Homework 6

To:	Andrew Ning
From:	Joshua Canlas
Date:	April 15, 2020
Subject:	Homework 6 Memo

Propeller Aerodynamics

Introduction

As stated in the homework prompt, the objective of this homework is to implement a basic blade element momentum (BEM) model and validate our results against experimental data for the APC thin electric 10 x 5. The problem is split up into two sections: airfoil corrections and blade element momentum theory. Both sections are talked about throughout the report. The results and discussion section have plots that show the results that were calculated from the BEM theory.

To make this problem easier to solve we made some simplifications:

- Our blade only uses one airfoil and it is of constant thickness ratio along the blade (NACA 4412).
- We restricted our attention to comparing against data that is relatively constant in Reynolds number (Re = 1.5×10^6).
- Rather than analyzing across various Mach numbers, we used the Prandtl-Glauert correction.

Methods

In order to accurately use the BEM theory, we first had to make some corrections to the airfoil data. The following steps show what I did before using the BEM theory for calculating the coefficient of thrust, coefficient of power, and efficiency:

1. Use XFOIL to find the coefficient of lift and drag as a function of angle of attack until just past stall.
2. Use the Viterna method to extend the data to higher angles of attack (-30 to 30 degrees).

We decided to solve apply the correction methods (Prandtl-Glauert and rotational) to the airfoil data as we went through the BEM theory (see Appendix A for implementation). The equations to find the thrust coefficient, power coefficient, and efficiency are:

$$C_T = \frac{T}{\rho n^2 D^4}$$

$$C_Q = \frac{Q}{\rho n^2 D^5}$$

$$C_P = \frac{P}{\rho n^3 D^5}$$

$$\eta = \frac{P_{out}}{P_{in}} = \frac{T V_\infty}{Q\Omega} = \frac{C_T \rho n^2 D^4 V_\infty}{C_P \rho n^3 D^5} = J\frac{C_T}{C_P}$$

where thrust, torque, and power are defined as

$$W^2 = (V_\infty(1+a))^2 + (\Omega r(1-a'))^2$$

$$T' = Bc_n\frac{1}{2}\rho W^2 c$$

$$Q' = Brc_t\frac{1}{2}\rho W^2 c$$

$$T = \int_{r_h}^{r_t} T' dr$$

$$Q = \int_{r_h}^{r_t} Q' dr$$

$$P = Q\Omega$$

In order to find torque and thrust, we need the normal and tangential coefficients. Those coefficients are found by solving the following residual function:

$$\begin{aligned}
&\text{function } \mathcal{R}(\phi) \\
&\qquad \alpha = \theta - \phi \\
&\qquad c_l = f(\alpha, Re, M) \\
&\qquad c_d = f(\alpha, Re, M) \\
&\qquad c_n = c_l \cos\phi - c_d \sin\phi \\
&\qquad c_t = c_l \sin\phi + c_d \cos\phi \\
&\qquad a = \frac{\sigma' c_n}{4F\sin^2\phi - \sigma' c_n} \\
&\qquad a' = \frac{\sigma' c_t}{4F\sin\phi\cos\phi + \sigma' c_t} \\
&\quad \text{return } \frac{\sin\phi}{1+a} - \frac{V_\infty}{\Omega r}\frac{\cos\phi}{(1-a')}
\end{aligned}$$

where σ' is defined as

$$\sigma' = \frac{Bc}{2\pi r}$$

and F is found by

$$\begin{aligned}
f_{tip} &= \frac{B}{2}\left(\frac{R-r}{r|\sin\phi|}\right) \\
F_{tip} &= \frac{2}{\pi}\arccos(\exp(-f_{tip})) \\
f_{hub} &= \frac{B}{2}\left(\frac{r-R_{hub}}{R_{hub}|\sin\phi|}\right) \\
F_{hub} &= \frac{2}{\pi}\arccos(\exp(-f_{hub})) \\
F &= F_{tip}F_{hub}
\end{aligned}$$

This residual function can be solved by a simple root finding method. As mentioned earlier, the Prandtl-Glauert and rotational corrections are implemented at this time. We use the Prandtl-Glauert correction to correct the lift and drag coefficient:

$$c_l = \frac{c_{l0}}{\sqrt{1 - M^2}}$$

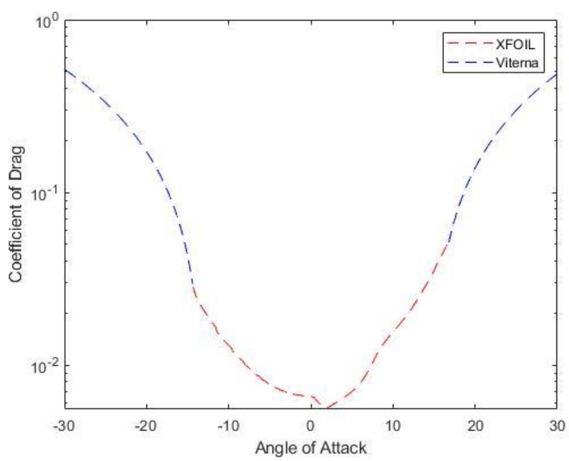
where M is the local Mach number. We then apply the rotational correction only to the normal coefficient

$$C_{n_{3-D}} = C_{n_{2-D}} + 1.5\left(\frac{c}{r}\right)^2 (C_{l,\text{pot}} - C_{l_{2-D}})\left(\frac{\omega r}{V_l}\right)^2$$

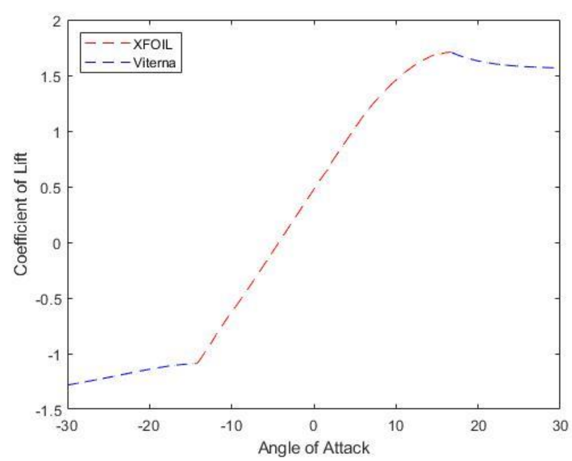
where $C_{l,pot}$ is the lift coefficient from thin airfoil theory for a symmetric section evaluated at the local angle of attack.

Results and Discussion

Figs. 1a and 1b show plots of the raw data collected from XFOIL and the extended data calculated using the Viterna method. As you can see, the extended data is continuous from the XFOIL data, which was one of the advantages of the Viterna method.



(a)



(b)

Figure 1: (a) Drag coefficient vs. angle of attack. (b) Lift coefficient vs. angle of attack. Both plots show the XFOIL data (red) and the extended data from the Viterna method (blue).

Figs. 2, 3, and 4 show the plots of thrust coefficient, power coefficient, and efficiency against the advance ratio respectively. The results that we got from the BEM theory are slightly off when compared to the experimental data. This is due to the fact that we ignored the changing Reynolds number throughout our calculations. However, taking this assumption into consideration, the results that we got are still fairly close.

Based on these results, we can conclude that the BEM theory is a very powerful tool when analyzing the performance of a propeller. One of the disadvantages of this theory is that it can get computationally expensive when we don't simplify things as much as we did for this problem. However, for our purposes, the BEM theory was quite successful in modeling the propeller's performance.

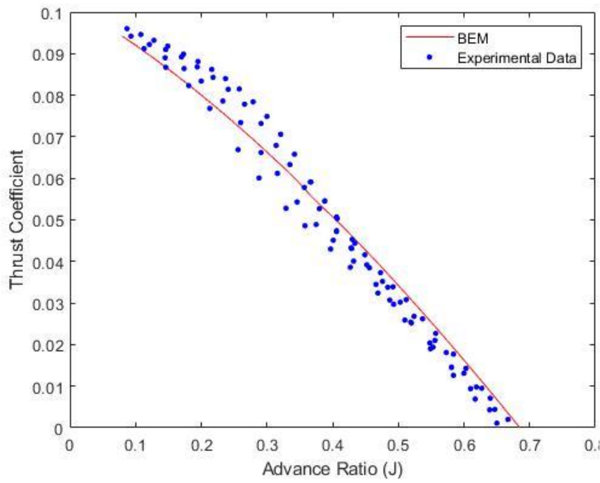


Figure 2: Thrust coefficient vs. advance ratio comparing experimental data and blade element momentum theory.

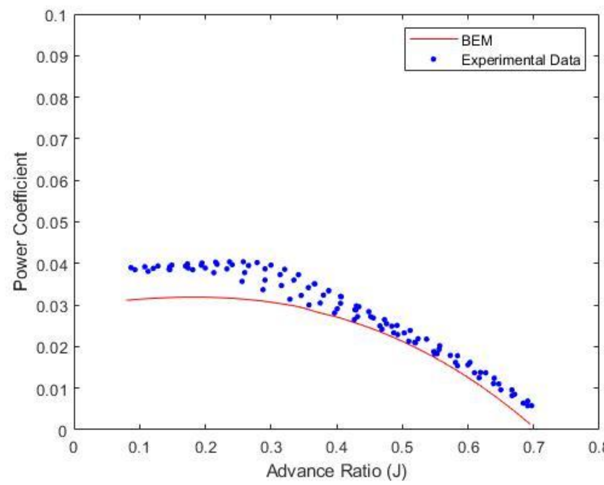


Figure 3: Power coefficient vs. advance ratio comparing experimental data and blade element momentum theory.

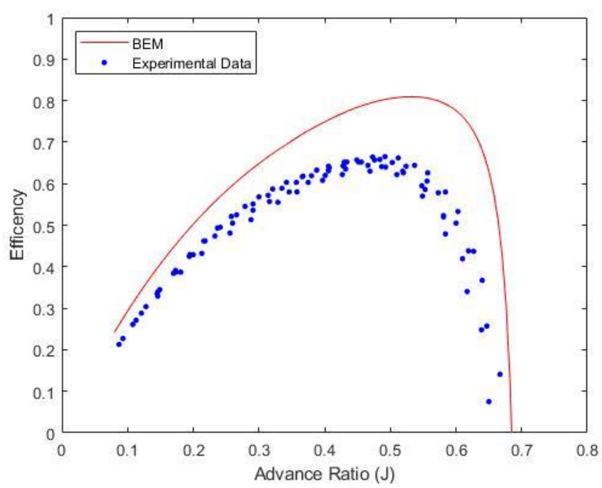


Figure 2: Plot of propeller efficiency vs. advance ratio comparing experimental data and blade element momentum theory.

Appendix A

```
clc
clear all

%6.1------------------------------------------------
xfoil = importdata("naca4412_FINAL.txt");
exp_data = importdata("exp_data.txt");
J_exp = exp_data(:, 1);
CT_exp = exp_data(:, 2);
CP_exp = exp_data(:, 3);
eta_exp = exp_data(:, 4);
alpha = xfoil(:, 1);
cl = xfoil(:, 2);
cd = xfoil(:, 3);
D_prop = 0.254; % meters
c_R = 0.128; % from UIUC data
AR = 1 / c_R;
R_prop = D_prop/2;
B = 2;  %number of blades
c_R_tot = [0.149;0.173;0.189;0.197;0.201;0.200;0.194;0.186;....
    0.174;0.160;0.145;0.128;0.112;0.096;0.081;0.061];
c = c_R_tot*R_prop; % chord length at any location of the propeller

% Viterna Method-----------------------------------------------
% Upper angles of attack
cd_max = 1.11 + 0.18 * AR;
B_1 = cd_max;
A_1 = B_1 / 2;
cl_stall = cl(end);
cd_stall = cd(end);
alpha_stall = alpha(end) * (pi/180);
A_2 = (cl_stall - cd_max * sin(alpha_stall) * cos(alpha_stall)) * ...
    sin(alpha_stall)/(cos(alpha_stall))^2;
B_2 = cd_stall - (cd_max*(sin(alpha_stall))^2 / cos(alpha_stall));
alpha_up = (16.8:0.1:30) .* (pi/180);
C_D_up = B_1 .* (sin(alpha_up)).^2 + B_2 .* cos(alpha_up);
C_L_up = A_1 .* sin(2 .* alpha_up) + A_2 .* ((cos(alpha_up)).^2 ./
sin(alpha_up));

% Lower angles of attack
cl_stall = cl(1);
cd_stall = cd(1);
alpha_stall = alpha(1) * (pi/180);
A_2 = (cl_stall - cd_max * sin(alpha_stall) * cos(alpha_stall)) * ...
    sin(alpha_stall)/(cos(alpha_stall))^2;
B_2 = cd_stall - (cd_max*(sin(alpha_stall))^2 / cos(alpha_stall));
alpha_low = (-30:0.1:-14.4) .* (pi/180);
C_D_low = B_1 .* (sin(alpha_low)).^2 + B_2 .* cos(alpha_low);
C_L_low = A_1 .* sin(2 .* alpha_low) + A_2 .* ((cos(alpha_low)).^2 ./
sin(alpha_low));

%Plot Cd and Cl
plotting1(alpha,cl,C_L_up,C_L_low,cd,C_D_up,C_D_low,alpha_up,alpha_low);
```

```
%6.2 BEM-----------------------------------------------

%setup
%Concantenate cl, cd, and alpha values (from xfoil and vaterni)----
cl_0 = horzcat(C_L_low, cl', C_L_up);
cd_0 = horzcat(C_D_low, cd', C_D_up);
alpha = horzcat(alpha_low, alpha' .* (pi/180), alpha_up);
N = 100; % number of points
J = linspace(0.08, 0.695, N);
omega = 1;
rho = 1;
r_R = [0.20;0.25;0.30;0.35;0.40;0.45;0.50;0.55;0.60;...
    0.65;0.70;0.75;0.80;0.85;0.90;0.95]; % position on propeller
r = r_R .* R_prop;

for R = 1:N              %advance ratios
    for K = 1:length(r)      %across blade
        V_inf = J(R) * omega * D_prop / (2 * pi);
        sigma_prime = B*c(K)/(2*pi*r(K));

        %solving for twist (theta)
        [theta] = twist(K);

        %solving residual equation
        % x = phi variable
        fun = @(x)
residuals(x,omega,sigma_prime,theta,r(K),V_inf,alpha,R_prop,cl_0,cd_0,c(K),B)
;
        x0 = [(1*10^-6) pi/2];

        phi = fzero(fun,x0);

        %solve for values using known phi
        alpha_fin = theta - phi;
        [cn_fin, ct_fin] = corrections(cl_0, cd_0,
alpha_fin,omega,R_prop,alpha,r(K),V_inf,phi,c(K));

        %Tip modificaitons
        R_hub = 0.15*R_prop;
        ftip = B/2*((R_prop-r(K))/(r(K)*abs(sin(phi))));
        Ftip = (2/pi)*(acos(exp(-ftip)));
        fhub = (B/2)*((r(K)-R_hub)/(R_hub*abs(sin(phi))));
        Fhub = (2/pi)*(acos(exp(-fhub)));
        F = Ftip*Fhub;

        %induction factors
        a_prime = sigma_prime*ct_fin /
(4*F*sin(phi)*cos(phi)+sigma_prime*ct_fin);
        a = sigma_prime * cn_fin/(4*F*(sin(phi))^2-sigma_prime*cn_fin);

        W_sq = (V_inf*(1+a))^2 + (omega*r(K)*(1-a_prime))^2;

        T_prime(K) = B*cn_fin*0.5*rho*W_sq*c(K);
        Q_prime(K) = B*r(K)*ct_fin*0.5*rho*W_sq*c(K);
    end
```

```
    T = trapz(r,T_prime);
    Q = trapz(r,Q_prime);
    P = Q*omega;

    n = omega/(2*pi);
    CT(R) = T/(rho*n^2*D_prop^4);
    CQ(R) = Q/(rho*n^2*D_prop^5);
    CP(R) = P/(rho*n^3*D_prop^5);

    eff(R) = J(R)*(CT(R)/CP(R));
end

%Plot CT,CP,Eff vs J
plotting2(J,CT,CP,eff,J_exp,CT_exp,CP_exp,eta_exp)

%******************Functions**********************
function [cn, ct] = corrections(cl_0, cd_0,
alpha_curr,omega,R_prop,alpha,r,V_inf,x,c)
    a = 343.0; % speed of sound in m/s
    % call interpolation
    alpha_max = 30 * pi /180;
    alpha_min = -30 * pi / 180;

    if alpha_curr > alpha_max
        alpha_curr = alpha_max;
    elseif alpha_curr < alpha_min
        alpha_curr = alpha_min;
    end

    cl_0 = interp1(alpha, cl_0, alpha_curr);
    cd_0 = interp1(alpha, cd_0, alpha_curr);

    %V_l or W
    V_l = sqrt((omega*r)^2+(V_inf)^2);

    % Prandtl-Glauert
    M = V_l / a; % local mach number
    cl = cl_0 ./ (1 - M^2);
    cd = cd_0 ./ (1 - M^2);

    cn_0 = cl * cos(x) + cd * sin(x);
    ct = cl * sin(x) - cd * cos(x);

    %Rotational correction (only for c_n)
    cn = cn_0 + 1.5*(c/R_prop)^2*(2*pi*alpha_curr-cl_0)*(omega*R_prop/V_l)^2;
end

function y =
residuals(x,omega,sigma_prime,theta,r,V_inf,alpha,R_prop,cl_0,cd_0,c,B)
    %alpha
    alpha_curr = theta - x;

    %cn and ct using alpha
```

```
    [cn, ct] = corrections(cl_0, cd_0,
alpha_curr,omega,R_prop,alpha,r,V_inf,x,c);

    %Tip modificaitons
    R_hub = 0.15*R_prop;
    ftip = (B/2)*((R_prop-r)/(r*abs(sin(x))));
    Ftip = (2/pi)*(acos(exp(-ftip)));
    fhub = (B/2)*((r-R_hub)/(R_hub*abs(sin(x))));
    Fhub = (2/pi)*acos(exp(-fhub));
    F = Ftip*Fhub;

    %induction factors
    a_prime = sigma_prime*ct / (4*F*sin(x)*cos(x)+sigma_prime*ct);
    a = sigma_prime * cn/(4*F*(sin(x))^2-sigma_prime*cn);

    %residual equation
    y = sin(x)/(1+a) - V_inf*cos(x)/(omega*r*(1-a_prime));
end

function [theta] = twist(J)
    %Twist Values
    twist_value = [37.19;33.54;29.25;25.64;22.54;20.27;...
        18.46;17.05;15.97;14.87;14.09;13.39;12.84;12.25;11.37;...
        10.19];

    theta = twist_value(J) * (pi/180); % radians
end

function
plotting1(alpha,cl,C_L_up,C_L_low,cd,C_D_up,C_D_low,alpha_up,alpha_low)
    %Plotting
    figure(1)
    plot(alpha, cl, 'r--')
    hold on
    plot(alpha_up .* (180/pi), C_L_up, 'b--')
    hold on
    plot(alpha_low .* (180/pi), C_L_low, 'b--')
    xlabel('Angle of Attack')
    ylabel('Coefficient of Lift')
    xlim([-30 30])
    ylim([-1.5 2.0])
    legend("XFOIL", "Viterna", "Location", "northwest")

    figure(2)
    plot(alpha, cd, 'r--')
    hold on
    plot(alpha_up .* (180/pi), C_D_up, 'b--')
    hold on
    plot(alpha_low .* (180/pi), C_D_low, 'b--')
    xlabel('Angle of Attack')
    ylabel('Coefficient of Drag')
    xlim([-30 30])
    legend("XFOIL", "Viterna")
    set(gca, 'YScale', 'log')
    end
```

```matlab
function plotting2(J,CT,CP,eff,J_exp,CT_exp,CP_exp,eta_exp)
    figure(3)
    plot(J,CT,'r')
    hold on
    plot(J_exp,CT_exp,'b.','MarkerSize', 10)
    xlabel('Advance Ratio (J)')
    ylabel('Thrust Coefficient')
    xlim([0.0 0.8])
    ylim([0.0 0.1])
    legend("BEM", "Experimental Data")

    figure(4)
    plot(J,CP,'r')
    hold on
    plot(J_exp,CP_exp,'b.','MarkerSize', 10)
    xlabel('Advance Ratio (J)')
    ylabel('Power Coefficient')
    xlim([0.0 0.8])
    ylim([0.0 0.1])
    legend("BEM", "Experimental Data")

    figure(5)
    plot(J,eff,'r')
    hold on
    plot(J_exp,eta_exp,'b.','MarkerSize', 10)
    xlabel('Advance Ratio (J)')
    ylabel('Efficency')
    xlim([0.0 0.8])
    ylim([0.0 1.0])
    legend("BEM", "Experimental Data", "Location", "northwest")
end
```