```
%Written by Tianjun Han, 03/15/2025
%function CL = steady(alpha)

%% Define parameters
alpha = 15/180*pi;
U = -1;
c = 1;
Np = 10;
DelT = 0.01*c/abs(U);
Nstep = 500;
xw = ones(1,Nstep);
yw = ones(1,Nstep);
Gammaw = ones(1,Nstep);
Gammap = zeros(2,Np);
L = ones(1,Nstep);
CL = ones(1,Nstep);

for k = 1:Nstep

%% Discreteize the foil
t = DelT*(k-1);
xb = linspace(0,1,Np+1)*c*cos(alpha) + U*t;
yb = linspace(0,1,Np+1)*c*(-sin(alpha));


%% Build vortex points and collocation points
xv = (xb(2:end)-xb(1:end-1))*0.25 + xb(1:end-1);
yv = (yb(2:end)-yb(1:end-1))*0.25 + yb(1:end-1);

xc = (xb(2:end)-xb(1:end-1))*0.75 + xb(1:end-1);
yc = (yb(2:end)-yb(1:end-1))*0.75 + yb(1:end-1);



%% Build vectors normal to the body
nb = ones(2,Np);
nb(1,:) = (yc - yv)./sqrt((yc-yv).^2 + (xc-xv).^2);
nb(2,:) = -(xc-xv)./sqrt((yc-yv).^2 + (xc-xv).^2);

%% Build vectors for foil's motion
Vp = ones(2,Np);
Vp(1,:) = U;
Vp(2,:) = 0;

%% Update the location of trailing-edge vortice
if k>1
Gammaw(2:k)=Gammaw(1:k-1);
xw(2:k) = xw(1:k-1);
yw(2:k) = yw(1:k-1);
end
```

```
xw(1) = (xb(end)-xc(Np))/sqrt((xc(Np)-xb(end))^2 + (yc(Np)-
yb(end))^2)*DelT*abs(U)*0.4+xb(end);
yw(1) = (yb(end)-yc(Np))/sqrt((xc(Np)-xb(end))^2 + (yc(Np)-
yb(end))^2)*DelT*abs(U)*0.4+yb(end);

%% Build the matrix for induced velocity coefficient
a = ones(Np,Np,2);
aw = ones(Np,k);

for i = 1:Np
for j = 1:Np

a(i,j,1) = 1/2/pi/sqrt((xc(i)-xv(j))^2 + (yc(i)-yv(j))^2)*(yc(i)-
yv(j))/sqrt((xc(i)-xv(j))^2 + (yc(i)-yv(j))^2);

a(i,j,2) = 1/2/pi/sqrt((xc(i)-xv(j))^2 + (yc(i)-yv(j))^2)*(-xc(i)
+xv(j))/sqrt((xc(i)-xv(j))^2 + (yc(i)-yv(j))^2);

end
end

for i = 1:Np
for j = 1:k

aw(i,j,1) = 1/2/pi/sqrt((xc(i)-xw(j))^2 + (yc(i)-yw(j))^2)*(yc(i)-
yw(j))/sqrt((xc(i)-xw(j))^2 + (yc(i)-yw(j))^2);

aw(i,j,2) = 1/2/pi/sqrt((xc(i)-xw(j))^2 + (yc(i)-yw(j))^2)*(-xc(i)
+xw(j))/sqrt((xc(i)-xw(j))^2 + (yc(i)-yw(j))^2);

end
end


%% Build matrices A and B

A = ones(Np+1,Np+1);
B = ones(Np+1,1);

for i = 1:Np

    A(i,1:Np) = a(i,:,1) * nb(1,i) + a(i,:,2) * nb(2,i);
    B(i) = Vp(1,i) * nb(1,i) + Vp(2,i) * nb(2,i) -
sum(aw(i,2:end,1).*Gammaw(2:k)*nb(1,i) +
aw(i,2:end,2).*Gammaw(2:k)*nb(2,i));
    A(i,Np+1) = aw(i,1,1)*nb(1,i) + aw(i,1,2)*nb(2,i);

end

A(Np+1,1:Np) = 1;
```

```
A(Np+1,Np+1) = 1;
B(Np+1) = -sum(Gammaw(2:k));


%% Solve Gamma and calculate lift
Gamma = inv(A)*B;

Gammap(2,:) = Gammap(1,:);
Gammap(1,:) = Gamma(1:end-1)';

Gammaw(1) = Gamma(end);

steadyterm = 0;
unsteadyterm = 0;
for i = 1:Np

    steadyterm = 1000*abs(U)*Gammap(1,i) + steadyterm;

    unsteadyterm = 1000*c/Np*(sum(Gammap(1,1:i))-sum(Gammap(2,1:i)))/
DelT + unsteadyterm;

end

L(k) = steadyterm+unsteadyterm;

CL(k) = L(k)/(0.5*1000*U^2*c);


%% Wake roll-up

rollup_matrix = ones(k,k+Np,2);
x_total = [xv,xw(1:k)];
y_total = [yv,yw(1:k)];
Gamma_total = [Gamma(1:Np)',Gammaw(1:k)];

for i = 1:k
for j = 1:k+Np

    rollup_matrix(i,j,1) = 1/2/pi/sqrt((xw(i)-x_total(j))^2 + (yw(i)-
y_total(j))^2 + (0.04*c)^2)*(yw(i)-y_total(j))/sqrt((xw(i)-
x_total(j))^2 + (yw(i)-y_total(j))^2 + (0.04*c)^2);
    rollup_matrix(i,j,2) = 1/2/pi/sqrt((xw(i)-x_total(j))^2 + (yw(i)-
y_total(j))^2 + (0.04*c)^2)*(-xw(i)+x_total(j))/sqrt((xw(i)-
x_total(j))^2 + (yw(i)-y_total(j))^2 + (0.04*c)^2);

end
end


for i = 1:k
```

```
    xw(i) = xw(i) + DelT*sum(rollup_matrix(i,:,1).*Gamma_total);
    yw(i) = yw(i) + DelT*sum(rollup_matrix(i,:,2).*Gamma_total);
end



%end

%% plot the flow field
if mod(k,10) == 0
plot(xb,yb,'k-','Linewidth',2);
hold on
scatter(xw(1:k), yw(1:k), 10, Gammaw(1,1:k)*c/abs(U), 'filled')
colormap(red_blue_color_scheme_qiang)
colorbar
axis equal
caxis([-0.01,0.01])
xlim([-5.5*c,1.5*c])
ylim([-c,c])
pause(1)
close all
end


end
```