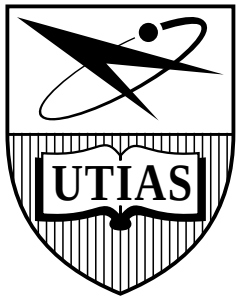

CYCLONE
OPTIMA–2D

User's Guide Version 4.0

by

M. Nemec

Prof. D. W. Zingg

L. Billing

C. Oldfield

M. Tabesh

M. Rumpfkeil

H. Buckley

University of Toronto Institute for Aerospace Studies
4925 Dufferin Street, Toronto, ON, M3H 5T6

August 2010

Contents

Chapter 1

Introduction

1.1 Objective

This guide provides a reference for the two-dimensional Navier-Stokes flow solver CYCLONE and the aerodynamic shape optimization algorithm OPTIMA–2D, both developed at the University of Toronto Institute for Aerospace Studies (UTIAS). This guide [15] replaces the "SC1 User's Guide, Version 1.0", written by De Rango and Zingg [6], and updates "CYCLONE OPTIMA-2D User's Guide" version 3.0 written in October 2008 by Nemec and Zingg [11] with contributions from Billing, Oldfield, Tabesh, and Rumpfkeil.

1.2 Optima2D Version Control

As of the current version 13, Optima2D version control is now handled with GIT. Prior to version 13, Optima2D version control was handled with CVS. The old CVS central repository has been migrated over to GIT in its entirety. This means that *all* versions of

the code, before and after the introduction of GIT, are contained in the GIT repository. The GIT central repository containing Optima2D is located on the oddjob server at:

```
/nfs/carv/d1/people/optima/optima.git
```

Given the remote possibility that some of you may still want to access the old CVS repository, it remains in its original location on the oddjob server at:

```
/nfs/carv/d1/people/optima/optima
```

It is highly recommended that all new users of Optima2D start using GIT for version control instead of CVS.

1.3 Installation of Optima2D

This section gives an overview of the procedure for installing Optima2D. The Optima2D source code must first be copied to a directory where the source code will be compiled. To accomplish this, clone the GIT central repository for Optima2D source code. There are two main scenarios for cloning the Optima2D central repository:

1. The machine intended for Optima2D installation is on the same local network as the Optima2D central repository, i.e. the oddjob network.
2. The machine intended for Optima2D installation is remotely located with respect to the oddjob network.

If the first scenario is applicable, then execute the following command:

```
git clone -o central /nfs/carv/d1/people/optima/optima.git optima
```

Otherwise:

```
git clone -o central <username>@oddjob.utias.utoronto.ca:
/nfs/carv/d1/people/optima/optima.git optima
```

After the cloning procedure has completed, go to the newly created directory named *optima*. Open the text file named *INSTALL* and follow the instructions for installing/compiling Optima2D.

1.4 Overview

CYCLONE solves the compressible thin-layer Navier-Stokes equations, which are discretized on structured grids for single-element airfoils. The flow solver is based on the approximate-factorization algorithm and second-order spatial discretization used in the NASA Ames flow solver ARC2D [18]. Turbulence is modeled using the algebraic Baldwin-Lomax and the one-equation Spalart-Allmaras [20] models. The solver includes matrix dissipation [9], CUSP dissipation [12], local-preconditioning [22], a dual-time-stepping subiteration algorithm [8] for unsteady calculations, and third-order spatial discretization [7, 5].

Recently, an aerodynamic shape optimization capability has been developed for the CYCLONE flow solver. The algorithm is referred to as OPTIMA–2D [13, 14], where a gradient-based numerical optimization approach is used to determine optimal airfoil shapes and configurations. The gradient is computed via the discrete-adjoint method. Furthermore, the convergence rate of OPTIMA–2D has been accelerated by a Newton–Krylov algorithm [17, 23]. Details on the numerical implementation of OPTIMA–2D are given by Nemec [10].

For aerodynamic shape optimization problems, the user is required to define design variables that control the shape of the airfoil. B-splines are used to parameterize the airfoil shape. The coordinates of the B-spline control points are used as design variables. The B-spline parameterization and the selection of design variables is performed by a separate program referred to as MODAIR, which is described in Chapter 5.

1.5 Limitations

The following limitations apply to the present version of CYCLONE & OPTIMA-2D:

1. Fully-turbulent flow is assumed for analysis runs with the Newton–Krylov algorithm and optimization runs. The laminar-turbulent trip terms can be used only with the approximate–factorization algorithm.

2. For analysis runs with the approximate-factorization algorithm, scalar, matrix and CUSP dissipation can be used. For the Newton–Krylov algorithm and optimization, only scalar dissipation is available.

3. For optimization runs, better convergence of the optimization is obtained with DIS2X and DIS2Y set to zero. For flow with moderate-strength shocks, one can increase DIS4X and DIS4Y to 0.02 in order to improve stability.

Chapter 2

INPUT

2.1 Overview

In this chapter, the input files required by CYCLONE & OPTIMA–2D are discussed. The program CYCLONE & OPTIMA–2D can be executed in three modes: 1) analysis, 2) sweep, and 3) optimization. In the analysis mode, the program solves for the flowfield resulting from a given airfoil geometry and freestream conditions. The sweep mode is essentially an analysis mode, except that the program automatically sweeps through a range of angles-of-attack or Mach numbers to generate polars. Since the flow solutions are "warm-started", the sweep mode is also useful for obtaining solutions at stall and post-stall conditions. In the optimization mode, the program seeks to find an airfoil shape that best attains specified performance criteria.

The following input files are required to execute the program in the analysis mode:

1. *CYCLONE &OPTIMA–2D INPUT CONTROL FILE, 'opt.inp'* (user supplied)

2. *GRID DATA FILE*

3. *FILE CONTAINING PREFIXES TO BE USED FOR RESTART OUTPUT FILES* (this file is a hack required to perform optimization restarts. Further description is provided in section 2.7)

To perform an angle-of-attack or a fixed-lift Mach number sweep, the user is required to supply a *POLAR DATA FILE* in addition to the above-mentioned analysis input files. To execute the program in the optimization mode, the following input files are required in addition to the analysis input files:

1. *DESIGN VARIABLES FILE* (created by MODAIR)

2. *B–SPLINE DATA FILE* (created by MODAIR)

3. *SNOPT SPEC FILE* (required only if using the SNOPT optimizer)

The following sections describe each input file in detail. Note that all input files, with the exception of the *CYCLONE &OPTIMA–2D INPUT CONTROL FILE*, should have the same file name root. This file name root is specified as the 'grid_file_prefix' in the input control file, as can be seen in the example provided in the next section.

2.2 CYCLONE & OPTIMA–2D Input Control File

The name of the input control file used with Optima-2D Version 13 is hard-coded in subroutine *cyclone/input.f*. The input control file must be named *opt.inp*.

The expectations of the format and contents of this file have changed in version 11 (as of September 2006), with the addition of work by Billing [2] and Oldfield [16]. Input files in the Version 10a format are superseded and no longer function. A detailed discussion of the differences is beyond the scope of this document.

The contents of this file have changed in version 13 (as of October 2009), with the addition of work by Buckley [3]. Most notably, off-design constraints and operating conditions can now be specified within the input control file. These parameters are relevant when solving multipoint practical aerodynamic design problems. The input parameters from version 12 are all still valid. The notable contributions to version 12 from Tabesh [21] and Rumpfkeil [19] are

as follows: PROBE is now able to handle unsteady flows and OPTIMA–MB is capable of unsteady optimization. The approximate–factorization startup can also be omitted by adding an implicit Euler time-marching term to the steady–state equations [4]. A detailed discussion of the differences is beyond the scope of this document. All previous versions of Optima2D are contained in the central git repository for Optima. Any version is accessible by cloning the central git repository and checking out the version of interest. For more details see section 1.2.

The *opt.inp* file contains all the user adjustable parameters that affect the execution of CYCLONE & OPTIMA–2D. The *opt.inp* file is split into six sections: 1) operating (design) conditions in free format, 2) *OPTIMA namelist*, 3) *CYCLONE namelist*, 4) *PROBE namelist*, which is used for the Newton–Krylov flow solver, 5) *EXTRA namelist*, which defines the names of the input/ output files, and 6) iteration and start-up control parameters. Most entries have default settings. A sample *opt.inp* file listing some default settings is provided below, see subroutine *cyclone/input.f* for additional comments. Note that where there is no default, <n.d.> is indicated.

```
MP-OPT
 <n.d.>
WT OFF_DES C_UPP C_LOW FSMACH CL_TAR WFL CD_TAR RE DVALFA ALPHA OBJ_FUNC CLALPHA CLALPHA2 CLOPT

<n.d.> FALSE <n.d.> <n.d.> <n.d.> <n.d.> <n.d.> <n.d.> <n.d.> <n.d.> <n.d.> 6  FALSE FALSE FALSE

&OPTIMA
   OPT_METH = 1, OPT_ITER = 80, OPT_TOL = 1.d-5,
   OPT_RESTART = FALSE, GRADIENT = 1, CDF = TRUE, FD_ETA  = 1.d-7,
   COEF_FRZ = FALSE, IFRZ_WHAT = 3, WFACTOR = 0.1, BSTEP = 0.1,
   AUTO_RESTART = FALSE, NUM_RESTARTS = 5,GRID_HIS = FALSE,
   SOL_HIS = FALSE, FLO_HIS = FALSE, SC_METHOD = 2,
   USE_QUAD_PENALTY_METH = TRUE,
   WT_CYCLE_ITER = 1000,

   JMSTART = 1, JMEND = 289, JMINC = 1, KMSTART = 2, KMEND = 65,
   KMINC = 1, RHOKS = 31,

   INORD = 2, IREORD = 2,
   ILU_METH = 2, LFIL = 6, PDC = 3.0,
   IM_GMRES = 85, EPS_GMRES = 1.d-7, ITER_GMRES = 500,
```

```
   REMNX=100, REMNY=100, ERRADAP=FALSE, ERRADAPTOL=1.d-6,
   REMXMIN = -5.d0, REMXMAX = 5.d0, REMYMIN = -5.d0, REMYMAX = 5.d0,

   ORIGINAL_GRID = TRUE, MOVETE = FALSE,
   GPTOL = 1.d-15, GPMAXIT = 5000,
   GATOL = 1.d-15, GAMAXIT = 5000,

   NTCON=<n.d.>, WFL=1.0, WFD=1.0, WTC=1.0,
   ctx     = <n.d.>,<n.d.>,<n.d.>,<n.d.>,<n.d.>,<n.d.>,<n.d.>,
   cty_tar = <n.d.>,<n.d.>,<n.d.>,<n.d.>,<n.d.>,<n.d.>,<n.d.>,

   NRTCON = 1, crtxl = <n.d.>, crtxt = <n.d.>, crtxn = <n.d.>,
   crthtar = <n.d.>,

   NRC=2, WRC_CON=0.d0,
   XRC_TAR = 0.d0, 1.d0,
   RC_TAR  = 0.d0, 0.d0,
&END
&CYCLONE
   JMAX=<n.d>, KMAX=<n.d.>, JTAIL1=<n.d.>, JTAIL2=<n.d.>,

   TRANSLO=0.0, TRANSUP=0.0,
   CLTOL=1.0d-11,

    CMESH=FALSE,
   IREAD=2, BCAIRF=TRUE, SHARP=FALSE, CUSP=FALSE, CIRCUL=FALSE,
   ISPEC=1, RESTART=TRUE, STORE=TRUE,
   PHIDT=0.0, THETADT=1.0, PREC=0, PRXI=0.0,PRPHI=1.0,
   VISCOUS=TRUE, VISETA=TRUE, VISXI=FALSE, VISCROSS=FALSE,
   ORIGINALSA=FALSE,

   IDMODEL=1,
   DIS2X   = 0.0,  DIS4X = 0.02,  DIS2Y = 0.0,  DIS4Y = 0.02,
   VLXI    = 0.25, VNXI  = 0.25,  VLETA = 0.25, VNETA = 0.25,
   LIMITER = 1,    EPZ   = 1.d-3, EPV   = 5.d0,

   PREC = 0, PRXI = 0.0, PRPHI = 1.0,

   TURBULNT = TRUE, ITMODEL = 2,    ISPBC = 1,     VISCEIG  = 1.d0,
   VISCOUS  = TRUE, VISETA  = TRUE, VISXI = FALSE, VISCROSS = FALSE,
```

```
   ORIGINALSA = TRUE,

   NCP = 1000, NQ = 1000,

   SNGVALTE   = FALSE, GRDSEQ_REST = FALSE, SV_GRDSEQ = FALSE,
   WRITERESID = FALSE, TIMING      = FALSE, WRITETURB = FASLE,
   FLBUD      = FALSE, PCHORD      = 0.32,  SKNFRC    = TRUE,

   PERIODIC=FALSE, DESALL = 1.d-5, SOBMAX = 16.d0,
   OMEGAA=0.0, OMEGAF=0.0, SBETA=90.0, SBBC=FALSE,

   JOBS=1, KOBS=1, NOBS=1, outFWH=0, XOBS=0.0, YOBS=0.0,

   UNSTEAD=FALSE, JESDIRK=1,
   OUTTIME=FALSE, NOUTEVERY=1, NOUTEVERYBIG=1, DTBIG=0.1d0,
&END
&PROBE
   NK_ITS = 20,   NK_ILU   = 2,  NK_LFIL  = 2, NK_PFRZ  = 1,NK_SKIP=0,
   NK_PDC = 6.d0, NK_IMGMR = 40, NK_ITGMR = 40,NK_IENDS=0,
&END
&EXTRA
   grid_file_prefix = '<n.d.>',
   output_file_prefix = 'rest00',
   restart_file_prefix = 'rest00'
&END
&GRAPH
   graphout = false, graph_mode = 1, interval = 0,
   num_graph= 0,
   graph_pt= 0
&END
1.E-7  | AF Convergence criteria: AF_MINR
1.E-15                          | Convergence criteria: MIN_RES
1.E-2                           | Min. res. in turb. mod. : SPMIN_RES
1.E-12                          | Min. res. for subiterations : SIMIN_RES
ISEQUAL
     1
   JMXI    KMXI    IENDS    DTISEQ   DTMIS   DTOW
   <n.d.>  <n.d.>  <n.d.>   <n.d.>   <n.d.>  <n.d.>
```

The *opt.inp* file is processed in the subroutine *cyclone/input.f*. The following sections explain each input parameter. If a parameter is marked as *A–F only*, this denotes

that the parameter can be used only with the approximate–factorization algorithm, i.e. the original CYCLONE, and consequently cannot be used in the optimization mode. Example input files are shown in Subsection 2.2.8.

2.2.1 Definition of operating conditions

MP-OPT Number of operating (design) points for multi-point optimization problems. For analysis, sweep, and one-point optimization cases, MP-OPT should be set to one. The program is dimensioned for up to 20 design conditions, which is controlled in *include/optcom.inc*.

WEIGHT Weight for each operating point. The weighted-sum method is used for multi-point optimization problems

$$\mathcal{J}_{\mathrm{m}} = \sum_{i=1}^{\mathrm{MP-OPT}} \mathrm{WEIGHT}_i \, \mathcal{J}_i \tag{2.1}$$

where $\mathcal{J}_i$ denotes the objective function at each operating point, see OBJ_FUNC in Subsection 2.2.2.

OFF_DES If TRUE, the corresponding design point represents off-design operating conditions within the context of a multipoint practical aerodynamic design problem. Default is FALSE.

C_UPP The upper bound on the constraint associated with the corresponding off-design point. Inactive when OFF_DES is FALSE.

C_LOW The lower bound on the constraint associated with the corresponding off-design point. Inactive when OFF_DES is FALSE.

FSMACH Freestream Mach number.

CL_TAR Target lift coefficient, C_{L}^*, see Eq. 2.2 in Subsection 2.2.2.

WFL Objective function weight for lift functional, ω_{L}, see Eq. 2.2 in Subsection 2.2.2.

CD_TAR Target drag coefficient, C_{D}^*, see Eq. 2.2 in Subsection 2.2.2.

RE Reynolds number based on freestream velocity.

DVALFA If TRUE, angle of attack is a design variable.

ALPHA Angle of attack in degrees, positive for pitch up.

OBJ_FUNC Objective function (see subroutines *control/calcobj.f* or *control/unsteadyobjandderiv.f*).

=0 Minimize mean drag.

=1 Inverse design. Input file *.cptar* is required to specify target pressures, in addition to the files listed in Section 2.1.

=2–4 Not Active.

=5 Maximization of lift-to-drag or mean lift-to-drag ratio. Thickness constraints are described in option 6 below.

=6 Lift-constrained drag minimization, drag-constrained lift enhancement, and lift maximization. The objective function is given by

$$\mathcal{J} = \begin{cases} \omega_{\mathrm{L}} \left(1 - \frac{C_{\mathrm{L}}}{C_{\mathrm{L}}^*}\right)^2 + \omega_{\mathrm{D}} \left(1 - \frac{C_{\mathrm{D}}}{C_{\mathrm{D}}^*}\right)^2 + \text{P.T.} & \text{if } C_{\mathrm{D}} > C_{\mathrm{D}}^* \\ \omega_{\mathrm{L}} \left(1 - \frac{C_{\mathrm{L}}}{C_{\mathrm{L}}^*}\right)^2 + \text{T.C.} & \text{otherwise} \end{cases} \tag{2.2}$$

where C_{D}^* and C_{L}^* represent target drag and lift coefficients, respectively, and T.C. represents thickness penalty terms. The weights ω_{D} and ω_{L} are user-specified constants. See Section 2.2.1 for the specification of targets and ω_{L}, and see below for ω_{D} and penalty weight WTC.

=7 Maximization of lift subject to a moment constraint. The formulation of the objective function is similar to Eq. 2.2.

=8 Lift-constrained drag minimization, using variable angle of attack in the flow solver and the coefficient of drag as the objective function. Target lift coefficient is given by CL_TAR. To ensure lift constraint is achieved, should be used with DVALFA set to FALSE, and CLALPHA, CLALPHA2, and CLOPT all TRUE. Can currently only be run with circulation correction OFF.

=9 Unsteady remote inverse design. Input files *tarXXXX* are required to specify target pressures where *XXXX* is the number of a time step, e.g. *0001*. These files can be produced by setting OPT_METH to 1 and OUTTIME to TRUE. The pressures are compared along the line with k = KLINECONST which is closed off in the wake with the grid line closest to $x(j, \text{KLINECONST}) \approx \text{REMXMAX}$.

=10 Unsteady remote inverse design using FW-H. Input file *.cptar* is required to specify target pressures at observer location. The values can be obtained by an unsteady optimization run (OPT_METH=2 and OBJ_FUNC=10) with the target shape by setting OUTFWH=3. The FW-H near-field surface is defined exactly as the line for comparison in OBJ_FUNC=9.

=11 Minimize total radiated acoustic power. The objective function is given by

$$\mathcal{J} = \overline{\left(\frac{\partial}{\partial t}\int_{\mathcal{S}} n_j p_{1j}(y,t)ds\right)^2} + \overline{\left(\frac{\partial}{\partial t}\int_{\mathcal{S}} n_j p_{2j}(y,t)ds\right)^2}. \qquad (2.3)$$

Here, p_{ij} is the compressive stress tensor, n_j are the normalized components of the outward normal to the airfoil surface $\mathcal{S}$, and y is the airfoil surface position vector. The overbar denotes time-averaging over the chosen time interval, and repeated indices follow the usual Einstein summation convention.

=12 Unsteady remote inverse design. Similar to OBJ_FUNC=9 but the pressures are compared along the sides of a rectangle given by REMXMIN, REMXMAX, REMYMIN, REMYMAX and REMNX, REMNY.

=13 Minimize the vorticity (only active for PERIODIC=TRUE)

=14 Minimize far-field pressure fluctuations. The objective function is given by

$$\mathcal{J} = \sum_{n=NK\_SKIP+1}^{NK\_SKIP+NK\_IENDS} (p^n_{obs} - \bar{p}_{obs})^2 = \sum_{n=NK\_SKIP+1}^{NK\_SKIP+NK\_IENDS} (p'^n_{obs})^2 \tag{2.4}$$

where $\bar{p}_{obs}$ is the mean pressure at the observer location, which is specified by XOBS and YOBS or if both are zero by JOBS and KOBS, and $p'^n_{obs} = p^n_{obs} - \bar{p}_{obs}$ is the pressure fluctuation in the observer location at time step n. The FW-H near-field surface is defined exactly as the line for comparison in OBJ_FUNC=9.

=15 Off-design constraint function, $\Psi_{\text{off-des}} = C_l$. Evaluates the coefficient of lift at the current design iteration. To be used when solving practical aerodynamic design problems with the SNOPT optimization algorithm. This constraint function is assigned to design points representing off-design operating conditions; operating condtion parameter OFF_DES must be set to TRUE. Typically, only a lower bound, C_LOW, is set for this constraint function.

=16 Off-design constraint function, $\Psi_{\text{off-des}} = M_{max}$. Evaluates the maximum Mach number in the flow field at the current design iteration. To be used when solving practical aerodynamic design problems with the SNOPT optimization algorithm. This constraint function is assigned to design points representing off-design operating conditions; operating condtion parameter OFF_DES must be set to TRUE. Typically, only an upper bound, C_UPP, is set for this constraint function. ***NOTE*** The maximum Mach number constraint function does not work well with SNOPT because it is discontinuous. Use OBJ_FUNC 17 instead.

=17 Off-design constraint function, $\Psi_{\text{off-des}} = M_{ks}$. Estimates the maximum Mach number in the flow field using the Kreisselmeier-Steinhauser (KS) function. The KS function is used as a means to aggregate the Mach number constraints at all nodes in the flow field into a single composite function that is continuously differentiable. To be used when solving practical aerodynamic design problems with the SNOPT optimization algorithm. This constraint function is assigned to design points representing off-design operating conditions; operating condtion parameter OFF_DES must be set to TRUE. Typically, only an upper bound, C_UPP, is set for this constraint function. C_UPP must be set in conjuction with the KS function parameter RHOKS to ensure the desired maximum Mach number constraint is satisfied.

CLALPHA If TRUE, the angle of attack varies in the approximate–factorization stage of the flow solver to ensure the desired lift coefficient is achieved. If FALSE, angle of attack remains constant in approximate–factorization stage.

CLALPHA2 If TRUE, the angle of attack varies in the Newton–Krylov stage of the flow solver to ensure the desired lift coefficient is achieved. If FALSE, angle of attack remains constant in Newton–Krylov stage.

CLOPT If TRUE, angle of attack variation from flow solver is carried through to adjoint gradient calculation.

2.2.2 Definition of essential *OPTIMA namelist* parameters

OPT_METH Selection of program mode (see subroutine *control/optima2D.f*).

=1 Analysis mode: all remaining flags in *OPTIMA namelist* are ignored, with the exception of INORD and IREORD.

=2 Unsteady aerodynamic shape optimization using the unconstrained L-BFGS quasi-Newton method in conjunction with a quadratic penalty method.

=3 Aerodynamic shape optimization using the unconstrained BFGS quasi-Newton method in conjunction with a quadratic penalty method.

=4 Mach number sweep at fixed lift. This is an analysis mode for a range of Mach numbers. Achieved by using angle of attack as a design variable in single design variable optimization.

=5 Angle of attack sweep. This is similar to option 4 above, but the lift is not constrained.

=6 Mach number sweep at fixed lift. This is an analysis mode for a range of Mach numbers. Achieved by allowing the angle of attack to vary in the flow solver to ensure the desired lift is achieved.

=7 Error estimation and/or time step adaptation for unsteady flows. The unsteady adjoint is used to estimate the value of a functional given by OBJ_FUNC using half the current time step size.

=8 Aerodynamic shape optimization using SNOPT. SNOPT is a sequential quadratic programming algorithm for constrained optimization problems.

OPT_ITER Maximum number of objective function evaluations.

OPT_TOL Optimization exit tolerance. If the L_2 norm of the gradient vector is below OPT_TOL, the program terminates.

OPT_RESTART TRUE if the optimization is being restarted from a previous optimization.

GRADIENT Method for gradient computation

=0 Finite-difference gradient, used mostly for diagnostic purposes.

=1 Adjoint gradient. Finite differences are used to form most terms in the adjoint equations.

=2 Matrix-free sensitivity gradient, used mostly for diagnostic purposes.

=4 Augmented adjoint method, default. The grid perturbation is explicitly included in the adjoint formulation, and all terms in the adjoint equations are formed analytically. CAUTION: Is only implemented for a few cases for unsteady optimization.

OBJ_RESTART TRUE if optimization is being restarted from previous optimization and previous initial objective functions are to be used in normalization. Only used if SCALE_OBJ is TRUE.

SCALE_OBJ Logical flag for objective function scaling for steady optimization. The objective is scaled by its value at the first design iteration for TRUE (default), and remains unscaled for FALSE.

ERRADAP Logical flag to indicate whether time step sizes are adapted if OPT_METH=7. Default is FALSE.

ERRADAPTOL If the estimated error in the functional is greater than ERRADAPTOL keep refining the time step sizes if ERRADAP=TRUE (default 10^{-6}).

WFD Objective function weight for drag functional, ω_{D} in Eq. 2.2.

WTC Objective function weight for thickness constraints It applies to both fixed and floating constraints.

WSTART Logical flag for warm-starts, only used in sweep and optimization modes

=TRUE Warm-start flow solutions, default.

=FALSE Initialize flow solutions to freestream.

BSTEP Stepsize for the first iteration of the optimization. The range of values is between $10^{-2} - 1.0$, with 1.0 being the default.

INCR Selects grid perturbation method.

$= 0$ Algebraic grid perturbation, default.

$= 1$ Elasticity grid perturbation.

> 1 Elasticity grid perturbation, using multiple increments and mesh stiffening between increments. INCR increments are used. Changing the maximum value (2) will require adjusting the parameter *MAXINCR* in the file *include/optcom.inc*.

WAC A switch used to enable airfoil optimization subject to an area constraint. For positive, non-zero values of WAC, the optimization problem will be subject to an area constraint (default = 0.0). If the BFGS unconstrained optimization algorithm is used, WAC also specifies the weight applied to penalty term associated with the area constraint.

AREAFAC The factor applied to the initial airfoil area when an area constraint is used. If the BFGS unconstrained optimization algorithm is used, the area constraint is satisfied if the area of the current design iteration is exactly equal to AREAFAC × AREAINIT. If the SNOPT constrained optimization alogorithm is used, the area constraint is satisfied if the area of the current design iteration is greater than or equal to AREAFAC × AREAINIT.

NTCON Number of **fixed** thickness constraints, up to 20 are allowed. Changing the maximum value will require adjusting the parameter *maxtcon* in the file *include/optcom.inc*.

CTX x/c locations of fixed thickness constraints. The locations are separated by a comma.

CTY_TAR Target thicknesses of the fixed constraints. The thicknesses correspond to respective constraints in CTX, and are separated by a comma.

NRTCON Number of **floating** thickness constraints, up to 10 are allowed. Changing the maximum value will require adjusting the parameter *nthc* in the file *include/optcom.inc*.

CRTXL The lower limit of the floating thickness constraint

CRTXT The upper x/c limit of the floating thickness constraint

CRTXN The number of stations where the constraints are applied

CRTHTAR The target thickness

NRC Number of points used to define the peacewize-linear radius of curvature constraint target. Changing the maximum value (20) will require adjusting the parameter *NRC_MAX* in the file *include/optcom.inc*.

WRC_CON Weight for radius of curvature constraint volation penalty.

XRC_TAR The x-coordinates of the points defining the peacewize-linear target radius of curvatrue. These must occur in ascending order (same targets apply to upper and lower surface), and the first and last entries should be 0 and 1.

RC_TAR The target radius of curvatrue at each vertex in the peacewize-linear relation.

WT_EXP The value of the exponent used in the off-design weight update formula. Applicable when performing practical multipoint aerodynamic design problems using the off-design weight update procedure.

WT_CYCLE_ITER The number of optimization iterations per weight update cycle. A weight update cycle consists of a number of optimization iterations followed by a procedure that modifies the weights of off-design objective functions. If weight update cycles are not required, set this value to 1000.

WT_AOA Logical flag to turn on angle of attack sweep during off-design weight update procedure to determine $C_{l,max}$ at off-design points with a high-lift constraint. Default = FALSE.

JMSTART The streamwise index of the first grid node used in the calculation of the KS function for estimating the maximum Mach number in a flow field. Applicable to cases using the SNOPT optimization algorithm (OPT_METH=8) and having maximum Mach number off-design constraints (OBJ_FUNC=17).

JMEND The streamwise index of the last grid node used in the calculation of the KS function for estimating the maximum Mach number in a flow field. Applicable to cases using the SNOPT optimization algorithm (OPT_METH=8) and having maximum Mach number off-design constraints (OBJ_FUNC=17).

JMINC The streamwise index of nodes used in the calculation of the KS function for estimating the maximum Mach number in a flow field is incremented by JMINC. Applicable to cases using the SNOPT optimization algorithm (OPT_METH=8) and having maximum Mach number off-design constraints (OBJ_FUNC=17).

KMSTART The normal index of the first grid node used in the calculation of the KS function for estimating the maximum Mach number in a flow field. Applicable to cases using the SNOPT optimization algorithm (OPT_METH=8) and having maximum Mach number off-design constraints (OBJ_FUNC=17).

KMEND The normal index of the last grid node used in the calculation of the KS function for estimating the maximum Mach number in a flow field. Applicable to cases using the SNOPT optimization algorithm (OPT_METH=8) and having maximum Mach number off-design constraints (OBJ_FUNC=17).

KMINC The normal index of nodes used in the calculation of the KS function for estimating the maximum Mach number in a flow field is incremented by KMINC. Applicable to cases using the SNOPT optimization algorithm (OPT_METH=8) and having maximum Mach number off-design constraints (OBJ_FUNC=17).

RHOKS The value of the draw-down parameter, ρ, used in the calculation of the KS function for estimating the maximum Mach number in a flow field. Applicable to cases using the SNOPT optimization algorithm (OPT_METH=8) and having maximum Mach number off-design constraints (OBJ_FUNC=17).

GRID_HIS Logical flag to turn on grid logging. When GRID_HIS is TRUE, the grid file (*.g* extension) at each design iteration is saved to the *history* folder.

SOL_HIS Logical flag to turn on flow solution logging. When SOL_HIS is TRUE, the flow solution file (*.q* extension) at each design iteration is saved to the *history* folder.

FLO_HIS Logical flag to turn on flow solution convergence history logging. When FLO_HIS is TRUE, the flow solution convergence history file (*.his* extension) at each design iteration is saved to the *history* folder.

2.2.3 Definition of optional *OPTIMA namelist* parameters

The following parameters should not require modification, they are mostly used for research and diagnostic purposes.

INORD Initial ordering of grid nodes.

=1 Natural ordering.

=2 Reverse (upstream) ordering, default.

=3 Double bandwidth for 3-block C-grids.

IREORD Reordering of grid nodes.

=0 No reordering.

=1 Cuthill–McKee reordering.

=2 Reverse Cuthill–McKee reordering, default.

COEF_FRZ "Freezing" non-differentiable functions in the residual equations.

IFRZ_WHAT Selection of freezing parameter.

ITER_GMRES Maximum number of GMRES iterations for gradient computation when using the adjoint and sensitivity methods. The default value is 500.

IM_GMRES Size of GMRES search space for gradient computation when using the adjoint and sensitivity methods. The default value is 85 and changing this value above the default value will require adjusting the parameter *iwk_gmres* in subroutines *control/adjoint.f* and *control/sensit_mf.f.*

EPS_GMRES Exit criterion for GMRES gradient computation when using the adjoint and sensitivity methods. The default value is 10^{-5}, i.e. a five order of magnitude reduction.

LFIL Level of fill in the preconditioner for gradient computation when using the adjoint and sensitivity methods. The default value is 6. Increasing this value above 7 will require adjusting the parameter *iwk* in subroutines *control/adjoint.f* and *control/sensit_mf.f.* This parameter has a large influence on memory usage.

PDC Constant used to enhance the diagonal of the preconditioner for gradient computation when using the adjoint and sensitivity methods. The default value is 3.0.

AUTO_RESTART Logical switch for using automatic restart if BFGS stalls, default FALSE. Automatic restart cannot be used with grid sequencing.

NUM_RESTART Set the total number of automatic restarts. Default = 0

ORIGINAL_GRID Logical flag for differentiation of algebraic grid perturbation. For TRUE, grid derivatives use arc lengths computed on the parent grid (default). For FALSE, grid derivatives use arc lengths computed on the grid found at the most recent optimizer iteration (this gives a gradient calculation that is inconsistent with the objective function).

MOVETE Logical flag only active if INCR is not zero. If true the cubic polynomial for the trailing edge is formed (default false).

GPTOL Residual tolerance for solving the elasticity grid perturbation systems (default 10^{-15}).

GPMAXIT Maximum number of congugate gradient iterations permitted in solving the elasticity grid perturbation systems (default 5000).

GATOL Residual tolerance for solving the grid adjoint systems (default 10^{-15}).

GAMAXIT Maximum number of congugate gradient iterations permitted in solving the grid adjoint systems (default 5000).

REMXMIN Left border of rectangle which defines near-field plane for OBJ_FUNC=12 (default=-5.0).

REMXMAX Right border of rectangle which defines near-field plane for OBJ_FUNC=12 (default=5.0). Also gives the location of the line in the wake for OBJ_FUNC = 9, 10 or 14.

REMYMIN Lower border of rectangle which defines near-field plane for OBJ_FUNC=12 (default=-5.0).

REMYMAX Upper border of rectangle which defines near-field plane for OBJ_FUNC=12 (default=5.0).

REMNX Number of points along lower and upper border (default=100).

REMNY Number of points along left and right border (default=100).

2.2.4 Definition of essential *CYCLONE namelist* parameters

UNSTED Logical flag to determine if the run is steady (UNSTED=FALSE) or unsteady (UNSTED=TRUE), default is FALSE.

JESDIRK Time-marching method.

=1 Second-order accurate backwards difference (BDF2), default.

=3 Explicit first stage, single diagonal coefficient, diagonally implicit Runge-Kutta scheme of third-order (ESDIRK3).

=4 Explicit first stage, single diagonal coefficient, diagonally implicit Runge-Kutta scheme of fourth-order (ESDIRK4).

DTBIG Time step size for the adjusting period if NK_SKIP is nonzero, default is 0.1.

PERIODIC If TRUE O-mesh for cylinder cases is assumed, default is FALSE.

METH Numerical Method : METH=1 for diagonal form with subiteration method, METH=2 for block pentadiagonal method, METH=3 for diagonal form (i.e. scalar pentadiagonal), METH=4 same as METH=1 except a modified dissipation scheme is used on the implicit side reducing the system to a scalar tridiagonal form, METH=5 same as METH=4 but also employing the Frozen Eigenvalue scheme (default =3).

CIRCUL Logical switch for application of circulation correction at farfield boundary.

=TRUE Circulation correction is ON. This is recommended for better accuracy.

=FALSE Circulation correction is OFF, results in faster convergence of optimization problems (recommended).

RESTART Logical flag to denote if a restart run is to be performed, default FALSE.

TRANSLO *A–F* only: transition location, (x/c), on lower surface (default = 0.0).

TRANSUP *A–F* only: transition location, (x/c), on lower surface (default = 0.0).

VISCOUS Logical flag to denote viscous flow - TRUE for viscous flow or FALSE for inviscid flow (default = TRUE). See also TURBULNT below.

IDMODEL Artificial dissipation model.

=1 Scalar dissipation, default.

=2 *A–F* only: Matrix dissipation.

=3 *A–F* only: Constant coefficient.

=4 *A–F* only: CUSP dissipation.

LIMITER Limiter used with CUSP scheme, default 3.

=1 3rd order dissipation

=2 1st order dissipation

=3 Zingg-Nemec limiter, pressure only

=4 Zingg-Nemec limiter

=5 Venkatakrishnan limiter

DIS2X Second-difference X coefficient for artificial dissipation (default = 1.0). For subsonic cases, this can often be set to 0.0. For optimization problems, setting DIS2X to 0.0 for subsonic and transonic flow results in a better optimization convergence.

DIS2Y Second-difference Y coefficient for artificial dissipation (default = 1.0). For subsonic cases, this can often be set to 0.0. For optimization problems, setting DIS2X to 0.0 for subsonic and transonic flow results in a better optimization convergence.

DIS4X Fourth-difference X coefficient for artificial dissipation (default = 0.01).

DIS4Y Fourth-difference Y coefficient for artificial dissipation (default = 0.01).

VLXI Matrix dissipation limiter used in ξ direction, scales amount of dissipation applied to first and second characteristic equations (default =0.1).

VLETA Matrix dissipation limiter used in η direction, scales amount of dissipation applied to first and second characteristic equations (default =0.1).

VNXI Matrix dissipation limiter used in ξ direction, scales amount of dissipation applied to third and fourth characteristic equations (default =0.1).

VNETA Matrix dissipation limiter used in η direction, scales amount of dissipation applied to third and fourth characteristic equations (default =0.1).

SMU Dissipation coefficient used on explicit side for constant coefficient dissipation (default =0.6).

SMUIM Dissipation coefficient used on implicit side for constant coefficient dissipation (default =0.6).

PREC Local preconditioning variable: PREC=0 for preconditioning off, PREC=3 for preconditioning on. Note that preconditioning is usually run with JACDT=2 (default =0).

PREXI Preconditioning limiter : scaling factor (related to local diffusion velocity) used in determining the amount of local-preconditioning used (default =0.0).

PRETA Preconditioning limiter : scaling factor (related to free-stream Mach number) used in determining the amount of local-preconditioning used (default =1.0).

IORD Integer value which determines the spatial order of the discretization scheme. These settings apply to both inviscid and viscous subroutines.

=2 Second-order centred differences are used, default.

=4 *A–F* only: Fourth-order centred differences.

INTEG Integer value which determines the spatial order of the force integration routine.

=2 Second-order scheme, default.

=3 *A–F* only: Third-order scheme.

SNGVALTE Logical flag to indicate if the trailing edge of each element should be single or multi-valued:

=TRUE *A–F* only: Averages the upper and lower surface values of the trailing edge point. All solution and turbulence variables are averaged. This ensures a single-valued trailing edge point. It is suggested that this flag be set to TRUE for the A–F algorithm, as single-valued points generally improve the residual convergence.

=FALSE Dual-value trailing edge, default. This is required for Newton–Krylov algorithm and optimization runs.

ISMODEL Subiteration method : ISMODEL=1 for single-time-step subiteration method, ISMODEL=2 for dual-time-stepping subiteration method [see reference [8] for details] (default =2). Only active for unsteady flows.

2.2.5 Definition of optional *CYCLONE namelist* parameters

The following parameters should not require modification, they are mostly used for research and diagnostic purposes.

BCAIRF Logical flag for solid wall boundary conditions. This is only used as a debugging or code verification tool following any modification.

=TRUE Airfoil body, default.

=FALSE *A–F only*: Uniform flow.

BCFARF Logical flag to activate the far-field boundary conditions. This is only used as a debugging or code verification tool following any modification.

TRUE Far-field ON, default.

FALSE Far-field OFF for the uniform flow test case.

DTRATE Scaling factor for the variable time step option based on the Jacobian. This allows for larger time steps when the Jacobian is large, near the body for instance (default = 1.0).

IREAD Denotes format of grid file.

=0 Generate cylinder grid internally.

=1 Formatted grid file.

=2 Unformatted grid file, default.

DSWALL Off-wall spacing for internally generated cylinder grids (IREAD=0), default is 10^{-5}.

SOBMAX Far-field boundary for internally generated cylinder grids (IREAD=0), default is 16.0.

NCP Number of iterations between successive pressure coefficient file updates (default = 100).

NQ Number of iterations between successive Q-file updates (default = 100).

ORDERXY *A–F* only: Order of operations for implicit integration - *true* for $\xi - \eta$ order or *false* for $\eta - \xi$ order (default = *true*).

JSUBMX Number of subiterations to be used (default =2) for unsteady flows (A-F only).

ISPMOD Time-marching method used in Spalart-Allmaras turbulence model (A-F only): ISPMOD=1 as described in [20] (use for steady-state runs), ISPMOD=3 for dual-time-stepping inside model (second-order in time so use for unsteady runs) (default =1).

PHIDT Time marching parameter

=0 THETADT=1 Euler implicit method, default.

=0 *A–F* only: THETADT=0.5 Trapezoidal method.

=0.5 *A–F* only: THETADT=1 Three- point backwards method.

THETADT Time marching parameter, see PHIDT.

RETINF Freestream turbulent Reynolds number. This variable is used in the turbulence models and need not be changed without good reason and understanding, default = 0.001.

STORE Logical to write restart data file, default=TRUE.

STRTIT Number of iterations over which the slow start is spread (default = 12, if MG=TRUE then STRTIT is set to 6). It is unlikely that this should be changed.

TINF Temperature at infinity in degrees Rankine (default = 460 degrees Rankine).

TURBULNT Logical flag to indicate flow type when VISCOUS =TRUE.

=TRUE Turbulent flow, default.

=FALSE Laminar flow.

ITURB Integer variable used to select the turbulence model.

=2 *A–F* only: Baldwin-Barth turbulence model.

=3 Spalart-Allmaras model, default.

=4 *A–F* only: Menter SST model.

ORIGINALSA Logical flag for using original S-A model.

=TRUE Use original S-A turbulence model, default.

=FALSE Use S-A model with Ashford fix [1]. Appears to be useful for unsteady cases.

NNIT The maximum number (integer) of subiterations performed by the Spalart-Allmaras turbulence model, default=1. Used when startup problems are encountered, suggested value is 3.

VISCROSS Logical flag for viscous cross terms - TRUE for explicit viscous cross terms or FALSE for no explicit viscous cross terms (defaul=FALSE).

VISETA Logical flag for viscous terms in eta - TRUE for explicit viscous terms in eta or FALSE for no explicit viscous terms in η (default = TRUE).

VISXI Logical flag for viscous terms in ξ - TRUE for explicit viscous terms in ξ or FALSE for no explicit viscous terms in ξ (default = FALSE).

WRITERESID Logical flag to write residuals to file, default = FALSE. This is an advanced debugging tool.

WTRAT Ratio of fixed wall temperature to temperature at infinity - WTRAT=0.0 for adiabatic wall conditions, WTRAT > 0.0 for isothermal wall conditions. The wall temperature will then be $WTRAT * TINF$, default = 0.0.

TIMING Logical flag to output the total cpu time at every iteration to file (units are in seconds), default=FALSE. This flag is presently not used, timing information is output in the residual file.

ZEROTURRE Logical flag to initialize the eddy viscosity in the Spalart-Allmaras turbulence model to RETINF. This variable is useful on restarts when there is concern over the uniqueness of the final solution, default = FALSE.

GRDSEQ_REST Perform restart from a grid sequencing run, default FALSE.

SV_GRDSEQ Save solution during a grid sequencing run, default FALSE.

FLBUD Output flux budget and velocity profile, default FALSE.

PCHORD Fraction of chord where to output flux budget.

SKNFRC Skin friction output, default FALSE.

OUTTIME Logical flag to write unsteady solution for each time step to harddisk if OUTTIME=TRUE, default is FALSE.

NOUTEVERY Writes unsteady solution every NOUTEVERY time steps to harddisk if OUTTIME=TRUE, default is 1.

NOUTEVERYBIG Writes unsteady solution in adjusting period every NOUTEVERYBIG time steps to harddisk if OUTTIME=TRUE and NK_SKIP is nonzero, default is 1.

OUTFWH Integer variable used to control the output from the FW-H solver.

=0 No output, default.

=1 Output for pressure fluctuation comparison plots between FW-H and CFD in time domain.

=2 Output for directivity plots, observers are located on a circle with radius XOBS.

=3 Output for target pressures in time domain for use with OBJ_FUNC=10.

XOBS X-coordinate of far-field observer location, default is 0.0.

YOBS Y-coordinate of far-field observer location, default is 0.0.

JOBS *J*-index of observer location if XOBS=YOBS=0.0, default is 1.

KOBS *K*-index of observer location if XOBS=YOBS=0.0, default is 1.

SBBC *Very preliminary*: Logical flag to indicate suction and blowing boundary condition, default is FALSE.

OMEGAA Amplitude of angular velocity for a rotating cylinder (PERIODIC=TRUE) and *very preliminary* amplitude of suction/blowing. Default is 0.0.

OMEGAF Frequency of angular velocity for a rotating cylinder (PERIODIC=TRUE) and *very preliminary* frequency of suction/blowing. If zero it is steady rotation or suction/blowing with amplitude OMEGAA (default is 0.0).

SBETA *Very preliminary*: Angle in degrees of the suction and blowing with respect to the perturbation slot (default is 90.0).

2.2.6 Definition of *PROBE (Newton–Krylov Solver) namelist* parameters

NK_ITS Number of Newton ("outer") iterations for steady runs and also number of Newton iterations per time step for unsteady flows. Setting NK_ITS to zero turns off the Newton–Krylov solver. Default value is 20 and should be adjusted to about 8 for unsteady runs. These iterations are in addition to the IENDS iterations, see subsection 2.2.7.

NK_LFIL Level of fill in flow solver preconditioner. Default value is 2.

NK_PFRZ Newton iteration after which the preconditioner is frozen. Default value is 1 which works great for unsteady cases, however, it should be equal to NK_ITS for steady runs.

NK_PDC Constant used to enhance the diagonal of the preconditioner for the flow solver. Default value is 6.0, and this value should not require adjustment.

NK_IMGM Size of GMRES search space for flow solution. Default value is 40, and this value should not require adjustment.

NK_ITGMR Maximum number of "inner" GMRES iterations (for each Newton iteration). Default value is 40, and this value should not require adjustment.

NK_IENDS Number of time steps for unsteady flows. Default value is 0.

NK_SKIP Number of time steps which do not contribute to objective functions to account for adjusting period. The total number of time steps is then NK_SKIP + NK_IENDS. Default value is 0.

DISSCON Logical flag to turn on dissipation-based continuation scheme (default = FALSE). Dissipation-based continuation is a scheme for globalizing the Newton-Krylov method used in the PROBE flow-solver. Currently, dissipation-based continuation is only implemented for inviscid flow solutions.

LAMDISSMAX The initial value of the dissipation-based continuation parameter LAMDISS (default = 0.01).

DCTOL Sets the level of modified residual reduction after which the dissipation-based continuation parameter LAMDISS is reduced (default = 0.1). For a given value of LAMDISS, a residual reduction ratio is calculated at each Newton iteration as:

$$\text{RFRATIO} = \frac{\text{RESIDF}}{\text{RESIDF}_0}$$

where RESIDF is the value of the modified residual evaluated using the current value of LAMDISS at the current Newton iteration and RESIDF_0 is the intial value of the modified residual evaluated using the current value of LAMDISS. If RFRATIO < DCTOL then LAMDISS is reduced.

LAMKILL Sets the value of the unmodified residual where the dissipation-based continuation scheme is turned off and pure Newton-Krylov iterations take over until residual is converged to the flow-solver termination value MIN_RES (default = 0.1e-5).

2.2.7 Iteration and startup control

ISEQUAL Integer number of sequences for time-stepping and grid-sequencing. This is not supported for optimization runs, but works well for analysis runs.

JMXI Integer maximum number of grid points in the J direction.

KMXI Integer maximum number of grid points in the K direction.

IENDS Integer number of iterations for execution of the approximate-factorization algorithm for steady runs. For most cases, 500 to 5,000 iterations produces reasonable convergence. Setting it to zero omitts the approximate–factorization startup by adding an implicit Euler time-marching term to the steady–state equations [4] (see subroutine *control/timevec.f*). It is automatically set to zero for unsteady runs.

DTISEQ Time-step size for steady and unsteady flows. A value of 5.0 works well for most cases. For turbulent cases, a reduced time-step of 2.0-3.0 may be required. For the case where JACDTSEQ=3 (i.e., constant CFL), DTISEQ is the CFL number.

DTMINS Minimum allowable time step when using local time-stepping. A value of 0.0 works well for most cases.

DTOW Second time-step used in dual-time-stepping subiteration scheme (unsteady runs). $DTOW$ (or $\Delta\tau$) is used to drive the subiteration process to convergence, whereas $DTISEQ$ (or Δt) governs the temporal accuracy of the iteration. If $ISMODEL \neq 2$, then $DTOW$ is not used and is therefore arbitrary.

2.2.8 Examples

Example input files are provided below to further demonstrate the use of CYCLONE & OPTIMA–2D. The first example shows time-step sequencing for a 5,000 iteration run using two grids and time-steps. The column with line numbers is provided for convenience. The first 1000 iterations are run using a time-step of 5.0 (Line 4) on a 269×65 grid. The next 4000 iterations use a time-step of 3.0 (Line 5), on a finer grid of 537×129.

```
Example 1
```

```
---------

Line #|
  1   |    ISEQUAL
  2   |        2
  3   |    JMXI    KMXI    IENDS    DTISEQ   DTMIS   DTOW
  4   |    269     65      1000      5.d0     0.0     1.d1
  5   |    537     129     4000      3.d0     0.0     1.d1
```

Example 2 shows the definition of operating conditions for a four-point optimization problem. The goal of the optimization is to determine an airfoil shape that attains a lift coefficient of 0.733 and a drag coefficient of 0.013 over the indicated Mach number range. This is a classic transonic optimization problem for the design of efficient cruise sections. The angle of attack is a design variable at each operating point (in addition to shape design variables, which are discussed Section 2.5 and also Chapter 5), and the initial angle of attack varies from 3.0° to 1.0°. For this case OPT_METH = 3, and WFD = 0.1. There are three thickness constraints defined at 35, 96, and 99 percent chord locations.

```
Example 2
---------

 MP-OPT
   4
 WT OFF_DES C_UPP C_LOW FSMACH CL_TAR WFL CD_TAR RE DVALFA ALPHA OBJ_FUNC CLALPHA CLALPHA2 CLOPT
 1.0    FALSE    0.00    0.00    0.68   0.733  1.0 0.013 2.7E6  TRUE  3.0   6  FALSE   FALSE  FALSE
 1.0    FALSE    0.00    0.00    0.71   0.733  1.0 0.013 2.7E6  TRUE  2.0   6  FALSE   FALSE  FALSE
 2.0    FALSE    0.00    0.00    0.74   0.733  1.0 0.013 2.7E6  TRUE  2.0   6  FALSE   FALSE  FALSE
 3.0    FALSE    0.00    0.00    0.76   0.733  1.0 0.013 2.7E6  TRUE  1.0   6  FALSE   FALSE  FALSE

.
.
.

 NTCON=3, WFL=1.0, WFD=0.1, WTC=1.0,
 TX1 = 0.35,   TX2 = 0.96,  TX3 = 0.99,   TX4 = 0.0, TX5 = 0.0,
 TY1 = 0.1204, TY2 = 0.005, TY3 = 0.0012, TY4 = 0.0, TY5 = 0.0
```

Example 3 shows a three-point optimization problem, which was motivated by a leading-edge design study for a single-element airfoil. The first operating point denotes a lift maximization at a low Mach number with the angle of attack as a design variable. Note that a high value of CD_TAR is specified such that the drag functional in Eq. 2.2 is not active. The second design condition represents cruise, where the airfoil shape is

required to maintain a specified lift (0.3) and attain low drag (0.012). The last operating point represents a dive condition, where the lift functional in not active (WFL=0.0) and the angle of attack is kept fixed at $-6.0°$. The drag is required not to exceed 0.022, which prevents separation on the lower surface of the airfoil.

```
Example 3
---------

 MP-OPT
   3
 WT OFF_DES C_UPP C_LOW FSMACH CL_TAR WFL CD_TAR RE DVALFA ALPHA OBJ_FUNC CLALPHA CLALPHA2 CLOPT

 4.0  FALSE    0.00    0.00   0.15  1.8    1.0 1.0   5.39E6 TRUE  14.0   6  FALSE FALSE FALSE
 1.0  FALSE    0.00    0.00   0.3   0.3    1.0 0.012 5.39E6 TRUE   1.0   6  FALSE FALSE FALSE
 1.0  FALSE    0.00    0.00   0.3   0.3    0.0 0.022 5.39E6 FALSE -6.0   6  FALSE FALSE FALSE
```

2.2.9 Optima2D Test Suite

Additional examples of input files for various optimization and anaylsis scenarios can be found in the Optima2D test suite. To obtain a copy of the Optima2D test suite, clone the GIT repository located on the oddjob server at:

```
/nfs/carv/d1/people/optima/optima2D_testSuite.git
```

For details on cloning a GIT repository, refer to a similar procedure presented in section 1.3. A summary of the test cases contained within the Optima2D test suite can be found on the web at the following link:

```
http://spreadsheets.google.com/pub?key=tIKe95EqCrUOvM6Looc1IsQ&single=true&gid=0&output=html
```

2.3 Grid Data File

This file contains the grid for the computations and is in a double precision version of *PLOT3D* unformatted 2d single-block format. It is intended that this file be generated by AMBER2d or DHGRD, but it may be generated using any suitable grid generator. The file name must have a *'.g'* extension. The format is shown in section 7.2. The file name root is specified as the grid_file_prefix, see section 2.2. It is important to use *checkgrid* for AMBER2d grids to ensure that blocks align to double precision before conversion to single-block grid.

2.4 Polar Data File

This file, denoted by the *'.pol'* extension, is used for angle of attack sweeps and Mach number sweeps, OPT_METH=4 and 5. The first line of the file contains an integer that indicates how many angles of attack (or Mach numbers) are considered, and the remaining lines list the values of interest. For example, the following file would specify 6 angles of attack of interest.

```
6
10.0
12.0
13.0
13.5
13.75
14.0
```

2.5 Design Variables File

This file, denoted by the *'.dvs'* extension, is generated by MODAIR, see chapter 5. The file contains the shape design variables that are used for an optimization problem. A typical example is shown below. The first column indicates the B-spline control point index (clockwise starting from the lower surface trailing edge), and the secong column is the design variable value, which is the y-coordinate of the B-spline control point.

```
 3 -0.7708487646681778E-02
 4 -0.4355552515410239E-02
 5 -0.3644850198856511E-02
 6 -0.1803831601831673E-01
```

2.6 B-Spline Data File

This file, denoted by the *'.bsp'* extension, is generated by MODAIR, see Chapter 5. The file contains the definition of the B-spline curves that parameterize the airfoil shape. The first number is the order of the B-splines, followed by the number of control points, and the number of grid points on the body of the airfoil. The control points are listed next, followed by the the parameter vector, and the B-spline knots.

2.7 *'file_prefixes'* File

The *'file_prefixes'* file contains the prefixes of the output and restart files that are needed to perform seamless restarts. At the beginning of a flow analysis or optimization run, the contents of this file must look exactly like this:

```
rest00
rest00
F
F
```

From the top, a description of each line is as follows: *output file prefix*, *restart file prefix*, *OPT_RESTART* flag, *OBJ_RESTART* flag. At the onset of an optimization run, the contents of the *'file_prefixes'* file are automatically modified. Manually resetting the contents to the above values is typically required when starting a fresh optimization run.

2.8 SNOPT Specs File

The SNOPT specs file has the file name *'SNOPT.spc'*. It contains a list of options and values used by the SNOPT optimizer. For a description of parameters that may be specified in this file, see the SNOPT 7 User's Guide. This file must be present in the working directory whenever the SNOPT optimizer is employed.

Chapter 3

EXECUTION

To perform a run, including restart runs, the syntax is:

$$OPTIMA2D\_xxx > output.file$$

where *OPTIMA2D_xxx* is the appropriate executable generated by the compiling procedure described in the *INSTALL* text file located in the *optima* directory.

For a flow solver or optimization restart, the RESTART flag in the input file should be set to TRUE and the appropriate restart prefix specified, i.e. the root of the file name of the run that the user would like to restart from. Note that one can first converge an analysis run and then restart from the solution and perform an optimization run. Alternatively, for an optimization restart the restart *.dvs* file and the original *.bsp* file should be linked to the new grid prefix with RESTART=FALSE. The optimization is then restarted from the freestream conditions. Presently, for optimization restarts the first search direction is set to the steepest descent direction.

Chapter 4

Output Files

4.1 Output and Restart Filename Prefixes

Previously, the prefixes of the output filename and restart filename were specified in the program input file, *opt.inp*, by the variables 'output_file_prefix' and 'restart_file_prefix'. In the current version, the prefixes of the output filename and restart filename are defined within the code as *rest(##)-(PP)* and *rest(##-1)-(PP)*, respectively. The 2 digit integer *##* represts the number of restarts that have been performed and *PP* is a 2 digit integer representing the processor number. For an optimization or flow analysis performed with N operating conditions (defined by the MP-OPT variable in the *'opt.inp'* file), N processors are required. The processor number *PP* may take on the range of values from 0 to $(N-1)$. Output filename prefixes with the *rest(##)-(PP)* format contain design-point-specific output, such as the flow solution contained in output files with extension *.q*. An additional set of output files with the *rest(##)* prefix format contain output that is common to all design points, such as design variable history and grid file contained in output files with extensions *.dvhis* and *.g* respectively.

4.2 Output and Restart Filename Extensions

The possible output and restart filename extensions are listed below, including the content and format of each file. This list describes only most important files. For additional details see *include/units.inc*, where all files are defined, and the subroutines *cyclone/ioall.f* and *cyclone/input.f*.

.ac History of airfoil (x, y) coordinates. This file is created only if OPT_METH=3, i.e. a steady optimization run, and is updated after each successful line search. During the optimization, the file holds the current design as the optimization progresses. If the optimization converges, the file holds the history of all designs examined.

.bcp B-spline control point values.

.best Points that define the most successful airfoil achieved in the steady optimization.

.bsp B-spline control point values, same format as input file.

.cf Skin friction coefficient and related viscous data, see file header.

.cp History of pressure distribution, similar to .ac file. The entries are: x, C_p.

.dbc Dissipation-based continuation output file. Prints out values of parameters relevant to the dissipation-based continuation scheme at each Netwon-Krylov interation.

.dvhis Design variable history for each iteration of a steady optimization run. The order of design variables is the same as defined in the input .dvs file, see Section 2.5. The last entry per iteration is the angle of attack design variable, if active.

.dvs Design variables for optimization runs. This file is updated after each successful line search, and hence, holds the best design variables obtained as the optimization progresses. Mostly used for restarts.

.g This file is a soft-link to the original input file, which is convenient for restarts and plotting.

.ghis Gradient convergence history for steady optimization runs, updated at the end of each search direction. Columns are: search direction, number of objective function evaluations, norm of gradient vector.

.gmres GMRES residual output for gradient computations during optimization runs.

.gvhis Gradient value for each design variable, updated at each iteration of a steady optimization run.

.hest Hessian estimates.

.his Flow solver convergence history. The definition of residual has changed. The residual value is defined as the L_2 norm of the total residual vector, i.e. Navier–Stokes equations and the turbulence model. The columns are: iteration, CPU time, residual, location of maximum density residual (j index, k index), maximum density residual, lift, drag, angle of attack, inviscid and viscous lift coefficients, as well as inviscid and viscous drag coefficients.

.ld Lift, drag, and moment values. Moment coefficient is based at $(0.25, 0.0)$ with a chord of unity.

.m Maximum Mach number history. The maximum Mach number in the flow field is calculated at each iteration for off-design points (OFF_DES = TRUE) with a maximum Mach number constraint. If the BFGS optimizer is used (OPT_METH = 3), the columns are: design iteration, search direction, maximum Mach number, j index, and k index of node where maximum Mach number occurs. If the SNOPT optimizer is used (OPT_METH = 8), the columns are: design iteration, search direction, maximum Mach number, estimate of maximum Mach number using KS function, j index, and k index of node where maximum Mach number occurs.

.obj Initial objective function values (for each design point). Used for restarting the steady optimization from the same initial objective function values.

.ohis Optimization convergence history. The columns for steady optimization are: iteration, search direction, objective function value, norm of gradient vector, lift, drag, moment, angle of attack, flow solver residual, and trailing edge angle. The columns for unsteady optimization are: iteration, number of function and gradient evaluations, objective function value, maximums and L_2 norm of gradient vector, norm of projected gradient vector, all design variables, all gradients.

.opt General output file for the optimization. Most error flags are directed to this file.

.q Binary file for solution vector Q. This contains the state vectors, i.e. $[\rho, \rho u, \rho v, e]^T$, for all nodes in double precision unformatted single-block 2d PLOT3D format. It is written upon execution completion and every NQ iterations to guard against pesky machine crashes. To read this file into PLOT3D or TECPLOT, the file must be converted into single precision (see Chapter 6).

.q2 Binary file for solution vector at previous time step (only written for UNSTED=TRUE).

.scr Summary or *screen* output from Optima2D. In previous versions of Optima2D, the contents of this file were printed directly to standard output on the terminal (aka the *screen*). Lots of useful information can be found here about the progress of an optimization or analysis run.

SNOPT.out*PP* Summary output from the SNOPT optimizer, where PP is a 2 digit interger representing the processor number. This output file is only created when the SNOPT optimizer is used (OPT_METH = 8). For a description of its contents, see the User's Guide for SNOPT Version 7.

.solv Flow solution output.

.svhis Search direction history. The columns are: design iteration, search direction, step size along the search direction, search direction vector. This output file is only generated for BFGS optimizations (OPT_METH = 3).

.t Binary file for viscous and turbulence variables. The format is double precision unformatted single-block 2d PLOT3D. This file is only written upon execution completion if TURBULNT, VISCOUS and WRITETURB are all TRUE. Also, this file is written to disk every NQ iterations provided the above holds.

.tcon Optimization thickness constraints history. The airfoil thickness values are output at each iteration of an optimization run.

.this Turbulence model convergence history. The format is: iterations, turbulence model residual, location of maximum residual, and maximum residual.

Chapter 5

Airfoil Shape Parameterization

The program MODAIR is used to parameterize the airfoil shape using B-spline curves and define design variables for optimization problems. See Nemec [10] for details of the algorithm.

An example input file is shown below:

```
&INPUTS
       JMAX=257, KMAX=57, JTAIL1=29, JTAIL2=229,
       NC = 21, IORDER = 4, MAXITER = 1000, TOL = 1.d-6,
       TECPLOT = TRUE
&END
```

where:

IORDER Order of the B-spline curve (degree = order - 1).

MAXITER Maximum number of iterations for the parameter correction algorithm used to improve the B-spline least-squares approximation of the original airfoil. This value can range from 100 to 5,000 and the value of 1,000 works well for most cases.

TOL Convergence criterion. This value represents the norm of the error vector that measures the deviation of the B-spline curve from the original airfoil. The convergence criterion is rarely satisfied, typically the program terminates at MAXITER.

TECPLOT Logical flag. When TRUE, an output file *airfit.tec* is generated that can be viewed with TECPLOT. It is strongly recommended that the user views this file to ensure reasonable control point locations and design variables.

NC Number of B-spline control points.

JMAX, KMAX Grid parameters.

JTAIL1, JTAIL2 Grid parameters.

To run the program, the GRID file must be linked (or copied) to a file name with the root 'grid'. MODAIR reads the input file, which must have the name *grid.inp*, and the grid file *grid.g*.

User input is required as the program executes. MODAIR generates a B-spline knot sequence that can be modified by the user. This is the first input required from the user. A good strategy is not to modify the knot sequence on the first pass, view the resulting *airfit.tec* file, and then make adjustments. Next, the user is asked to define design variables from a list of control points. Once the design variables are selected, the user is asked if the airfoil shape should be modified. Typically, the answer is *no*. This feature is used as an optimization validation test for inverse design problems.

Chapter 6

Plotting Grid And Solution Files

6.1 Overview

There are a number of ways to view the unformatted grid and solution files. The unformatted files are in double precision PLOT3D format. It should be noted that the PLOT3D format requires single precision. The grid and solution files use double precision in order to preserve accuracy for very fine meshes where grid points are so close together that single precision is not precise enough to resolve the points as distinct. It was deemed prudent to take this path rather than to use a completely new data format. Hence, it is first necessary to convert the files to single precision and then use a suitable graphics package. Both Plot3d and TECPLOT graphics packages can be used to view the single precision files.

6.2 Plotting Grids

To convert a grid file from double to single precision unformatted 2d single-block PLOT3D format, use the post processor *grd_d2s*. The usage is:

grd_d2s <input> <output>

where <*input*> is the input name in double precision 2d single-block PLOT3D format and <*output*> is the equivalent in single precision. Note that the *input* file name must have a .g extension and do not specify this extension in the command.

To view the grid file in TECPLOT, first use TECPLOT's PREPLOT command:

preplot *output* -plot3d -2d -f -gridonly

and then load *output.plt* into TECPLOT.

6.3 Plotting Grid And Solution Files

A conversion routine is provided that converts grid *and* q files *together* from the double precision unformatted single-block 2d PLOT3D format into the standard single precision version of the same. From here the converted grid and q files can be used in any PLOT3D compatible utility. Called *d2s*, the routine is run using to following protocol:

d2s <input> <output>

It is assumed that the input file names are of the form:

GRID FILE: <*input*>.g
Q FILE: <*input*>.q

while the output file names are of the form:

GRID FILE: <*output*>.g

Q FILE: <*output*>.q

where <*input*> is the input name of the grid and q files (i.e., the common prefix) and <*output*> is the output file name for the grid and q files. Note that the ".g" and ".q" extensions are assumed and are not included in the file name specification.

To view the files in TECPLOT, it is necessary to first use TECPLOT's PREPLOT utility:

preplot output -plot3d -2d -f

and then load the resulting *output.plt* file into TECPLOT.

Chapter 7

File Data Formats

7.1 Grid File

The grid file is in double precision unformatted single-block 2d *PLOT3D* format:

```
      READ(IUNIT) JMAX,KMAX
      READ(IUNIT) ((X(J,K), J=1,JMAX), K=1,KMAX),
     +    ((Y(J,K), J=1,JMAX), K=1,KMAX)
```

7.2 Q-File

The q-file data format is double precision unformatted 2d single-block *PLOT3D* format:

```
      READ(IUNIT) JMAX,KMAX
```

```
      READ(IUNIT) FSMACH,ALPHA,RE,TIME
      READ(IUNIT) (((Q(J,K,NX),
     +      J=1,JMAX),
     +      K=1,KMAX),
     +      NX=1,4)
```

References

[1] G. A. Ashford, *An Unstructured Grid Generation and Adaptive Solution Technique for High Reynolds Number Compressible Flows*, PhD thesis, University of Michigan, 1996.

[2] L. Billing, *On the development of an improved lift-constrained aerodynamic optimization algorithm*, Master's thesis, University of Toronto, September 2006.

[3] H. Buckley, *Numerical Shape Optimization of Airfoils With Practical Aerodynamic Design Requirements*, Master's thesis, University of Toronto, September 2009.

[4] T. Chisholm, *A Fully Coupled Newton-Krylov Solver with a One-equation Turbulence model*, PhD thesis, University of Toronto, 2005.

[5] S. De Rango, *Higher-Order Spatial Discretization for Turbulent Aerodynamic Flows*, PhD thesis, University of Toronto, 2001.

[6] S. De Rango and D. Zingg, *SC1 user's manual, version 1.0.* University of Toronto, Institute for Aerospace Studies, 1997.

[7] S. De Rango and D. W. Zingg, *Higher-order spatial discretization for turbulent aerodynamic computations*, AIAA Journal, 39 (2001), pp. 1296–1304.

[8] De Rango, S., and Zingg, D.W., *Improvements to a Dual-Time-Stepping Method for Computing Unsteady Flows*, AIAA Journal, 35 (1997), pp. 1548–1550.

[9] K. Frew, D. W. Zingg, and S. De Rango, *On artificial dissipation models for viscous airfoil computations*, AIAA Journal, 36 (1998), pp. 1732–1734.

[10] M. Nemec, *Optimal Shape Design of Aerodynamic Configurations: A Newton–Krylov Approach*, PhD thesis, University of Toronto, 2003.

[11] M. Nemec and D. Zingg, *CYCLONE OPTIMA-2D user's guide*. University of Toronto, Institute for Aerospace Studies, May 2003.

[12] M. Nemec and D. W. Zingg, *Aerodynamic computations using the convective-upstream split pressure scheme with local preconditioning*, AIAA Journal, 38 (2000), pp. 402–410.

[13] ——, *Newton–Krylov algorithm for aerodynamic design using the Navier–Stokes equations*, AIAA Journal, 40 (2002), pp. 1146–1154.

[14] M. Nemec, D. W. Zingg, and T. H. Pulliam, *Multi-point and multi-objective aerodynamic shape optimization*, AIAA Paper 2002–5548, Atlanta, GA, Sept. 2002.

[15] Nemec, M., Zingg, D.W., Billings, L. and Oldfield, C., *CYCLONE OPTIMA-2D user's guide*. University of Toronto, Institute for Aerospace Studies, Jan. 2007.

[16] C. Oldfield, *An adjoint method augmented with grid sensitivities for aerodynamic optimization*, Master's thesis, University of Toronto, September 2006.

[17] A. Pueyo and D. W. Zingg, *Efficient Newton–Krylov solver for aerodynamic computations*, AIAA Journal, 36 (1998), pp. 1991–1997.

[18] T. H. Pulliam, *Efficient solution methods for the Navier-Stokes equations*, tech. rep., Lecture Notes for the von Kármán Inst. for Fluid Dynamics Lecture Series: Numerical Techniques for Viscous Flow Computation in Turbomachinery Bladings, Brussels, Belgium, Jan. 1986.

[19] M. Rumpfkeil, *Airfoil Optimization for Unsteady Flows with Application to High-Lift Noise Reduction*, PhD thesis, University of Toronto, 2008.

[20] P. R. Spalart and S. R. Allmaras, *A one-equation turbulence model for aerodynamic flows*, AIAA Paper 92–0439, Jan. 1992.

[21] M. Tabesh, *Higher-Order Spatial and Temporal Discretization Using a Newton-Krylov Algorithm for Unsteady Aerodynamic Computations*, PhD thesis, University of Toronto, 2009.

[22] D. Unrau and D. W. Zingg, *Viscous airfoil computations using local preconditioning*, AIAA Paper 97-2027, June 1997.

[23] D. W. Zingg, M. Nemec, and T. Chisholm, *A Newton–Krylov algorithm for aerodynamic analysis and design*, keynote paper, International Conference on Computational Fluid Dynamics (ICCFD) 2, Sydney, AU, July 2002.