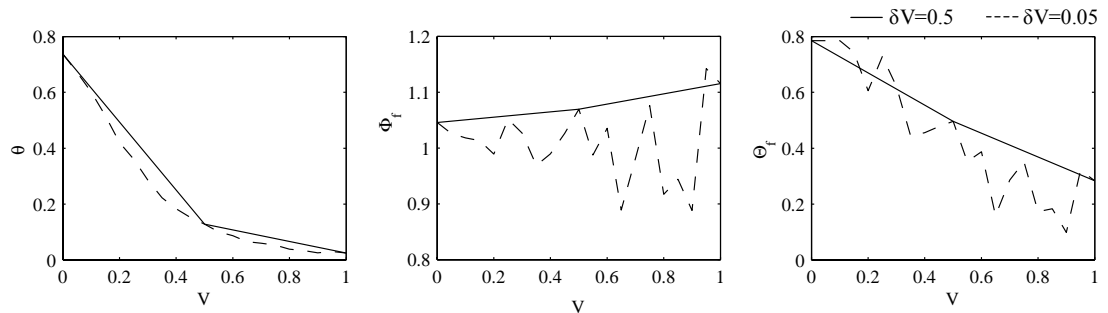


Figure 32: Base-model solution for varying velocity

The data from the base-model solution has been shown, whereby the velocity increases in increments, $\delta V = 0.5$. All flapping parameters show a relatively smooth transition across the data set. The body pitch attitude and wing twist amplitudes both decrease with increasing velocity, as expected from Preliminary Analysis (3.4). The wing flapping amplitude remains almost constant over the entire velocity range, which opposes the predictions made in Preliminary Analysis (3.4).

Figure 32 also shows the solution obtained using a smaller velocity increment, $\delta V = 0.05$. The variation of pitch attitude is still smooth and continuous; however the flapping and twisting amplitudes now follow a much more discontinuous route; it should be noted that at equal velocity values ($V=0$, $V=0.5$ and $V=1$) the results are equal to those for the larger velocity increment. It is clear that incrementing the velocity in large steps produces a smooth, though not necessarily accurate, set of results.

²⁰ Solutions are obtained using the standard base-model parameters for cost weighting etc. unless otherwise stated. Small F_x and F_z requirements are applied.

Part of the project requirements is to produce results that are fit for animation purposes, which essentially suggests smooth results. This can be achieved by obtaining solutions over a sparse set of velocity and force values. However, to accomplish the goals of numerical accuracy and strong physical understanding, it is necessary to consider alternative methods. Specifically, it is important to analyse the solution behaviour with small variations in input parameters to try and determine the validity of such oscillating results, and to assess the factors that induce this behaviour.

While the pitch attitude shows a feasible, smooth variation with velocity, the flapping and twisting amplitudes clearly do not. It is useful to consider the nature of the optimisation space: Figure 33 shows the variation in solution cost with flapping and twisting amplitudes for various velocity values.

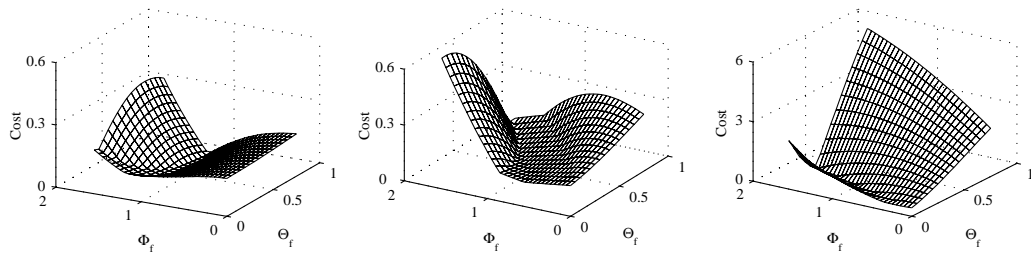


Figure 33: Optimisation spaces for $V=0.2, 0.7$ and 3.0

The results for the lower velocity, $V=0.2$, show that a distinct *valley* of low cost solutions exists across the optimisation space, in which the derived minimum values in Figure 32 are identified. In the higher velocity case, the results develop an almost planar region in which low cost solutions exists for a substantial range of parameter values. The minimum value was predicted in this case at $\Phi_f=0.98, \Theta_f=0.28$.

For reference, a higher velocity optimisation space has also been included. While the physicality of this case may not be strictly relevant, it does emphasise the variation in numerical behaviour of the results: the high velocity results again forming a valley of low cost solutions. In this case it is also more clearly defined where the global minimum lies. Due to the nature of gradient-based methods it seems apparent that obtaining the global minimum should be more straightforward in the high and low velocity cases, than in the intermediate velocity values. It seems reasonable to

assume that solutions will be more well-defined in the high and low velocity cases, than in the intermediate range. This assumption relates directly to the theory of flight regimes: the hover and cruise flight regimes are well defined, while the intermediate velocity is not, and is therefore less commonly observed in nature.

DISTANCE FUNCTIONS

While it is false to assume that the solution varies gradually with increasing velocity, it seems appropriate to expect a more continuous solution than seen in Figure 32. It is desirable to obtain a solution that is flexible enough to capture the true physics of the system, without being too sensitive to minor alterations in the optimisation space (and hence producing unstable data).

In order to restrict the degree of sensitivity that the solution has to the optimisation space, additional constraints should be applied to the model. The distance function is an initial attempt to help increase the stability of the solution.

As mentioned in The Gaits of Motion (3.2.5), physiological systems show preference to remain in a particular motion regime. The nature of this method is to restrict the *movement* of the minimum point in the optimisation space by increasing the values (i.e. the costs) at the surrounding nodes. This causes the optimiser to remain at or a near a current solution, therefore portraying a system's preference to remain in the current flight regime.

The method used, involves increasing the cost at all nodes by an amount proportional to their *distance* (i.e. the difference in flapping parameters) from the previous solution. Therefore the subsequent optimisation will be performed on an adapted optimisation space. The *distance-cost* function can be described at velocity value V_n mathematically by

$$C_{V_n}(\Phi_f, \Theta_f, \dots) = v \cdot \left| \begin{pmatrix} \Phi_f \\ \Theta_f \\ \vdots \end{pmatrix}_{V_n} - \begin{pmatrix} \Phi_f \\ \Theta_f \\ \vdots \end{pmatrix}_{V_{n-1}} \right|. \quad 93$$

where ν is a weighting parameter.

This additional cost is simply added to the previous solution, to produce results such as those shown in Figure 34:

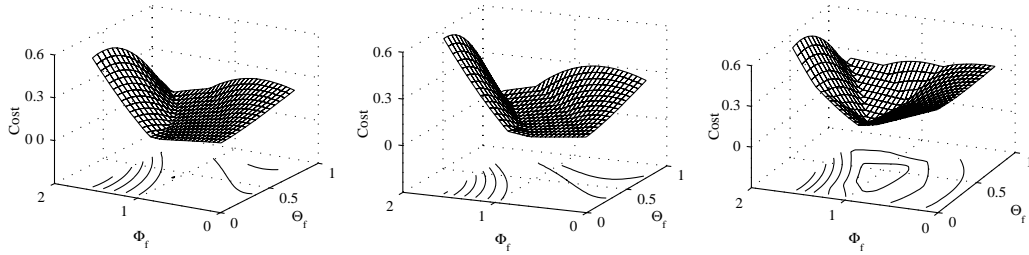


Figure 34: Optimisation (with contours) for $V=0.6$ (left), $V=0.7$ (centre) and $V=0.7$ with distance cost weighting ($\nu=0.2$) (right)

The results in Figure 34 show how the optimisation space varies as the velocity is increased from 0.6 to 0.7; results are shown for cases with and without the distance function. The results from Figure 32 showed that for a velocity of 0.6, the derived optimal flapping parameters are $\Phi_f=1.03$, $\Theta_f=0.40$. Without the use of the distance function, the solution flattens and hence offers a wide range of low cost solutions; the optimised solution gives $\Phi_f=0.97$ and $\Theta_f=0.26$. However, by incorporating the distance function with a weighting, $\nu=0.2$, the solution at $V=0.7$ is $\Phi_f=1.03$, $\Theta_f=0.38$, thus constraining the change in motion parameters.

The distance function clearly has the desired affect on constraining the solution evolution with varying velocity. Unfortunately, serious problems have also been cited when using this method. The first issue is that the exact variation of optimal parameters with velocity proves to be very sensitive to the choice of weighting of the distance function. As this function is an empirical measure, and is not directly related to the physical system, it would require substantial testing to determine a suitable value for use over a range of solutions. Details of the solutions using a range of distance functions are shown in Appendix 3

The second problem relates to the idea of altering the optimisation space. While this method does promote stability, it may restrict the movement of the solution to a

degree whereby important physical phenomena are not recognised. For example there may be a situation in which a large variation in a flapping parameter should occur at a critical velocity value, however the distance function may falsely act to shield the optimiser from this ‘new’ solution. In order to achieve accurate results a more subtle approach may be required, that does not prescribe a direct alteration of the optimisation space.

UPDATING SOLUTIONS

In the previous section the function `fmincon`, used in the optimisation, was supplied with an initial guess value in the centre of the optimisation space for each velocity value tested. While this is appropriate while the global minima lies near the centre, the analysis of Optimisation Methods (3.6) suggested that care should be taken when prescribing the starting point for the gradient based methods of optimisation. This section aims to consider the affect of using a new initial guess for each velocity value.

When performing large computations on-the-fly, optimisation routines often update the initial guess value according to past results, in order to improve efficiency. If an optimisation is deemed successful in producing an accurate solution, it is appropriate to use this value as a suitable starting point for the following case, providing that no substantial changes occur to the optimisation space. This approach is straightforward to encode into the current model, whereby the output from the optimisation function is stored in a temporary variable, and used as an input parameter for the following calculation.

The results for the optimisation of the base-model over a velocity range 0 to 1, using updated initial guess values is shown in Figure 35:

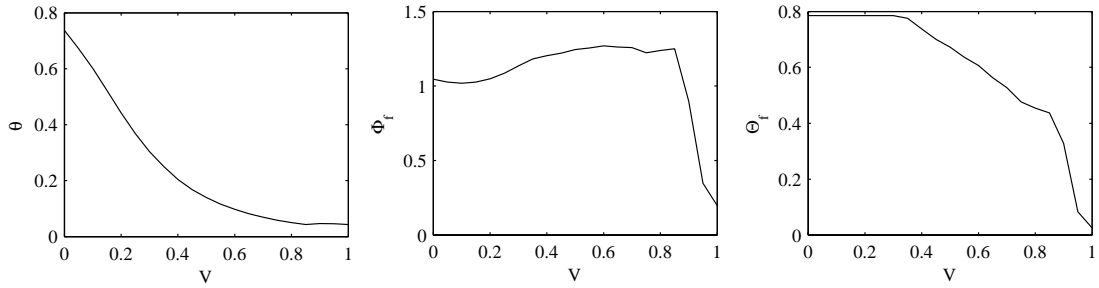


Figure 35: Base-model solution for varying velocity with updated initial guess values for optimisation

The method of updating the solution clearly has the desired affect of increasing the solution stability. The results show much more uniform behaviour (even when using a small velocity increment) with only one possible anomalous result for Φ_f at $V=0.8$; this will be discussed in due course. The design space has not been altered in any way, and it is therefore reasonable to assume that these results are offering a strong representation of the true physical behaviour of the system.

The variation of the body pitch attitude is similar to that seen previously in Figure 32: increasing the velocity causes the body pitch to decrease as expected. The body rotation acts to decrease the aerodynamic drag on the body and also to rotate the force vector in the positive X -direction. It should be noted that at $V=0$, the body is not pitched vertically as in true hovering, due to the prescription of the (constant) F_x force requirement.

The flapping amplitude now shows a slight decrease, followed by a more substantial increase as the velocity is increased from zero. At high velocity values the amplitude shows a rapid decrease as it enters the cruise regime. It is important to note that the exact physical limit to the maximum velocity attainable is not clear at this stage, as mentioned in Base-Model Design (3.3). It may therefore be possible that the flapping amplitude undergoes a less substantial change if the maximum velocity of a particular species was 0.9, for example. This type of limitation will be considered later in this chapter, when identifying particular flight regimes from the results.

The twist amplitude remains at its maximum value in the range $V \sim 0$ to $V \sim 0.3$, proving that in (or near) the hovering regime the twisting motion of the wing is beneficial. This correlates to the results shown in Preliminary Analysis (3.4). In the

intermediate velocity stages the twist amplitude shows an approximately linear decrease in value with increasing velocity. Approaching the cruise regime, the twist amplitude now shows a sudden decrease, and the solution offers negligible twist amplitude at $V=1$. This important result directly reflects the analytical solution shown in Preliminary Analysis (3.4). While the analytical method did not include effects of power expenditure, the method still predicted that wing twist is only beneficial up to a velocity value of unity, which matches the result obtained from this numerical solution.

COST FRACTIONS

The optimiser not only provides the lowest cost solution parameters, but also the numerical details of the how the net cost is derived. The Gaits of Motion (3.2.5) explained how different flapping regimes have specific force and velocity requirements; this idea was extended in Power Consumption (3.2.6) whereby it was suggested that the individual *cost fractions* (i.e. force deficit and power expenditure) incurred will vary according to flight regime. It is therefore useful to consider how the individual costs vary for the solution obtained for the varying velocity case; Figure 36 contains the results for the force and power costs as percentages of the total cost, for each velocity value:

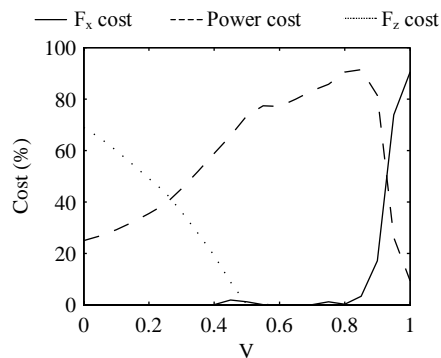


Figure 36: Cost fractions for varying velocity solution of base-kinematic model

While an in depth analysis of power expenditure is not required, the results do highlight some interesting points that should be noted. At low velocity the dominant cost is the force F_z , as the bird struggles to provide enough vertical force to meet the requirements; this process also requires significant power consumption. As the

velocity increases the relative cost of F_z is soon outweighed by the increased power cost that rises to around 90% of the total at $V=0.8$.

Further increase in velocity yields a drop in the cost of power due to the sudden rise in cost for force F_x : this can be explained by the result noted in Preliminary Analysis (3.4) whereby the production of horizontal force becomes increasingly difficult towards the limiting case of $V=w_f$. The sudden variation in costs at higher velocities identifies the boundaries of a flight regime, and strengthens the argument posed in the previous section with regards to the maximum (feasible) velocity value that should be used.

UNDERRELAXING

Having obtained a more stable solution, it is now useful to consider additional methods of smoothing out the results for the purpose of computer animation. Solving the problem over a smaller number of design points did show some degree of success; however it is felt that a more mathematical approach would be more reliable.

Considering again the notion that solution should be constrained into flapping regimes, a standard numerical method can be applied to limit the rate at which a solution changes (with varying velocity). Underrelaxation is a process that restricts the amount by which an iterative solution changes at each step. It requires the prescription of an underrelaxation parameter, δ , that limits that change in solution. A solution for given velocity, V_n , is obtained by summing fractions of the ‘new’ solution (from optimisation) and the previous solution (at V_{n-1}); with the fractions defined by δ and $(1-\delta)$ respectively.

In the context of this project, the formulation of this method requires an additional equation (93) to be solved at the end of solution for the given velocity value:

$$\begin{pmatrix} \Phi_f \\ \Theta_f \\ \vdots \end{pmatrix}_{V_n} = \delta \cdot \begin{pmatrix} \Phi_f \\ \Theta_f \\ \vdots \end{pmatrix}_{V_n}^* + (1 - \delta) \begin{pmatrix} \Phi_f \\ \Theta_f \\ \vdots \end{pmatrix}_{V_{n-1}} . \quad 94$$

By selecting an underrelaxation value of zero, no corrections are made and the solution remains constant, whereas an underrelaxation value of unity returns to the previous model. The results in Figure 37 show how the solution for Φ_f changes using three different underrelaxation values:

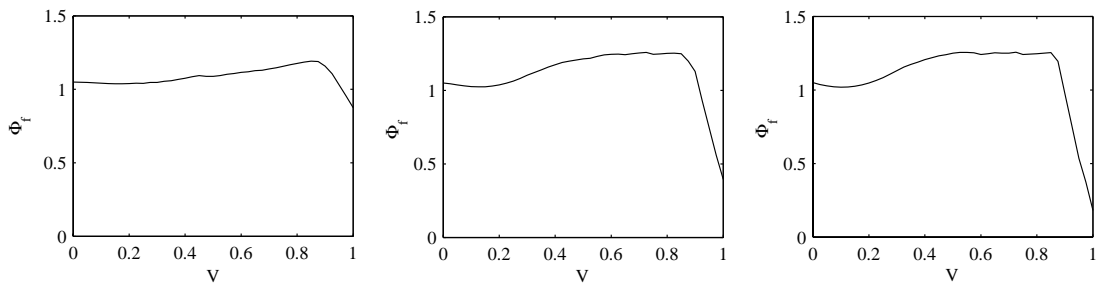


Figure 37: Base-model solution for Φ_f using underrelaxation values, $\delta = 0.1$ (left), 0.4 (centre) and 0.9 (right)

The results show the smoothing effect that this method has on the solution. It is important to note that both spurious data (from optimisation errors) and also genuine solution variations are both restricted equally.

This type of method may be useful if included in the final program to be developed; the underrelaxation could be linked directly to the degree of instability detected in a solution set.

Another important factor relates to the order in which the solutions are obtained. While the underrelaxation process has been developed for iterative processes, this system is not strictly iterative, as the previous, current and future results are all required to populate the results for varying V . The results in Figure 37 have been obtained by increasing the velocity from 0 to 1, however if the solutions were obtained in the reverse order the underrelaxation would have a different affect. This feature is not only important when using underrelaxation, but also when considering how the end set of results is to be populated. For this reason, the current analysis will avoid incorporating underrelaxation, while stable solutions are still attainable.

DIRECTIONAL PREFERENCE ON THE DESIGN SPACE

Successful results were obtained by updating the initial guess value for the optimisation as the velocity value increased; however a starting value for the first velocity being used must still be specified explicitly. It is important to assess the sensitivity of the results to the prescribed starting value, to determine whether it can be arbitrarily chosen, or whether a more technical approach is necessary.

Results shown in Figure 35 were produced by setting the starting value at $V=0$ to the parameter maximum values. The reason for this choice was that at zero velocity, previous results have suggested that the hovering regime is expected: in this motion the bird is likely to have maximum body pitch and twist amplitudes, with large flapping amplitudes. The initial guess is therefore linked to the physical characteristics of the problem, which is accepted as a valid approach.

Similarly, successful results can also be obtained by solving the problem in the reverse order (a backwards pass): starting at high velocities with low initial guess values to capture the characteristics of cruising flight. The results are shown in Figure 38.

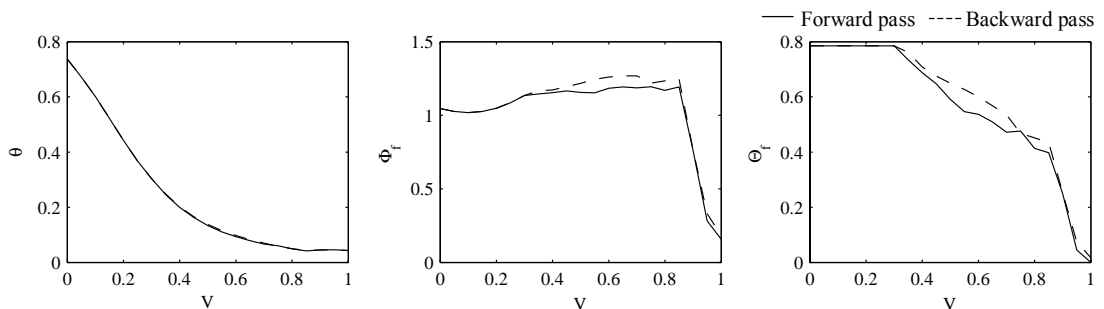


Figure 38: Successful base-model solution for forward and backward passes

The results show that at high and low velocity values there is a strong match between the forward and backward passing approaches. In the intermediate stages the more complicated solution behaviour causes a slight difference in the values of flapping and twisting amplitude. However, it is believed that results from either of the approaches would be acceptable in terms of physical accuracy.

Linking the initial guess value to physical characteristics of the problem appears to be a successful technique. Considering the converse case, Appendix 4 shows examples of choosing inappropriate starting values such as maximum parameter values for cruising regimes; this results in invalid results.

The solution procedure can also be tested over variations of the other input parameters, namely the forces F_x and F_z . By isolating the individual input parameters as independent variables, and considering the expected physical behaviour to prescribe a suitable starting value, similar results to those seen in Figure 35 can be obtained for varying forces. Appendix 5 gives details of trial solutions for varying forces F_x and F_z .

The results show that by performing a backwards pass for force F_x (i.e. $F_x=0.5 \rightarrow 0$) a stable, valid solution is attainable when applying the maximum initial guess parameters. A similar result is seen when performing a backwards pass for F_z ($F_z=-0.5 \rightarrow 0$). Solutions can now be obtained for all three independent input variables that govern the trajectory of the bird: F_x , F_z and V .

POPULATING LOOKUP TABLES

It has been shown that strong coupling exists between the order in which the input variables are incremented and the initial guess value for the optimisation. To obtain solutions over the entire range of forces and velocities, and hence fully populate the lookup tables for the motion parameters, the diagram in Figure 39 portrays the route that should be taken:

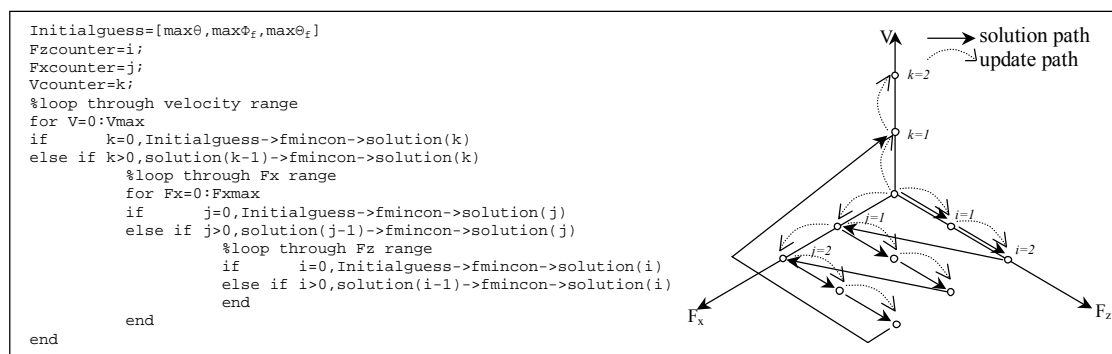


Figure 39: Pseudo code and depiction of the process of populating the solution space

It has been found that valid solutions can be obtained by sweeping through the input parameters in the order of F_z , F_x and V respectively; the initial guess values are updated accordingly. This method enables a solution to be obtained over the entire space of input variables and therefore populate a lookup table containing optimal values for all flapping parameters over the desired range of F_x , F_z and V . This is done by simply solving over the extremes of motion and described in Base-Model Design (3.3).

Background Information (2.) described the use of lookup tables as a suitable method of enhancing performance. Using this method, simulations produced in real-time would simply pick results from the lookup tables according to the current requirements; the actual costing of flapping motions would still have to be performed offline to populate the lookup tables initially.

POPULATED RESULTS

A populated solution space can now be obtained for the base kinematic model. The results shown in Figure 40 show the populated data for the parameters θ , Φ_f and Θ_f over a range of forces F_x and F_z , and for three velocity values.

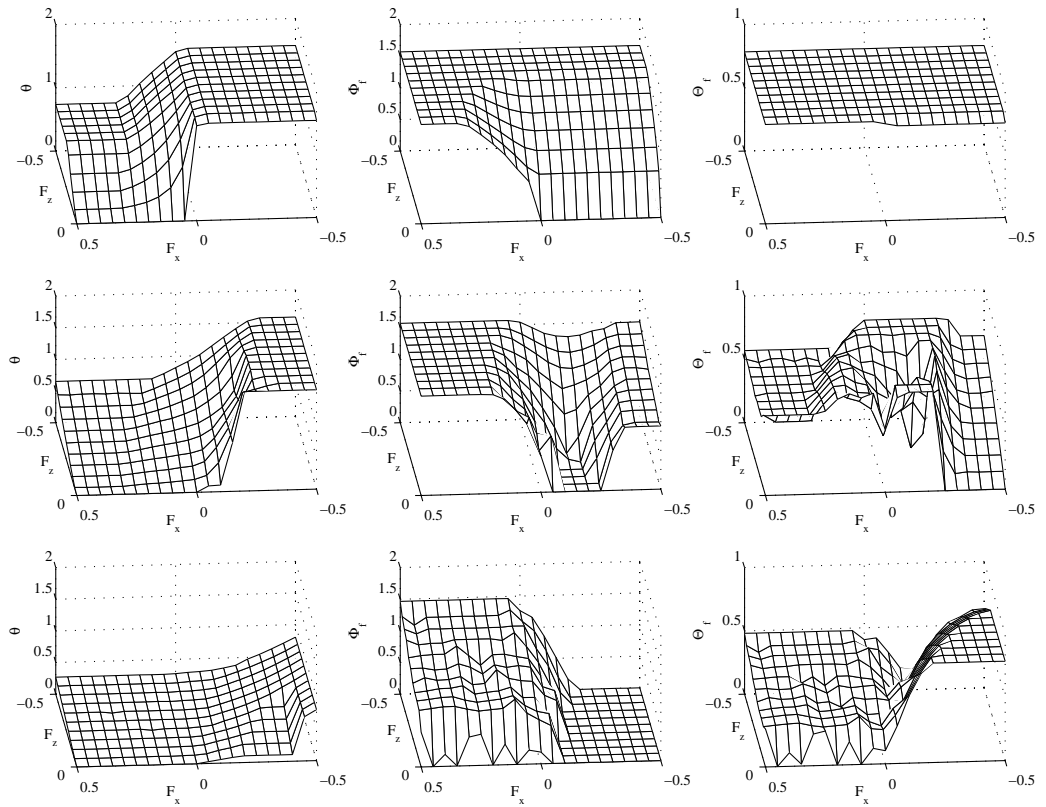


Figure 40: Populated lookup tables for base-model solution, $V=0$ (top row), $V=0.5$ (middle row) and $V=0.9$ (bottom row)

Some of the data shown in Figure 40 can be assessed simply by observation, and compared to what is commonly seen in nature. For example, considering the body pitch attitude, θ , the overall solution surface is seen to decrease with increasing velocity which is clearly what happens as birds accelerate. Also, the wing flap amplitude is seen to increase for increasing (negative) force F_z , therefore highlighting the effects seen in climbing, take-off and also steep landings.

An important point to consider about the results is the variation in parameters, specifically the twist amplitude, during the intermediate flights stages (given here when $V=0.5$). The resulting lookup tables show some degree of instability for low force values. It appears that with $V=0.5$, the global minimum is not well defined, perhaps due to flattening of the optimisation space as demonstrated previously. This leads to less stable results, even with updated the initial guess values. Physically this relates back to the discussion on flight regimes and the overlapping gaits of motion. It is clear that this phenomenon is most apparent in the wing twist behaviour.

It is important to appreciate that while the solution has been obtained for the full range of values, some combinations of forces and velocities are physically unrealistic. For example as the bird approaches its maximum velocity, it is clear that any additional F_x force it can produce will be negligible, as described in Preliminary Analysis (3.4). So while results are given over the entire input parameter range, some of the data will be redundant.

In order to analyse the results further, it is appropriate to examine a selection of flight regimes: the typical forces and velocity values can be predicted, and hence the resulting parameter values can be determined from Figure 40; the reader should refer back to Figure 5, The Gaits of Motion (3.2.5), for clarification. The three regimes to be considered are hover, take-off, and cruise:

HOVER ($V \sim 0$, $F_x \sim 0$, $F_z \sim F_{zmax}$)²¹

The case of hovering has been described as the most clearly defined flapping regime (Ellington, 1984). It has been shown that the objective of attaining the desired force F_z with no applied freestream velocity vastly outweighs the process of limiting power expenditure (Figure 36, Cost Fractions). Also, in this case, no F_x force is present, and hence the optimisation process is much more tightly bound.

To give a more physical understanding of the nature of the flapping behaviour in hovering, Figure 41 shows the path of the control point that is predicted from the input parameters $V=0$, $F_x=0$, $F_z=0.5$. The choice of maximum force F_z is made to exaggerate the requirements on hovering; while this force may be greater than the bird weight, the strong process of downwash would require additional force production. The wing chord line passing through the control point has also been added to Figure 41 for completeness.

²¹ The reader should refer back to The Gaits of Motion (3.2.5) for clarification of the suggested force and velocity values for different flight regimes.

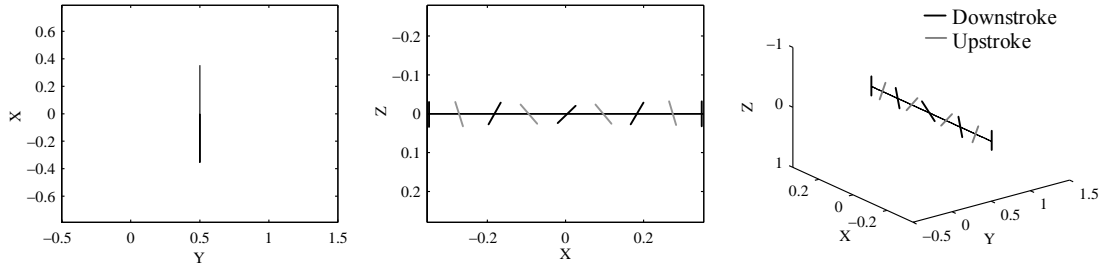


Figure 41: Control point paths and chord lines of base-model right wing during hovering ($\theta=1.5$, $\Phi_j=0.79$, $\Theta_j=1.5$): plan view (left), side view (centre) and perspective view (right)

The solution achieves the maximum upward force by moving the wings forwards and backwards with maximum flap, and twist amplitude. This motion is achieved by pitching the body vertically to incline the path of the wing accordingly. The result gives the maximum available lift coefficient and hence maximum lift force, as described in the Flapping Kinematics (3.5).

It should be noted (from Figure 40) that a slight increase in the force F_x , would cause a substantial reduction in body pitch angle. A decrease in F_x (to give negative force in the X -direction) appears to have no effect; this will be discussed further in the case of landing.

TAKE-OFF ($V \sim 0$, $F_x \sim F_{xmax}$, $F_z \sim F_{zmax}$)

The case of take-off represents the strongest acceleration affects of the bird. The slight velocity produced by the bird ‘jumping’ is neglected, and large vertical and horizontal forces requirements are used. Landing is similar, in that immediately before touching down the velocity is negligible, however large vertical and horizontal forces are required to decelerate the bird.

Figure 42 shows the path of the control point during take-off that is predicted from the input parameters $V=0$, $F_x=0.5$, $F_z=-0.5$:

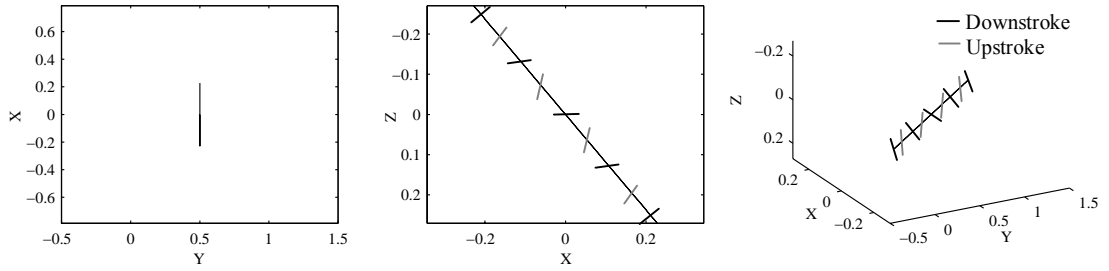


Figure 42: Control point paths and chord lines of base-model right wing during take-off ($\theta=0.7$, $\Phi_f=0.1.5$, $\Theta_f=1.5$): plan (left), side view (centre) and perspective view (right)

In a similar sense to hovering, take-off aims to find a solution that gives the maximum force available. For this reason, the flap and twist amplitudes both reach maximum values. However, the force from flapping must now be rotated to provide a force in the X -direction, which is done using the body pitch angle. Again the results comply with the type of behaviour observed in nature.

The results for the landing obtained from the lookup tables in Figure 40 show that the three parameters, ($\theta=0.7$, $\Theta_f=1.5$, $\Phi_f=0.1.5$), all share the same values as those used for hovering. While it would be desirable for the body pitch angle to extend further than $\pi/2$ rads when landing, this is clearly not realistic and is prevented by the imposed maximum pitch angle. It appears that the regime of landing is not distinguishable from that of hovering. While these two cases certainly show some degree of similarity in reality, the base-model does not capture any difference at all. Results may be separable when considering the more advanced kinematic models.

CRUISE ($V \sim V_{max}$, $F_x \sim 0$, $F_z \sim \text{weight}$)

Cruising represents the other end of the spectrum of flapping regimes. It has been shown by considering the cost fractions, that in cruise, the affect of limiting power expenditure becomes apparent. While the wing flap and twist amplitude relationship is now more complicated, the power constraint should help in generating the optimum result. Figure 43 shows the path of the control point during flapping that is predicted from the input parameters $V=0.9$, $F_x=0$, $F_z=-0.2$; the small ‘token’ F_z force is chosen to represent the vertical force needed to balance the bird weight. The value

of $V=0.9$ is taken from previous analysis that suggested this to be a suitable limiting boundary to the cruise regime.

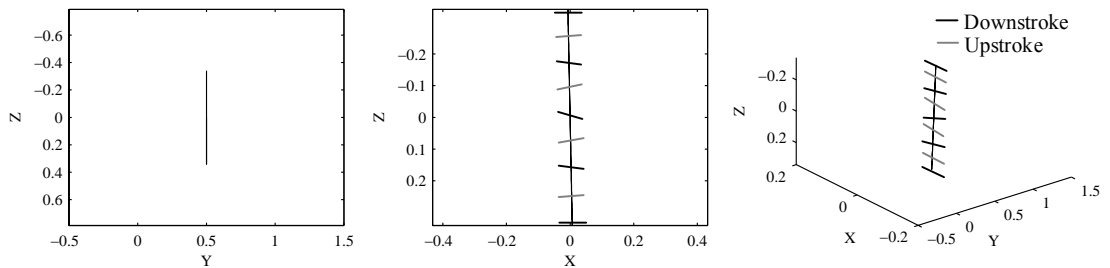


Figure 43: Control point paths and chord lines of base-model right wing during cruise ($\theta=0.02$, $\Phi_f=0.75$, $\Theta_f=0.25$): rear view (left), side view (centre) and perspective view (right)

The wing flapping amplitude is approximately half that used in hovering, landing and take-off. A small twist amplitude is still seen to be beneficial to the system: as shown in the Preliminary Analysis (3.4), this aids to increase the force production with less power expenditure than increasing the flap amplitude. The body pitch amplitude is negligible in this case, as the high velocity produces forces large enough to require only a small vertical component to satisfy the F_z requirement.

Referring to Figure 40, climbing at this high velocity can be achieved by increasing the force F_z , which in turn leads to an increase in body pitch, and flap and twist amplitudes. Alternatively a method of breaking can be achieved whereby a large negative F_x requirement (with zero lift requirement) yields an increased body pitch, and increased wing twist amplitude (with zero flapping). This behaviour is similar to that seen as a bird approaches a perched landing.

Having developed a suitable method for producing populated lookup tables for the system, focus can now be moved to the more advanced kinematic models.

4.1.2 KINEMATIC MODEL 1 SOLUTION

The second set of data to be populated incorporates the 2d flapping motion of the wing from kinematic model 1, developed specially for this project. This section will give details of the solution procedure for the new kinematic model. The method of updating the initial guess value when varying the input parameters will again be used.

While the advances in the kinematic model are important in improving the physics of the system, it is also important to consider where these advances will actually produce a noticeable change in results. For the case of transferring from the base-model to model 1, only a single additional rotation has been added, the effects of which were discussed at length in Kinematic Model 1 (3.5.3). Therefore rather than repeating results from the previous solution, this section will aim to only consider new information brought about by the solution of model 1.

In a similar manner to the base-model, a useful initial step is to consider the solution behaviour with varying velocity only; this is done with a small constant value of F_x and F_z . Figure 44 shows the results for the optimisation of the model 1, the data from the base-model has been included for reference:

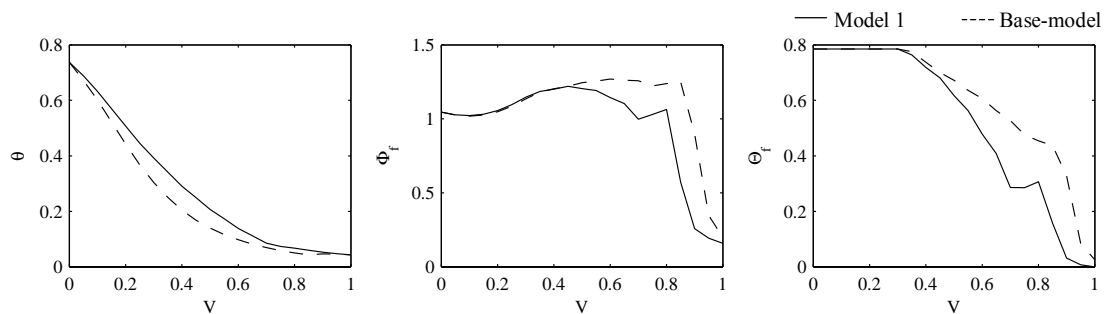


Figure 44: Comparison of base kinematic model and model 1 solutions for varying V

At high and low velocity values, the results from both models converge to similar values. However, in the intermediate velocity range there are noticeable differences.

In the hovering regime the base-model solution required maximum values of all parameters to achieve the required force F_z . It was shown in Kinematic Model 1 (3.5.3) that the equivalent motion with model 1 is less effective in producing force, due to a Y -component of force being generated. Therefore it is not surprising that this model also requires maximum values when hovering, as it attempts to generate the required force F_z .

At the intermediate velocity stages model 1 would again be expected to *work harder* than the base-model (i.e. use greater flap and/or twist amplitudes) in order to produce the required forces. However the results show that the base-model in fact uses slightly larger flapping and twisting amplitudes. This detail is believed to be another consequence of the flattening of the optimisation space in the intermediate velocity range, rather than a true physical phenomenon; this may also explain the sudden ‘jump’ in the value of flap and twist amplitude at $V=0.8$.

In order to properly assess the physical influences, the exact forces produced by the kinematic model (and not just the desired force inputs) would need to be considered more closely; this is not strictly necessary at this stage in the analysis. Perhaps more important to note is that both solutions show the general expected variation in the flapping parameters, and would both therefore fit the project requirements. It is also important to note that the slight difference between the two data sets would not have any detrimental affect on the results in terms of the graphical accuracy.

Rather than reproduce similar control point paths to those seen in Figure 41, Figure 42 and Figure 43, it is more useful to begin assessing the other kinematic models that are likely to reveal new information. For the fully populated lookup table for kinematic model 1, the reader should refer to Appendix 6.

4.1.3 KINEMATIC MODEL 3 SOLUTION

This section will give details of the solution procedure for kinematic model 3, which includes the sweep plane angle θ_{sp} . A similar solution method to the previous cases will be employed, with any new information described accordingly. The influence that the new solution parameter (θ_{sp}) has on the methodology and corresponding results will be analysed. Having obtained the full solution and populated the lookup tables, some example results will be given.

Considering again the solution variation with velocity, the procedure can be invoked as done previously: a forward pass starting at the hovering regime ($V=0$) with maximum initial guesses for all parameters. However, it is important to consider the physical influence of the sweep plane angle, and how it affects current model parameters.

It was shown in Kinematic Model 3 (3.5.5) that the body pitch attitude and the sweep plane angle can both be used to rotate the force vector from horizontal to vertical directions. Mathematically, this would suggest that multiple solutions would exist, which would inhibit the optimisation. It has been shown by Ellington (1984) that the inclination of the sweep plane is the preferred method of thrust vectoring when hovering, and this will be assumed in this solution. To achieve this, a small token cost, the *orientation-cost*, is added to rank the use of sweep plane inclination over the use of body pitch when the velocity is equal to zero. The results are shown in Figure 45:

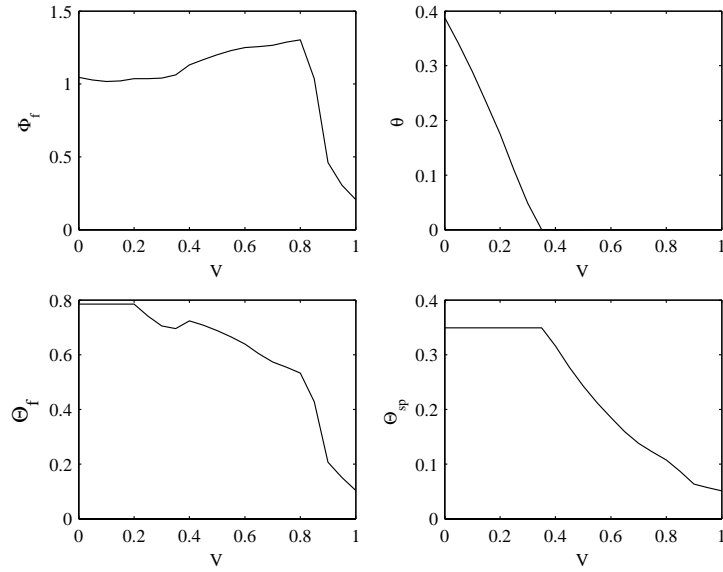


Figure 45: Kinematic model 3 solutions for varying V

At low velocities the solution now utilises the maximum sweep-plane angle, combined with a body pitch angle, to produce the required vertical force. It should be noted that the sum of the sweep plane angle and body pitch angle at $V=0$ is equal to the body pitch angle seen in the previous models, as expected.

With increasing velocity the solution shows preference to a reduction in body pitch angle, rather than a reduction in sweep plane angle, even though the *orientation-cost* is no longer present. This is due to the increased aerodynamic drag associated with the pitch angle of the body relative to the freestream flow, which makes body pitch less favourable.

The variations of flap and twist angles are again very similar to those seen in the previous two kinematic models (see Figure 44) with slight discrepancies again occurring for the intermediate velocity values.

To further examine the influence of the new parameter, it is useful to consider some example solutions from particular flight regimes. The three regimes to be considered are hovering, take-off and cruise. For the fully populated lookup tables the reader should refer to Appendix 7.

HOVER ($V \sim 0, F_x \sim 0, F_z \sim F_{zmax}$)

Figure 46 shows the path of the control point during hovering that is predicted from the input parameters $V=0, F_x=0, F_z=-0.5$.

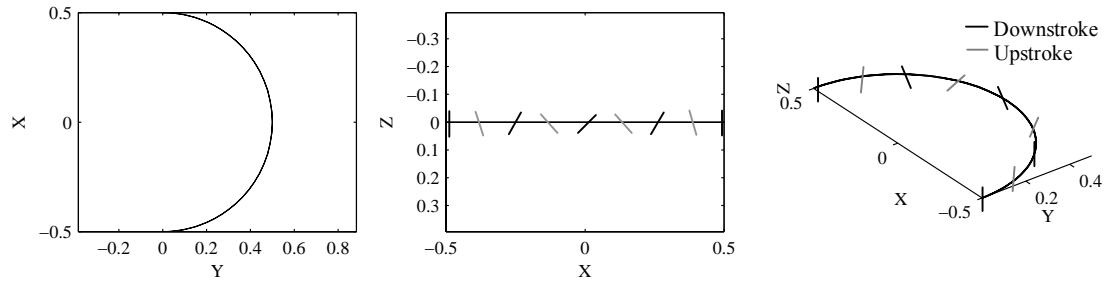


Figure 46: Control point paths and chord lines of model 3 during hover ($\theta=1.2, \Phi_f=0.75, \Theta_f=0.79, \Theta_{sp}=0.35$): plan view (left), side view (centre) and perspective view (right)

The results can be compared to those seen from the base-model in Figure 41. The optimisation procedure still achieves the maximum vertical force by selecting a horizontal flapping motion; this is done by using the maximum sweep-plane angle, and an additional amount of body pitch. The solution again selects maximum flap and twist amplitude for the motion.

The force and power expenditure will be equal to those for kinematic model 1; the body drag having no influence in hovering, with zero freestream velocity.

TAKEOFF ($V \sim 0, F_x \sim F_{xmax}, F_z \sim F_{zmax}$)

Figure 47 shows the path of the control point during take-off that is predicted from the input parameters $V=0, F_x=0.5, F_z=-0.5$:

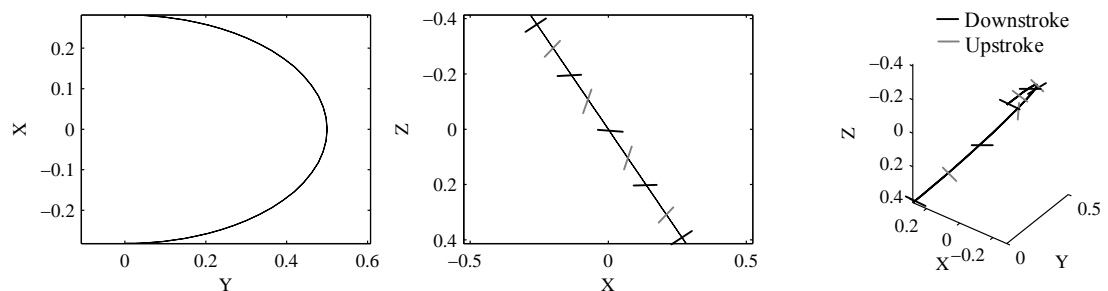


Figure 47: Control point paths during takeoff ($\theta=0.25, \Phi_f=1.5, \Theta_f=0.79, \Theta_{sp}=0.35$)

Comparing the results to those from the base-model in Figure 42, some important differences can be observed. While the flap and twist amplitudes still remain at the maximum values to obtain the maximum force, the sweep plane is now slightly less steeply inclined. The sum of the sweep plane angle and body pitch angles yields a net angle of 0.6 rads to the horizontal, in comparison the value of 0.7 rads for the base kinematic model. This would suggest that the 2d flapping wing is producing a greater thrust force than the 1d case, which is contrary to what was shown in Flapping Kinematics (3.5). This result must therefore be a factor of the numerical optimisation, which appears to be too heavily influenced by the additional orientation cost.

CRUISE ($V \sim V_{max}$, $F_x \sim 0$, $F_z \sim \text{weight}$)

Finally, considering the case of cruising, Figure 48 shows the path of the control point that is predicted from the input parameters $V=0.9$, $F_x=0$, $F_z=-0.25$:

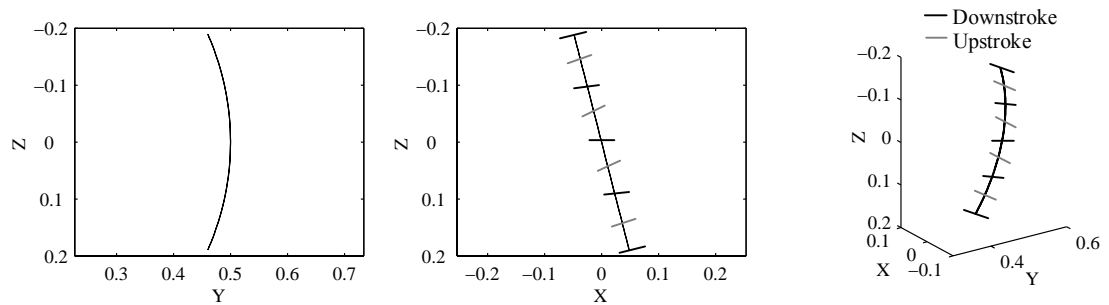


Figure 48: Control point paths of model 3 during cruise ($\theta=0.0$, $\Phi_f=0.4$, $\Theta_f=0.16$, $\Theta_{sp}=0.25$): rear view (left), side view (centre) and perspective view (right)

The cruise results demonstrate the simplicity in which the optimisation process is capable of manipulating the system parameters to gain desirable results. It is now seen that no body pitch angle is used, thus reducing the drag on the body. Instead, the sweep plane angle is used as the sole method for vectoring the thrust to fulfil the lift requirement.

A substantial value of θ_{sp} is used in comparison to the body pitch seen in the base-model, Figure 43. By doing this, the system now manages to resolve a larger component of the forward thrust into the vertical direction. The results of this is that

smaller flapping and twisting amplitudes can be used, which causes a dramatic reduction in power expenditure; this correlates directly to the behaviour predicted by considering the regimes of flight when assessing the cost fractions. This behaviour is also desired by the base-model, however the large drag penalty acting on the body during such pitch motions is too costly, and hence the larger flapping amplitude is chosen (with the an increased power penalty).

4.1.4 KINEMATIC MODEL 4 SOLUTION

This section will give details of the solution procedure for kinematic model 4, which includes an additional wing panel to represent the bird hand. As with the base-model, a detailed description of the solution procedure will be given to highlight the key factors involved in obtaining stable, accurate results.

While models 1 and 3 were relatively straightforward extensions of the base-model, the 2-panel model shows a much greater degree of complexity. The governing equations given in Kinematic Model 4 (3.5.6) explain how this model requires 3 additional free parameters to define the motion; namely the wrist flapping amplitude, phase lag and bias.²²

Not only do the new parameters increase difficulty in obtaining solutions, but also in the analysis of results. Visualising optimisation spaces, such as those shown in Figure 33, is no longer feasible as the cost is now a function of 7 parameters. Rather than attempting to optimise this solution with little knowledge of the model physics, a more strategic approach will be taken.

GROUPING PARAMETERS

A common scientific technique is to derive non-dimensional parameters that characterise physical behaviour by grouping the various contributing factors into a single term. An example of this has already been shown with the use of the advance

²² The wrist twist amplitude is not a strictly free parameter due to its algebraic prescription as a fraction of the shoulder twist

ratio. If a similar approach can be used to define the motion of the bird hand, then the three ‘new’ parameters can be grouped into a single value.

The analysis of the parameter range for kinematic model 4 (3.5.7) showed how the maximum values of wrist flapping amplitude, lag and bias can be derived mathematically, with limited physiological considerations. It was shown that the independent motion of the wrist is crucial in creating a net upward force in cruising. An individual parameter defining the motion of the wrist could be used to algebraically prescribe the values of the amplitude, lag and bias as fractions of their respective maximum values. The *wrist activity coefficient*, K , will be used to define the motion of the wrist using the following relationship

$$\begin{pmatrix} \Phi_w \\ \Delta \\ \Psi \end{pmatrix} = K \cdot \begin{pmatrix} \Phi_{w(\max)} \\ \Delta_{(\max)} \\ \Psi_{(\max)} \end{pmatrix}. \quad 95$$

To clarify this, an example of the effect of using the wrist activity parameter on the force F_z is shown in Figure 49. These results aim to demonstrate the influence of the parameter K on the basic kinematic system; this is achieved by considering only a limited number of model parameters.

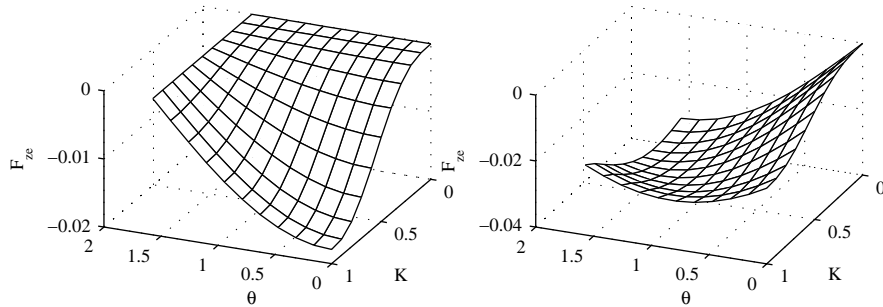


Figure 49: Variation in force F_z with wrist activity parameter and body pitch attitude, with $V=0$, $\Phi_f=\pi/4$, $\theta_{sp}=0$ and $\Theta_f=0$ (left), $\Theta_f=\pi/16$ (right)

In the case where no wing twist is employed, the fully pitched body ($\theta=\pi/2$ rads) yields zero force for all values of K , as expected. With reduced body pitch, the wrist activity now enables the force F_z to be increased (negatively): for $\theta<\pi/2$ rads, the maximum value of K can be used to obtain a greater force.

Considering the (more realistic) case with a wing twist of $\pi/16$, the force from the fully pitched body is now sensitive to the choice of K : greatest force is achieved when using no wrist motion. As the body pitch reduces, it becomes beneficial to use a greater value of K . These results simply confirm that the prescription of the wrist activity parameter correlates to the expected physical behaviour of the kinematic system; the cruise regime is thus likely to benefit most from use of the wrist.

A solution to the problem can now be obtained as done with the previous kinematic model; with varying velocity only. The focus will now be on achieving accurate results for the values of K . Figure 50 shows the optimisation spaces for three example velocities, whereby small F_x and F_z requirements have been specified:

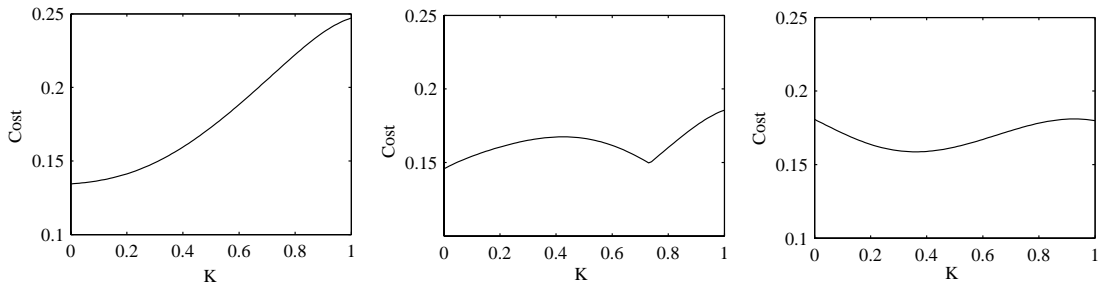


Figure 50: Optimisation spaces for kinematic model 4, with $V=0$ (left), $V=0.5$ (centre), 1 (right)

With zero freestream velocity, the optimiser will select $K=0$ as a solution, as expected. However at higher velocities a more irregular optimisation space can be seen. From a physical perspective, at $V=0.5$ it is unclear which of the two local minima represents the more valid solution. At $V=1$, the optimum value of K is selected as approximately 0.37, which yields much smaller values of Φ_w , Δ and Ψ than expected from previous analysis (3.5.7). It is believed that the optimisation spaces in Figure 50 are not a strong representation of the true physical behaviour of the system; other factors are believed to be interfering with the data.

One possible problem that has been cited relates to the size of the second panel. It has been shown previously (3.5.6) that the outer section of the wing has a strong influence in the force and power expenditure in the system. The high velocity of the outer panel yields large forces that can easily dominate those found on the inner panel, without substantial power expenditure. These initial solutions have been

obtained with an arbitrarily sized outer panel, which may be yielding forces and powers that are unrealistic. With the simple methods of deducing force and power results taken in this project, it is crucial to identify such factors that may inhibit the model and lead to spurious results.

Further attempts have been made, with reduced chord and span sizes for the outer panel. Example optimisation spaces, using half the original size of panel 2, are shown in Figure 51:

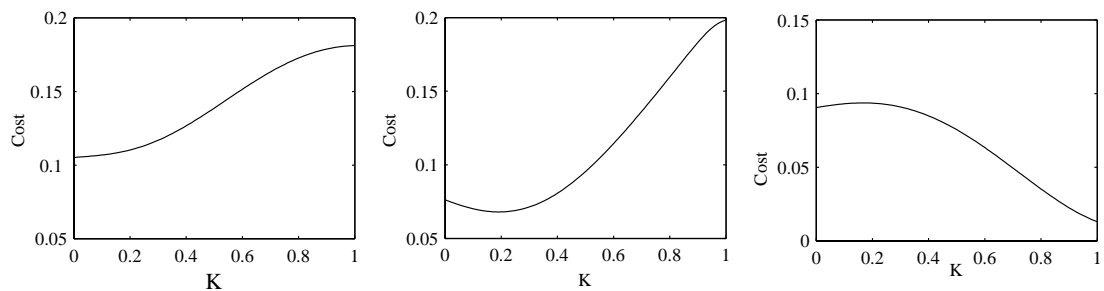


Figure 51: Optimisation spaces for kinematic model 4, with $V=0$ (left), $V=0.5$ (centre), 1 (right), using reduced panel 2 sizing ($s_2=0.5s_1$)

While the overall costs now appear lower, the variation in cost with K is now much more promising. At low velocities, the model favours $K=0$, however as the velocity increases the ‘valley’ in the optimisation space gradually moves to favour higher values of K . This occurs until $V=1$, where the optimum value of $K=1$ is chosen, as shown in Figure 51.

The data appears feasible with regards to the physical understanding of the use of the outer panel. More importantly, a smooth, continuous variation in results is now seen, which suggests that the problems noted in the initial solution were indeed a result of an unrealistic panel sizing. This also implies that the wrist activity parameter offers some degree of success in capturing the appropriate behaviour of the second panel. The results highlight what can be observed in nature, whereby the wrist motion is clearly beneficial at higher velocities, such as in the cruise regime.

POPULATING (FREE PARAMETER) RESULTS

The wrist activity parameter has been useful in confirming the use of the outer wing panel at different velocities, as well as highlighting the importance of correctly modelling the relative lengthscales of the bird physiology. Rather than continuing to use the grouped parameter K , the project will now revert back to the three separate defining parameters. The knowledge gained in the previous section has provided a firmer foundation for the analysis of the final kinematic model with all 7 free parameters.

While a similar strategy to the previous models can be employed when invoking the solution procedure, some consideration of the passing direction (of the solution space) must be made. It is now appropriate to perform a backward pass from $V=1$ to $V=0$, using minimum initial guess values for θ , Φ_f , θ_{sp} and Θ_f where they have been shown to be small²³. In this regime, the previous analysis confirmed that the values of Φ_w , Δ and Ψ are likely to be large; therefore maximum initial guess values can be applied.

A solution can now be obtained, and the results are shown in Figure 52; the sweep plane angle has been temporarily omitted to simplify the analysis. Also, as the true hovering behaviour is believed to be an important consideration in this model, no F_x requirement is used.

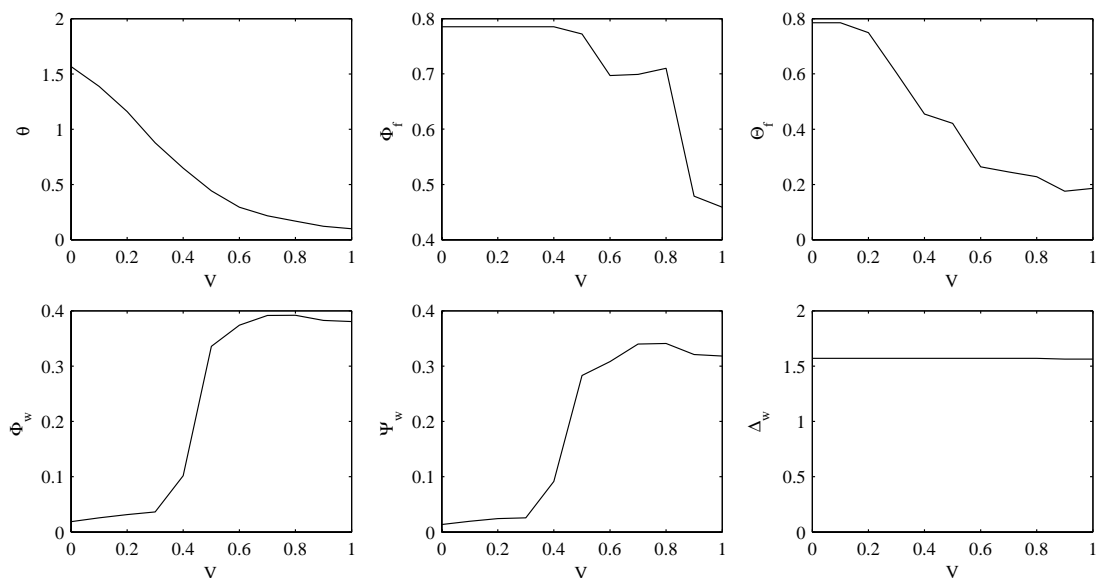


Figure 52: Kinematic model 4 solutions for varying V

²³ Forward passes were unsuccessful for this particular case

In order to retain the model realism, the size of the original panel has to be reduced to accommodate an additional wing section without increasing the net semispan. The result of this is that the smaller inner panel is itself not capable of producing the magnitudes of forces as done in previous models. Similarly, the maximum flapping amplitude is also reduced; $\Phi_{f(max)}=\pi/4$ rads.

Figure 52 shows that the model now utilises maximum body pitch, flapping and twisting amplitudes in the hovering regime: the smaller inner panel now has to ‘work harder’ to achieve the desired force F_z . Conversely, the amplitude and bias of the outer panel are small at low velocity values, confirming that the wrist motion is not beneficial in the hover flight regime, as shown in the previous section (Figure 51).

As the velocity increases θ , Φ_f and Θ_f all follow very similar paths to the results from the previous kinematic models. The more interesting result is the variation of the parameters governing the wrist motion. Both Φ_w and Ψ_w show a gradual increase in the range $V=0$ to $V=0.3$, followed by a much more rapid jump in the intermediate velocity range. The phase lag of the wrist remains constant at a value of $\pi/2$ rads. The results show that the wrist is crucial in the role of producing vertical force in intermediate and high velocity ranges. However, the results also suggest that the wrist motion is not significant in the hovering mode, even though it could be used to increase the net upward force as shown in section Kinematic Model 4 (3.5.6). It seems that the power expenditure in the wrist outweighs the force production benefits when in the hovering regime.

To further examine the results of the 2-panel model, it is useful to consider some example solutions from particular flight regimes. The three regimes to be considered are hovering, take-off, and cruise. For the fully populated lookup tables the reader should refer to Appendix 8.

HOVER ($V\sim 0$, $F_x\sim 0$, $F_z\sim F_{zmax}$)

Figure 53 shows the path of the control point during hovering that is predicted from the input parameters $V=0$, $F_x=0$, $F_z=-0.5$.

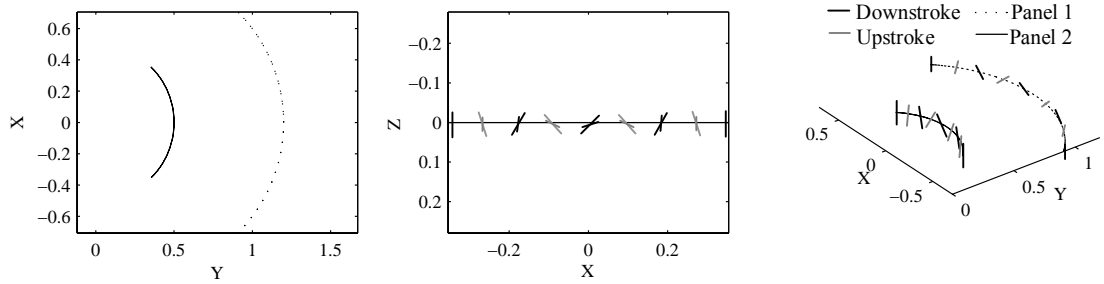


Figure 53: Control point paths during hover ($\theta=1.2$, $\Phi_f=0.79$, $\Theta_f=0.79$, $\Theta_{sp}=0.35$, $\Phi_w=0.01$, $\Psi_w=0.02$, $\Delta_w=1.55$) : plan view (left), side view (centre) and perspective view (right)

The 2-panel model utilises maximum flapping and twisting amplitudes of the inner panel as seen with previous cases. The optimised solution yields no wrist movement, so the motion of control point 2 is simply that induced by the inner panel. The method of prescribing the hand twist angle may be responsible for the lack of wrist movement in hovering. Due to the order in which the rotations are applied, an additional flapping motion of the wrist will yield a vertical motion of the hand: this may inhibit the magnitude of force F_z by increasing the local incidence angle to value greater than $\pi/4$ radians (thus reducing the lift coefficient). This aspect of the kinematic model would require further consideration in order to perform a more rigorous optimisation; which is not required at this stage.

TAKEOFF ($V \sim 0$, $F_x \sim F_{xmax}$, $F_z \sim F_{zmax}$)

Figure 54 shows the path of the control points of both panels during takeoff that is predicted from the input parameters $V=0$, $F_x=0.5$, $F_z=-0.5$:

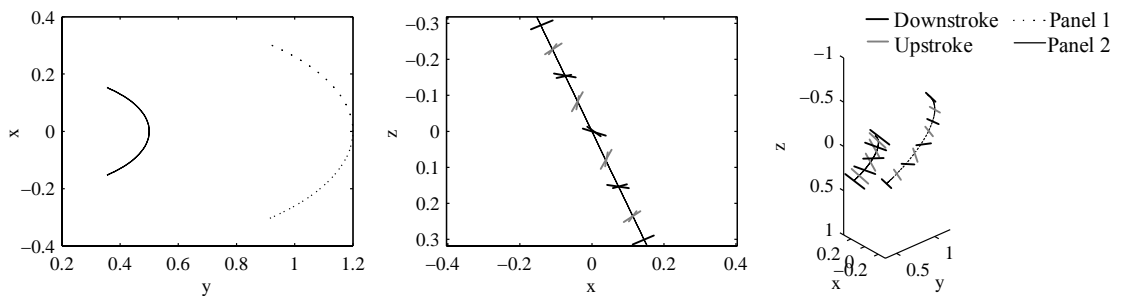


Figure 54: Control point paths during takeoff ($\theta=0.1$, $\Phi_f=0.79$, $\Theta_f=0.79$, $\Theta_{sp}=0.35$, $\Phi_w=0.01$, $\Psi=0.01$, $\Delta_w=1.55$) : plan view (left), side view (centre) and perspective view (right)

The results for takeoff show a smaller sweep plane angle than kinematic model 3 (Figure 47), with maximum flapping and twisting amplitude of the inner panel. It seems that the additional twist of the outer panel aids in producing the force F_z , therefore the force vector requires less rotation towards to vertical direction, and hence the smaller sweep plane angle. Again the wrist shows negligible movement, with all motion governed by the shoulder.

CRUISE ($V \sim V_{max}$, $F_x \sim 0$, $F_z \sim \text{weight}$),

Finally, considering the case of cruising, Figure 55 shows the path of the control point that is predicted from the input parameters $V=0.9$, $F_x=0$, $F_z=0.25$:

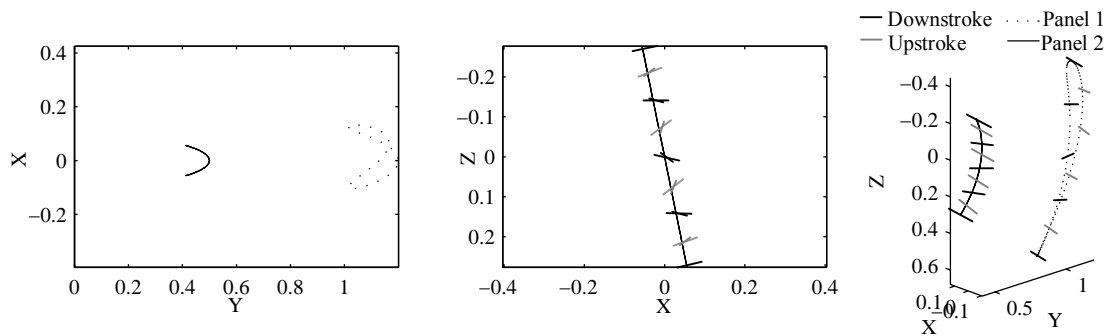


Figure 55: Control point paths during cruise ($\theta=0$, $\Phi_f=0.62$, $\Theta_f=0.41$, $\Theta_{sp}=0.21$, $\Phi_w=0.30$, $\Psi=0.29$, $\Delta_w=1.55$): plan view (left), side view (centre) and perspective view (right)

In cruise, the sweep plane is now inclined at a similar angle to the model 3 solution, however the flapping and twisting angles of the inner panel are substantially greater. The magnitude of twist amplitude is believed to be over predicted, which may be a factor of the optimisation process rather than a problem in the kinematic model. The sensitivity of the results to the prescribed panel sizes should still be considered.

The movement of the wrist is now a focal point of the control point paths. The combination of the derived amplitude, lag and bias yield an elliptical path of the second control point. This important result can be compared directly to what is found in nature; Ellington (1984) described such patterns in insect flight, and Rayner & Gordon (1998) observed similar results for pigeons.

While some problems have been cited with the populated results for this model, there has certainly been some degree of success, particularly with the cruise case. It is believed that while the populated results are offering a strong insight into the physics of the problem, further testing is required to fully eradicate any errors in the solution procedure. The results for varying V only, were much closer to what is expected from the model than the populated lookup tables, and these data are therefore assumed to be more reliable.

4.2 VISUALISATION AND PERFORMANCE

This chapter will be used to present the results in the context of the overall theme of the research program: computer animation. Visualisations of the flapping motion and complete system behaviour will be shown in order to provide an insight into the desired end product of this program²⁴. The differences between the kinematic models will be expressed both in terms of the final visual result, and also in terms of performance.

In order to observe the evolution of the model through the course of this project, visual data for the base-model and also the final 2-panel model will be shown. The solution times to populate lookup tables for four kinematic models will be compared. Discussions made will focus purely on the visual data, and avoid repetition of the more thorough analysis in the previous chapter.

4.2.1 ANIMATIONS

Figure 56 shows the visual result for the base-kinematic model during hovering. These results represent the effective starting point for the project in terms of the model sophistication. Having determined a suitable method of populating the lookup tables (4.1.1) animations of the form shown in Figure 56 can be produced. It should be noted that the slight body rotation and translation exist due to a prescribed reaction force from the flapping wing; this is purely an aesthetic factor.

Figure 57 shows the equivalent hovering motion for the 2-panel kinematic model, which represents the most sophisticated model in this project. The inclusion of the sweep plane and motion of the second panel are clearly visible. The shoulder joint has been depicted as a square to allow the sweep plane angle to be identified relative to the body.

²⁴ Visualisations are produced by applying the SIMULINK model shown in Appendix 1b, using VRML

The 2-panel kinematic model gives a more realistic interpretation of hovering flight, and the resulting simulations are comparable to the hovering behaviour of hummingbirds, as described by Ellington (1984).

A more striking comparison can be found between Figure 58 and Figure 59, which contain the corresponding cruise behaviour for the base and 2-panel kinematic models. Here, the addition of the sweep plane and the second wing panel has an even greater influence on improving the realism of the animations. The resulting behaviour shown in Figure 59 strongly resembles the cruising flight of a medium-sized²⁵ bird, such as thrush or blackbird.

²⁵ Birds are generally grouped into 7 size categories according to height; birds of height approximately equal to 30cm can be defined as "medium sized" (Baumel, 1993).

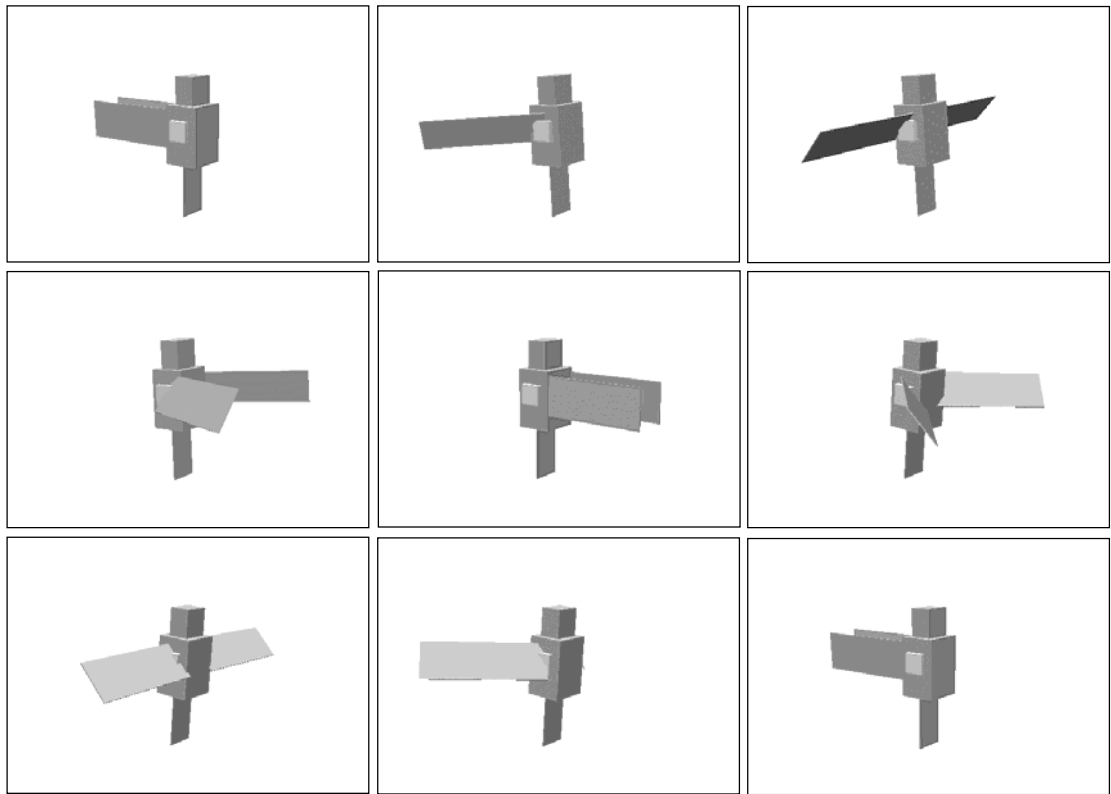


Figure 56: Hovering animation sequence for base kinematic model. From top left $t=0$, $T/8$, $T/4$, $3T/8$, $T/2$, $5T/8$, $3T/4$, $7T/8$, T .

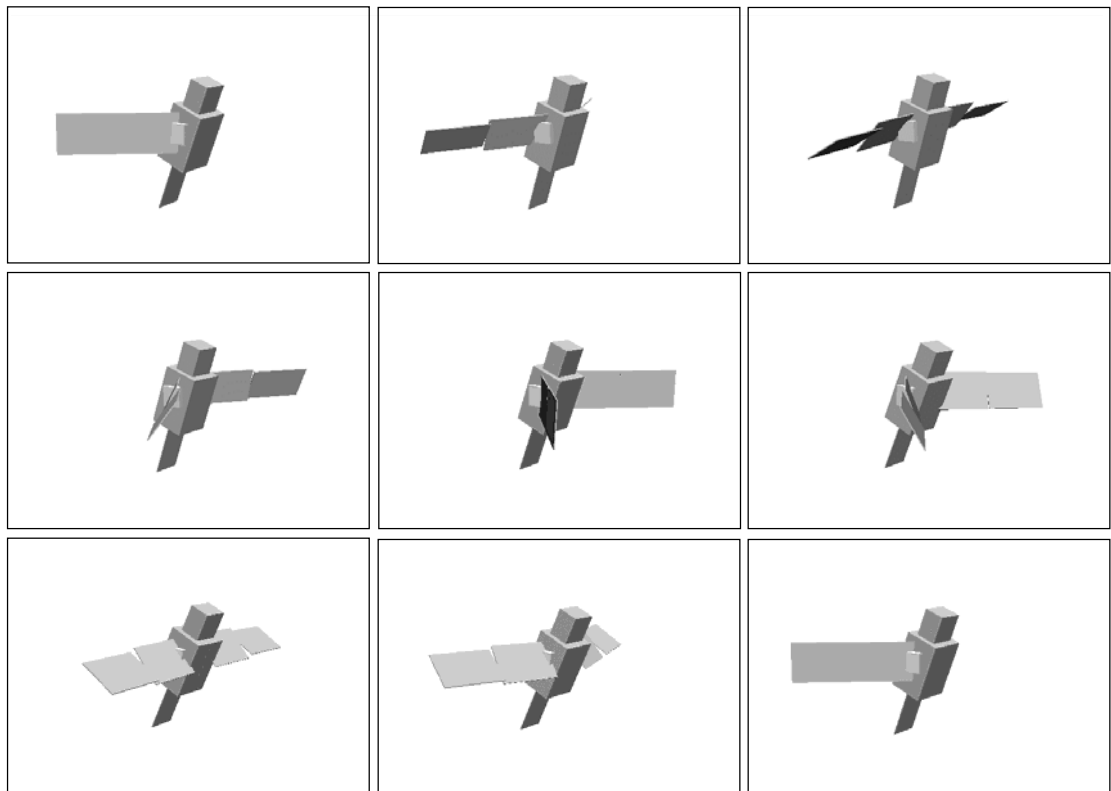


Figure 57: Hovering animation sequence for kinematic model 4. From top left $t=0$, $T/8$, $T/4$, $3T/8$, $T/2$, $5T/8$, $3T/4$, $7T/8$, T .

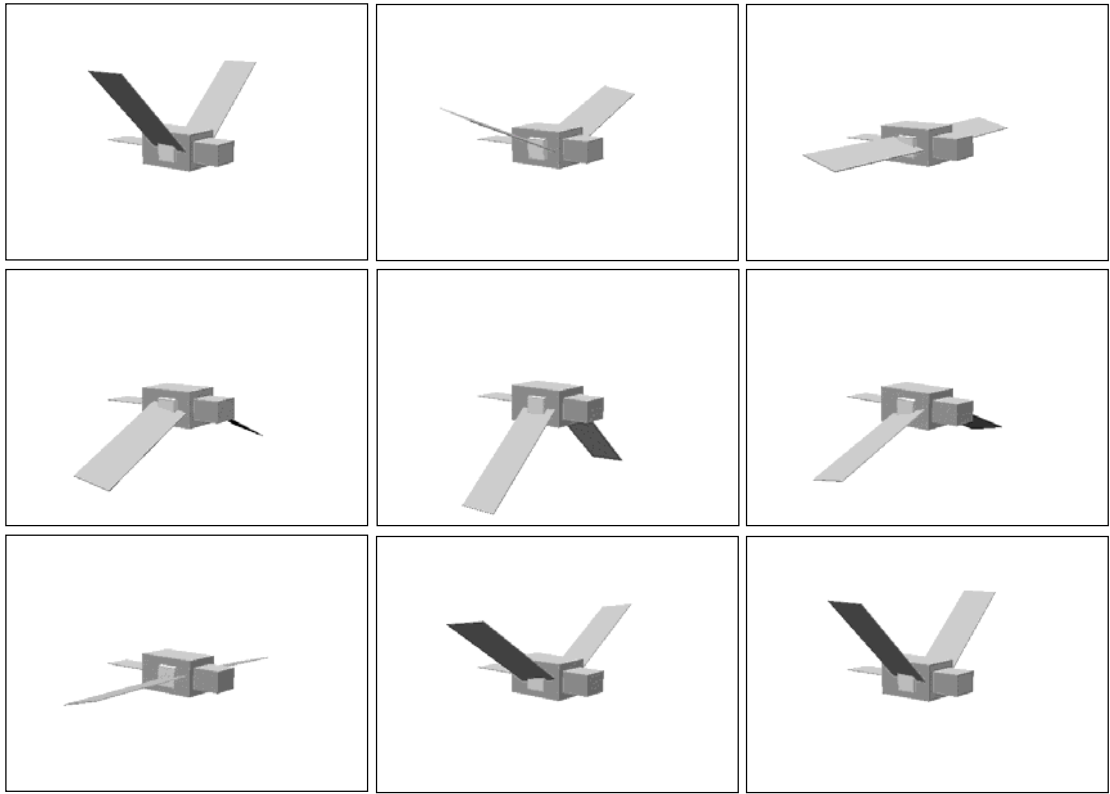


Figure 58: Cruising animation sequence for base kinematic model. From top left $t=0$, $T/8$, $T/4$, $3T/8$, $T/2$, $5T/8$, $3T/4$, $7T/8$, T .

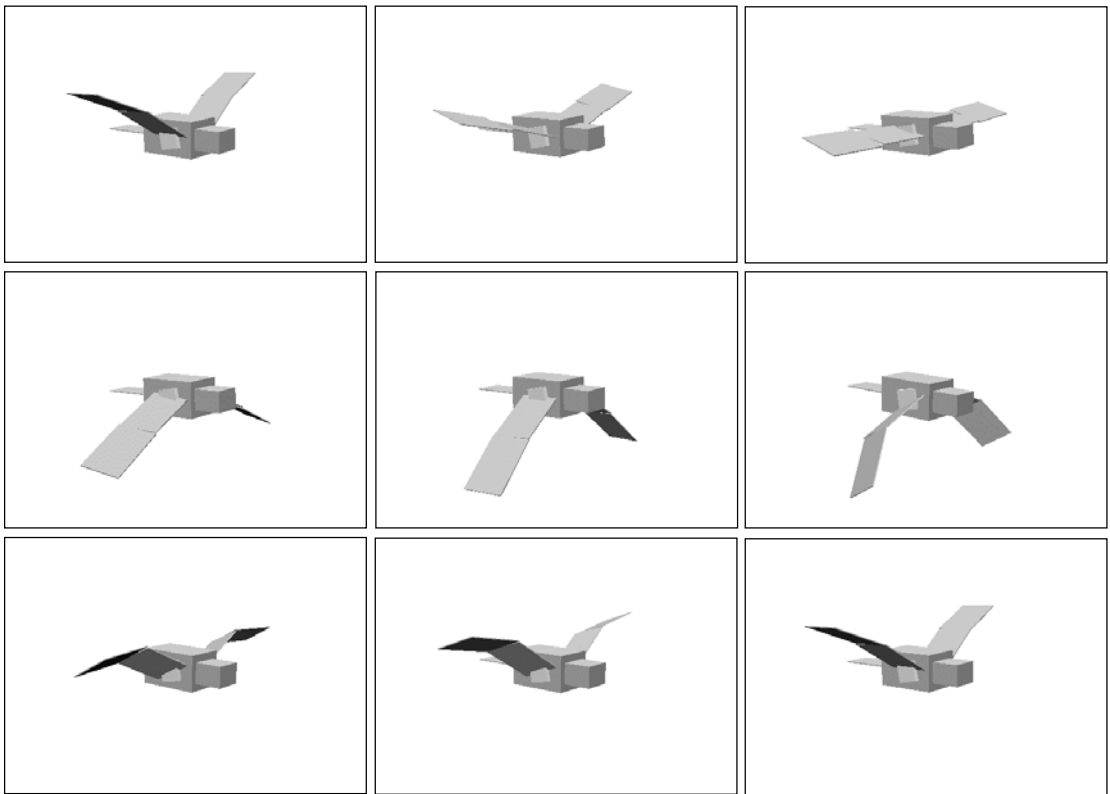


Figure 59: Cruising animation sequence for kinematic model 4. From top left $t=0$, $T/8$, $T/4$, $3T/8$, $T/2$, $5T/8$, $3T/4$, $7T/8$, T .

4.2.2 PATHLINES

Another useful visual result to consider is the effective pathlines created by the flapping motion. Pathlines from the control points were used to analyse the results throughout Numerical and Physical Results (4.1) and can now be included in the visualisation procedure. These results give a strong understanding of the somewhat complicated nature of the 3d motion. Figure 60 shows the paths lines computed for the base- kinematic model in cruise. Fig 61 shows the equivalent results for the 2-panel model, with additional paths lines for the second control point.

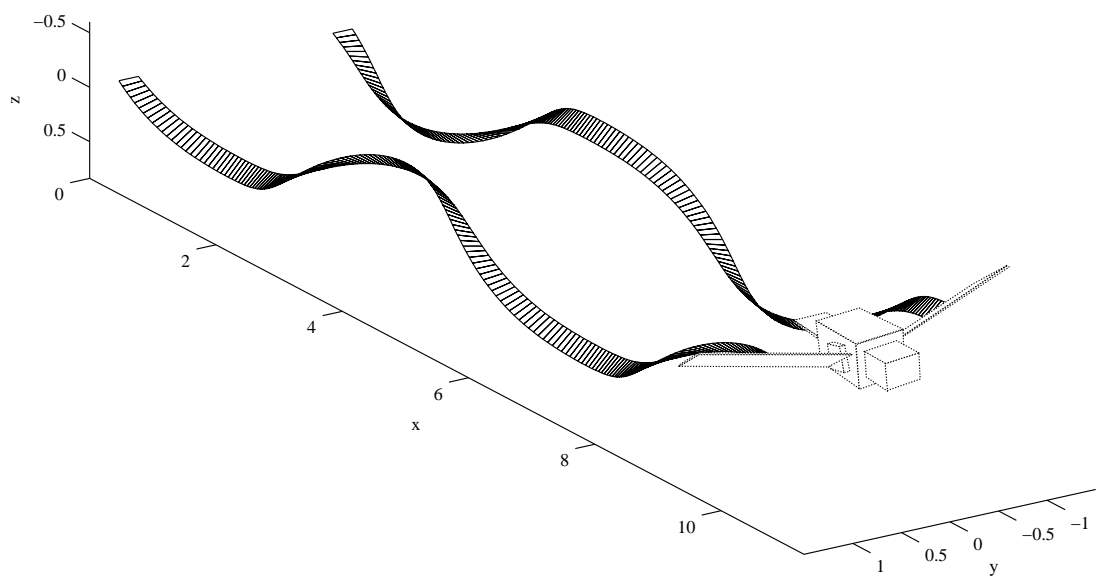


Figure 60: Pathlines for control points in the base-kinematic model during cruise

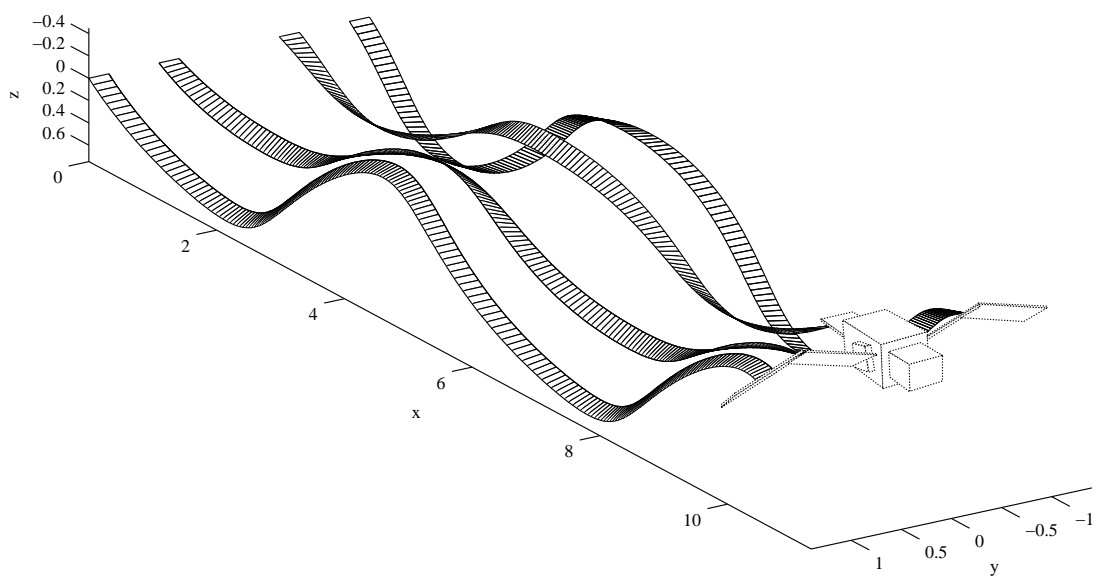


Figure 61: Pathlines for control points in kinematic model 4 during cruise

The pathlines represent a series of small surfaces, each being tangential to the wing control points at a particular instant in time. While the base-model shows a regular sinusoidal variation, the 2-panel model pathlines are somewhat more complicated. The effect of lag and bias of the second panel are now clearly visible as the surfaces for each control point are not coplanar. The difference in height between upstroke and downstroke can also be noted.

4.2.3 LOOKING AHEAD

The final visual results to be displayed will give the reader an insight into the future of this research program. The populated lookup tables can not only be used to consider individual flight cases, but also for varying input parameters to visualise manoeuvres. A simple example of a landing manoeuvre can be simulated by using the appropriate data from the lookup tables in Appendix 8 to provide increasing (negative) forces F_x and F_z , and reducing velocity; Figure 62 shows the visual results.²⁶

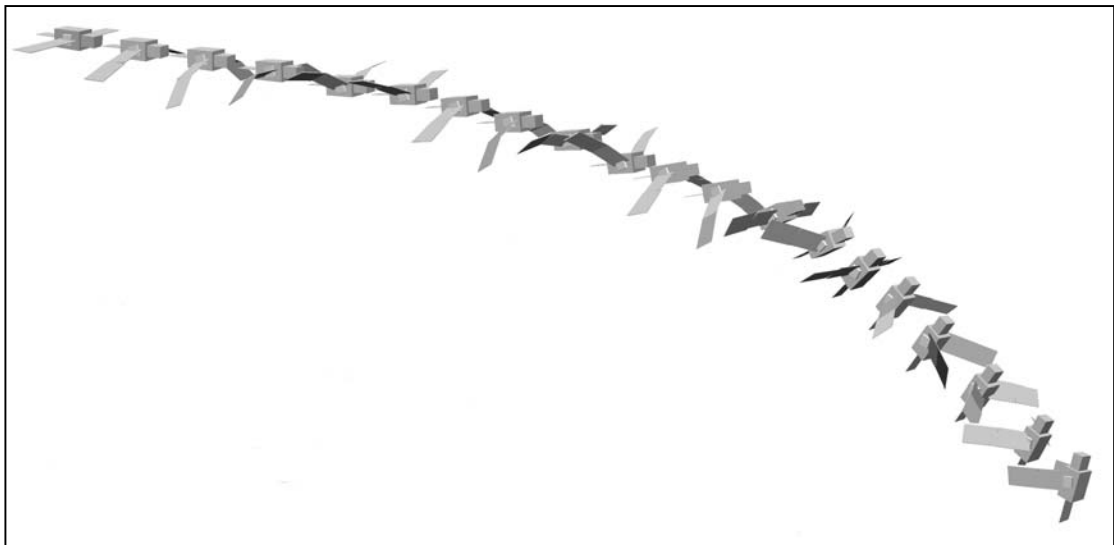


Figure 62: Simulated landing manoeuvre for kinematic model 4

The landing manoeuvre, as described throughout Numerical and Physical Results (4.1), is seen to follow the familiar pattern seen in nature; the simulation portrays a typical perched landing. The transition from the cruising regime with distinct motion

²⁶ It should be noted that the numerical solution to the landing routine assumes that the vertical velocity can be neglected without loss of accuracy.

of the wrist, to the hovering regime with large body pitch and flapping amplitudes, is clear from the animation.

4.2.4 PERFORMANCE

The quality of the animations that can be produced is closely coupled to the amount of computation time required. Figure 63 shows the time required to populate the entire results tables, and hence enable visualisations such as those shown in Figures 56-59, to be produced. This data allows the user of the program to offset the benefit of higher quality animations against the increased computational cost.

This project highlighted a problem in previous work of the large computational cost associated with model based animation. The crucial factor to be noted about the results in Figure 63 is that the solution times represent an initial setup time for the software, which, in theory, need only be performed once. Having gained fully populated results for a particular species, any number of simulations (within the bounds of the kinematic model) can now be performed in ‘real-time’.

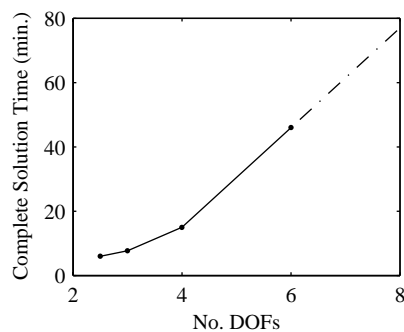


Figure 63: Computation times to populate results arrays for varying model complexity (2.8GHz Pentium 4 PC)

The performance details show an approximately quadratic increase in computation time with number of DOFs of the kinematic model. The data has been linearly extrapolated to give an indication of the computation time that would be required for an 8DOF model; which would represent a pitching body, with complete wing physiology. The solution times are still manageable, even on a single processor home computer, and are comparable to those required for a single simulation in the model of Wu & Popović (2003).

5. CONCLUSIONS AND FURTHER WORK

This chapter will be used to present conclusions made from the previous investigation and to offer suggestions for future work in this field. In addition to evaluating the technical results from the project, a brief assessment of the implementation strategy will also be given; the reader should refer to Success Criteria (1.2). Reference will be made to specific results from the project, with detailed proposals for continuation of this particular program and also other research in the field of flapping flight.

5.1 CONCLUSIONS

Considering firstly the numerical analysis, important advances have been made from the solution of the base-model. The original base-model produced somewhat inconclusive results, with spurious data that differed from expectations. A more general approach to obtaining accurate, stable solutions was derived by considering physical influences on the system. Specifically, the constraint of a solution when varying the input parameters was identified as crucial for stability and was successfully achieved by updating the initial guess for the optimisation routine.

The method of populating results tables was found to be coupled with the choice of initial guess for each solution parameter, which again required strong physical understanding. This part of the analysis also highlighted limitations of obtaining solutions in real-time and confirmed the use of precompiled lookup tables in the end program.

Further information regarding the physical nature of flapping flight, particularly the flight regimes, proved important in obtaining and analysing solutions. In the analysis of the base kinematic model solution, the low and high velocity cases were identified as having a smaller range of low cost solutions than the intermediate velocity range. Well-defined flight regimes offered distinct solutions that are comparable to those found in nature. The flight regime was also seen as the major contributor to the

magnitude of individual cost fractions. The prediction that motion gaits are not only important in aerodynamic analysis, but also in the general analysis of flapping flight, has been verified.

The method of grouping variables was used firstly in the application of the advance ratio, and also as a method of simplifying the 2-panel kinematic model. This approach enabled a straightforward route to uncovering the physical influence of the outer wing section in bird flight.

The solutions obtained were compared to typical flapping behaviour observed in nature. Results were assessed using intuitive reasoning, and also validated against previous analysis in this field. The overall body of results has been shown to be generally consistent with the literature examined. Examples such as the predicted hovering flight have been shown to mimic the analysis of well founded works (Ellington, 1984), with more sophisticated kinematic models offering the most physically accurate results.

Considering the results from the perspective of computer animation, the regimes of cruise, hover and landing shown in this project all appear highly realistic even with basic graphical representation. The computational cost associated with the animations is moderate, with solution times considerably lower than in similar works (Wu & Popović, 2003). The high system performance is largely attributed to the choice of blade-element theory as the method of aerodynamic modelling.

Numerical solution to the governing equations proved essential, with large numbers of calculations being required for a single case. Invoking the system in MATLAB proved highly efficient and easily extendable to allow a VRML simulation of the results.

The process of selection for optimisation methods provided valuable information that proved useful in the subsequent analysis. By carefully considering the exact nature of the gradient based optimisation methods, problems with the base-model solution were overcome.

The design methodology and implementation can be deemed successful in this project. By obtaining a balance between physical modelling and the objective of animation, a strong advance in the system sophistication has been achieved. The project managed to reach the specified objectives within the available time period, while providing thorough and reasoned analysis throughout. A suitable amount of time was allowed for research into the project, which proved essential in the subsequent implementation.

The strategic selection of simplifications to the kinematic model has led to a stable, accurate system of determining solution parameters. A gradual approach to the increase in model complexity allowed rigorous numerical analysis of the results throughout. The high modularity of the kinematic system has facilitated the development of more complicated physiological processes within the models. Overall, it has been shown that relatively simple physical models can be used to produce realistic solutions and simulations that capture the essence of avian motion.

5.2 FURTHER WORK

This project has highlighted a number of areas that require further investigation. From the analysis performed, extensions can be suggested to the current research program, and also to the general analysis of flapping flight.

Perhaps the most apparent area for consideration would be the validation of the fully-populated lookup tables for the final kinematic model. It was shown that while the varying velocity data is believed to be accurate, further confirmation of the effect of varying forces should be gained. It would be particularly useful to produce visual results that represent the entire data set, thus considering all possible flight cases.

Further extensions to the kinematic model are the next step to the development of the program sophistication. Inclusion of the wrist sweep and elbow bend motions would provide a complete 7 DOF representation of a bird wing skeleton. Following this, the second key aerodynamic controller, the tail, could be added to the model with an appropriate level of sophistication.

Inclusion of feathers into the model is still a possibility, though whether this would be done at the physical level, or as a purely graphical feature, is undecided. The physiological mechanism by which feathers are controlled would strongly influence the inclusion of feather motion as a free parameter in the solution; background research of bird physiology would be vital.

With regards to the overall dynamic model, extension to the full 6DOF (translating, rotating) body is desirable. This would require all force and velocity components to be considered in the solution; the final populated arrays would have to provide information for all motion parameters with forces F_x , F_y and F_z and velocities V_x , V_y and V_z . The resulting model would be capable of more advanced motion, including steep climb/descent and turning, to provide realistic modelling of all bird manoeuvres seen in nature.

The physical accuracy of the model could be increased by enhancing the aerodynamic method employed. As described in the Aerodynamic Modelling (2.4) the inclusion of downwash effects could be achieved by incorporating the axial momentum theory to the current model. This method could be applied to the function `flapcost` to include downwash effects ‘internally’ in the physical model, and repopulate the lookup tables. Alternatively the standard lookup tables could be used and the force and velocity requirements adjusted to account for the induced motion of the surrounding air, therefore applied ‘externally’ to the physical model.

The calculation of aerodynamic forces could be advanced by adapting the blade-element theory to utilise results from recorded bird wing data, obtained experimentally or otherwise. Furthermore, the prescription of forces for the body could also employ more realistic data to closely represent a chosen species.

A number of suggestions can be made for future work on the research program. While this project has focussed on the physical and numerical solution to the given problem, it has become evident how the exact method of modelling bird flight for animations will be achieved. The direct prescription of forces and velocity values into the optimisation routine should be overcome by including a method of specifying a flight trajectory and calculating the corresponding force and velocity targets. This also leads to the suggestion of including a flight control system to link the calculated results to the desired simulation path. Following this, the entire program will require assembly with an appropriate GUI.

Finally, with regards to the general analysis of flapping flight, a number of suggestions for continuation of studies in this field can be made. It would be particularly useful to draw parallels between the flight regimes observed from the results of this project, and the aerodynamic motion gaits observed in other works (Rayner & Gordon, 1998). To strengthen the link between model-based animation and physical understanding, experimental and numerical results of the problem domain are essential.

This leads to the suggestion for using advanced aerodynamic models, such as commercial CFD software packages, to gain general aerodynamic data (force,

moment coefficients) for flapping wings, and also information of the surrounding flow field. There is scope for constructing CFD models to analyse various technicalities of flapping flight such as unsteady wake behaviour, wing-body interaction effects and the aerodynamics of feathered wings.

APPENDICES

APPENDIX 1A - BASE-MODEL

Aerodynamic modelling of flapping motion for model-based animation

Bill Crowther

last updated 04/09/20053

Objective

To calculate aerodynamic forces acting on a flapping wing for the purposes of computer animation

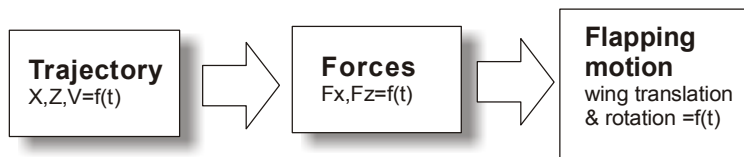


Figure 1 Desired calculation sequence for model-based animation of flapping flight

Axis systems

Earth fixed axes XYZ

Z aligned with the gravity vector

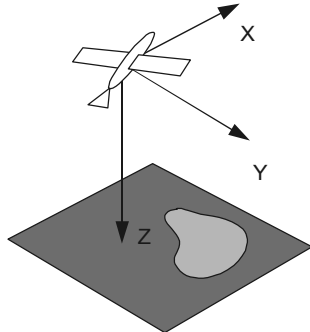


Figure 1 Earth axes

Body fixed axes xyz,

Fluid body aligned with x axis, figure 1. Flapping motion takes place in yz plane

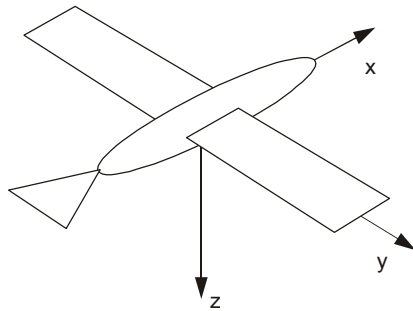


Figure 2 *Body axes*

Wind axes

Axis system aligned with the local wind vector

Velocity diagrams

In the general case, the fluid body orientation vector is non coincident with the free stream vector, figure 3, giving a free stream component w_1 in the flapping plane. To simplify the problem, it is proposed that the fluid body axis is always aligned with the free stream relative wind, figure 4. This means that

$$\gamma_1 = \theta_1 \text{ and}$$

$$\alpha_1 = 0$$

This is not unreasonable because most flying creatures try and fly with the body pointing into wind to reduce drag.

Aerodynamic forces are calculated in a frame aligned with the local wind direction, figure 5.

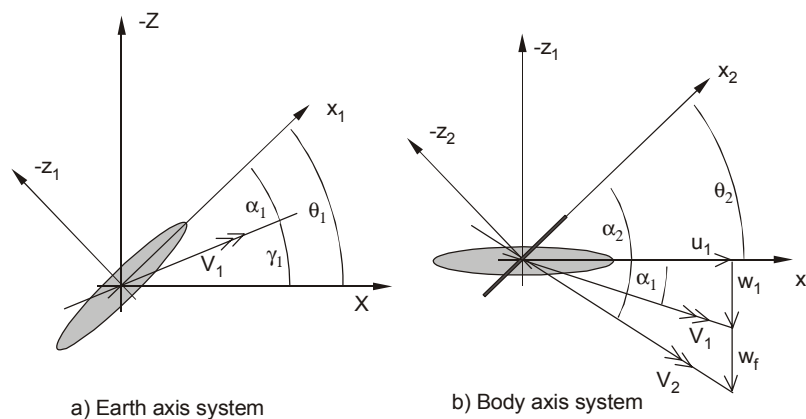


Figure 3 *Velocity components in Earth and body axes for general flapping flight (body x axis not aligned with free stream)*

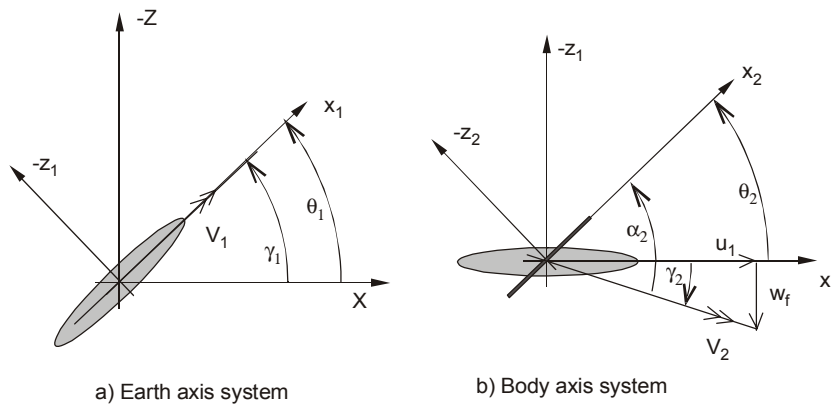


Figure 4 Velocity components in Earth and body axes for flapping flight with the body x axis aligned with the free stream

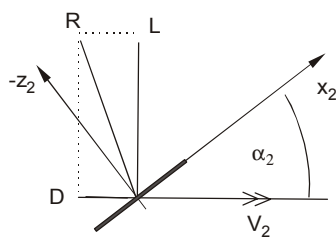


Figure 5 Aerodynamic force components in axis system aligned with the local free stream direction

Flapping kinematics

The flapping kinematic geometry used for the present work is shown in figure 6. The 2d wing motion is composed of simple harmonic translation (z_f) and rotation (θ_2) at a common frequency ω . The rotation is composed of an offset (θ_{2o}) and a time varying component defined by a maximum amplitude θ_{2max} . The phase between the translation and rotation is fixed at 180 degrees. This is the most sensible arrangement for steady aerodynamics. (For real unsteady aerodynamics phase lag or lead may be of benefit).

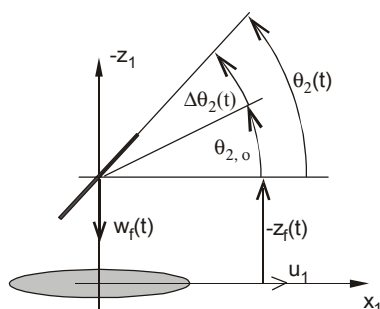


Figure 6 Definition of flapping kinematics

Calculation methodology

There are two main theoretical approaches to calculation of aerodynamics forces due to flapping motion: blade-element theory and actuator disc theory. The present work

uses both theoretical approaches in an iterative manner. Firstly, blade-element theory is used to calculate the thrust in the axial direction due to flapping, ignoring the induced velocity. Secondly, the induced velocity is calculated using the calculated thrust and actuator disc theory. The thrust is then recalculated and so on.

The calculation takes place over one complete flap cycle, starting with the wing at the beginning of the down stroke and ending with the wing at the end of the up stroke. Variables are typically vectors of values over a complete cycle for efficient Matlab implementation.

Amplitude of flapping motion:

$$z_f = -\hat{z}_f \cos(\omega t)$$

Velocity of flapping motion

$$w_f = \hat{z}_f \omega \sin(\omega t)$$

Pitch attitude of wing relative to x-axis

$$\theta_2 = \hat{\theta}_2 \sin(\omega t + \phi) + \theta_0$$

Velocity at actuator disc (yz plane) made up from V_{∞} and an ‘induced’ velocity from the mean axial (thrust) force. This equation comes from actuator disc theory for a climbing helicopter...

$$u_1 = -\frac{V_{\infty}}{2} + \left(\frac{V_{\infty}^2}{4} + \frac{\bar{F}_x}{2\rho A} \right)$$

Angle of the relative wind with respect to the x-axis

$$\gamma_2 = -\tan^{-1} \left(\frac{w_f}{u_1} \right)$$

Angle of attack of the wing

$$\alpha_2 = \theta_2 - \gamma_2$$

Wing lift coefficient

$$C_L = \hat{C}_L \sin(2\alpha_2)$$

Wing drag coefficient

$$C_D = \frac{1}{2} \hat{C}_D (1 - \cos(2\alpha_2))$$

Relative velocity at the wing

$$V_{RW}^2 = u_1^2 + w_f^2$$

$$q = \frac{1}{2} \rho V_{RW}^2$$

$$L = q S C_L$$

$$D = q S C_D$$

Mechanical power input

$$P = -F_z w_f$$

Thrust force

$$F_x = -D \cos(\gamma_2) - L \sin(\gamma_2)$$

Lift force

$$F_z = -L \cos(\gamma_2) + D \sin(\gamma_2)$$

$$C_{F_x} = \frac{F_x}{qS}$$

$$C_{F_z} = \frac{F_z}{qS}$$

$$C_P = \frac{P}{qV_{RW}S}$$

For trajectory simulation, purposes the body axis forces need to be resolved back into Earth axes:

$$F_X = F_x \cos(\theta_1) + F_z \sin(\theta_1)$$

$$F_Z = F_z \cos(\theta_1) - F_x \sin(\theta_1)$$

Dimensional analysis

If the flapping induced velocity is ignored, the aerodynamic lift, drag and power coefficients for a flapping wing system will be similar if the nondimensional flapping motions are similar and the ratio of the free stream speed to flapping speed is similar.

For propeller work, this ratio is known as the advance ratio

$$AdvR = \frac{\hat{w}_f}{V_\infty}$$

For zero free stream speed (hover) the advance ratio is infinite. An alternative metric is the advance angle defined as

$$AdvA = \tan^{-1}\left(\frac{\hat{w}_f}{V_\infty}\right)$$

which goes to 90 degrees as V_∞ tends to zero.

Results

Force and power coefficients histories for a complete flap cycle

Hovering flight, figure 7, forward flight figure 8. Note that for hovering, aerodynamic force is generated on both up and down strokes. For forward flight the flap kinematics have been adjusted to give zero net force on the up stroke.

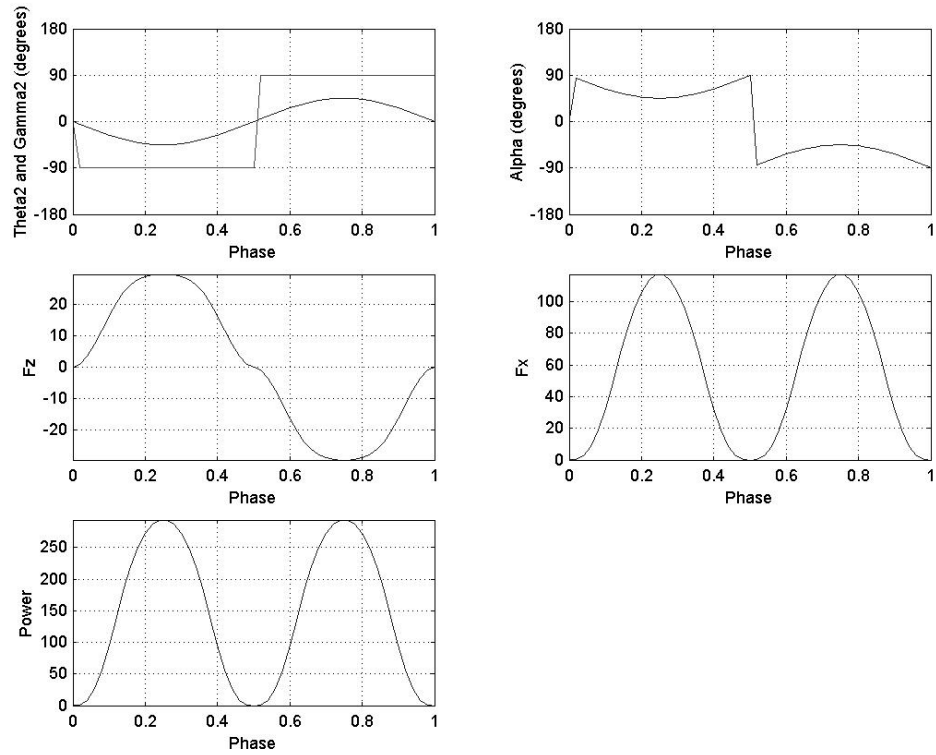


Figure 7 Time histories for one flap cycle during hover ($V_{inf}=0$, $\theta_{2o}=0$, $\theta_{2max}=45$ degrees, $w_{fmax}=10$)

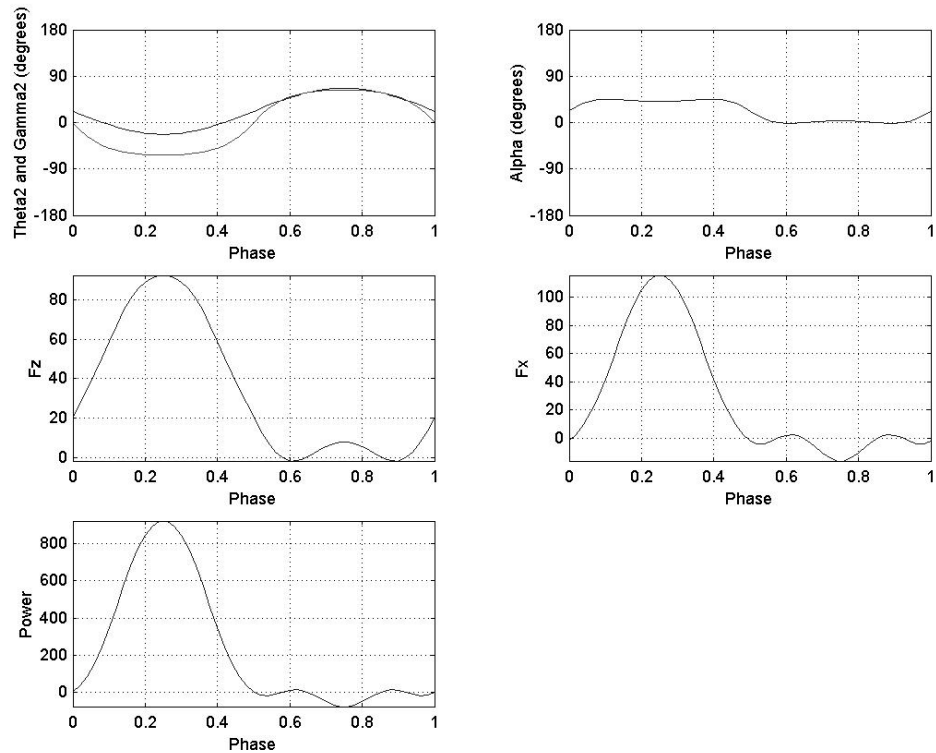


Figure 8 Time histories for one flap cycle during forward flight ($V_{inf}=5$, $\theta_{2o}=22$, $\theta_{2max}=45$ degrees, $w_{fmax}=10$)

Effect of flap kinematic parameters on mean aerodynamic forces and power

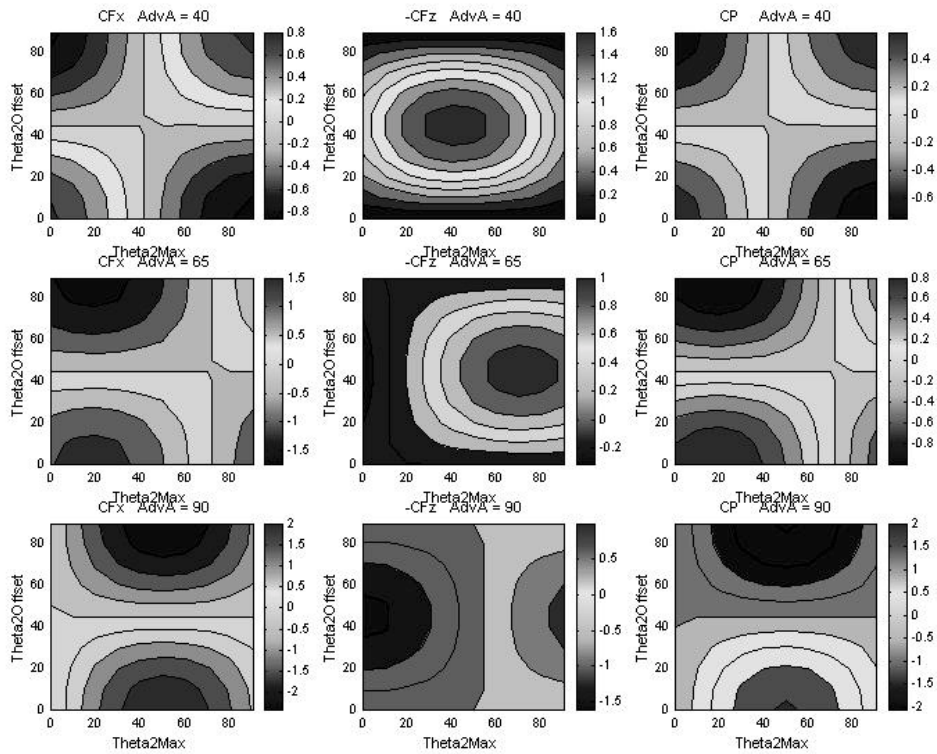


Figure 9 C_{fx} , C_{fz} and CP as a function of flap kinematic params and advance angle, induced velocity = 0

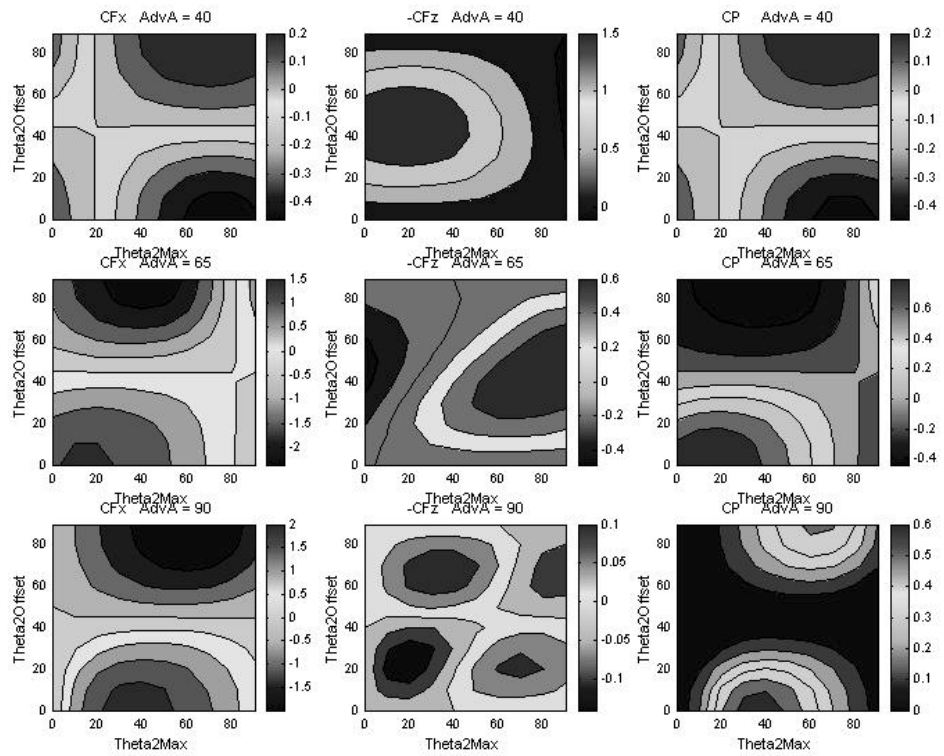


Figure 10 C_{fx} , C_{fz} and CP as a function of flap kinematic params and advance angle, induced velocity included in calc.

APPENDIX 1B – SOURCE CODE, SIMULINK MODEL AND SOLUTION SUPPLIED FOR BASE-MODEL

SOURCE CODE

FlapOptimizeMatrix

```
clear
global FlapParamsMatrix Fxrange Fzrange Vrange

Xo=[.5 .5 .5]; %initial guess for Theta Phifmax and Thetafmax
Xmin=[0 0 0]; % lower bound
Xmax=[pi/2 pi/2 pi/4]; %upper bound theta phif thetaf

CostWeight=[1 1 .5 0.00 0.0];%[CFX CFZ CP CPhif CThetaf]
options=optimset('Tolfun',1e-6,'Display','off','largescale','off');
tic

Fxrange=-.5:1:.5;
Fzrange=-.5:1:0;
Vrange=0:.5:1;

FlapParamsMatrix=zeros(size(Fxrange,2),size(Fzrange,2),size(Vrange,2),3);
i=0;
for Fxed=Fxrange; %Fxe demand
    i=i+1
    j=0;
    for Fzed=Fzrange; %Fze demand
        j=j+1;
        k=0;
        for V=Vrange;
            k=k+1;
            Veinf=[V 0 0]';
            [FlapParams] = fmincon(@Flapcost,Xo,[],[],[],[],[],Xmin,Xmax,[],options,Fxed,Fzed, Veinf, CostWeight);
            FlapParamsMatrix(i,j,k,:)=FlapParams;
        end
    end
end
end
toc
%print final values on screen
%FlapPrint(FlapParams, Veinf)
%Run Simulink model with params
ThetaTable=FlapParamsMatrix(:,:,1);
PhifTable=FlapParamsMatrix(:,:,2);
ThetaTable=FlapParamsMatrix(:,:,3);
```

```
OptimizeContourPlot;
DisplayFloidLookup; % open sim model. Press play to run
```

flapcost

```
function FlapCost=Flap(FlapParams,Fxed,Fzed, Veinf, CostWeight)
%flap calc
%xyze = earth fixed, xyz1 = body fixed, xyz2 = wing fixed xyz3 wind axes
% Omega=flapfrequency
% Phif=wing flap angle
% Thetaf = wing twist angle
% Theta = body pitch angle
%
Theta=FlapParams(1);
Phifmax=FlapParams(2);
Thetafmax=FlapParams(3);

Omega=1;

CDo=0.05; %zero lift drag coefficient
CDmax= 1; % max drag coefficient
CLmax=2; %max lift coeff
% Fxed=0;
% Fzed=0.1;

s=1; % wing semi span
c=1;
Rel=[cos(Theta) 0 -sin(Theta); 0 0 0; sin(Theta) 0 cos(Theta)];
T=2*pi/Omega;
dt=T/10;
n=1;
for t=0:dt:T;
    Phif=Phifmax*sin(Omega*t);
    Thetaf=Thetafmax*sin(Omega*t-pi/2);
    R12=[cos(Thetaf) 0 -sin(Thetaf); 0 0 0; sin(Thetaf) 0 cos(Thetaf)];
    V1f=[0 0 0.5*s*Phifmax*cos(Omega*t)];
    V1inf=Rel*Veinf;
    V1=V1f+V1inf;
    V2=R12*V1;
    alpha=atan2(V2(3),V2(1));
    q=0.5*1*(V2(1)^2+V2(3)^2);
    CD=CDo+CDmax*sin(alpha)^2;
    CL=CLmax*sin(2*alpha);
    F3=[-CD 0 -CL]'.*q*s*c;
    R32=[cos(alpha) 0 -sin(alpha); 0 0 0; sin(alpha) 0 cos(alpha)];
```

```

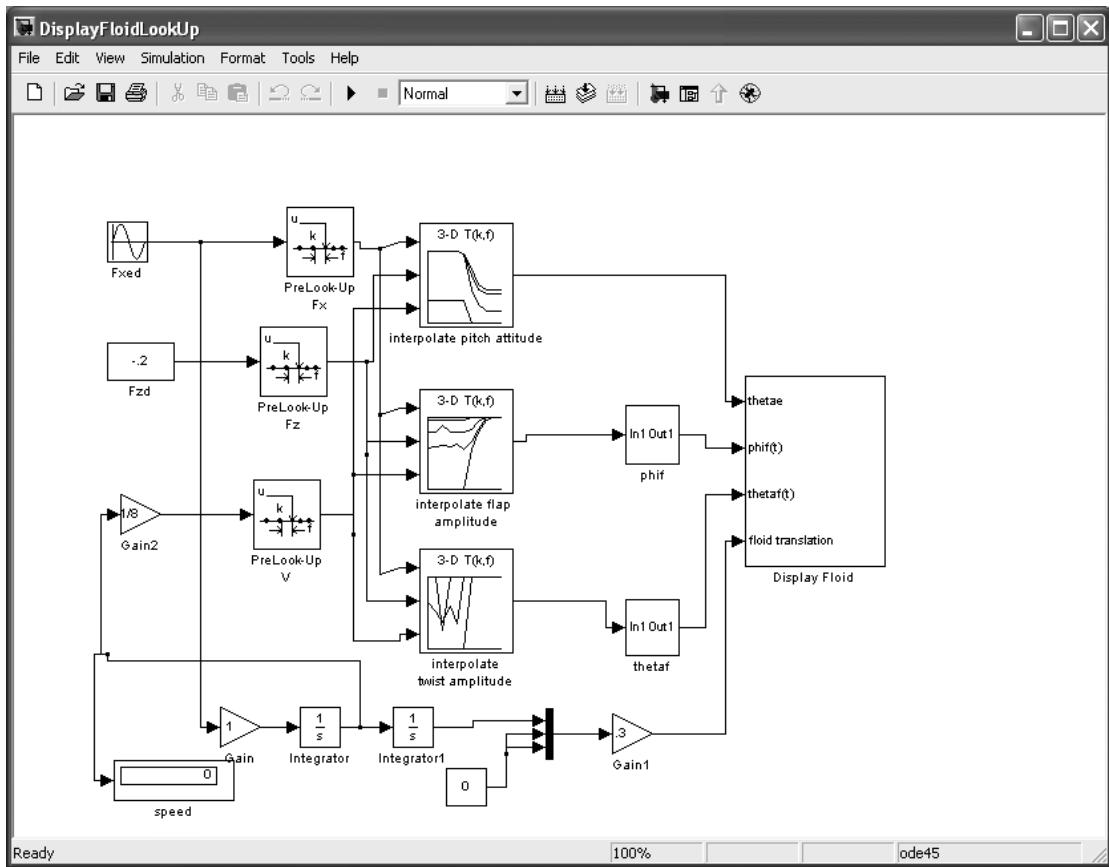
F2=R32*F3;
F1=R12.*F2;
Power(n)=F1(3)*V1f(3);
Fe(:,n)=Re1.*F1;
n=n+1;
end %t

Fxebody=-0.5*1*Veinf(1)^2*sin(Theta);% body drag coeff = 0, 1 when theta = pi/2
P=mean(Power);
Fxe=mean(Fe(1,:))+Fxebody;
Fze=mean(Fe(3,:));

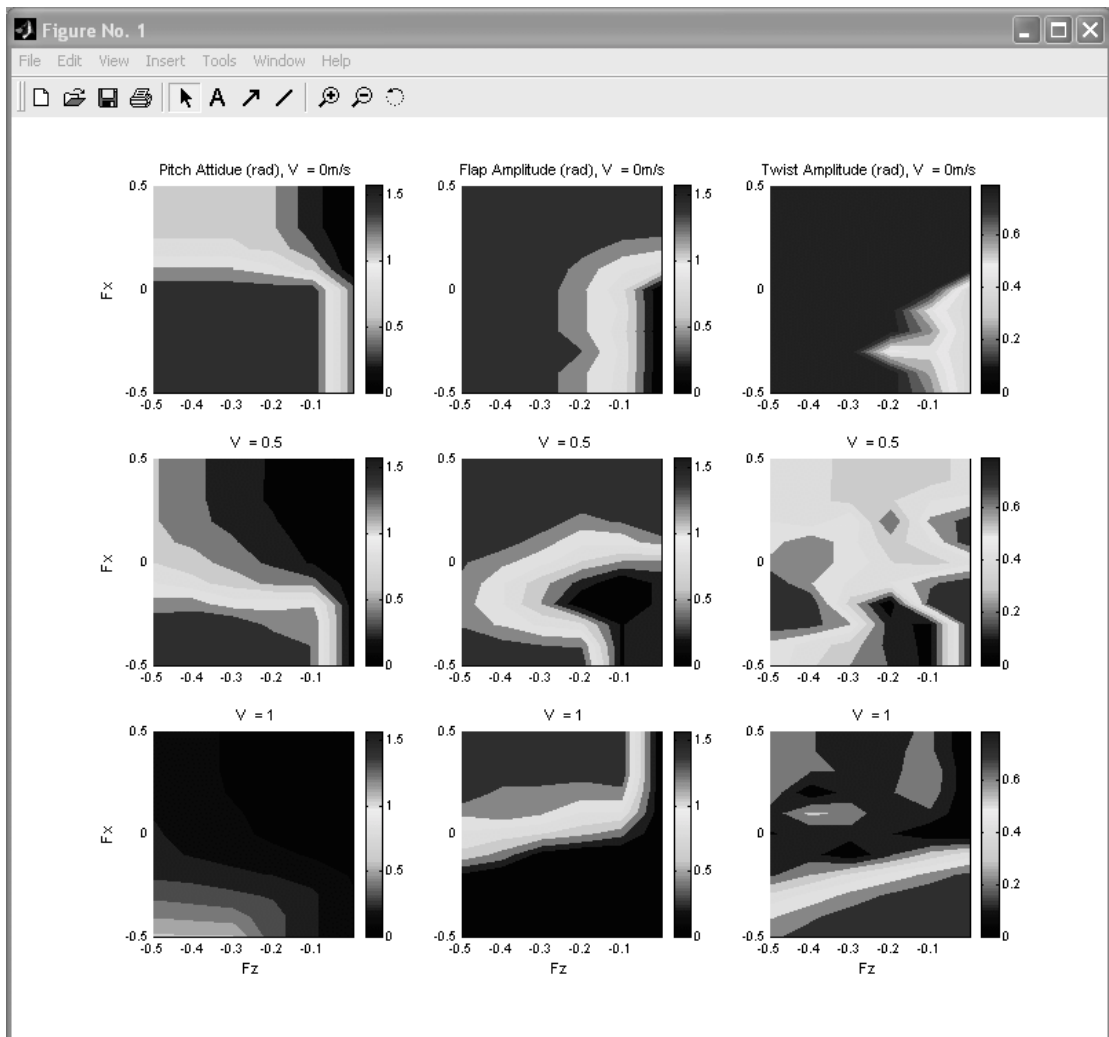
FxeCost=abs(Fxe-Fxed);
FzeCost=abs(Fze-Fzed);
PCost=abs(P); %ne
RawCost=[FxeCost FzeCost PCost Phifmax Thetafmax];
FlapCost=sum(RawCost.*CostWeight);% [CFX CFZ CP CPhif CThetaf]
% figure
% plot(Power);

```

SIMULINK MODEL



SOLUTION CONTOUR PLOTS FROM SUPPLIED SOURCE CODES



APPENDIX 2 – GENETIC ALGORITHM SOURCE CODE

```

clear all

for i=1:10
    for j=1:4
        bestgenes(i,j)=0;
    end
end
for i=1:10;
    for j=1:3;
        a(i,j)=rand(1);
    end
end
a ; %10 random genes (variable range 0-1)

%*****RANK GENES*****
Thetamax=10;
Thetafmax=10;
Phifmax=10;
maxangles = [Thetamax Thetafmax Phifmax];

for i=1:10;
    b(i,:)=a(i,:).*maxangles;
end
b ; %10 random genes (variable range of angles)

%test function to give design space:
[x,y] = meshgrid(0:0.1:10, 0:0.1:10);
test=2*sin(3*x)+0.2*(x-5).^2+cos(2*y)+0.3*(y-4).^2;
subplot(2,2,1)
surf(x,y,test)
xlabel('1');
ylabel('2');

for h=1:1000 %this starts the evolution cycle, this assumes that (later) the array 'c' always contains the actual angles
for i=1:10;
    func(i)=2*sin(3*b(i,1))+0.2*(b(i,1)-5).^2+cos(2*b(i,2)) +0.3*(b(i,2)-4).^2 %func is the test design space
    c(i,:)=b(i,1) b(i,2) b(i,3) func(i)]; %c now contains the variables and the cost function
end
c;
%c(1,4)

```

```

%*****sort by cost function*****
for n=2:10
for i=n:10
if c(i,4)<c(n-1,4)
swap(i,:)=c(n-1,:);
c(n-1,:)=c(i,:);
c(i,:)=swap(i,:);
end
end
end
%gene pool now sorted by cost
c;

Thetaarray(h)=c(1,1); %store best gene info in these arrays
Thetafarray(h)=c(1,2); %store best gene info in these arrays
Phifarray(h)=c(1,3); %store best gene info in these arrays

%*****selection*****
for i=1:10
c(i,4)=1/c(i,4);
end
bestgene(h,:)=c(1,:);

for k=1:10
for j=1:10
if c(j,4)>bestgenes(k,4)
bestgenes
bestgenes(k,:)=c(j,:);
end
end
end

c;
sumfitness=sum(c(:,4));
for i=1:10
f1(i)=c(i,4)/sumfitness ;
end
f1;
for i=1:10
f2(i)=f1(i)/(max(f1(:)));
end
f2;

for g=1:2:9
genepool
%start selection cycle - finish at stage 7 where new children added to new
%note g increments in 2 as each pass of g loop adds
%a pair of genes to pool
%swap array just used to store temporary data for swapping

%c() now contains 10 genes, with c(4) as ascending cost function

%c now contains 10 genes with c(4) as descending fitness function
%i.e. lower cost function = higher fitness = better solution

%loop through bestgenes array
%loop through current gene pool (c) array
%if any of the genes in the currentpool havebetter cost function that in
%then add this gene to the bestgenes array

%each gene is given a fraction of the roulette wheel according to fitness

%the fractions are scaled (so f2 max = 1) to allow random no. gene

%start selection cycle - finish at stage 7 where new children added to new
%note g increments in 2 as each pass of g loop adds
%a pair of genes to pool

```

```

r=rand(1) ;
for i=10:-1:1
    if r<=f2(i)
        p1=[c(i,1) c(i,2) c(i,3)] ;
        break
    else
        end
    end
r=rand(1) ;
for i=10:-1:1
    if r<=f2(i)
        p2=[c(i,1) c(i,2) c(i,3)] ;
        break
    else
        end
    end
end
%*****MATE / CROSSOVER*****%
r=3*rand(1) ;
for i=1:3
    if i<=r
        c1(i)=p1(i) ;
        c2(i)=p2(i) ;
    else
        c1(i)=p2(i) ;
        c2(i)=p1(i) ;
    end
end
c1 ;
c2 ;
%*****MUTATE*****%
%child 1 mutation:
for i=1:3
    r0=rand(1) ;
    r1=rand(1) ;
    r2=rand(1) ;
    if r0<=0.1
        if r1<=0.5
            r2=-1*r2;
        end
    end
    c1(i)=c1(i)+r2*maxangles(i) ;
end
%spin the wheel
%start at worst (lowest) lowest fitness gene
%if rand. no is in gene fitness interval
%parent1 selected as that gene

%parent2 selected

%to have random number on interval 0-3

%child 1 element i from parent 1 (i.e not crossed yet)
%child 2 element i from parent 2 (i.e not crossed yet)

%child 1 element i from parent 2 (i.e. crossed)
%child 2 element i from parent 1 (i.e. crossed)

%now have child1 and child2

%for each gene element
%mutation probability
%mutation increase or decrease
%size of mutation
%mutation probability (should be small !!!)

%apply mutation

```

```

if c1(i)<0
    %lower bound on angles
    c1(i)=0;
end
if c1(i)>maxangles(i)
    %upper bound on angles
    c1(i)=maxangles(i);
end
end
end
%child 1 now mutated
c1;

%child 2 mutation:
for i=1:3
    r0=rand(1);
    r1=rand(1);
    r2=rand(1);
    if r0<=0.01
        if r1<=0.5
            r2=-1*r2;
        end
        c2(i)=c2(i)+r2*maxangles(i);
        if c2(i)<0
            c2(i)=0;
        end
        if c2(i)>maxangles(i)
            c2(i)=maxangles(i);
        end
    end
end
end
end

%*****add children to new gene pool*****
g;
b(g,:)=c1(1,:); %add child 1 to gene pool
b(g+1,:)=c2(1,:); %add child 2 to gene pool
end
%go back to stage 4 to select two new parents

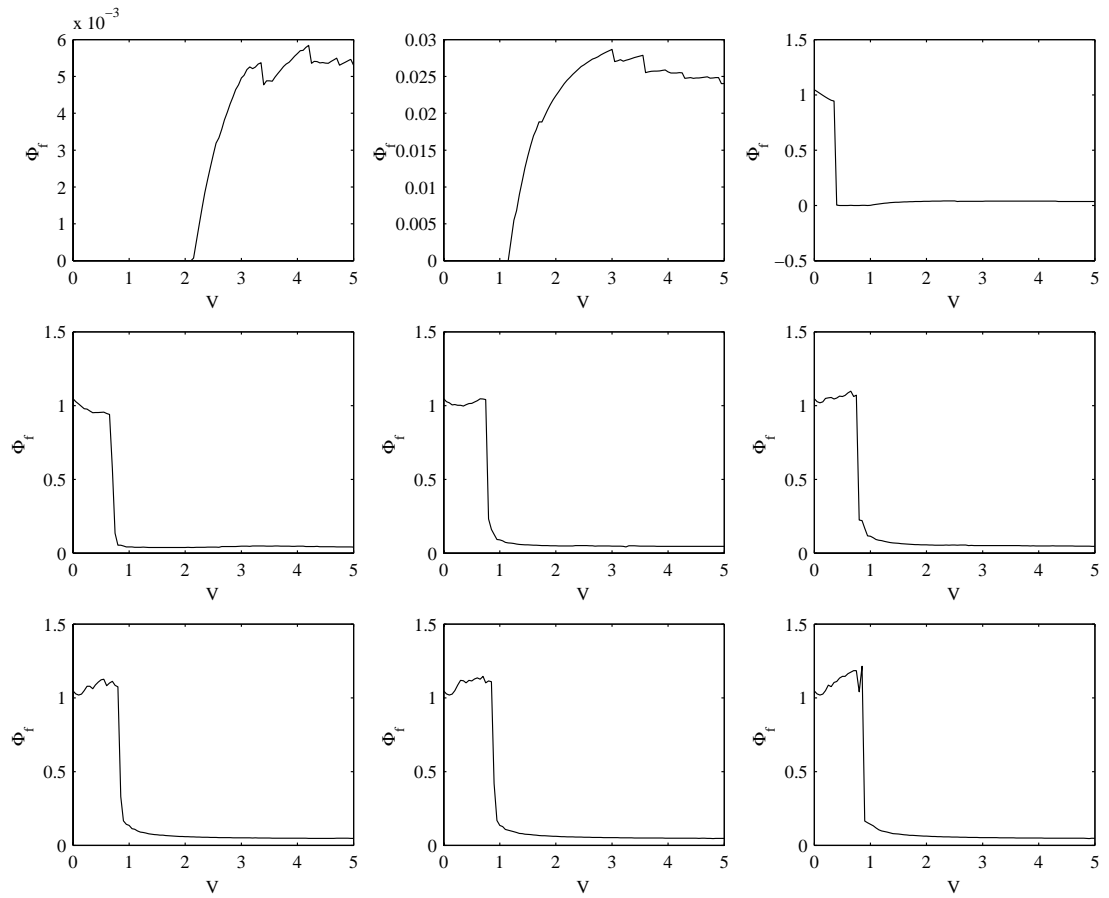
%now have a new gene pool stored in b
sumfitnessarray(h)=sumfitness;
h

end %end %go back to stage 2 with new gene pool - next generation
%plot(sumfitnessarray)
subplot(2,2,2);

```

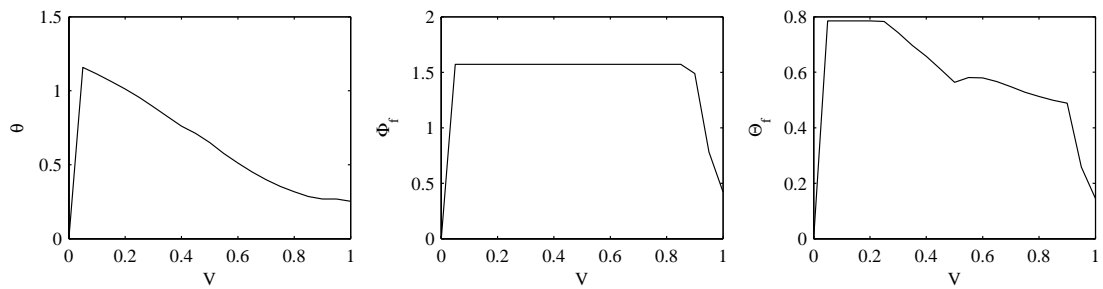
```
plot(Thetaarray, ':')
hold on
plot(Thetafarray, ':')
hold on
%plot(Phifarray)
%hold off
subplot(2,2,3)
plot(bestgene(:,4));
```

APPENDIX 3 – SOLUTIONS FOR VARYING DISTANCE-COST WEIGHTING



(from top, left to right) : $v=10^{-1}, 10^{-1.25}, 10^{-1.5}, 10^{-1.75}, 10^{-2}, 10^{-2.25}, 10^{-2.5}, 10^{-2.75}, 10^{-3}$

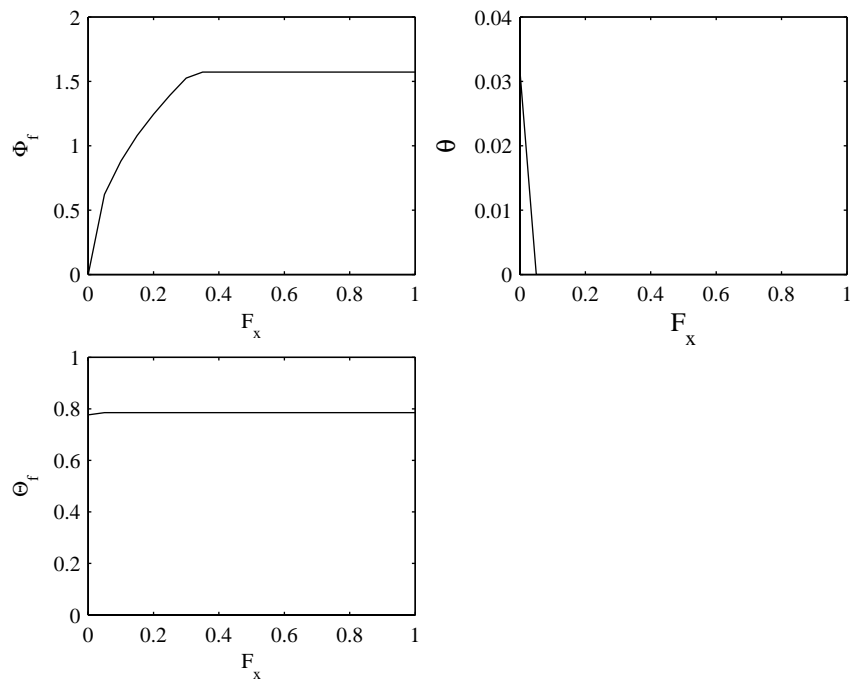
APPENDIX 4 – INAPPROPRIATE INITIAL GUESS VALUES FOR VARYING VELOCITY SOLUTIONS



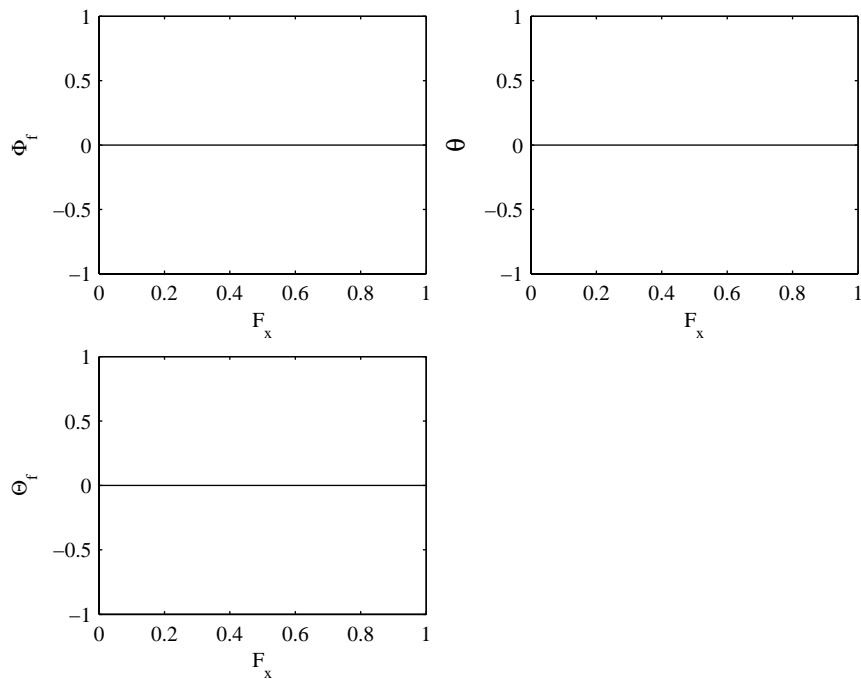
(Above, obtaining solutions for $V=0$ to $V=1$, using starting guess values $\theta=\Phi_j=\Theta_j=0$)

APPENDIX5 – SOLUTIONS FOR VARYING F_x AND F_z

VARYING F_x

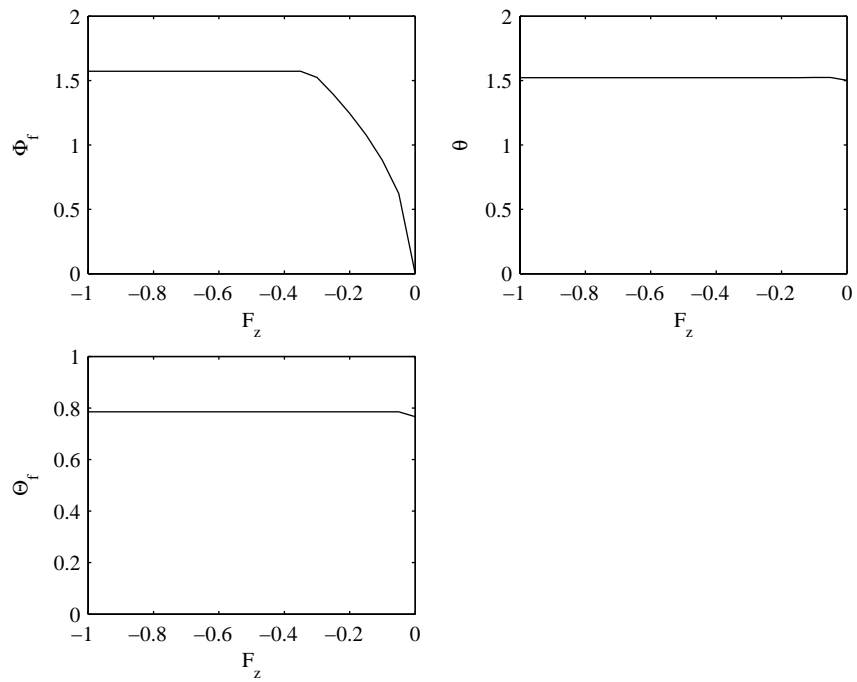


(Above, obtaining solutions for $F_x=1$ to $F_x=0$, using starting guess values $\theta=\pi/2$, $\Phi_f=\pi/2$, $\Theta_f=\pi/4$)

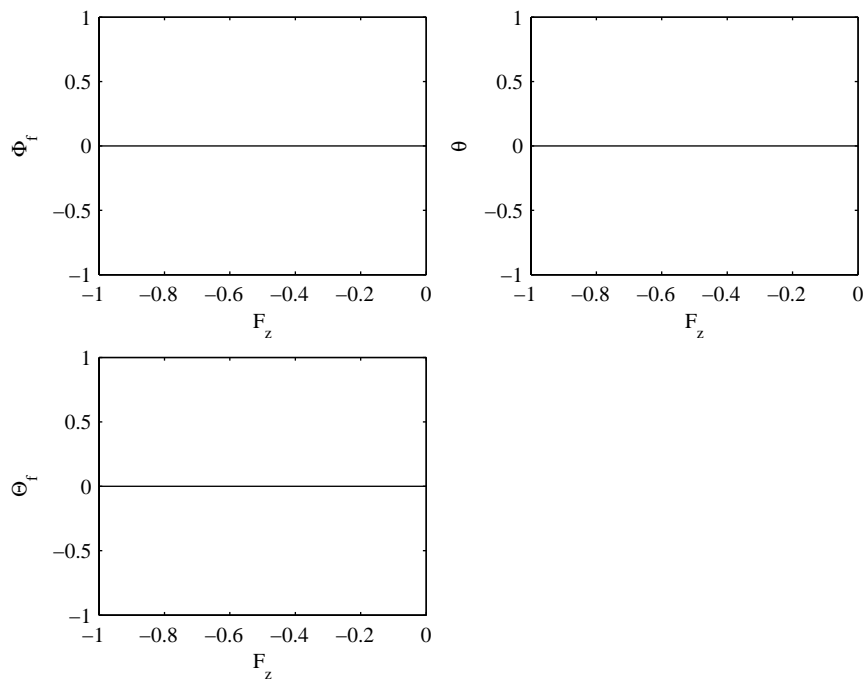


(Above, obtaining solutions for $F_x=1$ to $F_x=0$, using starting guess values $\theta=\Phi_f=\Theta_f=0$)

VARYING F_z

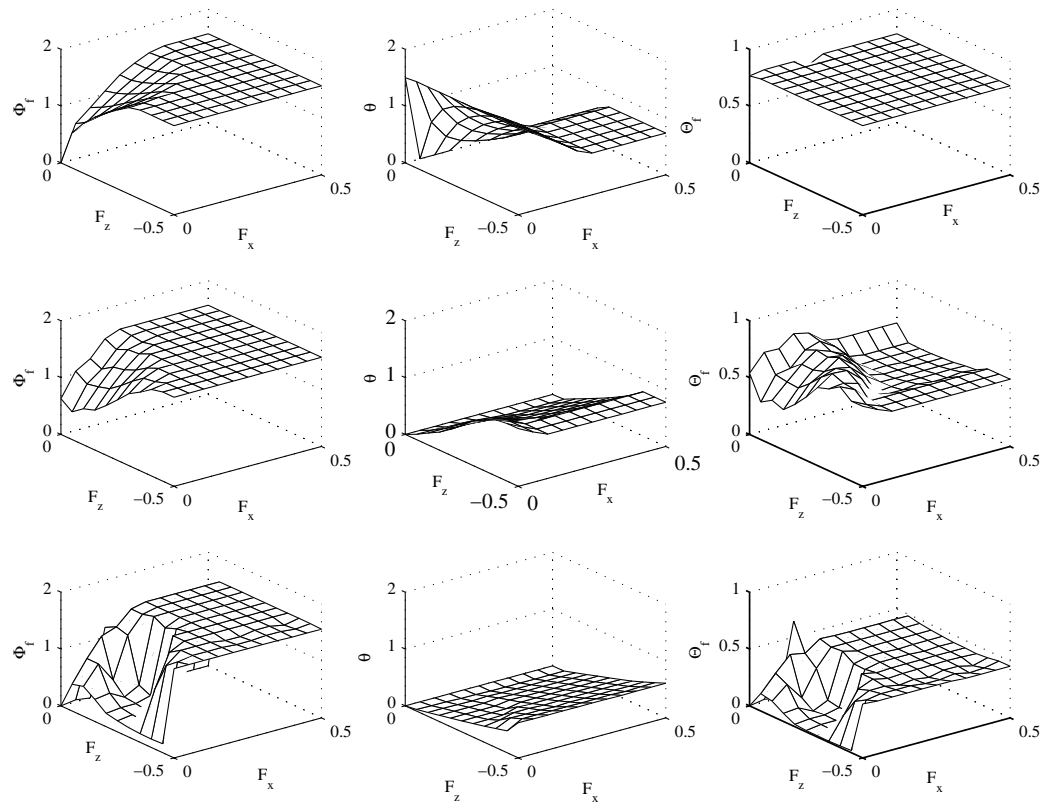


(Above, obtaining solutions for $F_z = -1$ to $F_z = 0$, using starting guess values $\theta = \pi/2$, $\Phi_f = \pi/2$, $\Theta_f = \pi/4$)



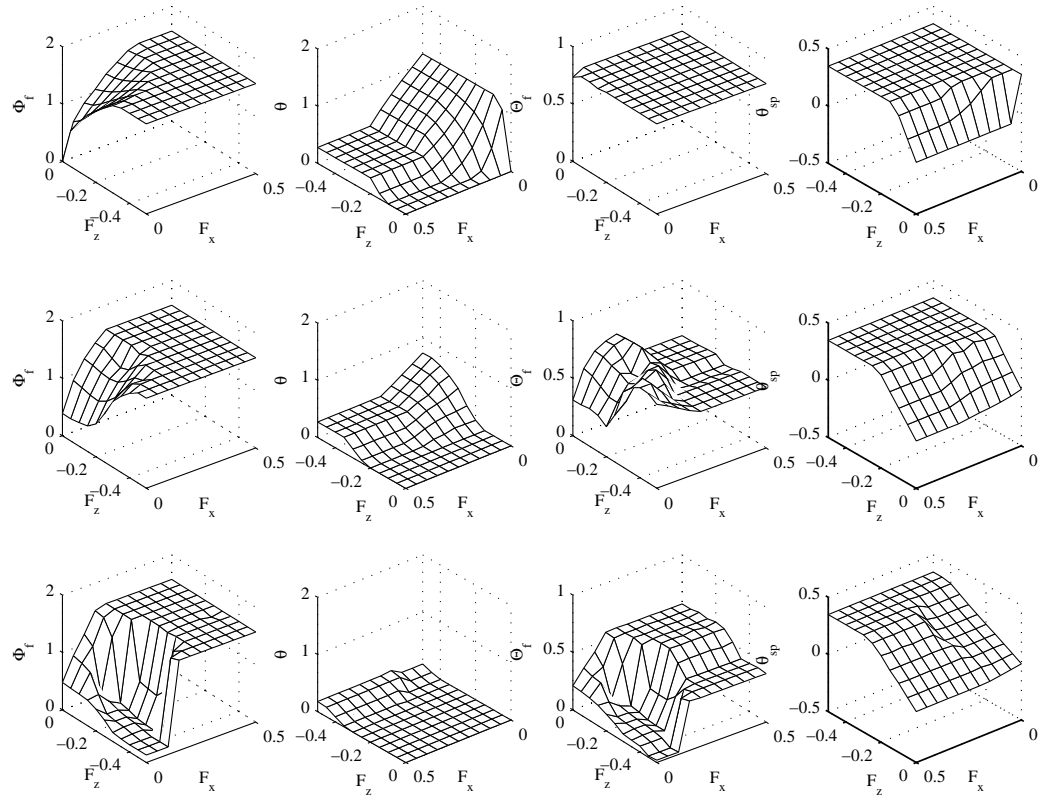
(Above, obtaining solutions for $F_z = -1$ to $F_z = 0$, using starting guess values $\theta = \Phi_f = \Theta_f = 0$)

APPENDIX 6 - FULLY POPULATED RESULTS FOR KINEMATIC MODEL 1



Populated lookup tables for model 1 solution, $V=0$ (top row), $V=0.5$ (middle row) and $V=0.9$ (bottom row)

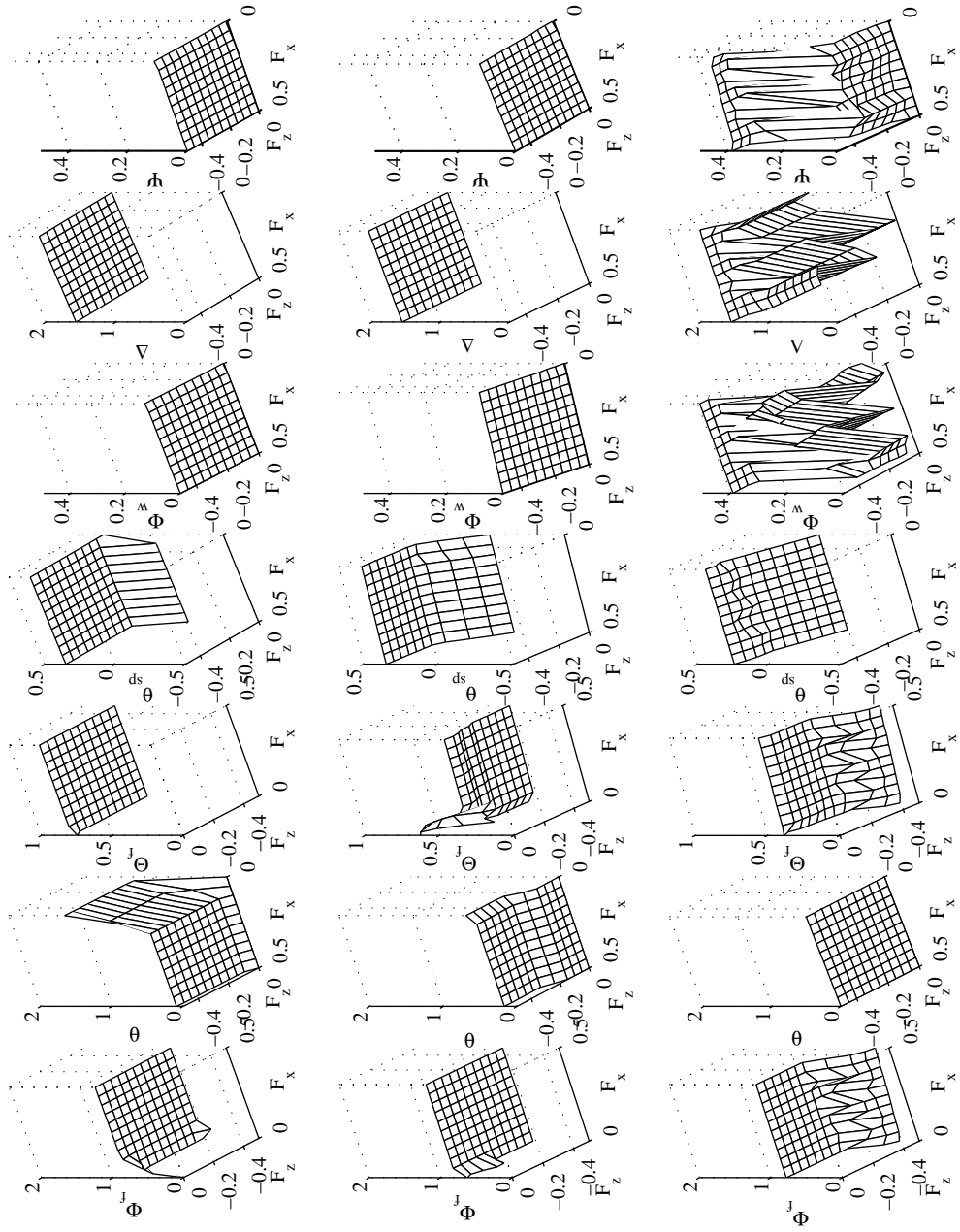
APPENDIX 7 – FULLY POPULATED RESULTS FOR KINEMATIC MODEL 3



Populated lookup tables for base-model solution, $V=0$ (top row), $V=0.5$ (middle row) and $V=0.9$ (bottom row)

APPENDIX 8 – FULLY POPULATED RESULTS FOR KINEMATIC MODEL 4

Populated lookup tables for model 4 solution, $V=0$ (top row), $V=0.5$ (middle row) and $V=0.9$ (bottom row)



BIBLIOGRAPHY

BAUMEL, J. (1993), *Handbook of Avian Anatomy: Nomina Anatomica Avium*. Cambridge, Ma: Harvard University Nuttall Ornithological Club.

ELLINGTON, C. P (1984), The Aerodynamics of Hovering Insect Flight. *Philosophical Transactions of the Royal Society of London*. **305**, 1-181.

FILIPPONE, A. (2005), *Advanced Topics in Aerodynamics*. Website. <http://aerodyn.org>. Accessed 1st August 2005.

HALL, K. C. & HALL, S. R. (2000), *A Rational Engineering Analysis of the Efficiency of Flapping Flight*. Notre Dame: IN.

KROEGER, R. (2003), *A Bug's Life Inspires Future Mocchi-Aircraft*, *Fluent News*. Website. http://www.fluent.com/about/news/newsletters/03v12i2_fall/a3.htm. Accessed 3rd August 2005.

LIGHTHILL, J. (1987), *Mathematical Biofluidynamics*. Philadelphia.: SIAM Society for Industrial & Applied Mathematics.

LIU, H. & KAWACHI, K. (2000), *Leading-Edge Vortices of Flapping and Rotary Wings at Low Reynolds Number*. Cited in Mueller, T. J. (2001), Fixed and Flapping Wing Aerodynamics for Micro Air Vehicle Application. Virginia: AIAA.

MITCHELL, M. (1998), *An Introduction to Genetic Algorithms*. Cambridge, MA: The MIT Press.

NEEF, M. F. & HUMMEL, D. (2001), *Euler Solutions for a finite-span flapping wing*. Cited in Mueller, T. J. (2001), Fixed and Flapping Wing Aerodynamics for Micro Air Vehicle Application. Virginia: AIAA.

RINGHAM, M. L. & HOUSE, D. H. (1997), Aerodynamic Bird Flight: A Physically-Based Approach to Behavioural Flocking. *Proceedings of CAD & Graphics*.

RAMAKRISHNANANDA, B. & WONG, K. C. (1999), Animated Bird Flight Using Aerodynamics. *The Visual Computer*. **15**, 494-508.

RAYNER, J. M. V & GORDON, R. (1998), Visualization and modelling of the wakes of flying birds. *Biona Report. 13, Motion Systems*, 165-173.

SMITH, M. J. C., WILKIN, P. J. & WILLIAMS, M. H. (1996), The Advantages of an Unsteady Panel Method in Modelling the Aerodynamic Forces on Rigid Flapping Wings. *Journal of Experimental Biology*, **(199)**, 11073-10831.

THE MATHWORKS, INC. (2002), MATLAB 6.5 Product Documentation.

VON KARMAN, T. & BURGERS, J. M. (1935), problems of non-uniform and curvilinear motion. *Aerodynamic Theory*. **2**, 304-310.

WEIS-FOGH, T (1956), Biology and physics of locust flight. II. Flight performance of the Desert Locust. *Philosophical Transactions of the Royal Society of London*. **239**, 459-520.

WU, J. & POPOVIĆ, Z. (2003), Realistic Modelling of Bird Flight Animations. *ACM Transactions on Graphics*. **22**, 888-895.