CProp

Generated by Doxygen 1.8.11

Contents

1 CProp **1**

2 Class Index **3**

2.1 Class List . . . 3

3 File Index **5**

3.1 File List . . . 5

4 Class Documentation **7**

4.1 propinfo Struct Reference . . . 7

4.1.1 Detailed Description . . . 7

4.1.2 Member Data Documentation . . . 7

4.1.2.1 angvelocity . . . 7

4.1.2.2 blades . . . 8

4.1.2.3 cruisespeed . . . 8

4.1.2.4 HUB_FLAG . . . 8

4.1.2.5 hubradius . . . 8

4.1.2.6 panels . . . 8

4.1.2.7 radius . . . 8

4.1.2.8 thrust . . . 8

4.1.2.9 waterdensity . . . 8

4.2 threetuple Struct Reference . . . 8

4.3 XfoilInterface Class Reference . . . 9

4.3.1 Detailed Description . . . 9

4.3.2 Member Function Documentation . . . 9

4.3.2.1 angleOfAttack(double angle) . . . 9

4.3.2.2 loadFoilFile(char fpath[], char foilname[]) . . . 9

4.3.2.3 NACA(int code) . . . 10

4.3.2.4 setViscosity(int Re) . . . 10

4.3.2.5 start() . . . 10

5 File Documentation **11**

5.1 /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/inducedvelocities.h File Reference 11

5.1.1 Detailed Description 11

5.1.2 Function Documentation 12

5.1.2.1 asympf(double y, double y0, double U, int a, int Z) 12

5.1.2.2 asympF1(double rc, double rv, double beta, int Z) 12

5.1.2.3 asympF2(double rc, double rv, double beta, int Z) 12

5.1.2.4 axialVelocity(double rc, double rv, double beta, int Z) 13

5.1.2.5 axialVelocityStar(int m, int p, double ∗rc, double ∗rv, double ∗beta, int Z, double rh, int HUB_FLAG) 13

5.1.2.6 tangentialVelocity(double rc, double rv, double beta, int Z) 13

5.1.2.7 tangentialVelocityStar(int m, int p, double ∗rc, double ∗rv, double ∗beta, int Z, double rh, int HUB_FLAG) 14

5.1.2.8 transformVars(double rc, double rv, double beta, int Z) 14

5.2 /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/matrixsolver.h File Reference 14

5.3 /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/xfoilinterface.h File Reference 15

5.3.1 Detailed Description 15

Index **17**

Chapter 1

CProp

Lifting line propeller optimization routine using C.

For a long time now, mathematical procedures to optimize the efficiency of marine propellers operating in nonuniform ship wakes have been known. Several implementations of these procudures exist, notably OpenProp (written in M↩ ATLAB) and JavaProp. However, OpenProp limits itself to single-stage propulsors, and JavaProp is mainly used for plane propellers.

This project aims to implement Coney's (1989) optimization procedure for multi-stage propulsors, contra-rotating propellers in particular, in a single program. The eventual goal is to provide the complete propeller geometry, starting with a lifting line optimization program, and including 2.5D viscous flow analysis with XFOIL, strenght/vibration analysis, and cavitation analysis.

Most of the routines will be written in the C and C++ programming languages, with exceptions made for graphical applications.

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

propinfo
Struct containing constant propeller info blades : const uint8_t : number of propeller blades radius : const double : radius of propeller, [m] angvelocity : const double : angular velocity of propeller, [rad/s] cruisespeed : const double : cruise speed of ship, [m/s] hubradius : const double : radius of propeller hub, [m] thrust : const double : required thrust of propeller, [N] panels : const uint8_t : number of lifting line panels waterdensity : const double : density of water, [kg/m^3] HUB_FLAG : const int : sets whether to simulate a hub 7
threetuple 8
XfoilInterface
Class that interface with XFoil 9

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

/mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/inducedvelocities.h
Functions for calculating induced velocities on lifting line 11
/mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/matrixsolver.h
Functions for solving the linear system that arises when optimizing the single propeller circulation distribution 14
/mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/xfoilinterface.h
Library for calling XFoil from a C++ program 15

Chapter 4

Class Documentation

4.1 propinfo Struct Reference

struct containing constant propeller info blades : const uint8_t : number of propeller blades radius : const double : radius of propeller, [m] angvelocity : const double : angular velocity of propeller, [rad/s] cruisespeed : const double : cruise speed of ship, [m/s] hubradius : const double : radius of propeller hub, [m] thrust : const double : required thrust of propeller, [N] panels : const uint8_t : number of lifting line panels waterdensity : const double : density of water, [kg/m^3] HUB_FLAG : const int : sets whether to simulate a hub

```
#include <matrixsolver.h>
```

Public Attributes

- const uint8_t blades
- const double radius
- const double angvelocity
- const double cruisespeed
- const double hubradius
- const double thrust
- const uint8_t panels
- const double waterdensity
- const int HUB_FLAG

4.1.1 Detailed Description

struct containing constant propeller info blades : const uint8_t : number of propeller blades radius : const double : radius of propeller, [m] angvelocity : const double : angular velocity of propeller, [rad/s] cruisespeed : const double : cruise speed of ship, [m/s] hubradius : const double : radius of propeller hub, [m] thrust : const double : required thrust of propeller, [N] panels : const uint8_t : number of lifting line panels waterdensity : const double : density of water, [kg/m^3] HUB_FLAG : const int : sets whether to simulate a hub

4.1.2 Member Data Documentation

4.1.2.1 const double propinfo::angvelocity

angular velocity of propeller, [rad/s]

4.1.2.2 const uint8_t propinfo::blades

number of propeller blades

4.1.2.3 const double propinfo::cruisespeed

cruise speed of ship, [m/s]

4.1.2.4 const int propinfo::HUB_FLAG

∗sets whether to simulate a hub

4.1.2.5 const double propinfo::hubradius

radius of propeller hub, [m]

4.1.2.6 const uint8_t propinfo::panels

number of lifting line panels

4.1.2.7 const double propinfo::radius

∗radius of propeller, [m]

4.1.2.8 const double propinfo::thrust

required thrust of propeller, [N]

4.1.2.9 const double propinfo::waterdensity

density of water, [kg/m^3]

The documentation for this struct was generated from the following file:

- /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/matrixsolver.h

4.2 threetuple Struct Reference

Public Attributes

- double **x**
- double **y**
- double **z**

The documentation for this struct was generated from the following file:

- /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/inducedvelocities.c

4.3 XfoilInterface Class Reference

Class that interface with XFoil.

`#include <xfoilinterface.h>`

Public Member Functions

- XfoilInterface (bool _plot, string _paccfile="")

  Constructor for XfoilInterface class.
- bool start ()

  Starts xfoil interface.
- void configure ()

  Configures xfoil with constructor parameters.
- void quit ()

  Quits xfoil.
- void loadFoilFile (char fpath[ ], char foilname[ ])

  Loads airfoil coordinates from file.
- void NACA (int code)

  Selects a NACA airfoil to input to xfoil.
- void setViscosity (int Re)

  Enables viscous mode.
- void angleOfAttack (double angle)

  Starts Xfoil analysis of single angle of attack.

4.3.1 Detailed Description

Class that interface with XFoil.

4.3.2 Member Function Documentation

4.3.2.1 void XfoilInterface::angleOfAttack (double *angle*)

Starts Xfoil analysis of single angle of attack.

Parameters

angle	angle of attack to analyze

4.3.2.2 void XfoilInterface::loadFoilFile (char *fpath[ ]*, char *foilname[ ]*)

Loads airfoil coordinates from file.

Parameters

fpath	File to load coordinates from
foilname	Airfoil name

4.3.2.3 void XfoilInterface::NACA (int *code*)

Selects a NACA airfoil to input to xfoil.

Parameters

input	4-digit naca airfoil code

4.3.2.4 void XfoilInterface::setViscosity (int *Re*)

Enables viscous mode.

Parameters

Re	Reynolds number of flow

4.3.2.5 bool XfoilInterface::start ( )

Starts xfoil interface.

Returns

Whether XFoil was initialized succesfully

The documentation for this class was generated from the following files:

- /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/xfoilinterface.h
- /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/xfoilinterface.cpp

Chapter 5

File Documentation

5.1 /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/inducedvelocities.h File Reference

Functions for calculating induced velocities on lifting line.

Macros

- #define **PI** 3.14159265359

Functions

- double asympf (double y, double y0, double U, int a, int Z)
  asymptotic formulae F1 and F2, encoded in single formula with parameter +/- 1
- struct threetuple transformVars (double rc, double rv, double beta, int Z)
  transforms variables rc, rv, beta to y, y0, U
- double asympF1 (double rc, double rv, double beta, int Z)
  asymptotic F1
- double asympF2 (double rc, double rv, double beta, int Z)
  asymptotic F2
- double axialVelocity (double rc, double rv, double beta, int Z)
  calculates axial velocity as per Coney
- double tangentialVelocity (double rc, double rv, double beta, int Z)
  calculates tangential velocity as per Coney
- double axialVelocityStar (int m, int p, double ∗rc, double ∗rv, double ∗beta, int Z, double rh, int HUB_FLAG)
  axial velocity induced at point m by vortex at p, including hub effects
- double tangentialVelocityStar (int m, int p, double ∗rc, double ∗rv, double ∗beta, int Z, double rh, int HUB_↩ FLAG)
  tangential velocity induced at point m by vortex at p, including hub effects

5.1.1 Detailed Description

Functions for calculating induced velocities on lifting line.

Author

Jesse van Rhijn

5.1.2 Function Documentation

5.1.2.1 double asympf (double *y*, double *y0*, double *U*, int *a*, int *Z*)

asymptotic formulae F1 and F2, encoded in single formula with parameter +/- 1

Parameters

y	see Coney
y0	see Coney
U	see Coney
Z	blade number
a	1 for F1, -1 for F2

Returns

value of asymptotic Bessel function formulae

5.1.2.2 double asympF1 (double *rc*, double *rv*, double *beta*, int *Z*)

asymptotic F1

Parameters

rc	control point radius
rv	vortex point radius
beta	pitch angle of trailing vortex
Z	blade number

Returns

result of asymptotic F1

5.1.2.3 double asympF2 (double *rc*, double *rv*, double *beta*, int *Z*)

asymptotic F2

Parameters

rc	control point radius
rv	vortex point radius
beta	pitch angle of trailing vortex
Z	blade number

Returns

result of asymptotic F2

5.1.2.4 double axialVelocity (double *rc*, double *rv*, double *beta*, int *Z*)

calculates axial velocity as per Coney

Parameters

rc	control point radius
rv	vortex point radius
beta	pitch angle of trailing vortex
Z	blade number

Returns

value of axial velocity at r = rc due to helical vortex line at radius r = rv

5.1.2.5 double axialVelocityStar (int *m*, int *p*, double ∗ *rc*, double ∗ *rv*, double ∗ *beta*, int *Z*, double *rh*, int *HUB_FLAG*)

axial velocity induced at point m by vortex at p, including hub effects

Parameters

m	index of control point
p	index of vortex point
rc	array of control point radii
rv	array of vortex point radii
beta	array of vortex pitch angles
Z	blade number
rh	hub radius
HUB_FLAG	sets whether to include hub effects

Returns

axial induced velocitiy at m due to helical horseshoe vortex at p

5.1.2.6 double tangentialVelocity (double *rc*, double *rv*, double *beta*, int *Z*)

calculates tangential velocity as per Coney

Parameters

rc	control point radius
rv	vortex point radius
beta	pitch angle
Z	blade number

Returns

value of tangential velocity at r = rc due to helical vortex line at radius r = rv

5.1.2.7 double tangentialVelocityStar (int *m*, int *p*, double ∗ *rc*, double ∗ *rv*, double ∗ *beta*, int *Z*, double *rh*, int *HUB_FLAG*)

tangential velocity induced at point m by vertex at p, including hub effects

Parameters

m	index of control point
p	index of vortex point
rc	array of control point radii
rv	array of vortex point radii
beta	array of vortex pitch angles
Z	blade number
rh	hub radius
HUB_FLAG	sets whether to include hub effects

Returns

tangential induced velocitiy at m due to helical horseshoe vortex at p

5.1.2.8 struct threetuple transformVars (double *rc*, double *rv*, double *beta*, int *Z*)

transforms variables rc, rv, beta to y, y0, U

Parameters

rc	control point radius
rv	vortex point radius
beta	pitch angle of trailing vortex
Z	blade number

Returns

three-tuple of transformed variables

5.2 /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/matrixsolver.h File Reference

Functions for solving the linear system that arises when optimizing the single propeller circulation distribution.

```
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <gsl/gsl_linalg.h>
#include <gsl/gsl_matrix.h>
```

Include dependency graph for matrixsolver.h:

5.3 /mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/xfoilinterface.h File Reference

Library for calling XFoil from a C++ program.

```
#include <iostream>
#include <fstream>
#include <cstdlib>
#include <cstring>
#include <string>
#include <thread>
```

Include dependency graph for xfoilinterface.h:

Classes

- class XfoilInterface

    Class that interface with XFoil.

5.3.1 Detailed Description

Library for calling XFoil from a C++ program.

Author

Jesse van Rhijn

Index

/mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/inducedvelocities.↩
h, 11
/mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/matrixsolver.↩
h, 14
/mnt/c/Users/Jesse/Dropbox/Git/CProp/libs/headers/xfoilinterface.↩
h, 15

angleOfAttack
XfoilInterface, 9
angvelocity
propinfo, 7
asympF1
inducedvelocities.h, 12
asympF2
inducedvelocities.h, 12
asympf
inducedvelocities.h, 12
axialVelocity
inducedvelocities.h, 13
axialVelocityStar
inducedvelocities.h, 13

blades
propinfo, 7

cruisespeed
propinfo, 8

HUB_FLAG
propinfo, 8
hubradius
propinfo, 8

inducedvelocities.h
asympF1, 12
asympF2, 12
asympf, 12
axialVelocity, 13
axialVelocityStar, 13
tangentialVelocity, 13
tangentialVelocityStar, 14
transformVars, 14

loadFoilFile
XfoilInterface, 9

NACA
XfoilInterface, 10

panels
propinfo, 8

propinfo, 7
angvelocity, 7
blades, 7
cruisespeed, 8
HUB_FLAG, 8
hubradius, 8
panels, 8
radius, 8
thrust, 8
waterdensity, 8

radius
propinfo, 8

setViscosity
XfoilInterface, 10
start
XfoilInterface, 10

tangentialVelocity
inducedvelocities.h, 13
tangentialVelocityStar
inducedvelocities.h, 14
threetuple, 8
thrust
propinfo, 8
transformVars
inducedvelocities.h, 14

waterdensity
propinfo, 8

XfoilInterface, 9
angleOfAttack, 9
loadFoilFile, 9
NACA, 10
setViscosity, 10
start, 10