Longitudinal trimming of a helicopter in forward flight

Nicholas Cheong

APRIL 22, 2018

UNIVERSITY OF GLASGOW
Dr George Barakos

Contents

Nomenclature

a_0	Coning angle
a	Lift-slope of aerofoil
a_1	Rearward rotor shaft tilt or longitudinal disk tilt
A_1	Lateral cyclic pitch component
B_1	Longitudinal cyclic pitch component
A_M, B_M	Summations of moments of helicopter system
V	External or oncoming velocity
$L_{section}$	Lift at blade section
$L_{\psi=0}$	Lift at $\psi = 0$ where $V \approx 0$
θ	Fuselage attitude
θ_0	Collective pitch
ψ	Azimuthal position
L_F	Lift of fuselage
D_F	Drag of fuselage
M_F	Moment of fuselage
L_{100}	Lift at 100 m/s (wind tunnel data)
D_{100}	Drag at 100 m/s (wind tunnel data)
M_{100}	Moment at 100 m/s (wind tunnel data)
σ	Solidity ratio
M_H	Moment about hub
M_F	Moment about fuselage

W	Weight of helicopter
x_i	Coordinates in the x plane
z_i	Coordinates in the z plane
θ_F	Backwards tilt of rotor
a_s	Shaft tilt angle
λ_β	Blade flapping frequency
λ_i	Downwash
γ	Lock number
μ_i	Advanced ratio
C_T	Coefficient of thrust
ω	Angular velocity
R	Total rotor radius
c	Chord
I_β	Flapping inertia of the blade
A	Total rotor plane area

Objectives and report overview

The objective of this assignment was to gain insight into longitudinal trimming of a helicopter and obtain the collective, cyclic, fuselage attitude and disk tilt from the blade flapping equation of a semi-rigid rotor through the methodology described in chapter 2 and implementing it into the multi-paradigm computing software, Matlab.

The report is split into 3 main sections:

Chapter 1:

This is a brief overview of trimming for a helicopter and the importance of certain concepts such as blade flapping, cyclic and collective pitch in order to understand the numerical procedure of calculating the family of angles.

Chapter 2:

This is the overview of the procedure used for the calculating the family of angles, much of which is referred to from the insightful book, “The foundations of helicopter flight”, by Simon Newman. This is followed by a flow chart which outlines the procedure.

Chapter 3:

This is the result section, which will explore the validity of the results obtained by the procedure.

Chapter 1

Trimming: overview and important concepts

1.1 Introduction

Trim entails the balancing of forces and moments in all three axes commonly referred to as so-called wind axes which entail roll, pitch and yaw. This initially gives six degrees of freedom in translational and rotational on each axis and related plane. However for a helicopter, the forces can be coupled. For example, the longitudinal and pitch moment are coupled as a change in longitudinal force invokes forwards or backwards motion in conjunction with a change in pitching moment. This is also the cause for lateral force and rolling moment [1]. The result of this coupling is a resulting four degrees of freedom in which are controlled by longitudinal, lateral, yaw and vertical controls of the helicopter.

Furthermore, as the lateral or longitudinal control entail a change in the overall thrust vector of the rotor, coupling is present between the two controls and therefore this further constrains the degrees of freedom of the helicopter. The result of this is that no exact trim solution is derivable when considering trim simultaneously for both cases [1]. However, this report will focus on the solely on longitudinal and therefore uncoupled equations can be used to obtain an approximate solution.

1.2 Rotor blade flapping

Rotor blade flapping is significant to the stability and trim of helicopter flight as the blades advancing and receding into an external flow, where the external flow is synonymous to wind or the helicopter moving, causes a disbalance of lift on either side of the rotor plane [1]. This enacts a rolling moment onto the helicopter which is required to be counteracted to maintain stability and trim; rotor flapping interrupts the transfer of the moment to the fuselage by enacting the moment onto the blades instead of the fuselage. Consider a helicopter in edgewise flight where the rotor plane is parallel to the external flow direction, figure 1.1 displays the effects of local external flow velocity for sections along the a rotor blade rotating clockwise at a ratio of external flow to tip speed, the advanced ratio, of $\frac{V}{\omega R} = 0.375$, where ω is the rotational rate and R the radius of the rotor blade. $L_{section}/L_{\psi=0}$ is the ratio of the lift produced by the rotor blade with and without external flow, which is insightful as it depicts the influence of the external flow in relation to the rotational speed of the blade sections.

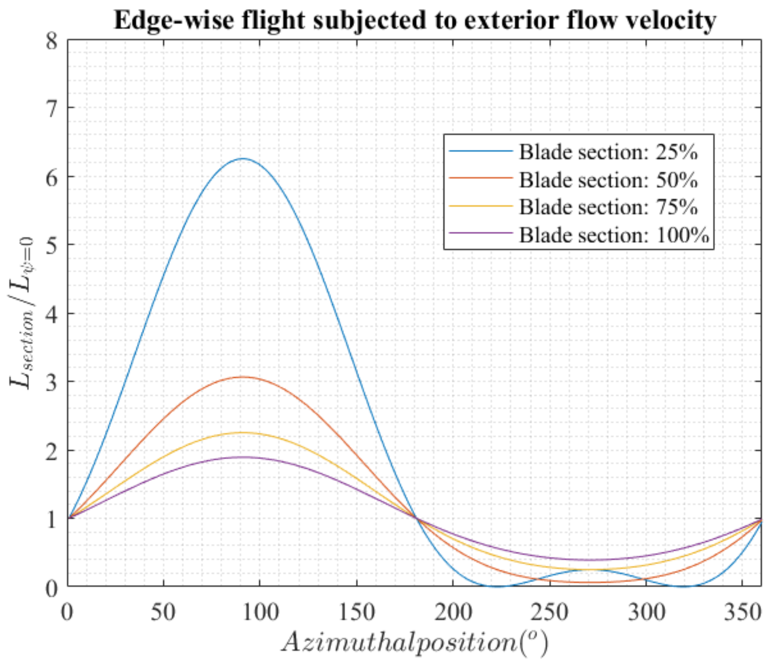

Figure 1.1 – Lift ratio $L_{section}/L_{\psi=0}$ for a rotor blade sections for different azimuthal positions subjected to an external flow at an advanced ratio, $\mu = 0.375$.

The unbalance of lift force on either side of the rotor plane ($0^o - 180^o, 180^o - 360^o$, for this example) is clearly depicted. Furthermore, the inboard blade sections are significantly more sensitive to the external flow velocity than the outboard and lift resurgence occurs at 270^o. This azimuthal position is where the maximum dynamic head occurs as the rotor is perpendicular to the flow direction; this also occurs on the opposite side of the rotor plane at 90^o, in which produces the opposite effect [1]. Therefore, the resurgence is due to the external flow overcoming the relative rotational velocity of the inner section which is significantly low due to the blade sections being near the centre of rotation. Figure 1.2 shows the disbalance of lift for $\frac{V}{\omega R} = 0.125$, which highlights the concept above as it is clear that the rotational velocity surpasses the slower external flow velocity. Additionally, if there is no external flow, this distribution becomes a horizontal, linear line with a value of $\frac{L_{section}}{L_{\psi=0}} = 1$.

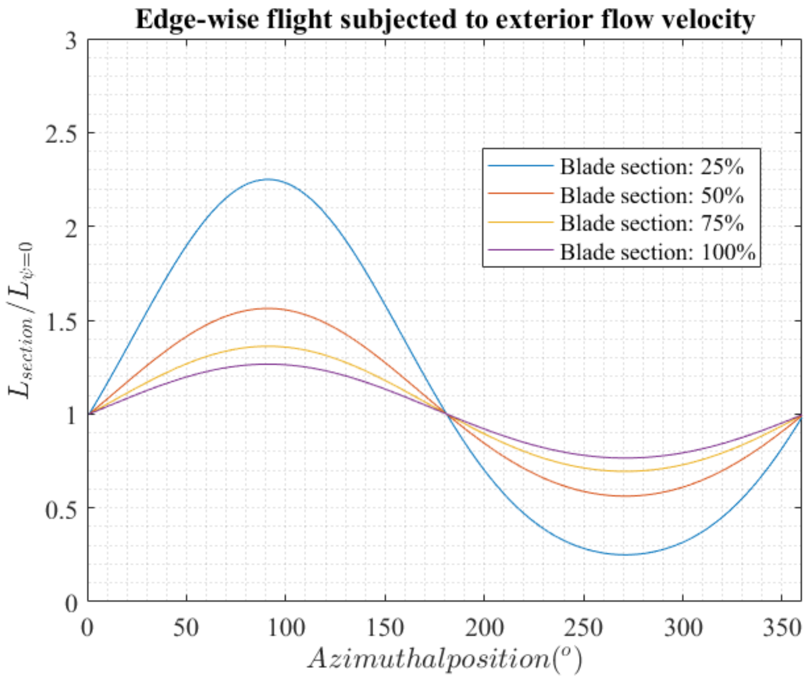

Figure 1.2 – Lift ratio $L_{section}/L_{\psi=0}$ for a rotor blade sections for different azimuthal positions subjected to an external flow at an advanced ratio, $\mu = 0.125$.

1.3 Cyclic and collective pitch

Further considering the unbalanced aerodynamic force and flapping motion, if rotating clockwise, the rotor plane has the maximum flapping acceleration upwards and downwards at 90^o and 270^o respectively due to the maximum dynamic head. The effect of this is an oscillatory motion where the blades begin to tilt upwards at $0-180^o$ and downwards at $180-360^o$, however, the time taken for the motion to complete must be considered; typically the rotor does not complete the upwards are downward motion till after approximately 90^o after the maximum upwards and downwards accelerations [1]. The significance of this is that the rotor will always tend into a backwards rotor disc tilt relative to the direction of the flight path, creating a positive pitching moment (positive for nose-up in forward flight) and shifting the thrust vector to oppose the desired direction. Helicopters typically tilt the rotor plane forward when in forward flight, which counteracts this effect as it shifts the thrust vector forwards, however, an additional means of counteraction is necessary as the blades will always oppose this motion and decelerate the helicopter [1]. This introduces cyclic pitch, which alters the aerodynamic forces locally over the rotor plane by varying the coefficient of lift depending on the azimuthal position of the blades. Cyclic pitch is set such that at the advancing side of the rotor plane, the pitch will decrease to reduce the aerodynamic forces, and the contrary on the receding side and thus flapping motion is reduced [1].

Additionally, pitch controls also entail the collective pitch of a helicopter, which is the base pitch angle of the rotor which is independent on azimuthal position and therefore, used for controlling the magnitude of the thrust vector [1]. The total pitch is the summation of the lateral and longitudinal cyclic pitch and the collective pitch in reference to the rotor plane.

$$\theta = \theta_o - A_1 sin\psi - B_1 cos\psi \tag{1}$$

Where θ_o is the collective pitch, A_1 and B_1 the cyclic pitch components applied laterally and longitudinally with respect to the fuselage, and ψ the azimuthal position [1].

Chapter 2

Methodology of calculating longitudinal trim

2.1 Establishing fuselage aerodynamics

Accounting for the aerodynamics of the fuselage, L_F, D_F and W are the fuselage lift, drag and weight respectively in which are calculated by reference values of the variables at 100 m/s.

$$\begin{bmatrix} L_F \\ D_F \\ M_F \end{bmatrix} = \frac{\sigma V^2}{10^4} \begin{bmatrix} L_{100} \\ D_{100} \\ M_{100} \end{bmatrix} \quad (2)$$

The reference values are typically set by wind tunnel tests of the fuselage and are taken at a single value of fuselage attitude; therefore, it is important to note that using these values is an assumption in which may become invalid if using significant values of fuselage attitude occur during any calculation procedure [1].

2.2 Establishing fuselage attitude

Fuselage attitude is the angle of the fuselage with respect to the wind axis. This is executed by using Newton Raphson's on equation (3) which is derived from the moment balance equation of helicopter:

$$A_M cos\theta + B_M sin\theta + M_H a_1 + M_F = 0 \quad (3)$$

Where a_1 is the rearward rotor shaft tilt (or longitudinal disk tilt), M_H,M_F the main rotor hub and fuselage moment respectively, and θ_F, α_s the backward tilt of the rotor thrust and shaft tilt angle [1].

$$a_1 = \theta_F - \theta - \alpha_s \tag{4}$$

$$tan\theta_F = \frac{D_F}{L_F - W} \tag{5}$$

A and B are the summation of moment about the rotor hub where $x_r, z_r, x_{cg}, z_{cg}, x_a$ and z_a are the coordinates of the rotor hub, centre of gravity and fuselage aerodynamic centre respectively [1].

$$A_M = W\left(x_{cg} - x_r\right) - L_F(x_a - x_r) - D_F(z_r - z_a) \tag{6}$$

$$B_M = -W\left(z_r - z_{cg}\right) - D_F(x_a - x_r) + L_F(z_r - z_a) \tag{7}$$

Newton Raphson's method is a numerical procedure of solving non-linear equations and the general equation for this method is [2]:

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})} \tag{8}$$

$$x: \quad f(x) = 0 \tag{9}$$

Where here the dash superscript indicates the derivative. The method is solved iteratively [2], where convergence can be deemed by comparing the values of proceeding and current iteration. This method in terms of the fuselage attitude equation surmounts to:

$$\theta_i = \theta_{i-1} - \frac{f(\theta_{i-1})}{f'(\theta_{i-1})} \tag{10}$$

An initial guess (θ_{i-1}) is required which should be a reasonable as the method will not converge if the absolute value of the initial condition is to high relative to the final solution. An initial guess of $\theta = 1^o$ was used.

2.3 Blade flapping equation

The blade flapping equation is used to determine the flapping behaviour of a rotor blade and is derived from the aerodynamic and centrifugal forces enacted onto the blade by the external flow and rotation [1]. The resultant flapping equation is displayed below:

$$\begin{bmatrix} \frac{\gamma}{8} & 0 & -\frac{\gamma}{6}\mu_x & -\lambda_\beta^2 \\ 0 & -\frac{\gamma}{8} & 0 & -\frac{\gamma}{6}\mu_x \\ \frac{\gamma}{3}\mu_x & 0 & -\frac{\gamma}{8} & 0 \\ \frac{1}{3} & 0 & -\frac{\mu_x}{2} & 0 \end{bmatrix} \cdot \begin{bmatrix} \theta_0 \\ A_1 \\ B_1 \\ a_o \end{bmatrix} = \begin{bmatrix} \frac{\gamma}{6}\mu_{zD} \\ (1-\lambda_\beta^2)a_1 - \frac{\gamma}{8}b_1 \\ \frac{\gamma}{8}a_1 + (1-\lambda_\beta^2)b_1 \\ \frac{C_T}{sa} + \frac{\mu_{zD}}{2} \end{bmatrix} \tag{11}$$

Where:

(2.3.1) a_0 is the coning angle,

(2.3.2) μ_i the advanced ratios on the disk axis,

(2.3.3) λ_β the flapping frequency

(2.3.4) γ the Lock number

(2.3.5) s and a the solidity and lift slope

b_1 is the disk tilt advancing side down (lateral side) which is set to zero for solely longitudinal analysis of the trim state. From this equation, the cyclic, collective, and cone angles can be calculated [1].

2.3.1 *Coning angle*

The coning angle is where the rotor blades on both sides of the rotor plan tilt upwards by a_0, forming a dihedral rotor (or downwards for an anhedral.) Likewise to the cyclic pitch, this feature is useful as it counteracts any rolling motion by tilting the thrust vectors of each blade inward and further, decreases the resultant lift on the rising side.

2.3.2 *Advanced ratios*

The advance ratio is where the external flow velocity, V, is normalized by the tip speed, ωR. μ_x and μ_z is the advanced ratio formed by the x and z components of V and μ_{zD} is formed with total resultant z velocity by including the effects of downwash, λ_i.

$$\mu_x = \frac{V_x}{\omega R} \quad (12) \qquad \mu_z = \frac{V_z}{\omega R} \quad (13) \qquad \mu_{zD} = \mu_z + \lambda_i \quad (14)$$

Where the downwash equates as:

$$\lambda_i = \frac{C_T}{4}\frac{1}{\sqrt{\mu_x^2+(\mu_{zD})^2}} \quad (15)$$

Which can be solved iteratively using Newton Raphson's method likewise to the fuselage pitch attitude. The advanced ratios must also be calculated in disk axis as the downwash is within this coordinate system [1]. The x and z components of velocity for disk axes are:

$$V_x = V cos(\alpha_s + \theta + a_1) \quad (16)$$

$$V_z = V sin(\alpha_s + \theta + a_1) \quad (17)$$

This results in the final equation for the advanced ratios which can be implemented into the blade flapping equation [1]:

$$\mu_{xDisk} = \mu\cos(\alpha_s + \theta + a_1) \tag{18}$$

$$\mu_{zDisk} = -\mu\sin(\alpha_s + \theta + a_1) \tag{19}$$

Where the total advanced ratio in the z direction now equivalent to, $\mu_{zD} = \mu_{zDisk} + \lambda_i$.

2.3.3 Natural flapping frequency

Firstly, a semi-rigid rotor is where the bending is modelled using a torsional spring placed at the flap hinge to simulate an arbitrary dynamic model. The natural flapping frequency is a normalized dynamic parameter of the blade which relates the flapping equation to the theoretical stiffness normalized by the flapping inertia and rotational speed [1].

$$\lambda_\beta^2 = \frac{k_\beta}{I_\beta \Omega} \tag{20}$$

Where k_β, I_β and Ω are the flapping spring stiffness, inertia and rotational speed respectively.

2.3.4 Lock number

The significance of the Lock number is that it also relates the aerodynamics to the dynamics of a rotor when considered in the flapping equation in a normalized form [1]. The equation for the Lock number is comprised of the ratio of aerodynamic to inertial forces, where *c* is the chord.

$$\gamma = \frac{\rho a c R^4}{I_\beta} \tag{21}$$

2.3.5 Solidity and lift slope

The solidity of a rotor is the ratio of the effective lift area (blade area) over total rotor plane area. Typically for momentum theory, the coefficient of thrust is dependent on the total rotor plane area; by dividing the coefficient of thrust by the solidity this changes this dependence from rotor plane to rotor blade area [1]. The equation for solidity is displayed below, where N is the number of blades and A is the total rotor plane area.

$$s = \frac{A_{Blade(s)}}{A_{Rotor}} = \frac{NcR}{A} \tag{22}$$

The lift slope is the ratio of the coefficient of lift and inflow angle. For this report, thin aerofoil theory can be used giving a lift slope of 2π, however, experiments have shown that the value is typically $5.5 - 6\ /rad$ in which will be used [3]. Notably, this is an assumption and further, the lift slope is assumed to be constant over all variants of inflow angles; therefore, the results will be incorrect if high angle of attacks occur.

$$C_l = a\alpha \tag{23}$$

Where a is the lift slope and α the total angle of attack

2.4 Numerical implementation

The methodology described was executed on Matlab and the flow chart depicts the code in figure 2.1. The procedure was tested with case data displayed in table 2.1 below:

Table 2.1 – Case 1 helicopter parameters

Parameters	Values
$[x_{cg}, z_{cg}]$	[0,0]
$[x_r, z_r]$	[0,2]
$[x_a, z_a]$	[0,1]
L_{100}	0 (N)
D_{100}	12,000 (N)
M_{100}	0 (Nm)
$C_{L\alpha}$	5.75
N	5
λ_β	1
I_β	2400 $(kg\ m^3)$

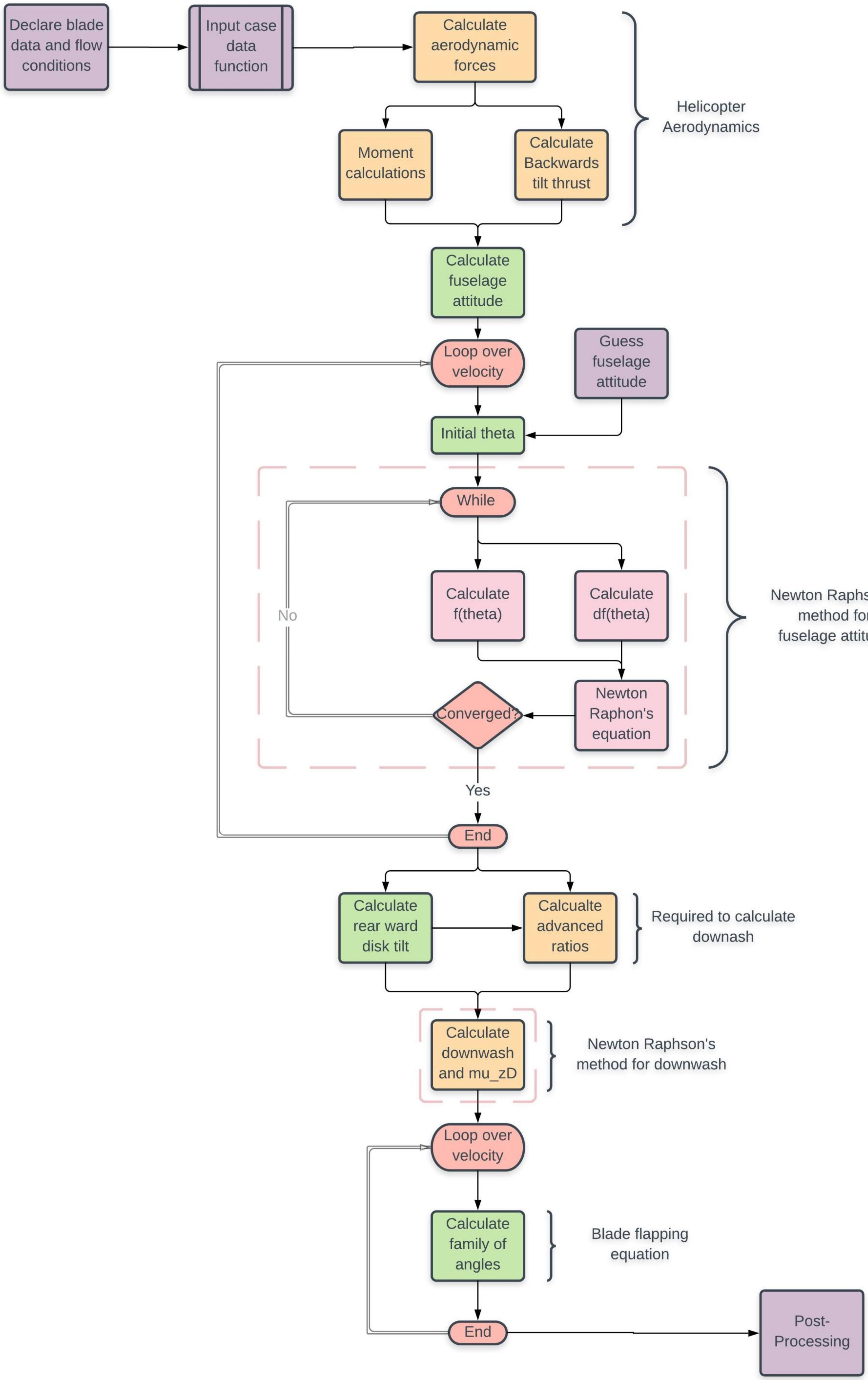

Figure 2.1 – Flow chart outline the procedure executed on Matlab for longitudinal trim

Chapter 3

Results of the longitudinal trim procedure

The results for case 1 can be seen in figure 3.1 below:

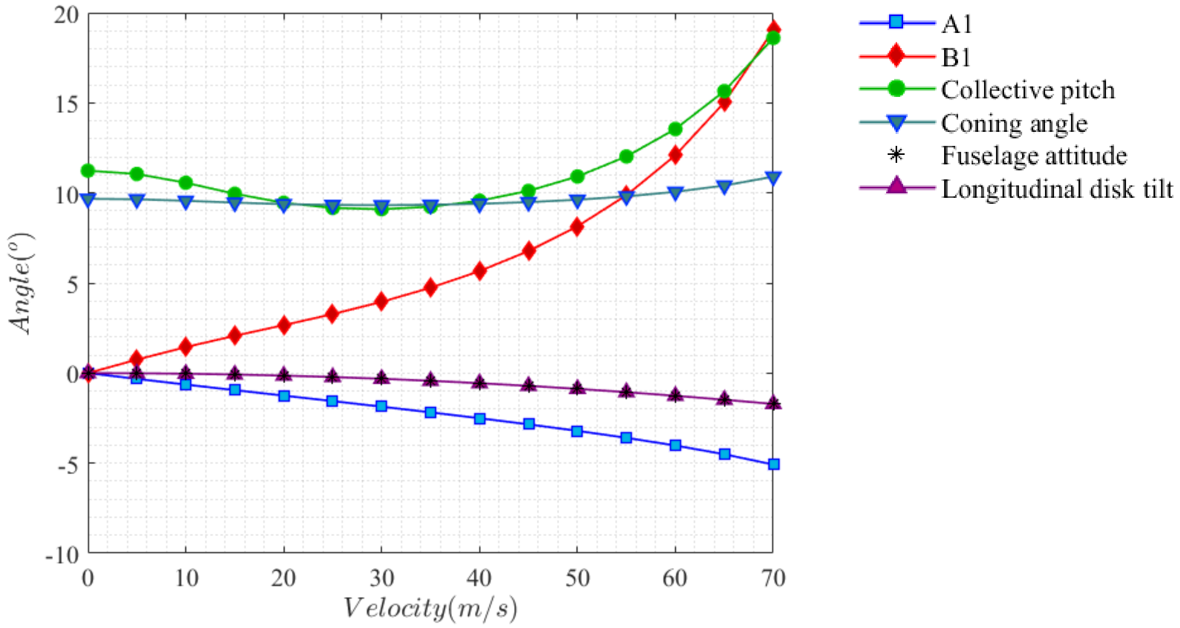

Figure 3.1 – Family of angles over a range of velocity from $0 \rightarrow 70\ m/s$

From the graph, the behaviour of the trim state is as follows:

- The collective pitch initially decreases from the initial value which is required to maintain hover. This is due to the external flow velocity increasing and therefore yielding a larger dynamic head and therefore lift. However, pitch increases due to the downwards motion of the fuselage attitude; this leads to external velocity acting more in the z component of the resultant velocity vector which decreases lift [1].
- The fuselage attitude decreases giving a nose down motion as the forward flight speed increases from $0^o \rightarrow -1.7^o$.
- The longitudinal cyclic pitch, B_1, increases to counteract the drag due to the increase in fuselage attitude [1].
- The lateral cyclic pitch, A_1, decreases to counteract the effects of the lateral disk tilt from the coning angle [1].

3.1 Validity considerations

The fuselage attitude is zero initially, however, at higher speeds this angle becomes an absolute value of approximately 1.7^o. This will result in a different fuselage aerodynamic force as the reference values are set at a different angle, likely 0^o. Although this force is likely to be small, any unbalanced force within a supposed equilibrium system will have effect over time and therefore can be deemed significant if the helicopter was moving forwards for a large period. The assumption that the lift slope was constant is also in question if the blades are near or in the stall region. Consider equation 1 which depicts the total pitch angle in the rotor plane:

$$\theta = \theta_o - A_1 sin\psi - B_1 cos\psi \qquad (1)$$

When the blades are at an azimuthal position of 180^o, the resultant value of pitch will be $\theta = \theta_o + B_1$, which at a forward velocity of 70 m/s, it is clear that the blades will be in the stall region (see figure 3.2). If the blades stall it will result in a significant reduction in lift on the front side of the rotor plane, which in turn, will produce a pitch moment and therefore will not be trimmed. This indicates that assuming a constant value of lift slope is not a safe assumption, and more complex models should be used to account for blade stall. Furthermore, it is an indication of the limitations for helicopter forward flight speeds.

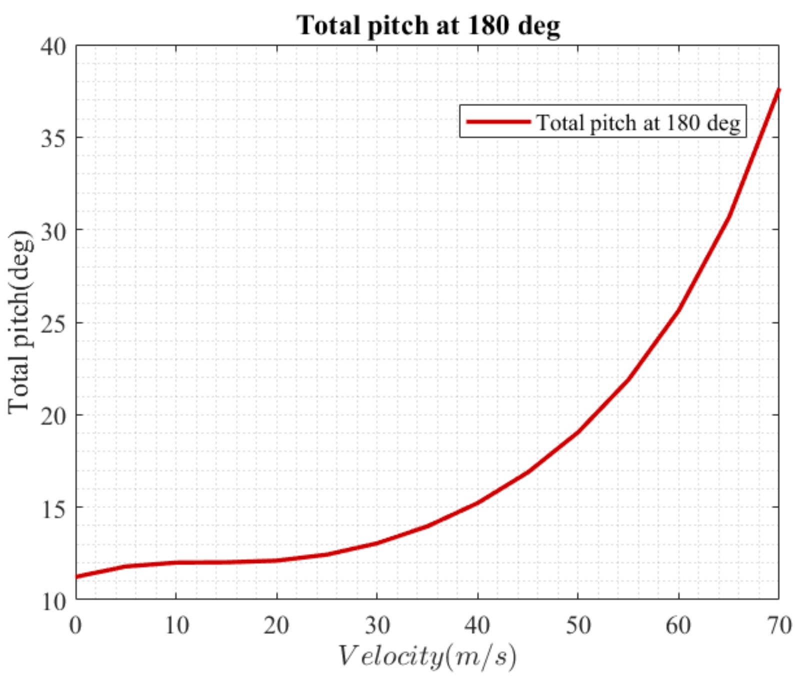

Figure 3.2 – Total pitch at an azimuthal position of $\psi = 180^o$ indicating blade stall at this region.

3.2 Comparing different fuselage drags

The comparison of fuselage drag was between a D_{100} of 12,000 and 10,000 respectively, where the former is the results above. The results for the fuselage attitude, disk tilt and collective pitch are displayed in figures 3.3 → 3.5.

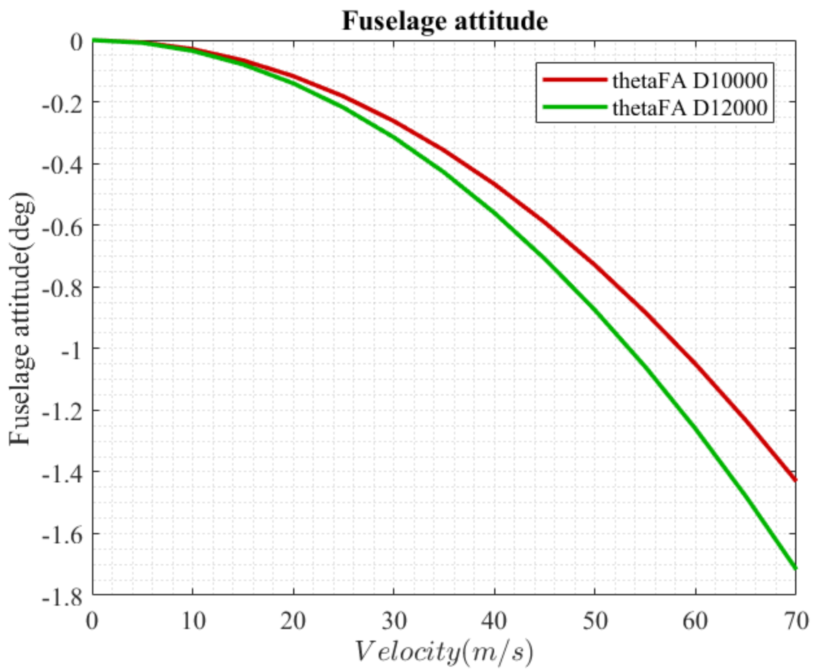

Figure 3.3 – Fuselage attitude comparison between different fuselage drags, $D_{100} = 12{,}000$ and $10{,}000$.

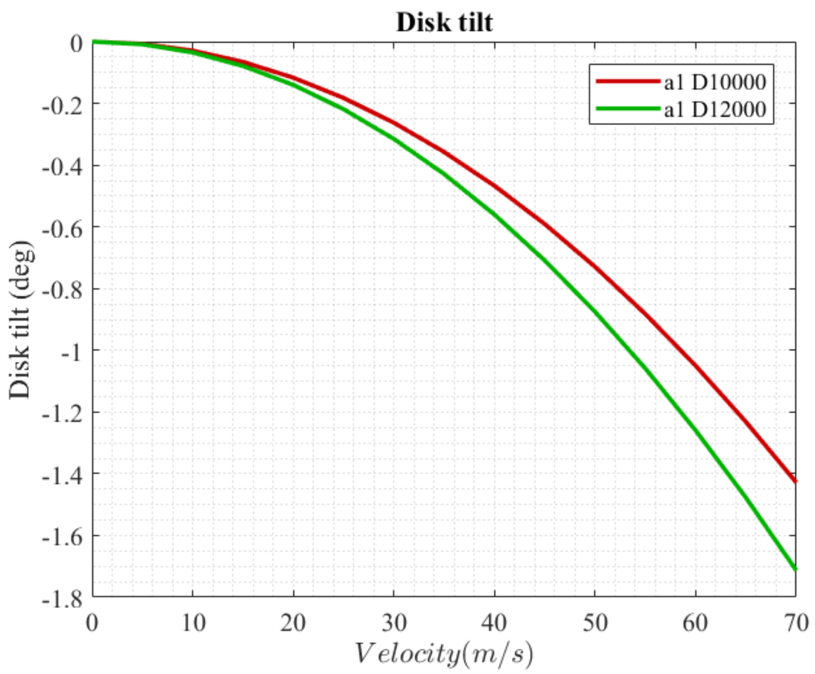

Figure 3.4 – Disk tilt comparison between different fuselage drags, $D_{100} = 12{,}000$ and $10{,}000$.

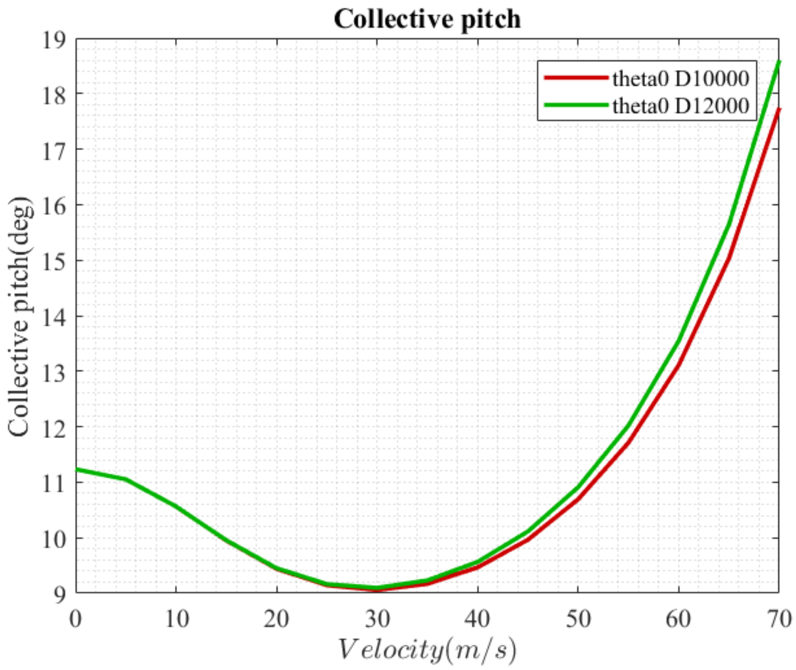

Figure 3.5 – Collective comparison between different fuselage drags, $D_{100} = 12{,}000$ *and* $10{,}000$.

With a decrease in fuselage drag, all angles decreased. Notably, the absolute fuselage attitude decreased by 16.6 % which is significant as it reduces the approximation error of the empirical aerodynamic assumptions; additionally, with a fuselage drag of zero the fuselage attitude surmounts to zero.

The reduction in fuselage attitude is the reason for the decrease in all the angles such as the disk tilt and collective pitch. This increases the size of the flight envelope the helicopter can fly in by increasing the maximum allowable forwards speed which is due to the lower required angle of attacks for trim state. Furthermore, the decrease in disk tilt is more efficient for forward flight as the thrust vector is driven forwards which results in less opposing force.

4. References

[1] S. Newman, The Foundations of Helicopter flight, Oxford: Elsevier Science, 1994, pp. 144,150,79,80,111,157,152,158.

[2] T. J. Ypma, “Historical Development of the Newton-Raphson Method,” *Society for Industrial and Applied Mathemati*, vol. 37, no. 4, pp. 531-551, 1995.

[3] C. Wright, Introduction to Aicraft Aeroelasticity and loads, Chichster: John Wiley & Sons ltd., 2007, p. 75.

5. Appendices

5.1 Main code

```
%% -------- ROTORCRAFT ASSIGNMENT -----------------
% Nicholas Cheong


close all; clear; clc
%% Controls
CTRL.casetype = '1';          % case '1','2','3'.
CTRL.incD     = '12000';      % '12000', '10000', '0'    (Drag)
CTRL.inittheta      = 1;      % Initial fuselage attitude for newton raphson
CTRL.initlambda_i   = 0;      % Initialize downwash, zero for zero velocity at start

%% Establish helicopter data

% Get case data
[WV] = rotorcraftinput(CTRL);

% Aerodynamics
L100        = 0;          % Lift at 100 m/s
M100        = 0;          % Moment at 100 m/s
V           = 70:-5:0;    % Velocity
sigma       = 1;          % Density ratio

switch CTRL.incD
    case '20000'
    D100    = 20000;
    case '12000'
    D100    = 12000;      % Drag at 100 m/s #
    case '10000'
    D100    = 10000;
    case '0'
    D100    = 0;
    case '5000'
    D100    = 5000;
    case '500'
    D100    = 500;
end

% Rotor blade parameters
Cla         = 5.75;      % Lift slope 5.5 - 6 from experimental
omega       = 20;        % Angular velocity (rad/s0
N           = 5;          % Number of blades
R           = 10;        % Radius
Vt          = omega*R;   % Tip speed (m/s)
alpha_s     = 0;         % Shaft tilt angle
c           = 0.5;       % Chord

gamma       = sigma*Cla*c*(R^4)/WV.I_b; % Lock number
K_b         = 0;                        % Spring stiffness
Mh          = 0.5*N*K_b;                % Main rotor hub moment per unit disc tilt

% Calculate fuselage lift, drag and moment
Lf = sigma*(L100.*(V.^2))/10000;
Df = sigma*(D100.*(V.^2))/10000;
Mf = sigma*(M100.*(V.^2))/10000;

% Moments about reference (rotor hub)
A =  WV.W*(WV.Xcg - WV.Xrh) - Lf*(WV.Xa - WV.Xrh) - Df*(WV.Zrh - WV.Za);
B = -WV.W*(WV.Zrh - WV.Zcg) + Lf*(WV.Zrh - WV.Za) - Df*(WV.Xa - WV.Xrh);
```

```
%% Fuselage attitude calculations with Newton Raphson's method
% Initializing while loop and theta
j = 2;
conv = 0;                               % Setting convergence factor for while loop
theta(1,1) = CTRL.inittheta*pi/180;     % Initial guess for theta

% Calculate equation components
theta_f = atan(Df./(Lf - WV.W));             % Backwards tilt thrust

for i = 1:length(V)

    if i > 1
    theta(i,1) = theta(i-1,end);
    end

    while conv < 1
        fun_theta(i,j)      = A(i)*cos(theta(i,j-1)) + B(i)*sin(theta(i,j-1)) +
Mh*(theta_f(i) - theta(i,j-1) - alpha_s) + Mf(i);
        dfun_theta(i,j)     = -A(i)*sin(theta(i,j-1)) + B(i) * cos(theta(i,j-1)) -
Mh;

        theta(i,j)          = theta(i,j-1) - fun_theta(i,j)/dfun_theta(i,j);

        % Convergence condition
        if j > 20 && abs(theta(i,j)-theta(i,j-1)) < 10^-3
            conv = 1;
            thetaFA(1,i) = theta(i,j);
        end
        j = j + 1;
    end

    % Reset while loop conditions!
    conv = 0;
    j = 2;
end

%% Finding downwash with Newton Raphson's method

% Initialize while loop and lambda
j = 2;
conv = 0;
lambda_i(1,1) = CTRL.initlambda_i;

% Establish advanced ratios and coefficient of thrust.
Ct      = repmat((WV.W/(0.5*sigma*(Vt^2)*pi*(R^2))),1,length(V));   % Coefficient
of thrust
a1      = theta_f - thetaFA - alpha_s;                              % Longitudinal
disk tilt
b1      = zeros(length(V),1);                                       % Lateral disk
tilt
mu      = V/Vt;                                                     % Advance ratio
mu_xd   =  mu.*cos(thetaFA + alpha_s + a1);                         % x ADV in disk
axes
mu_zd   = -mu.*sin(thetaFA + alpha_s + a1);                         % z ADV in disk
axes

for i = 1:length(V)

    if i >1
        lambda_i(i,1) = lambda_i(i-1,end); % Subsequent velocities use last value
as guess
    end

    while conv < 1
        fun_lambda_i(i,j)   = lambda_i(i,j-1) - (0.25*Ct(i))/((mu_xd(i)*mu_xd(i)...
                              + (mu_zd(i)+lambda_i(i,j-1))*(mu_zd(i)+lambda_i(i,j-
1)))^(1/2));
```

```
        dfun_lambda_i(i,j)  = 1 + (0.25*Ct(i)*(mu_zd(i)+lambda_i(i,j-
1)))/((mu_xd(i)...
                                  *mu_xd(i)) + ((mu_zd(i)+lambda_i(i,j-
1))*(mu_zd(i)+lambda_i(i,j-1))))^1.5;

        lambda_i(i,j)        = lambda_i(i,j-1) -
fun_lambda_i(i,j)/dfun_lambda_i(i,j);

        if j > 20 && abs(lambda_i(i,j)-lambda_i(i,j-1)) < 10-3
            conv = 1;
            lambda_iF(1,i) = lambda_i(i,j);
        end
        j = j + 1;
    end

    j = 2;
    conv = 0;
end


%% Calculating blade flapping equation

% Advanced ratio in z WITH downwash
mu_zD = lambda_iF + mu_zd;
% Solidity
sol   = (N*c*R)/(pi*R^2);

for i = 1:length(V)
    % Right hand side of blade flapping equation
    RHSM          = [gamma*mu_zD(i)/6;
                     (1-(WV.lam_b^2))*a1(i) - gamma*b1(i)/8;
                     (gamma*a1(i)/8) + (1 - (WV.lam_b^2))*b1(i);
                     (Ct(i)/(sol*Cla))+0.5*mu_zD(i)];

    % Left hand side
    LHSM              = [ gamma/8             0       -gamma*mu_xd(i)/6       -
(WV.lam_b^2);
                          0                -gamma/8          0                -
gamma*mu_xd(i)/6;
                      gamma*mu_xd(i)/3        0           -gamma/8
0;
                          1/3                 0        -mu_xd(i)/2
0             ];

    % Evaluate for family of angles
    BFEM(:,:,i) = LHSM\RHSM;
end

% Squeeze function used to extract data from N dimension of matrice
A1       = squeeze(BFEM(2,1,:));         % A1 lateral cyclic pitch
B1       = squeeze(BFEM(3,1,:));         % B1 longitudinal cyclic pitch
theta0   = squeeze(BFEM(1,1,:));         % Collective pitch
a0       = squeeze(BFEM(4,1,:));         % Coning angle

figure
set(gcf,'color','white')
plot(V,A1*180/pi,'b-s','LineWidth',1,'MarkerFaceColor',[0 0.7
0.9],'MarkerEdgeColor',[0 0 1]);     hold on
plot(V,B1*180/pi,'r-d','LineWidth',1,'MarkerFaceColor',[0.8 0
0],'MarkerEdgeColor',[1 0 0]);     hold on
plot(V,theta0*180/pi,'-o','color',[0 0.7 0],'LineWidth',1,'MarkerFaceColor',[0 0.7
0],'MarkerEdgeColor',[0 0.8 0]); hold on
plot(V,a0*180/pi,'-v','color', [0.2 0.5 0.5],'LineWidth',1,'MarkerFaceColor',[0.2
0.5 0.5],'MarkerEdgeColor',[0 0.2 1]);     hold on
plot(V,a1*180/pi,'-^','color',[0.5 0 0.5],'LineWidth',1,'MarkerFaceColor',[0.7 0
0.5],'MarkerEdgeColor',[0.5 0 0.5])
plot(V,thetaFA*180/pi,'k*'); hold on
xlabel('$$Velocity (m/s)$$','Interpreter', 'Latex')
```

```
ylabel('$$Angle (^o)$$','Interpreter', 'Latex')
set(gca,'FontName','Times New Roman','FontSize',12);
legend('A1','B1','Collective pitch','Coning angle','Fuselage
attitude','Longitudinal disk tilt')
grid minor

a0_D10 = a0;
a1_D10 = a1;
theta0_D10 = theta0;
A1_D10 = A1;
B1_D10 =B1;
thetaFA_D10=thetaFA;
```

5.2 Input function

```
function [WV] = rotorcraftinput(CTRL)

switch CTRL.casetype
    case '1'
% Inertial
    WV.Xcg      = 0;         % X coordinate of CG
    WV.Zcg      = 0;         % Z coordinate of CG
    WV.Xrh      = 0;         % X coordinate of hug
    WV.Zrh      = 2;         % Z coordinate of hub
    WV.Xa       = 0;         % X coordinate of aerodyn centre of fuse
    WV.Za       = 1;         % Z coordinate of aerodyn centre of fuse
    WV.lam_b    = 1;         % Blade flapping Frequency
    WV.I_b      = 2400;      % Blade Flapping inertia of blade (kgm^2)
    WV.m        = 10000;     % Mass
    WV.g        = 9.81;      % Gravity
    WV.W        = WV.m*WV.g;  % Weight

    case '2'
    WV.Xcg      = 0.5;       % X coordinate of CG
    WV.Zcg      = 0;         % Z coordinate of CG
    WV.Xrh      = 0;         % X coordinate of hug
    WV.Zrh      = 2;         % Z coordinate of hub
    WV.Xa       = 0;         % X coordinate of aerodyn centre of fuse
    WV.Za       = 1;         % Z coordinate of aerodyn centre of fuse
    WV.lam_b    = 1;         % Blade flapping Frequency
    WV.I_b      = 2400;      % Blade Flapping inertia of blade (kgm^2)
    WV.m        = 10000;     % Mass
    WV.g        = 9.81;      % Gravity
    WV.W        = WV.m*WV.g;  % Weight

    case '3'
    WV.Xcg      = 0;         % X coordinate of CG
    WV.Zcg      = 0;         % Z coordinate of CG
    WV.Xrh      = 0;         % X coordinate of hug
    WV.Zrh      = 2;         % Z coordinate of hub
    WV.Xa       = 0;         % X coordinate of aerodyn centre of fuse
    WV.Za       = 1;         % Z coordinate of aerodyn centre of fuse
    WV.lam_b    = 1.5;       % Blade flapping Frequency
    WV.I_b      = 2400;      % Blade Flapping inertia of blade (kgm^2)
    WV.m        = 10000;     % Mass
    WV.g        = 9.81;      % Gravity
    WV.W        = WV.m*WV.g;  % Weight


end

end
```

5.3 Code for calculating lift for a helicopter in edge-wise flight

```
% Why flapping is required
% edgwise rotor to wind

clear; clc; close all;
a            = 5.75; % Lift slope
rho          = 1.225;
Uw           = 10;
c            = 1;
R            = 1:1:4;
omega        = 20;
TSR         = omega*R(end)/Uw;
alpha        = 2*pi/180;


% Initilize rotor
psi  = 0;    % Azimuthal position. Start from 0 being parralell to tail.
NCYC = 1;
azi_inc = 1*pi/180;     % 1 degree azimuthal increment
azi_ind = (NCYC*2*pi)/azi_inc;

for i = 1:azi_ind


  for j = 1:length(R)   % Loop over blade increment

      Ul = omega*R(j) + Uw*sin(psi);
      L(j,i) = 0.5*rho*c*(Ul^2)*a*alpha;
      if psi == 0
          Lat0(j,i) = L(j,i);
      end
  end  % End of blade incremeners

  psi = psi + azi_inc;
  psistore(i) = psi;
  if psi >= 2*pi
      psi = 0;
  end
end

figure
set(gcf,'color','white')
plot(psistore*180/pi,L(:,1:end)./Lat0)
xlabel('$$Azimuthal  position ({^o})$$','Interpreter', 'Latex')
ylabel('$$L_{section}/L_{\psi = 0}$$','Interpreter', 'Latex')
title('Edge-wise flight subjected to exterior flow velocity')
axis([0 360 0 3])
legend('Blade section: 25%','Blade section: 50%','Blade section: 75%', 'Blade
section: 100%')
set(gca,'FontName','Times New Roman','FontSize',12);
grid minor
```

5.4 Code for comparing D12,000 and D10,000

Note: data files have been provided in the zip file.

```
clear; clc; close all

% Comparison code as differences are small
V = 70:-5:0;

% Load data
load('a0_d12.mat')
load('aa1_D12.mat')
load('A1_D12.mat')
```

```
load('B1_D12.mat')
load('theta0_D12.mat')
load('thetaFA_D12.mat')
load('a0_D10.mat')
load('aa1_D10.mat')
load('A1_D10.mat')
load('B1_D10.mat')
load('theta0_D10.mat')
load('thetaFA_D10.mat')


figure
set(gcf,'color','w')
plot(V,a1_D10*180/pi,'color',[0.8 0 0],'LineWidth',2); hold on
plot(V,a1_D12*180/pi,'color',[0 0.7 0],'LineWidth',2)
xlabel('$$Velocity (m/s)$$','Interpreter', 'Latex'); ylabel('Disk tilt (deg)')
title('Disk tilt')
legend('a1 D10000','a1 D12000')
set(gca,'FontName','Times New Roman','FontSize',12);
grid minor


figure; set(gcf,'color','w')
plot(V,A1_D10*180/pi,'color',[0.8 0 0],'LineWidth',2); hold on
plot(V,A1_D12*180/pi,'color',[0 0.7 0],'LineWidth',2)
title('Lateral cyclic pitch')
xlabel('$$Velocity (m/s)$$','Interpreter', 'Latex'); ylabel('Lateral cyclic pitch
(deg)')
legend('A1 D10000','A1 D12000')
set(gca,'FontName','Times New Roman','FontSize',12);
grid minor

figure; set(gcf,'color','w')
plot(V,B1_D10*180/pi,'color',[0.8 0 0],'LineWidth',2); hold on
plot(V,B1_D12*180/pi,'color',[0 0.7 0],'LineWidth',2)
title('Longitudinal cyclic pitch')
xlabel('$$Velocity (m/s)$$','Interpreter', 'Latex'); ylabel('Longitudinal cyclic
pitch (deg)')
legend('B1 D10000','B1 D12000')
set(gca,'FontName','Times New Roman','FontSize',12);
grid minor

figure; set(gcf,'color','w')
plot(V,thetaFA_D10*180/pi,'color',[0.8 0 0],'LineWidth',2); hold on
plot(V,thetaFA_D12*180/pi,'color',[0 0.7 0],'LineWidth',2)
title('Fuselage attitude')
xlabel('$$Velocity (m/s)$$','Interpreter', 'Latex'); ylabel('Fuselage
attitude(deg)')
legend('thetaFA D10000','thetaFA D12000')
set(gca,'FontName','Times New Roman','FontSize',12);
grid minor

figure; set(gcf,'color','w')
plot(V,theta0_D10*180/pi,'color',[0.8 0 0],'LineWidth',2); hold on
plot(V,theta0_D12*180/pi,'color',[0 0.7 0],'LineWidth',2)
title('Collective pitch')
xlabel('$$Velocity (m/s)$$','Interpreter', 'Latex'); ylabel('Collective
pitch(deg)')
legend('theta0 D10000','theta0 D12000')
set(gca,'FontName','Times New Roman','FontSize',12);
grid minor

psi = 180;
theta = theta0_D12 - A1_D12*sind(psi) - B1_D12*cosd(psi);

figure; set(gcf,'color','w')
plot(V,theta*180/pi,'color',[0.8 0 0],'LineWidth',2);
title('Total pitch at 180 deg')
```

```
xlabel('$$Velocity (m/s)$$','Interpreter', 'Latex'); ylabel('Total pitch(deg)')
legend('Total pitch at 180 deg')
set(gca,'FontName','Times New Roman','FontSize',12);
grid minor
```