Wind-US Utilities Guide*

The NPARC Alliance

NASA Glenn Research Center
Cleveland, Ohio

USAF Arnold Engineering Development Center
Tullahoma, Tennessee

*This is an unnumbered version of this document, created August 5, 2016. Please send corrections, additions, ideas, etc., Dennis Yoder at *Dennis.A.Yoder@nasa.gov*.

Contents

1 Introduction

In addition to MADCAP, GMAN, and CFPOST, several pre- and post-processing utilities are supplied as part of the Wind-US tools distribution. None of these are required in order to run Wind-US, but may be useful when preparing input files and/or examining the results for a Wind-US run.

Another very useful utility is CGNSview, developed as part of the CGNS (CFD General Notation System) project. *CGNSview* is a viewer/editor for CGNS files (which are similar to Wind-US common files), and allows access to any node in the file using a Windows-like GUI with a collapsible node tree. Nodes and data may be added, deleted, and modified. *CGNSview* is *not* part of the Wind-US tools distribution, but instead is part of the *CGNStools* package, available from `http://cgns.github.io/`.

The utilities in the Wind-US tools distribution are listed below.

	Supported Utilities
adf_revert	Convert a common flow (*.cfl*) file created by Wind-US 3.0 to one that can be read by Wind-US 2.0.
cfaverage	Average multiple common flow (*.cfl*) files.
cfbeta	Add zones to a common grid (*.cgd*) or common flow (*.cfl*) file that are symmetric to existing zones.
cfcnvt	Convert between common files and various other file types.
cfcombine	Combine multiple structured zones in a common grid (*.cgd*) or common flow (*.cfl*) file into a single zone.
cflistnum	List zone sizes and grid units in a common file.
cfpart	Partition a single-zone unstructured grid into multiple zones.
cfreorder	Re-order and/or delete zones in a common grid (*.cgd*) or common flow (*.cfl*) file.
cfreset_iter	Reset the iteration count in a common flow (*.cfl*) file to zero.
cfspart	Re-partition unstructured common flow (*.cfl*) and boundary data (*.tda*) files.
cfsplit	Split a structured zone (or zones) in a common grid (*.cgd*) or common flow (*.cfl*) file into multiple zones.
decompose	Split a Wind-US structured grid system and, optionally, the corresponding flow file, into smaller grid zones to improve the parallel processing efficiency.
fpro	Operate on data in common flow (*.cfl*) files.
gpro	Operate on data in common grid (*.cgd*) files.
jormak	Find boundary points in a common grid (*.cgd*) file, and create a journal file containing subset information for CFPOST or PLOT3D.
mpigetnzone	Get number of zones in a *.cgd* file.
recombine	Convert a "split" structured common flow (*.cfl*) file, originally created by *decompose*, back to the original grid system.
resplt.pl	Create GENPLOT files containing convergence data from a list output (*.lis*) file.

thplt	Create GENPLOT files containing time history data from a list output (*.lis*) file. This utility must be used with *.cth* files created using the improved time history capability introduced in Wind alpha 5.52.
tmptrn	Create a point-by-point wall temperature distribution and/or boundary layer transition data, and write it into the common flow (*.cfl*) file, for use with Wind-US's `TTSPEC` keyword.
USintrpltQ	Interpolate solution from one unstructured grid to another.
windpar	Compute an estimate of the potential for parallel speed-up of a particular Wind-US case as a function of the number of processors, based on the likely number of grid points per processor.

Depricated Utilities

cfappend	Append one common file to another. Use *cfcnvt* instead.
resplt.exe	Old Fortran version of the *resplt* utility used to create GENPLOT files containing convergence data from a list output (*.lis*) file. The *resplt* command now defaults to the Perl script (*resplt.pl*) version of this utility.

Obsolete Utilities

adfedit	View (as text) the structure and contents of ADF files. Use CGNSview instead.
b4wind	Convert between various types of grid and solution files, and/or compute an initial flow field.
cfnav	Explore the contents of common files via textual output. Use CGNSview instead.
cfrevert	Convert a common flow (*.cfl*) file created by Wind-US 2.0 to one that can be read by Wind-US 1.0.
cfsequence	Remove grid points from a structured common grid (*.cgd*) or common flow (*.cfl*) file.
cfsubset	Remove specified grid points from a structured common grid (*.cgd*) or common flow (*.cfl*) file.
cfunsequence	Add grid points to a structured common flow (*.cfl*) file.
cfview	View (as text) the structure and contents of common files. Use CGNSview instead.
chmgr	Create and manipulate chemistry (*.chm*) files.
gridvel	Read the output from a 6DOF program and use that information to set grid velocities in a common grid (*.cgd*) file.
timplt	Create GENPLOT files containing time history data from a list output (*.lis*) file. This utility must be used with *.cth* files created using versions of the flow solver prior to Wind 5.52.
usplit-hybrid	Split a single-zone unstructured common grid (*.cgd*) file into multiple zones. Use *cfpart* instead.

2 Supported Utilities

2.1 adf_revert

In Wind-US 3.0, the format and variable names for some of the data in common files changed. The flow solver will automaticall update grid (*.cgd*) and solution (*.cfl*) files to the new format. However, it is sometimes necessary to revert back to the old format in order to use utilities and other third party software compiled with the old library files. Incompatible software usually reports a common file error while trying to read the new file format.

The *adf_revert* utility may be used to convert common files to the old format. Simply type "*adf_revert*" and you will be prompted for the name of the existing file to convert. Enter the full name, including the extension. The filename may also be provided on the command line. Note that the converted file will overwrite the original, so if you need to retain the original file, copy it to a different name first.

2.2 cfaverage

cfaverage may be used to average the values in two or more existing common flow (*.cfl*) files, and write the averaged values to a new *.cfl* file. The maximum number of files that can be averaged is 256, and the maximum number of zones in a file is 512. All the files must have the same reference conditions.

Input to *cfaverage* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and parameters may be specified.

`CHECK` — Checks the input file for errors without performing any operations.

`AVERAGE FILE` *file1.cfl*
`AVERAGE FILE` *file2.cfl*
...
The input *.cfl* files to be averaged. The maximum number of files that can be averaged is 256.

`OUTPUT` *fileave.cfl* — The name of the output *.cfl* file containing the averaged values (or summmed values, in `ACCUMULATE` mode).

`AMODE {DEFAULT|ACCUMULATE|FINALIZE}`
The averaging mode. The options are:

- `DEFAULT` — Values in the specified input files are averaged, and the results are written to the specified output file. This is the default averaging mode.
- `ACCUMULATE` — Values in the specified input files are summed, and the results are written to the specified output file, along with a running total of the number of input files represented in the summed values. This averaging mode may be used multiple times with the same output file to accumulate additional results. When results from all the desired input files have been accumulated, *cfaverage* is run one last time in `FINALIZE` mode to compute the averaged values.
- `FINALIZE` — Summed values in the specified output file, previously written using `ACCUMULATE` mode, are averaged over the total number of input files. The results are written back to the same file, overwriting the summed values. Note that once `FINALIZE` mode is used, the same output file can't be used for `ACCUMULATE` mode as the results will no longer be valid. Thus, to compute intermediate averages while using `ACCUMULATE` mode, copy the output file to a new intermediate file, and use `FINALIZE` mode with that intermediate file.

`RMS` — Compute a root mean square average instead of a simple average.

`AFILE` *filebase.cfl* — A "base" *.cfl* file for use with RMS averaging. Values in this file will be subtracted from those in each input file during the averaging process, like a standard deviation.

Example

The following input file for *cfaverage*, named *cfave3.inp*, will average the values in the files *case1.cfl*, *case2.cfl*, and *case3.cfl*, and write the averaged values into the new file *caseave.cfl*.

```
/ Files to be averaged
/
AVERAGE FILE case1.cfl
AVERAGE FILE case2.cfl
AVERAGE FILE case3.cfl
/
/ File containing averaged values
/
OUTPUT caseave.cfl
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfaverage

                              ***** cfaverage *****

          Select the desired version from the following list.

0) END
1) cfaverage

Single program automatically selected.

Omit the .inp extension!
Enter cfaverage INPUT FILE ............. (<CR>=cfaverage.inp) : cfave3
 cfaverage - Version 1.8 (last changed 2009/05/01 23:18:38)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  INF: AVERAGING ZONE   1 FROM FILE case1.cfl
  INF: AVERAGING ZONE   1 FROM FILE case2.cfl
  INF: AVERAGING ZONE   1 FROM FILE case3.cfl
  INF: AVERAGING ZONE   2 FROM FILE case1.cfl
  INF: AVERAGING ZONE   2 FROM FILE case2.cfl
  INF: AVERAGING ZONE   2 FROM FILE case3.cfl
  INF: AVERAGING ZONE   3 FROM FILE case1.cfl
  INF: AVERAGING ZONE   3 FROM FILE case2.cfl
  INF: AVERAGING ZONE   3 FROM FILE case3.cfl
```

2.3 cfbeta

cfbeta may be used to add zones to a common grid (*.cgd*) or common flow (*.cfl*) file that are symmetric to existing zones. For example, a grid or flowfield for a complete aircraft configuration may be created from an existing half-span grid or flowfield, for additional calculations at non-zero yaw angles.

Input to *cfbeta* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and parameters may be specified.

`CHECK` Checks the input file for errors without performing any operations.

`FILE` *file_in* The input *.cgd* or *.cfl* file.

`OUTPUT` *file_out* The output common file, with the added symmetric zones. If the file is a *.cgd* file, the `reflection` (i.e., symmetry) boundaries are converted to `coupled` boundaries, and coupling data is generated and included in the output file. Other boundary types in the newly-added zones will be the same type as the corresponding boundaries in the original zones.

`BETA PLANE [X|Y|Z]` *value*
The location in physical space of the symmetry plane, which must be a constant x, y, or z plane.

Warning — all the `reflection` (i.e., symmetry) boundaries in the original zones will be coupled to the corresponding boundaries in the newly-added zones. No check is made to verify that they lie on the specified symmetry plane. It may thus be necessary to make corrections using GMAN.

`SWITCH DIRECTION [I|J|K] IN ZONES` *nzone1 nzone2*
This keyword specifies the index (for structured grids) or direction (for unstructured grids) that should be reversed in the newly-added zones, so that they have the same "handedness" as the grids in the original zones. For structured grids, `I`, `J`, and `K` correspond to the i, j, and k indices; for unstructured grids, they correspond to the x, y, and z directions. If *nzone1* = 0, the specified index or direction will switch in all the newly added zones, and *nzone2* need not be specified. Otherwise the specified index or direction will switch in zones *nzone1* through *nzone2*. A choice must be specified for all the zones.

Note: Users have reported problems with this utility when the original grid contains bleed regions. It is therefore recommended that GMAN be used to delete the bleed regions in the original grid (i.e., redefine them as some other type of boundary, such as a viscous wall) before using *cfbeta*. Then, after using *cfbeta*, use GMAN to recreate the bleed regions at the desired locations in the new grid.

Example

Suppose we have the three-zone structured grid shown in Figure 1, for the upper half of a simple two-dimensional diverging duct. The $y = 0$ plane is a symmetry plane, and the $j = 1$ boundaries in zones 1 and 3 are `reflection` boundaries.

A six-zone grid for the full duct may be created using the following input file for *cfbeta*, named *cfbeta3.inp*. Note that even though this is a two-dimensional configuration, it's still necessary to switch one of the index directions.

```
/ Input grid file
/
```

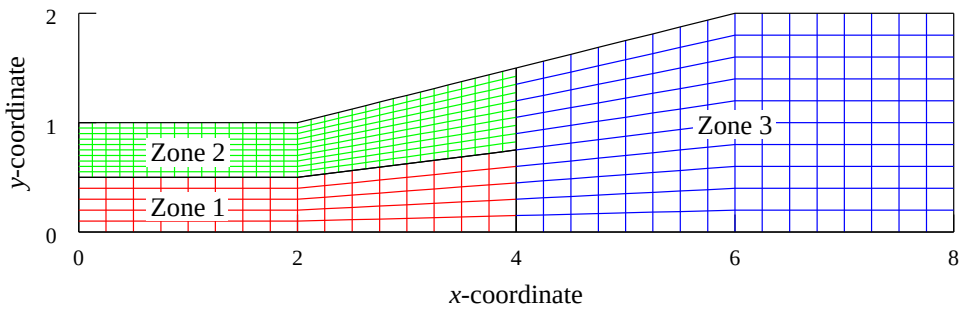

Figure 1: Three-zone mesh, input to *cfbeta*

```
FILE testa.cgd
/
/ Output grid file
/
OUTPUT testb.cgd
/
/ Symmetry plane at y = 0.
/
BETA PLANE Y 0.0
/
/ Switch k index direction in new zones
/
SWITCH DIRECTION K IN ZONES 0
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfbeta

                              ***** cfbeta *****

          Select the desired version from the following list.

0) END
1) cfbeta

Single program automatically selected.

Omit the .inp extension!
Enter cfbeta INPUT FILE ............. (<CR>=cfbeta.inp) : cfbeta3
 cfbeta - Version 1.8 (last changed 2007/02/13 22:16:01)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  CREATING BETA GRID FROM ZONE   1
  CREATING BETA GRID FROM ZONE   2
```

```
CREATING BETA GRID FROM ZONE   3
CREATING BETA BOUNDARIES FOR ZONE   1
CREATING BETA BOUNDARIES FOR ZONE   2
CREATING BETA BOUNDARIES FOR ZONE   3
```

The resulting six-zone grid is shown in Figure 2.

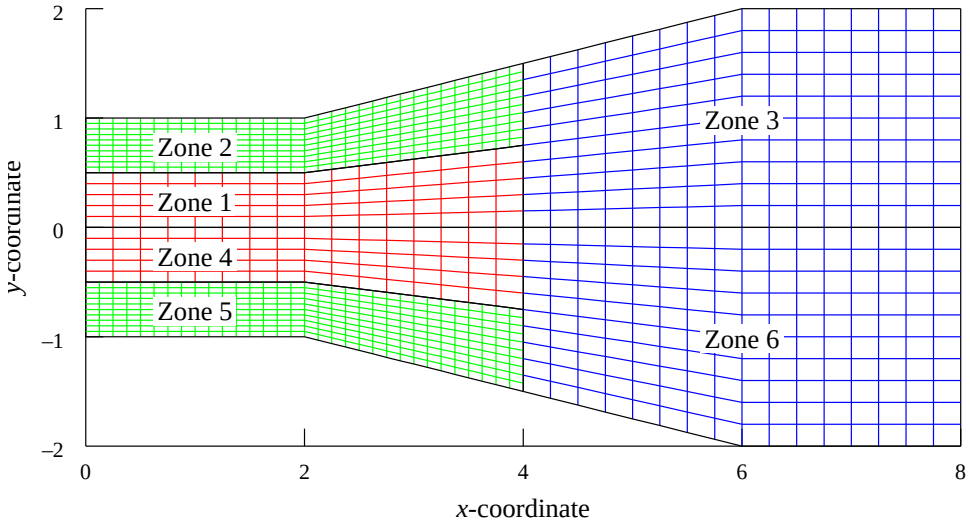

Figure 2: Six-zone mesh, output from *cfbeta*

2.4 cfcnvt

cfcnvt may be used to convert a variety of file formats, including PLOT3D files, to/from common file format, and to do some limited manipulation of common files.

The following menu choices are available when version 1.54 of *cfcnvt* is invoked:

```
 0: Exit program
 2: Import   a Common File
 3: Compress a Common File
 4: Break Common File into multiple transfer files
 5: Combine multiple transfer files into Common File
 6: Append one Common File to another
 7: Convert Common File binary to a text file
 8: Convert Common File text to a binary file
11: Convert PLOT3D/Pegsus file to Common File
12: Convert GASP     file to Common File
13: Convert OVERFLOW file to Common File
14: Convert Common File to OVERFLOW file
15: Convert CFPOST GPU file to Common File GPC
16: Convert ascii rake to Common File rake CGF
17: Convert Pegsus 4.0 files to Common File
18: Convert Common File CFL to Plot3d Q
```

For Wind-US users, *cfcnvt* is probably used most often when a grid has been created using software that can create a PLOT3D xyz file format, but not a common grid (*.cgd*) file. In this case, *cfcnvt* is used to convert the PLOT3D xyz file to a *.cgd* file, for input into GMAN. It could also be used to convert a PLOT3D q file to a common flow (*.cfl*) file, for use as initial conditions when running Wind-US.

When converting to a common file from another format (menu choices 11, 12, 13, and 17), for most data the scaling information written into the common file assumes that the FSS system of units (i.e., foot, slug, second, degrees Rankine, pound force, foot-pound force) is being used. There are two exceptions—the coordinates x, y, and z are assumed to be in inches, and the pressure p is assumed to be in $\mathrm{lb_f/in^2}$. If the data in the original file is actually in some other units, the `UNITS` command in GMAN (for *.cgd* files) and option 11 in the *fpro* utility (for *.cfl* files) may be used to change the scaling information for the correct units.

For more information about supported systems of units, see the description of `CFUNIT` in the *Common File User's Guide*, and the `units` command in the *CFPOST User's Guide*. For details about the use of reference and scaling data in common files, see the *Common File User's Guide*.

A detailed example illustrating the use of *cfcnvt* to convert from a PLOT3D xyz file to a *.cgd* file is included in the "Tutorial" section of the *Wind-US User's Guide*.

2.5 cfcombine

cfcombine may be used to combine structured zones in a common grid (*.cgd*) or common flow (*.cfl*) file into a single zone. The zones being combined must have abutting faces with contiguous grid points, or (for *.cfl* files) be overlapping in a manner consistent with the `NOLAP` option in the *cfsplit* utility.

Input to *cfcombine* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "`/`" are comments. The following keywords and parameters may be specified.

`CHECK` Checks the input file for errors without performing any operations.

`FILE` *file_in* The input *.cgd* or *.cfl* file, containing zones to be combined.

`OUTPUT` *file_out* The output common file, with the combined zones.

`NOLAP` *nplanes* The number of planes by which the split boundaries are overlapped. The overlapping must be in a manner consistent with the `NOLAP` specification in the *cfsplit* utility. However, in *cfcombine* `NOLAP` applies to all zones. Common flow files split with *cfsplit* can only be recombined if the same `NOLAP` value was used for all zones in *cfsplit*. This keyword may only be used with common flow (*.cfl*) files.

`HALFCELL` Indicates that the common boundary between the two zones being combined lies halfway between the adjacent grid surfaces, consistent with the `HALFCELL` specification in the *cfsplit* utility. When the two zones are combined, the common grid surface between the original two zones is removed. This option overrides any `NOLAP` specification. This keyword may only be used with common flow (*.cfl*) files.

`COMBINE ZONE` *nzone1 face1* `TO ZONE` *nzone2 face2*
The `COMBINE` keyword tells *cfcombine* what zone and face to combine to what other zone and face. The parameters *nzone1* and *nzone2* are the numbers of the zones to be combined. The parameters *face1* and *face2* specify the adjoining faces of the zones to be combined. The adjoining faces must be specified in the form `IMAX` (or `JMAX` or `KMAX`) to `I1` (or `J1` or `K1`), and in that order, with `I` faces connecting to `I` faces, etc. When multiple zones are being combined into a single zone, only a minimal connection set is required.

Example

Suppose we have the three-zone grid shown in Figure 3.

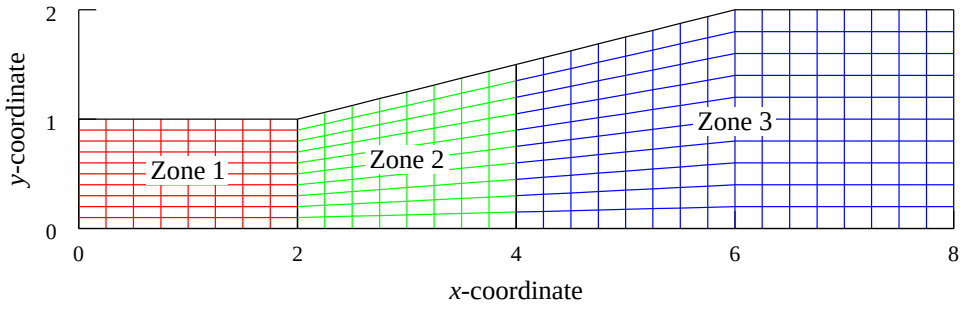

Figure 3: Input three-zone grid for *cfcombine*

A two-zone grid may be created by combining zones 1 and 2 using the following input file for *cfcombine*, named *cfcombine12.inp*.

```
/ Input grid file
/
FILE testa.cgd
/
/ Output grid file
/
OUTPUT testb.cgd
/
/ Combine zones 1 and 2
/
COMBINE ZONE 1 imax TO ZONE 2 i1
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfcombine

          Select the desired version from the following list.

0) END
1) cfcombine

Single program automatically selected.

Omit the .inp extension!
Enter cfcombine INPUT FILE ............. (<CR>=cfcombine.inp) : cfcombine12
 cfcombine - Version 1.11 (last changed 2007/02/13 23:23:07)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
 COMBINING ZONES FOR NEW ZONE   1 SIZE= ( 17, 11,  1)
 COMBINING ZONES FOR NEW ZONE   2 SIZE= ( 17, 11,  1)
  COMBINING BOUNDARIES OF ZONE   1
  COMBINING BOUNDARIES OF ZONE   2
```

The resulting grid is shown in Figure 4.

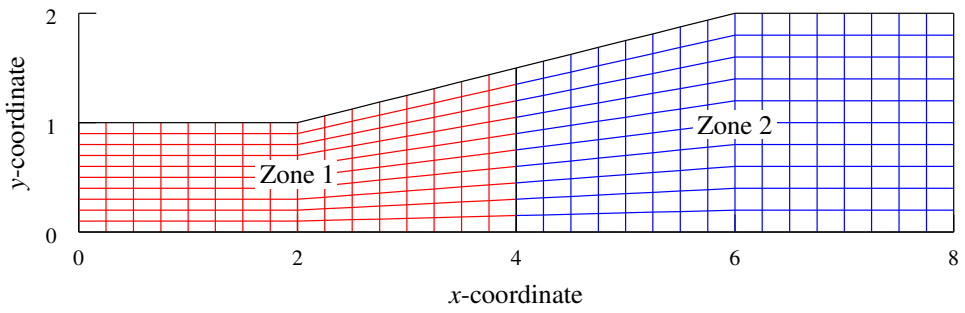

Figure 4: Output two-zone grid from *cfcombine*

To create a single-zone grid from the original three-zone grid, the following two `COMBINE` commands would be used, instead of the single `COMBINE` command shown above.

```
COMBINE ZONE 1 imax TO ZONE 2 i1
COMBINE ZONE 2 imax TO ZONE 3 i1
```

2.6 cflistnum

cflistnum may be used to list the zone sizes in a common grid (*.cgd*) or flow (*.cfl*) file. For structured grids, the list includes the number of grid points in each direction for each zone, the total number of grid points in each zone, and the total number of grid points in the entire grid. For unstructured grids, the list includes the number of nodes, faces, and cells in each zone, and in the entire grid. For *.cgd* files, the grid units are also listed.

The syntax is simply

`cflistnum` *filename*

where *filename* is the name of the file, including the *.cgd* or *.cfl* extension The user will be prompted for the file name if it isn't specified.

As an example, for a three-zone structured grid *.cgd* file named *my_file_struct.cgd*, with grid units in inches and zone sizes $101 \times 76 \times 51$, $101 \times 76 \times 67$, and $51 \times 26 \times 21$:

```
% cflistnum my_file_struct.cgd

Size report for my_file_struct.cgd

Grid units are: in

ZONE   1:                 is (     101,      76,      51), NUMPOINTS=   391476
ZONE   2:                 is (     101,      76,      67), NUMPOINTS=   514292
ZONE   3:                 is (      51,      26,      21), NUMPOINTS=    27846

Totals:  Npnts=   933614  Nface=        0  Ncell=        0
```

For a three-zone unstructured grid *.cgd* file named *my_file_unstruct.cgd*, with grid units in feet and 57,552 nodes, 385,172 faces, and 171,144 cells in zone 1, 61,017 nodes, 385,341 faces, and 168,333 cells in zone 2, and 61,941 nodes, 385,078 faces, and 167,398 cells in zone 3:

```
% cflistnum my_file_unstruct.cgd

Size report for my_file_unstruct.cgd

Grid units are: ft

ZONE   1:                 is (   57552,  385172,  171144)
ZONE   2:                 is (   61017,  385341,  168333)
ZONE   3:                 is (   61941,  385078,  167398)

Totals:  Npnts=   180510  Nface=  1155581  Ncell=   506875
```

2.7 cfpart

cfpart is used to partition a single-zone unstructured grid into multiple zones for parallel processing. Boundary conditions and surfaces defined in the original file are preserved in the partitioned file. Additional "crinkly" surfaces will appear at the boundaries between the newly-created zones, with negative surface ID numbers. The appropriate coupled boundary conditions are automatically set for these zonal boundaries.

Mapping data is written into the partitioned *.cgd* file, that maps the split grid to the original single-zone grid. This data is used by the *cfspart* utility when re-partitioning multi-zone common flow (*.cfl*) and boundary data (*.tda*) files into a different number of zones.

Input to *cfpart* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and values may be specified.

`CHECK` Checks the input file for errors without performing any operations.

`FILE` *file_in.cgd* The input common grid (*.cgd*) file (80 characters max), containing the unstructured grid to be partitioned.

`OUTPUT` *file_out.cgd* The output *.cgd* file (80 characters max), containing the partitioned grid.

`CREATELINES` Turns on the global line calculation, required for the line Gauss-Seidel implicit solver (i.e., the `LINE` option in the Wind-US `IMPLICIT` keyword). This is especially important for viscous calculations.

`MINLINE` *nmin* The minimum length of a line. The default value is 20.

`MAXLINE` *nmax* The maximum length of a line. The default value is 1000. This value is not currently used.

`ANGLELIMIT` *angle* The maximum turning angle allowed when creating lines, in degrees. The default value is 85.

`ASPECTRATIOMIN` *ar* The minimum aspect ratio allowed for cells to be eligible for inclusion in lines. The default value is 1.6.

`MINFACESPERCELLINLINE` *nfaces*
This keyword controls which cell types are included when creating lines. Set *nfaces* = 4 to include all cell types, 5 to only include non-tetrahedral cells, and 6 to only include hexahedral cells. The default value is 4.

`LINEMODE` *mode* This specifies the approach used to determine lines. Setting *mode* = 0 corresponds to the basic way of evaluating the transformation matrix to a canonical element; set *mode* = 1 to use projected face areas to determine an aspect ratio. The default value is 0.

`WRITELINES` If this keyword is specified, the lines will be written to a file named *lines.fvp*. Fieldview can read and display these lines as particle paths.

`LOWMEM` If this keyword is specified, data required by *cfpart* will be written to disk instead of being kept in memory. This will increase the CPU time required, and should only be used on systems with insufficient memory.

`GROUPFILENAME` *filename* `[MODE {APPEND|REPLACE}]`
Read surface group definitions from a text file and include these group definitions in the output grid (*.cgd*) file. Mode `APPEND` will append to the list of existing groups read from the input grid (*.cgd*) file and give an error if the specified groupname already exists. Mode `REPLACE` will

ignore any existing groups from the input grid file. The format of the group definition file is as follows:

```
BEGIN [GROUP] groupname
   { [SURFACE U surfID [ZONE zone]] | \
     [GROUP groupname] }
   ...(repeat as necessary for additional surfaces)
END [GROUP]
...(repeat as necessary for additional groups)
```

At present, surface groups can only be defined from unstructured surfaces.

Example

The following example shows how *cfpart* may be used to partition a single-zone grid into five zones, suitable for parallel processing on five CPUs. The keyword input file, named *M2129_cfpart.inp* is:

```
/ Input single-zone grid
/
FILE M2129_1zone.cgd
/
/ Output grid
/
OUTPUT M2129_5zone.cgd
/
/ Split for 5 CPUs
/
NCPUS 5
/
/ Create lines for line Gauss-Seidel solver
/
CREATELINES
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfpart

                              ***** cfpart *****

          Select the desired version from the following list.

0) END
1) cfpart

Single program automatically selected.

Omit the .inp extension!
Enter cfpart INPUT FILE ............. (<CR>=cfpart.inp) : M2129_cfpart
 CFPART - Version 1.2 (last changed 2008/09/05 19:58:42)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
```

```
* et seq.) or Executive Order 12470. Violation of these export-control     *
* laws is subject to severe criminal penalties. Dissemination of this      *
* software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
*****************************************************************************
beginning of us_main
Step Read grid
Time=  0.2639600      0.2639600
Step Face Connectivity
Time=  0.8398730       1.103833
Step Partitoning
 *** Line statistics report ***
 num lines           =        16695
 zero length lines   =            0
 minimum line length =           20
 maximum line length =          149
 average line length =    26.3946091644205
 ****************************
Time=   1.734736       2.838569
Step Global-Local cell function
Time=  3.9999485E-03   2.842569
Step Surface and face pointers           1
Time=  2.6996136E-02   2.869565
Step Volume data           1
Time=  2.3995876E-02   2.893561
Step CGD write           1
Time=  3.6993980E-02   2.930555
Step Surface and face pointers           2
Time=  2.0997047E-02   2.951552
Step Volume data           2
Time=  2.5995970E-02   2.977548
Step CGD write           2
Time=  3.8993120E-02   3.016541
Step Surface and face pointers           3
Time=  2.1996975E-02   3.038538
Step Volume data           3
Time=  2.7996063E-02   3.066534
Step CGD write           3
Time=  3.8993835E-02   3.105528
Step Surface and face pointers           4
Time=  2.1996975E-02   3.127525
Step Volume data           4
Time=  2.4996042E-02   3.152521
Step CGD write           4
Time=  3.8994074E-02   3.191515
Step Surface and face pointers           5
Time=  1.8996954E-02   3.210512
Step Volume data           5
Time=  2.4996996E-02   3.235509
Step CGD write           5
Time=  3.6995173E-02   3.273503
Total time   3.278502
```

```
end of us_main
in Linelet- Remapping cells
Total time   3.326493
end of cfpart_main
```

2.8 cfreorder

cfreorder may be used to reorder and/or delete zones in a common grid (*.cgd*) or common flow (*.cfl*) file.

Input to *cfreorder* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and parameters may be specified.

`CHECK`	Checks the input file for errors without performing any operations.
`FILE` *file*	The *.cgd* or *.cfl* file containing zones to be reordered and/or deleted. *The output is written to the same file name, overwriting the original file.*
`REORDER ZONE` *nzone1* `TO` *nzone2*	Zone number *nzone1* in the original grid will be renumbered as *nzone2*. You may also specify a range of zones whose numbers are to be shifted. (See the example below). All zones not specified will keep the same number.
`DELETE ZONE` *nzone*	Zone number *nzone* in the original grid will be deleted. *nzone* may also specify a range of zones to be deleted. A zone being deleted may not appear as *nzone1* in a `REORDER` keyword line. Zones will be automatically reordered to account for the deleted zones. Disk space used by the deleted zones is not automatically recovered. To recover this space, use *cfcnvt* and select the "Compress a Common File" menu option.

Example

The following input file for *cfreorder*, named *reorder.inp*, will switch numbers for zones 1 and 3, renumber the original zone 30 as zone 4, renumber the original zones 4–29 as zones 5–30, and delete zones 31–40.

```
/ Input/output grid file
/
FILE testa.cgd
/
/ Renumber zones
/
REORDER ZONE 1    TO ZONE 3
REORDER ZONE 3    TO ZONE 1
REORDER ZONE 30   TO ZONE 4
REORDER ZONE 4-29 TO ZONE 5-30
DELETE ZONE 31-40
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfreorder

                          ***** cfreorder *****

          Select the desired version from the following list.

0) END
1) cfreorder
```

```
Single program automatically selected.

Omit the .inp extension!
Enter cfreorder INPUT FILE ............. (<CR>=cfreorder.inp) : reorder
 cfreorder - Version 1.8 (last changed 2007/02/13 23:41:36)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  RENAMING ZONE   1 TO ZONE   3
  RENAMING ZONE   2 TO ZONE   2
  RENAMING ZONE   3 TO ZONE   1
  RENAMING ZONE   4 TO ZONE   5
  RENAMING ZONE   5 TO ZONE   6
  ...
  RENAMING ZONE  30 TO ZONE   4
  DELETING ZONE   31
  DELETING ZONE   32
  ...
  RENAMING BOUNDARIES OF ZONE   1
  RENAMING BOUNDARIES OF ZONE   2
  RENAMING BOUNDARIES OF ZONE   3
  ...
  To recover space of deleted zones use CFCNVT.
```

2.9 cfreset_iter

cfreset_iter is used to reset the iteration count in a common flow (*.cfl*) file to zero. This may be useful when you are using an existing solution to initialize a new simulation, and would like to start the iteration count at zero. Simply type "`cfreset_iter`" and you'll be prompted for the *.cfl* file name, which should include the *.cfl* extension.

More exactly, *cfreset_iter* sets the following data in the *.cfl* file to zero:

- The number of the last zone completed, the number of cycles completed, and the number of Newton time levels completed (stored in the root node header)
- The maximum residual for all zones (stored in the root node)
- For each zone, the maximum residual, the number of iterations completed, and the integrated time (stored in each zone node)

cfreset_iter does *not* reset the convergence data associated with global Newton iteration.

Note that the original *.cfl* file is actually modified by *cfreset_iter*, not copied. If the file containing the original iteration count and residual data will be needed in the future, you'll need to manually copy it to a different file name before using *cfreset_iter*.

Note also that at the start of a Wind-US calculation with structured grids, the boundary condition at viscous walls is gradually changed from slip to no-slip over a specified number of iterations (the default is 50). If you're restarting a structured grid case with viscous walls after using *cfreset_iter*, you probably don't want this behaviour. The `WALL SLIP` keyword should be used in the input data (*.dat*) file to specify that a no-slip boundary condition is to be applied from the first iteration, as follows:

```
WALL SLIP ITERATIONS 1
```

A terminal session illustrating the use of *cfreset_iter* is shown below, for a common flow file named *case3.cfl*. Lines in a slanted font are typed by the user.

```
% cfreset_iter

                          ***** cfreset_iter *****

          Select the desired version from the following list.

0) END
1) cfreset_iter optimized version

Single program automatically selected.

 cfreset_iter - Version 1.8 (last changed 2007/02/13 23:54:58)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  Enter name of the common flow file.
case3.cfl
```

```
Working on ZONE   1
Working on ZONE   2
Working on ZONE   3
All fixed!!!!!!!!!
```

2.10 cfspart

cfspart is used to re-partition multi-zone common flow (*.cfl*) and boundary data (*.tda*) files for an unstructured grid into a different number of zones. The primary purpose is to allow restarting an unstructured grid case using a different number of processors, while still having one zone per processor.

cfspart requires as input two multi-zone common grid (*.cgd*) files, both created from the same single-zone unstructured grid using the *cfpart* utility. One of the *.cgd* files corresponds to the "old" grid used for the existing *.cfl* and *.tda* files, and the other corresponds to the "new" grid, with a different number of zones.

The user is prompted for the names of the old and new files.

2.11 cfsplit

cfsplit may be used to split structured zones in a common grid (*.cgd*) or common flow (*.cfl*) file into multiple zones. This may be useful in splitting a large zone into smaller zones for parallel computation on multiple processors.

Input to *cfsplit* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and parameters may be specified.

`CHECK` Checks the input file for errors without performing any operations.

`FILE` *file_in* The input *.cgd* or *.cfl* file, containing zones to be split.

`OUTPUT` *file_out* The output common file, with the split zones. If the file is a *.cgd* file, coupling data is also generated and included in the output file. *Note, though, that coupling data is not generated if originally self-coupled boundaries are uncoupled by splitting a self-closing zone in the self-closing direction.*

`NOLAP` *nplanes* The number of planes to overlap the zones when `OVERLAPPING` split mode is being used. (See the `SPLIT` keyword below.) Note that this value can be changed between `SPLIT` commands to change the overlap size for particular zones.

`FRGSIZ` *nplanes* The number of planes to create fringe points on when `OVERLAPPING` split mode is being used. (See the `SPLIT` keyword below.) Note that this value can be changed between `SPLIT` commands. The value of `FRGSIZ` must be between 1 and `NOLAP`.

`HALFCELL` Indicates that the split will occur halfway between the specified "split" index and the next higher index. This option overrides and ignores any `OVERLAPPING` or `NOLAP` specifications. Using this option allows more accurate coupling, particularly when using the `COUPLING MODE ROE HIGH` keyword when running Wind-US. For *.cfl* files, the split zones may be re-combined using the *cfcombine* utility with its `HALFCELL` keyword.

`SPLIT [OVERLAPPING] ZONE` *nzone* `AT [I` *irange*`] [J` *jrange*`] [K` *krange*`]`
A series of `SPLIT` keywords tell *cfsplit* what zone to split, and where. Any zones not specified will be copied unsplit into the output file.

The *irange*, *jrange*, and *krange* parameters are each of the form *n1 n2*, where *n1* and *n2* specify the indices in the relevant direction to be included in the split-off zone. A value of 0 may be used for *n2* to specify the maximum value in that direction. Any *irange*, *jrange*, or *krange* parameter not specified defaults to the entire range.

A zone must be split in contiguous sections covering the whole zone. Thus, at least two `SPLIT` keywords must be used.

If the `OVERLAPPING` parameter is included, the split zones will be overlapped by the number of planes specified by `NOLAP`. The number of overlapping planes is added to the *n2* values.

Example 1

Suppose we have the simple single-zone two-dimensional grid shown in Figure 5. A three-zone grid may be created using the following input file for *cfsplit*, named *cfsplit1to3.inp*.

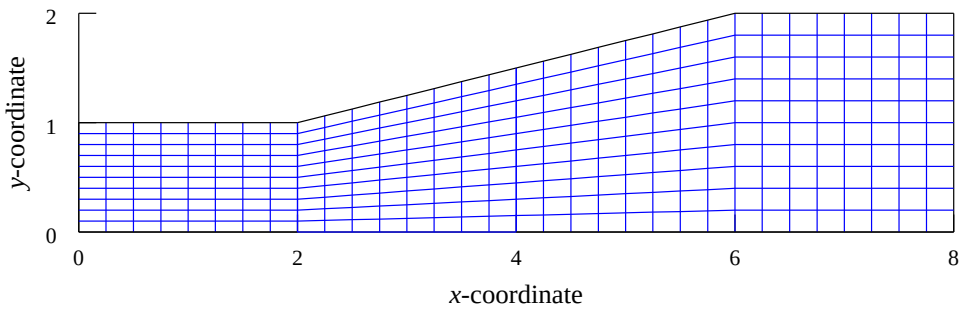

Figure 5: Single-zone mesh, input to *cfsplit*

```
/ Input grid file
/
FILE testa.cgd
/
/ Output grid file
/
OUTPUT testb.cgd
/
/ Split into three zones
/
SPLIT ZONE 1 AT I 1   9
SPLIT ZONE 1 AT I 9  17
SPLIT ZONE 1 AT I 17  0
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfsplit

                          ***** cfsplit *****

          Select the desired version from the following list.

0) END
1) cfsplit

Single program automatically selected.

Omit the .inp extension!
Enter cfsplit INPUT FILE ............. (<CR>=cfsplit.inp) : cfsplit1to3
 cfsplit - Version 1.15 (last changed 2007/03/01 21:04:55)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  SPLITTING ZONE   1
  SPLITTING BOUNDARIES OF ZONE   1
```

The resulting three-zone grid is shown in Figure 6.

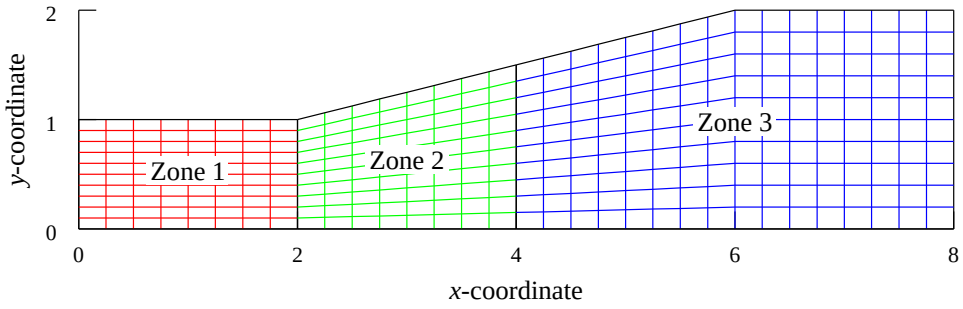

Figure 6: Three-zone mesh, output from *cfsplit*

Example 2

In three dimensions, the following `SPLIT` commands would be used to split a $23 \times 15 \times 19$ zone into a $10 \times 15 \times 19$ zone, a $14 \times 15 \times 5$ zone, and a $14 \times 15 \times 15$ zone.

```
SPLIT ZONE 2 AT I  1 10  K 1 0
SPLIT ZONE 2 AT I 10  0  K 1 5
SPLIT ZONE 2 AT I 10  0  K 5 0
```

Example 3

The following three commands will split zone 3 into two zones; the first containing `I` planes 1–23 from the original zone, and the second containing planes 21–`IMAX`.

```
NOLAP 2
SPLIT OVERLAPPING ZONE 3 AT I  1 21
SPLIT OVERLAPPING ZONE 3 AT I 21  0
```

2.12 decompose

decompose automatically splits a Wind-US structured grid system and, optionally, the corresponding flow file, into smaller grid zones to improve the parallel processing efficiency. By default the *.mpc* file is read to determine the number of processors. Alternatively, the user may specify the number of processors. The default values for decompose are generally sufficient to generate a grid system with a theoretical parallel efficiency of 90% or greater.

decompose is run from the command line, with the following syntax:

`decompose` *casename* [`-help`] [`-nprocs` *number*] [`-olap` [*number*]] [`-halfcell`] [`-frgpts` [*number*]] [`-minpts` *number*] [`-maxpts` *number*] [`-nosplit`] [`-npass` *number*] [`-dat` *datfile*] [`-grid` *cgdfile*] [`-flow` *cflfile*] [`-mpc` *mpcfile*]

The only required input parameter is *casename*, the base name for the Wind-US files for the original grid system. The various command-line options for *decompose* are described below.

`-help`	Display usage information and quit, ignoring any other options.
`-nprocs` *number*	Desired number of processors. If specified, this option overrides the number of processors defined by the *.mpc* file.
`-olap` [*number*]	Number of planes to overlap between zones; must be ≥ 0. The default value, if *number* is not specified, is 1.
`-halfcell`	Split zones halfway between the original grid points. This is the default behavior, if neither `-olap` nor `-halfcell` is specified. If both `-olap` and `-halfcell` are specified, the `-halfcell` splitting is used.
`-frgpts` [*number*]	Number of fringe planes for overlapped zones; must be > 0 and $\leq$ the number of overlap planes. The default value, if *number* is not specified, is 1.
`-minpts` *number*	Desired minimum number of points in any direction. The default value is 10.
`-maxpts` *number*	Desired maximum total number of points in a zone. The default value is 999999999.
`-nosplit`	Generate the splitting information, but do not actually split. Performs one pass through the splitting procedure.
`-npass` *number*	Number of passes through splitting procedure. The default value is 2.
`-dat` *datfile*	The original Wind-US input data (*.dat*) file, entered without the extension (e.g., *wing*, not *wing.dat*). The default is the name specified for the input parameter *casename*.
`-grid` *cgdfile*	The original Wind-US grid (*.cgd*) file, entered without the extension (e.g., *wing*, not *wing.cgd*). The default is the name specified for the input parameter *casename*.
`-flow` *cflfile*	The original Wind-US flow (*.cfl*) file, entered without the extension (e.g., *wing*, not *wing.cfl*). The default is the name specified for the input parameter *casename*.
`-mpc` *mpcfile*	The original Wind-US multi-processing control (*.mpc*) file, entered without the extension (e.g., *wing*, not *wing.mpc*). The default is the name specified for the input parameter *casename*.

Some keywords in the *.dat* file may not be generated correctly for the split grid system. Keywords not yet fully supported include:

- `ACTUATOR | SCREEN`
- `ARBITRARY INFLOW`
- `BL_INIT`
- `BOUNDARY-DAMP | BDAMP`
- `HISTORY`
- `LOADS`
- `TTSPEC`

Please check the modified .dat file carefully for warning messages.

A *.cfl* file with a split grid system originally created by *decompose* may be processed by the *recombine* utility to convert back to the original grid system.

decompose creates the following files:

split.cgd	The output grid file
split.cfl	The output flow file (if an original flow file exists)
split.dat	The modified input data file. *Check this file carefully for warning messages.*
split.mpc	A copy of the original multi-processor control file (if the original exists)
split.map	Detailed mapping of the original zone structure to the split structure
splitcgd.inp	Input to the *cfsplit* utility for the grid file. (*cfsplit* is used internally by *decompose*.) This file may safely be deleted.
splitcfl.inp	Input to the *cfsplit* utility for the flow file. (*cfsplit* is used internally by *decompose*.) This file may safely be deleted.
split.xxx.*	Intermediate files used for multiple splits. These files may safely be deleted.
combinecfl.xxx.*inp*	Input to the *cfcombine* utility, used by *recombine* to convert a "split" *.cfl* file back to the original grid system. (*cfcombine* is used internally by *recombine*.) *These files should not be deleted.*

2.13 fpro

fpro may be used to perform various operations on data in common flow (*.cfl*) files. The following menu is presented upon starting *fpro*:

```
----- Main Menu -----------------------
      0.....Exit This Menu
      1.....Fill With Constant Values
      2.....Scale Current Values
      3.....Rotate/Translate Velocity
      4.....Copy   From Another CFL or Zone
      5.....Insert From Another CFL or Zone
      6.....Interpolate From Another CFL or Zone
      7.....Map From Another CFL or Zone
      8.....Smooth Subset Using Index TFI
      9.....Change Species Concentrations
     10.....Redefine e0 based on ideal gas
     11.....Set Reference and I,F,Cpar Data
     12.....Rescale using New Freesream Conditions
     13.....Set Gas Properties
 Enter your choice:
```

Depending on the item selected, additional menus and prompts for information are presented, and the selected action is performed.

Example

A typical use of *fpro* is to interpolate from one *.cfl* file to another (menu option 6). This can be used, for example, when creating a new *.cfl* file after adding mesh points in a particular region to better resolve some flow feature found in an earlier Wind-US run.

In the following example, *case_coarse.cgd* and *case_coarse.cfl* are three-zone grid and solution files from an initial Wind-US run. Based on the results from that run, a new grid was created with additional grid points in zone 2, and stored in the grid file *case_fine.cgd*.

To run Wind-US with the new grid, but utilizing the already-computed flow field from the coarse-grid solution, we must first create an initial *.cfl* file corresponding to the new grid. To do this, simply run Wind-US using the new grid file *case_fine.cgd*, with the number of cycles set to zero in the input data (*.dat*) file. This will create the new *case_fine.cfl* file, with the flow field set to initial conditions (which are the freestream conditions by default).

Next, since the grids in zones 1 and 3 haven't changed in this example, the flow field in those zones can be copied directly from *case_coarse.cfl* to *case_fine.cfl*. This is done using option 4 from the *fpro* menu, and is fairly straightforward.

Finally, option 6 from the *fpro* menu is used to interpolate the results in zone 2 from the coarse grid onto the fine grid. The terminal session for the interpolation is shown below. Lines in a slanted font are typed by the user. Note that the zone number is entered twice in succession when specifying both the "interpolate from" and "interpolate to" files.

```
% fpro

             FPRO - Version 1.11 (last changed 2007/02/14 01:05:27)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
```

```
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ****************************************************************************

----- Main Menu ------------------------
      0.....Exit This Menu
      1.....Fill With Constant Values
      2.....Scale Current Values
      3.....Rotate/Translate Velocity
      4.....Copy   From Another CFL or Zone
      5.....Insert From Another CFL or Zone
      6.....Interpolate From Another CFL or Zone
      7.....Map From Another CFL or Zone
      8.....Smooth Subset Using Index TFI
      9.....Change Species Concentrations
     10.....Redefine e0 based on ideal gas
     11.....Set Reference and I,F,Cpar Data
     12.....Rescale using New Freesream Conditions
     13.....Set Gas Properties
 Enter your choice: 6

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 1
 Enter filename (include suffix):
case_coarse.cfl

Wind-US test case, 3 zones

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
```

```
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 2
 Enter filename (include suffix):
case_coarse.cgd

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 3
 Enter the zone number. (Max=           3 )
2

 Enter the zone number. (Max=           3 )
2

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 5
 Enter filename (include suffix):
case_fine.cfl

Wind-US test case, 3 zones

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
```

```
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 6
 Enter filename (include suffix):
case_fine.cgd

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 7
 Enter the zone number. (Max=           3 )
2

ZONE   2

 Enter the zone number. (Max=           3 )
2

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 9
 Interpolate all variables (Y/N)?
y
```

```
----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 10
 Current Status:
     Input CFL File: case_coarse.cfl
     Input CGD File: case_coarse.cgd
     Input Zone:    2  with IxJxK dimensions:   77  x  50  x  47
     Input Subset:   1,  1,  1  to   0,  0,  0
     Output CFL File: case_fine.cfl
     Output CGD File: case_fine.cgd
     Output Zone:    2  with IxJxK dimensions:  95  x 110  x  47
     Output Subset:   1,  1,  1  to   0,  0,  0
     Gamma=1.4000  Gas Constant=  2.8708E+02
     Variable rho                              interpolated
     Variable rho*u                            interpolated
     Variable rho*v                            interpolated
     Variable rho*w                            interpolated
     Variable rho*e0                           interpolated
 &lt;Return&gt; to continue:

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 11
 ***Number of points not interpolated =            0

----- Interpolation Menu ---------------
      0.....Exit This Menu
      1.....Select: Flow File To Interpolate From
```

```
      2..... Grid File To Interpolate From
      3..... Zone To Interpolate From
      4..... Subset To Interpolate From
      5.....Select: Flow File To Interpolate To
      6..... Grid File To Interpolate To
      7..... Zone To Interplolate To
      8..... Subset To Interpolate To
      9.....Select Variables To Interpolate
     10.....Show Current Settings
     11.....EXECUTE INTERPOLATION
 Enter your choice: 0

----- Main Menu -----------------------
      0.....Exit This Menu
      1.....Fill With Constant Values
      2.....Scale Current Values
      3.....Rotate/Translate Velocity
      4.....Copy   From Another CFL or Zone
      5.....Insert From Another CFL or Zone
      6.....Interpolate From Another CFL or Zone
      7.....Map From Another CFL or Zone
      8.....Smooth Subset Using Index TFI
      9.....Change Species Concentrations
     10.....Redefine e0 based on ideal gas
     11.....Set Reference and I,F,Cpar Data
     12.....Rescale using New Freesream Conditions
     13.....Set Gas Properties
 Enter your choice: 0
%
```

2.14 gpro

gpro may be used to perform various operations on data in common grid (*.cgd*) files. The following menu is presented upon starting *gpro*:

```
          ************************ Main Menu ***************************
            I/O OPERATIONS                     BLOCK OPERATIONS
              A - Read Input File                G - Combine Blocks
              B - Read Journal File              H - Reorder Blocks
              C - Write Output File              I - Delete Blocks
                                                 J - Extract part of Block
            BLOCK TRANSFORMATIONS                K - Exchange Block Indices
              D - Translate,Rotate,Scale         L - Block Statistics
              E - Exchange Coord Directions      M - Block Titles
              F - Generate Exploded View       BLOCK GENERATION
                                                 N - Gen Block by Symmetry
                                                 O - Gen Interior Grid
              T - RECONFIGURE GPRO               P - Gen Block Boundaries
              S - STOP GPRO                      Q - Redistribute Points
           ** ? - GPRO HELP SYSTEM ***************************************
      Option [Default=A]:
```

Depending on the item selected, additional menus and prompts for information are presented, and the selected action is performed.

2.14.1 Limitations

When rotating a grid with *gpro*, the boundary condition data is not written to the output grid file. This limitation was originally intended to force the user to re-apply coupling to the new grid, but it is unclear why other boundary types are not maintained. GMAN is the recommended tool to use for grid transformations such as rotation.

2.15 jormak

jormak may be used to find boundary points in a common grid (*.cgd*) file, and create a file containing subset information for use as a journal file for CFPOST or PLOT3D. If a CFPOST journal file is specified, *jormak* writes the appropriate `ZONE` and `SUBSET` commands. If a PLOT3D journal file is specified, *jormak* writes `wall` keywords with the appropriate attributes and indices.

jormak is typically used to identify viscous or inviscid wall points. However, it can be used to locate other boundary types as well by specifying the appropriate numeric boundary condition code. A list of common boundary condition codes is provided in Table 1.

Table 1: Numeric Boundary Condition Codes

n>0	Coupled to Zone *n*
0	Undefined
-1	Reflection
-3	Freestream
-4	Viscous Wall
-7	Arbitrary Inflow
-8	Outflow
-9	Inviscis Wall
-10	Self-Closing
-11	Singular Axis
-15	Bleed
-16	Pinwheel Axis
-17	Frozen

Example

The two-dimensional grid shown in Figure 7 is used for the Tutorial test case in the *Wind-US User's Guide*. There are three zones, with grid sizes of 17 × 6, 33 × 11, and 17 × 11, respectively. The *.cgd* file is *case4.cgd*, and the top and bottom boundaries are defined as inviscid walls.

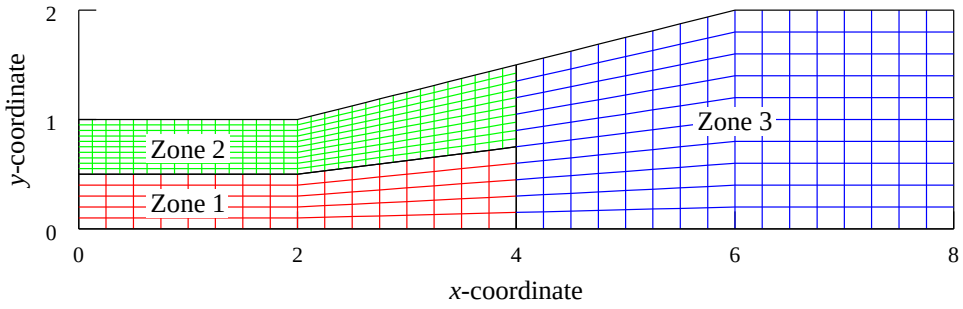

Figure 7: Input grid for *jormak*

jormak could be used to set up the `ZONE` and `SUBSET` commands for use by CFPOST as follows. Lines in slanted type are typed by the user.

```
% jormak

                    ***** jormak *****
```

```
          Select the desired version from the following list.

0) END
1) jormak optimized version

Single program automatically selected.

 jormak - Version 1.8 (last changed 2007/02/14 16:52:27)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************

 JOURNAL CREATE UTILITY
 ----------------------

 Common File:Not Open
 Zone Range :  0-  0

 1) Open common File
 2) Create CFPOST 2D-Journal File
 3) Create CFPOST 3D-Journal File
 4) Create PLOT3D Journal File
 5) Create partial FIELDVIEW fvbnd file
 6) Change Active Zone Range
 7) Exit
1
 Enter the cgd file name
case4.cgd
 Opening File: case4.cgd

 JOURNAL CREATE UTILITY
 ----------------------

 Common File:case4.cgd
 Zone Range :  1-  3

 1) Open common File
 2) Create CFPOST 2D-Journal File
 3) Create CFPOST 3D-Journal File
 4) Create PLOT3D Journal File
 5) Create partial FIELDVIEW fvbnd file
 6) Change Active Zone Range
 7) Exit
2
 Enter file name for CFPOST journal file
case4.cfpost.jou
```

```
 Search for Viscous, Inviscid, Both or Other? [V]:
i
 Searching for wall points in zone:   1
 Searching for wall points in zone:   2
 Searching for wall points in zone:   3

 JOURNAL CREATE UTILITY
 ----------------------

 Common File:case4.cgd
 Zone Range :  1-  3

 1) Open common File
 2) Create CFPOST 2D-Journal File
 3) Create CFPOST 3D-Journal File
 4) Create PLOT3D Journal File
 5) Create partial FIELDVIEW fvbnd file
 6) Change Active Zone Range
 7) Exit
7
```

The resulting file *case4.cfpost.jou* contains

```
ZONE   1
SUBSET K   1   1 I   1  17 J   1   1
ZONE   2
SUBSET I   1  33 K   1   1 J  11  11
ZONE   3
SUBSET K   1   1 I   1  17 J   1   1
ZONE   3
SUBSET I   1  17 K   1   1 J  11  11
```

This file could be used as a journal file during a CFPOST session to specify the solid walls as the subset for subsequent processing.

2.16 mpigetnzone

mpigetnzone may be used to get the number of zones in a common grid (*.cgd*) file. E.g., for a 12-zone *.cgd* file named *my_file.cgd*,

```
% mpigetnzone my_file.cgd
12
```

The user will be prompted for the file name if it isn't specified.

mpigetnzone is used by the Wind-US run scripts when MPI message passing is being used in a parallel run, and the number of zones isn't specified on the command line using the `-nzones` option.

2.17 recombine

recombine may be used to convert a "split" structured common flow (*.cfl*) file, created using the *decompose* utility, back to the original grid system.

The syntax is simply:

```
recombine
```

recombine uses the *combine*.xxx.*inp* files and the *split.cfl* file, created by *decompose*, to reconstruct the original zone structure. The "combined" *.cfl* file is named *combine.cfl*. It does not recombine the grid file, as it is assumed to already exist.

2.18 resplt.pl

resplt.pl is a Perl script that replaces the older Fortran version of resplt. Except for the "average mode" capability, resplt.pl has the same or greater capability as the older version. It is invoked automatically by the "resplt" command. The older Fortran version can be executed with the "resplt -exe" command.

resplt.pl may be used to extract residuals and/or integrated quantities from the list output (*.lis*) file created during a Wind-US run. A GENPLOT file is created for post-processing using CFPOST, containing the coordinates for curves of the selected quantities vs. iteration number, cycle, or time level, depending on the quantity selected. (See the *CFPOST User's Guide*.) This is extremely useful in monitoring convergence of the solution during a Wind-US run.

To run *resplt.pl*, simply issue the command `resplt.pl`. Command-line options (Section 2.18.2) may be used to specify some or all of the necessary input. If the `-batch` option is specified, all input must be specified via command-line options; otherwise, the user is prompted interactively. It should be noted that some features of *resplt.pl* are only available using command-line options, while others are only available in interactive mode.

2.18.1 Interactive Mode

After issuing the *resplt.pl* command, the user is prompted for the full name of the *.lis* file, unless it was specified on the command line. The following menu will then appear:

```
Exit                 0
Select Plane(s)     90
Select Zone(s)      91
Select Frequency    92
Select Summing mode 97
Select Direction    98
Select Average mode 99
Confined Outflow             Bleed        Region
  Mass Flow Ratio   15         Plenum p      71
  Back Pressure     16         Plenum mdot   72
  Average p0        93       Angle of Attack 73
Residuals     Big   L2       Integ. Planes Zone Grand Sum
  NS            1    2         Force     11    5    8   28
  k-e           3    4         Lift      17   18   19   29
  B-B          20   21         Moment    12    6    9
  S-A          22   23         Aer Mmnt  63   64   65
  SST          24   25         Momentum  13    7   10
  P-W          35   36         Mass      14   26
NEWTON NS      51   52         Heat Flx  54   55
Convergence   Zone Global    Adjoint  Zone Grand
  Force        61   62         ADM       67   68
Optional Var  Zone Global      TLM       69   70
  Value        56   57
```

To choose a specific option, enter the corresponding number at the "`Enter Selection`" prompt. Note that the data being requested must be consistent with the data that was written into the *.lis* file during the Wind-US run, as controlled by the keywords `LOADS`, `MASS FLOW`, etc.

The various options from the *resplt.pl* menu are described below.

`Exit` — Create the GENPLOT file based on the specified options, and exit *resplt.pl*.

`Select Plane(s)` — Extract integrated data only for selected "planes". The "planes" are actually the surfaces and/or subsets specified using the `SURFACE` and `SUBSET` keywords in the `LOADS` keyword block. When used, this option must be specified before choosing the particular integrated data to be extracted (i.e., from the `Planes` column of the `Integ.` section). You'll be prompted for the desired plane numbers. Multiple planes and/or a range of planes may be specified using syntax like "`2,3,6-8`". The default is all planes.

Planes are numbered in the order specified in the Wind-US input data (*.dat*) file, and are "global", not "zonal". Thus, the `Select Zone(s)` menu option does not apply.

For example, imagine a five-zone grid with two `SURFACE` keywords specified in the `LOADS` block for each zone, in order. In *resplt.pl*, specifying zone 3 with `Select Zone(s)`, then planes 1-2 with `Select Plane(s)`, will result in the integrated values for the two surfaces in zone 1 being written into the GENPLOT file, not the two surfaces in zone 3. To write the values for the two surfaces in zone 3 into the GENPLOT file, specify planes 5-6 using `Select Plane(s)`.

`Select Zone(s)` — Extract data only for selected zones. When used, this option must be specified before choosing the particular data to be extracted. You'll be prompted for the desired zone numbers. Multiple zones and/or a range of zones may be specified using syntax like "`2,3,6-8`". The default is all zones.

`Select Frequency` — Frequency for extracting data. When used, this option must be specified before choosing the particular data to be extracted. You'll be prompted for the desired frequency. For a frequency value of n, every n'th value found in the *.lis* file, starting with the first value, will be extracted. I.e., if $n = 10$, values 1, 11, 21, etc., will be extracted. The default is 1 (i.e., all values are extracted).

`Select Summing mode` — Toggle switch for turning summing mode on and off. When used, this option must be specified before choosing the particular data to be summed or not summed. Values of an extracted residual or quantity for the selected planes, zones, or bleed regions will be summed, and a single curve will be written into the GENPLOT file, rather than separate curves for each plane, zone, or bleed region. The default is off.

`Select Direction` — Direction of forces, moments, or momentum fluxes to be extracted. When used, this option must be specified before choosing the particular data to be extracted. You'll be prompted for the desired direction. The valid responses, all of which are case-insensitive, are described below. Note that not all of these are applicable to all the quantities that may be extracted. The default is to extract values for all applicable directions.

`X`, `Y`, `Z`	Forces, moments, or momentum fluxes in the Cartesian x, y, or z direction (applicable to `Force`, `Moment`, and `Momentum` in the `Integ.` section of the *resplt.pl* menu)
`Drag`, `Lift`, `Side`	Drag, lift, or side forces (applicable to `Lift` in the `Integ.` section)
`Roll`, `Yaw`, `Pitch`	Roll, yaw, or pitch moments (applicable to `Aer Mmnt` in the `Integ.` section). When `Roll`, `Yaw`, or `Pitch` is specified as a direction, *resplt.pl* actually extracts the x, y, or z moment, respectively, and takes the negative.
`2d`, `symm`	Only extract 2-D (i.e., symmetric) loads. For `Force` and `Momentum`, only the x and y components are extracted; for `Moment`, only the z component is extracted.

If the specified direction is inconsistent with the integrated quantities to be extracted (e.g., if `Drag` is specified as the direction, but `Force` is specified from the `Integ.` section instead of `Lift`), *resplt.pl* will try to recover by automatically redefining the direction as indicated in the following table.

Specified Direction	**Integ. Choice**	**Direction Redefinition**
`X`, `Y`, or `Z`	`Lift`	`X -> Drag, Y -> Lift, Z -> Side`
`X`, `Y`, or `Z`	`Aer Mmnt`	`X -> Roll, Y -> Yaw, Z -> Pitch`
`Drag`, `Lift`, or `Side`	`Force` or `Aer Mmnt`	`Drag -> X, Lift -> Y, Side -> Z`
`Roll`, `Yaw`, or `Pitch`	`Force` or `Lift`	`Roll -> X, Yaw -> Y, Pitch -> Z`

`Select Average mode` Not yet implemented.

`Confined Outflow` The options in this section of the *resplt.pl* menu allow extraction of various quantities related to boundary conditions at an outflow boundary (formerly called "confined outflow").

If the `MASS FLOW` keyword was used in the Wind-US run to specify the mass flow at an outflow boundary, the ratio of the computed mass flow to the desired mass flow (`Mass Flow Ratio`), and the resulting back pressure (`Back Pressure`), may be extracted. Note that when the `DIRECT` option is used with the `MASS FLOW` keyword, the pressure at the boundary isn't spatially-constant; in this case the pressure that's extracted will be the pressure at the first boundary point.

For any of the outflow boundary conditions (i.e., `COMPRESSOR FACE`, `DOWNSTREAM MACH`, `DOWNSTREAM PRESSURE`, `MASS FLOW`, and `OUTFLOW NON-REFLECTING`), if `TEST` option 123 was specified, the ratio of the average total pressure at the boundary to the freestream value (`Average p0`) may be extracted.

For these quantities, the abscissa written into the GENPLOT file is simply an integer, starting at zero, that corresponds to the order of the data in the *.lis* file, not the iteration or cycle number. The `-start` and `-stop` command-line options don't apply.

`Residuals` The options in this section allow extraction of the log of the residuals

for the various equation sets as a function of iteration number. The maximum residual (the column labeled `Big`) and the maximum L2 norm of the residual (`L2`) may be extracted for the mean flow Navier-Stokes and chemistry equations (`NS`), and for the equations in the k-ϵ (`k-e`), Baldwin-Barth (`B-B`), Spalart-Allmaras (`S-A`), SST (`SST`), and pointwise (`P-W`) turbulence models.

`NEWTON NS` — For cases run using Newton iteration, with the same time step size in all zones, `NEWTON NS` may be used to extract the log of the maximum residual, and the log of the maximum L2 norm of the residual, as a function of time level for any equation group in all zones.

`Convergence` — If `CONVERGE LOAD` was used in the Wind-US run, the change in the zonal and global forces and moments over a cycle, in the Cartesian x, y, and z directions, may be extracted as a function of cycle number. Pressure and viscous forces and moments are summed, not extracted separately. The values are expressed as a percentage of the maximum force or moment value in all the zones.

`Optional Var` — Not yet implemented.

`Bleed` — Not applicable to Wind-US.

`Angle of Attack` — Extract the angle of attack history as a function of cycle number, written to the *.lis* file when the `FIXED_CL` keyword is used in a Wind-US run.

`Integ.` — The options in this section allow extraction of integrated data specified using the `LOADS` keyword block in a Wind-US run, as follows:

- `Force` — Pressure and viscous forces in the Cartesian x, y, and z directions. In the `LOADS` block, the surface/subset specification must include `FORCE`, plus `VISCOUS` if viscous forces are desired. Pressure and viscous forces are extracted separately, not summed.
- `Lift` — Pressure and viscous drag, lift, and side forces. In the `LOADS` block, the surface/subset specification must include `FORCE`, plus `VISCOUS` if viscous forces are desired, and `LIFT` or `DRAG` must be specified with the `PRINT` keyword. Pressure and viscous forces are extracted separately, not summed.
- `Moment` — Pressure and viscous moments in the Cartesian x, y, and z directions. In the `LOADS` block, the surface/subset specification must include `MOMENT`, plus `VISCOUS` if viscous moments are desired. Pressure and viscous moments are extracted separately, not summed.
- `Aer Mmnt` — Aerodynamic pressure and viscous moments (i.e., roll, yaw, and pitch). In the `LOADS` block, the surface/subset specification must include `MOMENT`, plus `VISCOUS` if viscous moments are desired. (For roll, yaw, and pitch, *resplt.pl* actually extracts the x, y, and z moment, respectively, and takes the negative.)
- `Momentum` — Momentum in the Cartesian x, y, and z directions.

In the `LOADS` block, the surface/subset specification must include `MOMENTUM`.

`Mass` Mass flux in the Cartesian x, y, and z directions. In the `LOADS` block, the surface/subset specification must include `MASS`.

`Heat Flx` Heat flux. In the `LOADS` block, the surface/subset specification must include `HEAT`.

The extracted values may be the integrated values over the selected surfaces or subsets (the column labeled `Planes`), the totals for the selected zones (`Zone`), or the global totals over all the zones (`Grand`). The menu choices in the `Planes`, `Zone`, and `Grand` columns correspond to the data written into the *.lis* file when the `PLANES`, `ZONES`, and `TOTALS` options, respectively, are used with the `PRINT` keyword in the `LOADS` block.

The choices in the `Sum` column apply only to integrated loads computed when `CONVERGE LOAD` is used in a Wind-US run. For `Force`, the `Sum` value corresponds to the global sum (i.e., over all the zones) of both forces and moments in the Cartesian x, y, and z directions. For `Lift`, it corresponds to the global sum of the drag, lift, and side forces. Pressure and viscous forces and moments are summed, not extracted separately.

For `Lift`, the `Sum` choice (unlike `Planes`, `Zone`, and `Grand`), does not require that `LIFT` or `DRAG` be specified with the `PRINT` keyword in the `LOADS` block. In fact, the opposite is true; `LIFT` and `DRAG` must not be specified with the `PRINT` keyword. The directions for the drag, lift, and side forces are determined by the angles of attack and yaw specified with the `FREESTREAM` keyword.

For the menu choices in the `Planes` and `Zone` columns, the values are written into the GENPLOT file as a function of iteration number. For the `Grand` and `Sum` columns, the values are written as a function of cycle number.

`Adjoint` Not applicable to Wind-US.

2.18.2 Command-Line Options

The command-line options for *resplt.pl* are listed in the following subsections. Multiple names for the same option are separated by a vertical bar (`|`). Optional arguments are enclosed in brackets, with a vertical bar used to separate choices (e.g., `[arg1|arg2]`). There are additional options to *resplt.pl* that are either not yet fully implemented, or don't apply to Wind-US, and thus are not listed here.

General

The following options apply to the use of *resplt.pl* as a whole.

`-help|h|?` Display usage information and quit, ignoring any other options.

`-batch|b` Run in batch mode, with all input specified via command-line options. The default is to run in interactive mode.

`-list` *file*	Name of the list output (*.lis*) file, including the *.lis* extension.
`-gen` *file*	Name of the GENPLOT file to be created (or extended; see `-reuse`), including the *.gen* extension.
`-multiy|multi`	When there are multiple curves on a plot (e.g., L2 residuals in multiple zones), create a GENPLOT file with multiple sets of y values vs. a single set of x values (i.e., Format 2). The default is to create a GENPLOT file with multiple sets of y vs. x values (Format 1).
`-reuse|r`	Append to an existing GENPLOT file, specified using the `-gen` option, when extracting data from a *.lis* file that has grown after a restarted Wind-US calculation. When using this option, you must either: (1) use the same options, in the same order, that were used with *resplt.pl* to create the existing GENPLOT file; or (2) not use any options defining what's to be extracted, and let *resplt.pl* attempt to determine it automatically from the contents of the existing GENPLOT file. This option is ignored if the specified GENPLOT file doesn't exist.
`-statistics|stat`	Write statistical summary to standard output.

Selections

These options may be used to limit the amount of data to be extracted by selecting specific surfaces/subsets, zones, directions, etc.

`-plane` *list*	Extract integrated data only for the listed "planes". The "planes" are actually the surfaces and/or subsets specified using the `SURFACE` and `SUBSET` keywords in the `LOADS` keyword block. Multiple planes and/or a range of planes may be specified using syntax like "`2,3,6-8`". The default is all planes. Planes are numbered in the order specified in the Wind-US input data (*.dat*) file, and are "global", not "zonal". Thus, the `-zone` option does not apply. See `Select Plane(s)` in Section 2.18.1 for an example.
`-zone` *list*	Extract integrated data only for the listed zones. Multiple zones and/or a range of zones may be specified using syntax like "`2,3,6-8`". The default is all zones.
`-start` n	Iteration or cycle number to begin extracting data. For the loads (i.e., the integrated quantities in the `Integ.` section of the *resplt.pl* menu, or in the "Loads" section below), the value n corresponds to the variable used for the abscissa in the GENPLOT file. I.e., n is an iteration number for `plane` and `zone`, and a cycle number for `grand` and `sum`. However, for the residuals n is a cycle number, even though the abscissa in the GENPLOT file is iterations. The default for n is the first value in the *.lis* file.
`-stop` n	Iteration or cycle number to stop extracting data. See the `-start` option for details. The default is the last value in the *.lis* file.
`-skip` n	Frequency for extracting data. Every n'th value found in the *.lis* file, starting with the first value (assuming `-start` isn't used), will

be extracted. I.e., if $n = 10$, values 1, 11, 21, etc., will be extracted. The default is 1 (i.e., all values are extracted).

`-sum` Turn summing mode on. Values of an extracted residual or quantity for the selected planes, zones, or bleed regions will be summed, and a single curve will be written into the GENPLOT file, rather than separate curves for each plane, zone, or bleed region. The default is off.

`-direction|dir` *direction* Direction of forces, moments, or momentum fluxes to be extracted. The valid values for *direction*, all of which are case-insensitive, are described below. Note that not all of these are applicable to all the quantities that may be extracted. Note also that to extract 2-D (i.e., symmetric) loads, the command-line option `-symm` is used; thus, unlike in interactive mode, `2d` and `symm` are not valid values for *direction*. The default is to extract values for all applicable directions.

`X`, `Y`, `Z`	Forces, moments, or momentum fluxes in the Cartesian x, y, or z direction (applicable to the `-force`, `-moment`, and `-momentum` options)
`Drag`, `Lift`, `Side`	Drag, lift, or side forces (applicable to the `-lift` option)
`Roll`, `Yaw`, `Pitch`	Roll, yaw, or pitch moments (applicable to the `-aeromom` option). When `Roll`, `Yaw`, or `Pitch` is specified as a direction, *resplt.pl* actually extracts the x, y, or z moment, respectively, and takes the negative.

If the specified direction is inconsistent with the integrated quantities to be extracted (e.g., if `Drag` is specified as the direction, but the `-force` option is used instead of `-lift` to specify the integrated quantity), *resplt.pl* will try to recover by automatically redefining the direction. See `Select Direction` in Section 2.18.1 for details.

`-symm|symmetric` Only extract 2-D (i.e., symmetric) loads. For `-force` and `-momentum`, only the x and y components are extracted; for `-moment`, only the z component is extracted. The `-direction` option does not apply.

Confined Outflow

The following options allow extraction of various quantities related to boundary conditions at an outflow boundary.

`-mfr` For outflow boundaries where the `MASS FLOW` keyword was used in the Wind-US run, extract the ratio of the computed mass flow to the desired mass flow.

`-backp` For outflow boundaries where the `MASS FLOW` keyword was used in the Wind-US run, extract the back pressure. Note that when the `DIRECT` option is used with the `MASS FLOW` keyword, the pressure at the boundary isn't spatially-constant; in this case the pressure that's extracted will be the pressure at the first boundary point.

`-avgp0` For boundaries with any outflow boundary conditions (i.e., COMPRESSOR FACE, DOWNSTREAM MACH, DOWNSTREAM PRESSURE, MASS FLOW, and OUTFLOW NON-REFLECTING), extract the ratio of the average total pressure at the boundary to the freestream value. This option requires that `TEST` option 123 be specified in the Wind-US run.

For these quantities, the abscissa written into the GENPLOT file is simply an integer, starting at zero, that corresponds to the order of the data in the *.lis* file, not the iteration or cycle number. The `-start` and `-stop` command-line options don't apply.

Residuals

The following options allow extraction of the log of the residuals for the various equation sets.

`-ns [big|l2]` Extract the maximum residual (`big`) or the maximum L2 norm of the residual (`l2`) for the mean flow Navier-Stokes and chemistry equations. The default is `big`.

`-ke [big|l2]` Extract the maximum residual (`big`) or the maximum L2 norm of the residual (`l2`) for the k-ϵ turbulence model equations. The default is `big`.

`-bb [big|l2]` Extract the maximum residual (`big`) or the L2 norm of the residual (`l2`) for the Baldwin-Barth turbulence model equation. The default is `big`.

`-sa [big|l2]` Extract the maximum residual (`big`) or the L2 norm of the residual (`l2`) for the Spalart-Allmaras turbulence model equation. The default is `big`.

`-sst [big|l2]` Extract the maximum residual (`big`) or the maximum L2 norm of the residual (`l2`) for the SST turbulence model equations. The default is `big`.

`-pw [big|l2]` Extract the maximum residual (`big`) or the L2 norm of the residual (`l2`) for the pointwise turbulence model equation. The default is `big`.

`-newton [big|l2]` For cases run using Newton iteration, with the same time step size in all zones, extract the maximum residual (`big`) or the maximum L2 norm of the residual (`l2`) for any equation group in all zones. The default is `big`.

With the `-newton` option, the residuals are written into the GENPLOT file as a function of time level. With the others, they're a function of iteration number.

Angle of Attack

This option only applies when the FIXED_CL keyword is used in a Wind-US run.

`-alpha` Extract the angle of attack history as a function of cycle number.

Loads

The options in this section allow extraction of integrated data specified using the LOADS keyword block in a Wind-US run.

`-force [plane|zone|grand|sum]` Extract forces in the Cartesian x, y, and z directions. In the `LOADS` block, the surface/subset specification must include `FORCE`, plus `VISCOUS` if viscous forces are desired. Pressure and viscous forces are extracted separately, not summed. The default argument is `grand`.

`-lift [plane|zone|grand|sum]`	Extract drag, lift, and side forces. In the `LOADS` block, the surface/subset specification must include `FORCE`, plus `VISCOUS` if viscous forces are desired, and `LIFT` or `DRAG` must be specified with the `PRINT` keyword. Pressure and viscous forces are extracted separately, not summed. The default argument is `grand`.
`-moment [plane|zone|grand]`	Extract moments in the Cartesian x, y, and z directions. In the `LOADS` block, the surface/subset specification must include `MOMENT`, plus `VISCOUS` if viscous moments are desired. Pressure and viscous moments are extracted separately, not summed. The default argument is `grand`.
`-aeromom|pitch [plane|zone|grand]`	Extract aerodynamic pressure and viscous moments (i.e., roll, yaw, and pitch). In the `LOADS` block, the surface/subset specification must include `MOMENT`, plus `VISCOUS` if viscous moments are desired. (For roll, yaw, and pitch, *resplt.pl* actually extracts the x, y, and z moment, respectively, and takes the negative.) The default argument is `grand`.
`-momentum [plane|zone|grand]`	Extract momentum in the Cartesian x, y, and z directions. In the `LOADS` block, the surface/subset specification must include `MOMENTUM`. The default argument is `grand`.
`-mass [plane|zone]`	Extract mass flux in the Cartesian x, y, and z directions. In the `LOADS` block, the surface/subset specification must include `MASS`. The default argument is `plane`.
`-heatflux|hflux [plane|zone]`	Extract heat flux. In the `LOADS` block, the surface/subset specification must include `HEAT`. The default argument is `plane`.
`-noflip`	When `-lift` is specified, extract the side force values from the *.lis* file without changing the sign. The default is to extract the negative of the values in the file.
`-nopressure|nopress`	For forces and moments, only extract the viscous values. The default is to extract both pressure and viscous values.
`-noviscous|novisc`	For forces and moments, only extract the pressure values. The default is to extract both pressure and viscous values.

When specifying the integrated values to be extracted, the argument `plane`, `zone`, `grand`, or `sum` determines the region, or scope, of the integration, as described below.

`plane` Extract values for the surfaces or subsets specified using the `-plane` option. This corresponds to the data written into the *.lis* file when the `PLANES` option is used with the `PRINT` keyword in the `LOADS` block.

`zone` Extract total values for the zones specified using the `-zone` option. This corresponds to the data written into the *.lis* file when the `ZONES` option is used with the `PRINT` keyword in the `LOADS` block.

`grand` Extract global total values over all the zones. This corresponds to the data written into the *.lis* file when the `TOTALS` option is used with the `PRINT` keyword in the `LOADS` block.

`sum` This applies only to integrated loads computed when `CONVERGE LOAD` is used in a Wind-US run. For the `-force` option, the global sum (i.e., over all the zones) of both forces and moments in the Cartesian x, y, and z directions will be extracted. For the `-lift` option, the global sum of the drag, lift, and side forces will be extracted. Pressure and viscous forces and moments are summed, not extracted separately.

For the `-lift` option, using `sum` (unlike using `plane`, `zone`, or `grand`), does not require that `LIFT` or `DRAG` be specified with the `PRINT` keyword in the `LOADS` block. In fact, the opposite is true; `LIFT` and `DRAG` must not be specified with the `PRINT` keyword. The directions for the drag, lift, and side forces are determined by the angles of attack and yaw specified with the `FREESTREAM` keyword.

For `plane` and `zone`, the values are written into the GENPLOT file as a function of iteration number. For `grand` and `sum`, the values are written as a function of cycle number.

2.19 thplt

thplt may be used to extract information stored in time history (*.cth*) files, and create GENPLOT files for post-processing. (See the *CFPOST User's Guide* for a description of the format of GENPLOT files.) Time history files are created during Wind-US runs by using the `HISTORY` keyword. This is useful in tracking the values of certain specified parameters over time in an unsteady flow.

The thplt utility is used with .cth files created usingthe improved time history capability introduced in Wind alpha 5.52. It will not work with .cth files created using earlier versions of Wind; for those, use the timplt utility.

Example

Suppose an unsteady flow case has been computed in three Wind-US runs (i.e., an initial run and two restarts), with the following `HISTORY` keyword block in an input data file named *case4_3d.dat*:

```
History
   Variable p
   Frequency 5
   Region 1 17 17  1  1 1 1
   Region 2 17 17 11 11 5 5
   Region 3  9  9 11 11 7 7
```

The resulting time history files will contain the static pressure at the points (17,1,1), (17,11,5), and (9,11,7) in zones 1, 2, and 3, respectively. There will be three files, one for each run, and Wind-US automatically appends the creation date and time to the end of the file name. The file names will thus be something like *case4_3d.cth.10-23-07h15m53* (i.e., created on Oct. 23, 2007, at 3:53 PM), *case4_3d.cth.10-24-07h09m16*, and *case4_3d.cth.10-25-07h10m04*.

The *thplt* utility would be used as follows to create three GENPLOT files containing the unsteady pressure data, one file for each of the three points specified by the `Region` keywords in the `History` keyword block. Lines in slanted type are typed by the user. Note that when multiple *.cth* files will be processed, all of them must be opened before selecting the grid points and variables.

```
% thplt

                          ***** thplt *****

          Select the desired version from the following list.

0) END
1) thplt optimized version

Single program automatically selected.

 thplt - Version 1.5 (last changed 2005/05/18 22:35:51)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
 Enter name of time history file: case4_3d.cth.10-23-07h15m53
```

```
          1  Select grid point
          2  Select grid range
          3  Select variable
          6  Open another time history file
          7  Clear all selected points
          8  End

 Enter menu selection: 6
 WARNING: Entering multiple CTH files.
          It is your responsibility to ensure that they are increasing in time
Enter name of time history file: case4_3d.cth.10-24-07h09m16

          1  Select grid point
          2  Select grid range
          3  Select variable
          6  Open another time history file
          7  Clear all selected points
          8  End

 Enter menu selection: 6
 WARNING: Entering multiple CTH files.
          It is your responsibility to ensure that they are increasing in time
Enter name of time history file: case4_3d.cth.10-25-07h10m04

          1  Select grid point
          2  Select grid range
          3  Select variable
          6  Open another time history file
          7  Clear all selected points
          8  End

 Enter menu selection: 1

Enter zone number of point to track: 1
Enter I, J, K of point to track: 17,1,1

 Currently selected points:
ZONE   1 (17,   1,  1)

          1  Select grid point
          2  Select grid range
          3  Select variable
          6  Open another time history file
          7  Clear all selected points
          8  End

 Enter menu selection: 3
     1  p

  You may select up to            5 variables.
```

```
 Enter selection: 1

 Currently selected points:
ZONE   1 (17,   1,  1)

 Currently selected variables are
p

          1  Select grid point
          2  Select grid range
          3  Select variable
          4  Create GENPLOT file
          5  Analyze data
          6  Open another time history file
          7  Clear all selected points
          8  End

 Enter menu selection: 4
Enter name of genplot file: press1.gen
 Reading data...
 complete.

 Currently selected points:
ZONE   1 (17,   1,  1)

 Currently selected variables are
p

          1  Select grid point
          2  Select grid range
          3  Select variable
          4  Create GENPLOT file
          5  Analyze data
          6  Open another time history file
          7  Clear all selected points
          8  End

 Enter menu selection: 7

 Currently selected variables are
p

          1  Select grid point
          2  Select grid range
          3  Select variable
          6  Open another time history file
          7  Clear all selected points
          8  End
```

[*Repeat "Select grid point," "Select variable," and "Create GENPLOT file" steps for the*

points at (17,11,5) and (9,11,7) in zones 2 and 3, using different names for the GENPLOT files.]

```
Enter menu selection: 8
```

2.20 tmptrn

In the Wind-US input data (*.dat*) file, the TTSPEC keyword block is used when a point-by-point surface temperature boundary condition is being used, and/or when point-by-point boundary layer transition information is being specified. The *tmptrn* utility is normally used to add the specified wall temperature and/or boundary layer transition information to an existing common flow (*.cfl*) file.

Within a zone, boundary layer transition and/or wall temperature data may be specified on a j or k boundary surface as a function of the i index. Boundary layer transition is indicated by a number from 0.0 to 1.0, indicating laminar and fully turbulent flow, respectively. Wall temperature is specified in degrees Rankine, with 0.0 indicating an adiabatic wall, and non-dimensionalized using the reference conditions from the *.cfl* file before writing it to the file. The defaults, for zones not specified, are fully turbulent and adiabatic wall.

In earlier versions of *tmptrn*, the temperature written into the *.cfl* file was in K. This error was fixed in version 1.8 of *tmptrn*. However, some versions of Wind and Wind-US had coding to accomodate the dimensional temperature value, and these versions will not work with temperature distributions written using *tmptrn* 1.8 and above. The versions affected are shown in the following table.

Wind/Wind-US Versions		*tmptrn* Version
Wind	5.1 – 5.207	1.8 and up
	5.208 – 5.213	1.7 and below
	5.213 and up	1.8 and up
Wind-US	1.1 – 1.42	1.8 and up
	1.43 – 1.100	1.7 and below
	1.101 and up	1.8 and up
Wind-US	2.1 – 2.16	1.7 and below
	2.17 and up	1.8 and up

.cfl files containing temperature distributions written using *tmptrn* 1.7 and below can easily be modified to work with Wind versions 5.1 – 5.207, 5.213 and up, etc., by re-setting the temperature distribution using *tmptrn* 1.8.

Four functional distributions are allowed.

- Constant.
- Constant for $i < i_1$ and $i > i_2$, and variable in between. The variation between i_1 and i_2 is determined by a user-specified polynomial exponent.
- Constant within i_1 to i_2 ranges. With this option, an arbitrary distribution may be input to Wind-US by specifying the value at each i grid point.
- Piecewise linear between specified i locations.

Example

Suppose we have the three-zone configuration shown in Figure 8, and we want to run a viscous case with a boundary layer transition region from laminar to turbulent flow between $x = 1.5$ and 2.5. This corresponds to $i = 7$ to 11 on the j_1 surface in zone 1, and $i = 13$ to 21 on the j_{max} surface in zone 2.

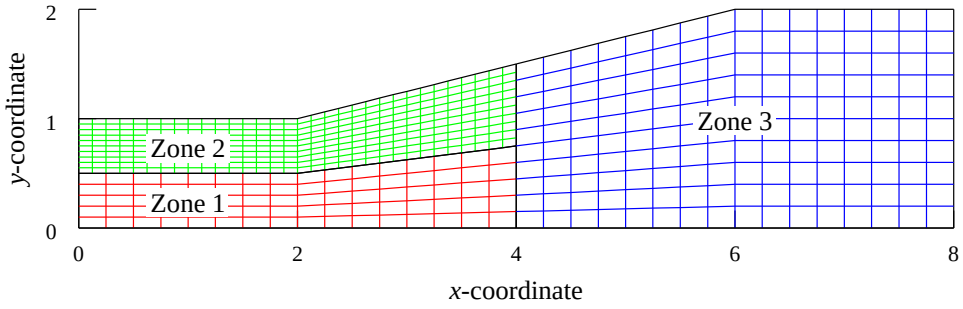

Figure 8: Input grid for *tmptrn*

We could create the boundary layer transition information, and add it to the *.cfl* file, as shown below. Lines in slanted type are typed by the user.

```
% tmptrn
                              ***** tmptrn *****

          Select the desired version from the following list.

0) END
1) tmptrn optimized version

Single program automatically selected.


 tmptrn - Version 1.9 (last changed 2007/02/14 19:49:23)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************

  TEMPERATURE/TRANSITION FILE GENERATION PROGRAM
  NOTE: (I=1,I=IMAX DISABLED FOR NOW)
  NOTE: Date is stored directly into the cfl file.

  ital the solution file name.
case4.cfl

 Input the zone to be changed (0=no more changes, MXZONE=           3 )
1

    3: FACE  J1
    4: FACE  JMAX
 Enter a number from the list above:
3
```

```
  ARE YOU GENERATING 0 - TRANSITION OR
                     1 - TEMPERATURE FILE?
  Enter 0 or 1
0

 Input values may range from 0.0 to 1.0 as follows:
  0.0 = Laminar
  1.0 = Fully Turbulent

  Distribution type menu
  0 - constant
  1 - Constant i<i1, transition, constant i>i2
  2 - Constant within i1,i2 ranges
  3 - Piecewise linear between i locations

  Enter distribution type for
     Zone            1  Boundary            3
1

  Entering i outside i=1,IMAX allows starting
    or ending zone within the transition region
  For instantaneous: start = last i at upstream
                     end   = next i location
  INPUT I STATION TO START TRANSITION, IMAX=          17
7
  INPUT I STATION TO END TRANSITION, IMAX=          17

11

  Polynomial transition parameter
  0  - not allowed
  <1 - concave down
  1. - linear transition
  >1 - concave up
  INPUT POWER OF POLYNOMIAL
1

  Save Trans    data for zone            1  Surface FACE  J1 (Y/N, <CR>=Y)


 Trans    data successfully written!!!!!

 Input the zone to be changed (0=no more changes, MXZONE=            3 )
2

    3: FACE  J1
    4: FACE  JMAX
 Enter a number from the list above:
4

  ARE YOU GENERATING 0 - TRANSITION OR
```

```
                    1 - TEMPERATURE FILE?
  Enter 0 or 1
0

 Input values may range from 0.0 to 1.0 as follows:
  0.0 = Laminar
  1.0 = Fully Turbulent

  Distribution type menu
  0 - constant
  1 - Constant i<i1, transition, constant i>i2
  2 - Constant within i1,i2 ranges
  3 - Piecewise linear between i locations

  Enter distribution type for
     Zone            2  Boundary            4
1

  Entering i outside i=1,IMAX allows starting
    or ending zone within the transition region
  For instantaneous: start = last i at upstream
                     end   = next i location
  INPUT I STATION TO START TRANSITION, IMAX=          33
13
  INPUT I STATION TO END TRANSITION, IMAX=          33

21

  Polynomial transition parameter
  0  - not allowed
  <1 - concave down
  1. - linear transition
  >1 - concave up
  INPUT POWER OF POLYNOMIAL
1

  Save Trans    data for zone            2  Surface FACE  JM (Y/N, <CR>=Y)


 Trans    data successfully written!!!!!

 Input the zone to be changed (0=no more changes, MXZONE=           3 )
0
Thats all she wrote!!!!!!!!!!
```

2.21 USintrpltQ

USintrpltQ may be used to interpolate the solution from one unstructured grid to another. The user is prompted for the existing *.cgd* and *.cfl* files containing the original grid and solution, for the existing *.cgd* file with the new grid, and for the *.cfl* file that will be created containing the interpolated solution.

Example

Lines in slanted type are typed by the user.

```
% USintrpltQ

                    ***** USintrpltQ *****


          Select the desired version from the following list.

0) END
1) USintrpltQ   (32-bit version with CPU optimization)

Single program automatically selected.

Enter unstructured grid common file to interpolate from..... : SSjet.cgd
Enter unstructured solu common file to interpolate from..... : SSjet.cfl
Enter unstructured grid common file to interpolate to..... : SSjet2.cgd
Enter new unstructured solu common file to interpolate to..... : SSjet2.cfl
 USintrpltQ - Version 1.4 (last changed 2001/12/18 16:15:37)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
in ReadInp: GridFile1 = SSjet.cgd

            FlowFile1 = SSjet.cfl

            GridFile2 = SSjet2.cgd

            FlowFile2 = SSjet2.cfl

 in intrpltQ, xmin(1) =   -2.65000350720223
              xmax(1) =    7.98333350720223
              ymin(1) =  -0.166666502483524
              ymax(1) =    3.49999650248353
              zmin(1) =  -0.159937593951805
              zmax(1) =   0.159937593951805
 in intrpltQ, zone =            1
 in GetXyzVrtcs2, xmin =   -2.16667000000000
                  xmax =    7.50000000000000
```

```
                ymin =   0.000000000000000E+000
                ymax =    3.33333000000000
                zmin =  -0.116331539343348
                zmax =   0.116331539343348
in intrpltQ, DistMax =    1.22618905074406
             DistMin =   0.000000000000000E+000
in IntrpltQ, nqv          =           10
             nVrtcsZ      =        72421
             nVrtcsZmax   =        72421
             nTotalVrtcs1 =        94923
CPU time =    1.352794
```

2.22 windpar

windpar may be used to compute an estimate of the potential for speed-up of a particular Wind-US case when run in parallel mode. The calculation is based on the likely number of grid points per processor, as determined from the number of zones and the grid size in each zone. Grid sizes are read from a common grid (*.cgd*) or common flow (*.cfl*) file. It is assumed that each processor is equally powerful, and zonal differences in solution procedure are not taken into account.

Example

Suppose we have the simple three-zone configuration shown below, with grid sizes 17×6, 33×11, and 17×11 in zones 1, 2, and 3, respectively.

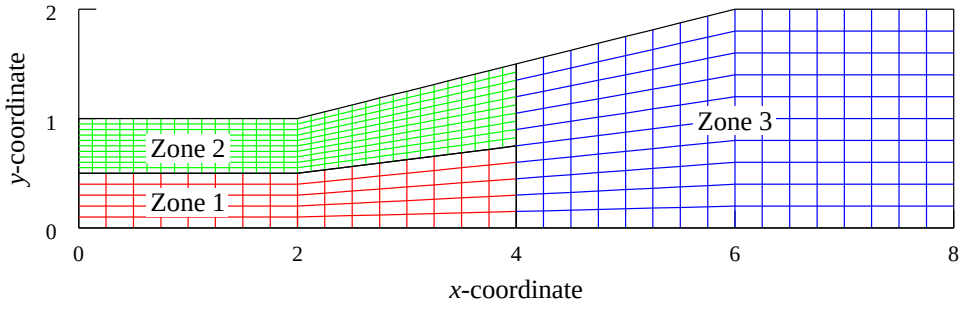

Figure 9: Input grid for *windpar*

Running *windpar* for this configuration gives the following results. Lines in slanted type are typed by the user.

% *windpar*

```
                          ***** windpar *****

          Select the desired version from the following list.

0) END
1) windpar

Single program automatically selected.

 windpar - Version 1.5 (last changed 2007/02/14 20:05:48)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
 Please input CFL or CGD filename
case4.cgd

 Proc Speedup Proc Speedup Proc Speedup Proc Speedup
    1   1.000    2   1.796    3   1.796
```

```
Maximum speedup attainable is
a speedup of   1.796 on   2 processors
with an efficiency of  89.807%

 Proc   Eff.  Proc   Eff.  Proc   Eff.  Proc   Eff.
   1 100.000    2  89.807    3  59.871
```

The speedup is defined as the total number of grid points in the problem, divided by the maximum number of grid points on any processor. The efficiency is then defined as the speedup divided by the number of processors, expressed as a percentage.

For this particular three-zone case, we have

Number of Processors	*Points on Processor 1*	*Points on Processor 2*	*Points on Processor 3*	*Speedup*	*Efficiency*
1	652	-	-	1.000	100%
2	363	289	-	1.796	89.807%
3	363	187	102	1.796	59.871%

Thus, the best scenario for this case is to run on two processors. Because the total number of grid points in zones 1 and 3 combined (289) is less than the total number in zone 2 (363), the maximum speedup is the same whether two or three processors are used.

3 Depricated Utilities

3.1 resplt.exe

The Fortran version of this utility, documented below, is now depricated. It has been superceded by a Perl script version (resplt.pl) which is automatically executed by the "resplt" command. To run the Fortran version, one must use "resplt.exe".

resplt may be used to extract residuals and/or integrated quantities from the list output (*.lis*) file created during a Wind-US run, and create a GENPLOT file for post-processing. (See the *CFPOST User's Guide* for a description of the format of GENPLOT files.) This is extremely useful in monitoring convergence of the solution during a Wind-US run.

After invoking "*resplt.exe*", and entering the name of the *.lis* file, the following menu choices appear:

```
Exit                 0
Select Zone(s)      91
Select Frequency    92
Select average mode 99
Confined Outflow
  Mass Flow Ratio   15
  Back Pressure     16
  Average p0        93
Residuals     Big   L2       Integ. Planes Zone Grand
  NS            1    2         Force    11    5    8
  k-e           3    4         Lift     17   18   19
  B-B          20   21         Moment   12    6    9
  S-A          22   23         Momentum 13    7   10
  SST          24   25         Mass     14   26    -
NEWTON NS      51   52         Heat Flx 54   55    -
Time History        53
```

Convergence data may be extracted for selected zones using `Select Zone(s)`, and at a selected cycle interval using `Select Frequency`. The defaults are all zones and every cycle.

The `Select average mode` choice may be used to calculate a composite convergence number for the entire solution by averaging over all the zones, providing that the same residuals are computed in each zone.

If the `MASS FLOW` keyword is being used, specifying the mass flow at an `outflow` boundary (formerly called "confined outflow"), the ratio of the computed mass flow to the desired mass flow (`Mass Flow Ratio`), and the resulting back pressure (`Back Pressure`), may be extracted. Note that when the `DIRECT` option is used with the `MASS FLOW` keyword, the pressure at the boundary isn't spatially-constant; in this case the pressure that's extracted will be the pressure at the first boundary point.

For any of the outflow boundary conditions (i.e., `COMPRESSOR FACE`, `DOWNSTREAM MACH`, `DOWNSTREAM PRESSURE`, `MASS FLOW`, and `OUTFLOW NON-REFLECTING`), if `TEST` option 123 was specified, the ratio of the average total pressure at the boundary to the freestream value (`Average p0`) may be extracted.

For these "Confined Outflow" quantities, the abscissa written into the GENPLOT file is simply an integer, starting at zero, that corresponds to the order of the data in the *.lis* file, not the iteration

or cycle number.

The maximum residual (the column labeled `Big`) and the L2 norm of the residual (`L2`) may be extracted for the mean flow Navier-Stokes equations (`NS`), and for the equations in the Chien k-ϵ (`k-e`), Baldwin-Barth (`B-B`), Spalart-Allmaras (`S-A`), and Mentor SST (`SST`) turbulence models.

For Wind-US cases run using Newton iteration, with the same time step size in all zones, `NEWTON MS` may be used to extract the maximum residual, or the maximum L2 norm of the residual, for any equation group in all zones, as a function of the Newton time level.

The `Time History` menu choice was used with an earlier version of Wind, with time history results written into the *.lis* file. It does not apply to files created with Wind 5 and later.

Integrated convergence parameters specified during a Wind-US run using the `LOADS` keyword may also be extracted. Depending on the options specified in the `LOADS` input block, these parameters may include pressure forces, lift, moments, momentum, mass flow, and (for unstructured grids) heat flux. The integrated values may be over subsets or surfaces specified in the `LOADS` input block (the column labeled `Planes`), totals over the selected zones (`Zone`), or grand totals over all the zones (`Grand`).

A detailed example illustrating the use of *resplt* to extract the maximum residual from the *.lis* file is included in the "Tutorial" section of the *Wind-US User's Guide*.

4 Obsolete Utilities

4.1 adfedit

> *The adfedit utility is now obsolete and no longer being maintained. The CGNSview utility available as part of the CGNS (CFD General Notation System) project (http://cgns.github.io/) is recommended for examining the contents of an ADF file. This tool provides a graphical user interface that makes navigating the file structure much easier.*

adfedit may be used to view (as text) the structure and contents of an ADF file. ADF (Advanced Data Format) files are created using the ADF I/O library routines, developed as part of the CGNS (CFD General Notation System) project. Version 3 common files, such as the common grid (*.cgd*) and common flow (*.cfl*) files currently used with Wind-US, are ADF files. The version of *adfedit* in the Wind-US tools distribution is essentially the same (identical to?) an earlier version available with the CGNS software.

Issuing the command `adfedit` causes the ADF main menu to be presented.

```
  ADF Utilities Main Screen
Selections:
  tr              : ADF translators.
  br              : ADF browser
  q               : quit program
  ? [command]     : Help.


ADFmain>
```

At any point in an *adfedit* session, typing "`?`" by itself redisplays the current menu. Typing "`?` *command*" gives information on the named command. And, typing "`q`" will exit the *adfedit* session.

The ADF translators, intended to translate between ADF and plot3d files, apparently do not currently work, for at least for some types of plot3d and Version 3 common files, and are therefore not documented here.

The ADF browser is started by typing "`br`" at the `ADFmain` prompt, and causes the ADF browser menu to be presented.

```
  ADF Browser:
Selections:
  o filename      : Open an existing database.
  z               : Close the open database.
  pwd             : Print current node.
  cd              : Change current node.
  ls              : List children of current node.
  dd              : Print description of node data.
  pd              : Print node data.
  t               : Miscellaneous ADF tools menu.
  b               : Backup to previous menu.
  q               : Quit this program.
  ? [command]     : Help.


ADFbrowse>
```

Additional detail on the available browser commands is presented below. This information is taken from the help files supplied with *adfedit*.

`o` *filename*	Opens the ADF file *filename* as OLD in NATIVE format.
`z`	Closes an open ADF file.
`pwd`	Prints the name of the current node.
`cd [`*name*`]`	Imitates the Unix *cd* command. The browser maintains a notion of the current node. This command changes the current node to that of the named node. If no node is named, it sets the current node to the root node. It understands "`.`", "`..`", and relative and absolute pathnames. It will accept (unambiguous) name abbreviations at the end of pathnames but not regular expressions.
`ls -lt [`*names*`]`	Imitates the Unix *ls* command. If no arguments are given, the names of the children of the current node are given. Otherwise the children of the named nodes are listed. Recognized names are the same as for *cd*. The `-l` option causes the label of the named node to be included. The `-t` option causes the listing to indicate whether or not the node is a link.
`dd`	Displays the type and dimensions of data at the current node.
`pd`	Prints the data at the current node.
`t`	Displays a menu of some additional tools for working with ADF files.
`b`	Returns to the previous menu.

The browser command "`t`" causes a menu of additional tools to be presented.

```
  ADF Miscellaneous Tools
Selections:
```

```
pt  [filename] : Print file hierarchy to a file.
lc             : Check for loops in hierarchy.
td             : Report longest path in hierarchy.
fl             : List all files involved in database.
cdb [filename] : Collect database into one file.
sdb filename   : Spread database into multiple files.
b              : Backup to previous menu.
q              : quit program
?   [command]  : Help.

ADFtools>
```

From the above list, the `lc`, `td`, `fl`, `cdb`, and `sdb` commands are currently not available. The `pt` command causes a view of the hierarchy in the ADF file to be writted into the file *filename*. If the file name is omitted, the information is displayed on the screen. There are two options to the `pt` command: `-l` causes the node labels to be included, and `-d` causes the data type and size information to be included.

4.2 b4wind

The b4wind utility is now obsolete and no longer being maintained.

b4wind is an interactive utility that may be used for a variety of tasks that may be helpful to users of NPARC Alliance software. Its features are accessed through a GUI written in *Tcl/Tk*, which must be installed on the user's system.

The options available in *b4wind* are listed below, along with a brief explanation. More detail may be found in the *b4wind User's Guide* by Don Todd, the author of *b4wind*.

- Convert files

  This option may be used to convert between various types of grid and solution files. Three types of grid and solution files are supported — NPARC restart files, PLOT3D files (all types), and common grid (*.cgd*) and common flow (*.cfl*) files.

- Compute initial conditions

  b4wind may also be used to compute an initial flow field. Three types of grid and solution files are supported — NPARC restart files, PLOT3D files (all types), and common grid (*.cgd*) and common flow (*.cfl*) files.

  After reading a grid file, initial conditions may be computed using one of three methods.

  - Uniform flow
  - Interpolation between given values at the ends of each block
  - One-Dimensional flow

- Interpolate

  Given a grid file and the corresponding solution file, this option may be used to interpolate the solution onto the grid in a second grid file, and create the new solution file. Trilinear interpolation is used. Again, three types of grid and solution files are supported — NPARC restart files, PLOT3D files (all types), and common grid (*.cgd*) and common flow (*.cfl*) files.

- Compute Reynolds number

  Various reference conditions and gas properties, and their units, may be interactively specified, and the resulting Reynolds number will be displayed.

- Scan a common file

  This option may be used display the header data and structure information for common grid (*.cgd*) and common flow (*.cfl*) files. It can also be used to delete selected turbulence model variables from a *.cfl* file.

- Prepare NPARC or XAIR NAMELIST

  This option may be used to create a namelist input file for NPARC (the flow solver formerly supported by the NPARC Alliance) or XAIR (a flow solver used primarily at AEDC).

4.3 cfrevert

The cfrevert utility is now obsolete and no longer being maintained.

The format and variable names for some of the data in common flow (*.cfl*) and boundary data (*.tda*) files written by Wind-US changed between the release of Wind-US 1.0 and Wind-US 2.0. As a result, *.cfl* and *.tda* files written by Wind-US 2.0 cannot be read by Wind-US 1.0.

Should it be necessary to restart a calculation with Wind-US 1.0 that was previously run with Wind-US 2.0, the *cfrevert* utility may be used to convert the current *.cfl* file to one that may be read with Wind-US 1.0. The *.tda* file written by Wind-US 2.0 should be deleted, and will automatically be recreated when the calculation is restarted.

To use *cfrevert*, simply type "`cfrevert`" and you'll be prompted for the existing *.cfl* file name, which should include the *.cfl* extension. The converted file will overwrite the original, so if you need to retain the original file, copy it to a different name before using *cfrevert*.

It should be noted that Wind-US 2.0 can read *.cfl* and *.tda* files written by Wind-US 1.0, so no conversion is necessary when using older files to restart a calculation with Wind-US 2.0.

A terminal session illustrating the use of *cfrevert* is shown below, for a common flow file named *case3.cfl*. Lines in a slanted font are typed by the user.

```
% cfrevert

                          ***** cfrevert *****

          Select the desired version from the following list.

0) END
1) cfrevert optimized version

Single program automatically selected.

  Enter name of the common cfl file.
case3.cfl
  All done!!!!!!!!!
```

4.4 cfsequence

The cfsequence utility is now obsolete and no longer being maintained.

cfsequence may be used to remove grid points in specified zones in a structured common grid (*.cgd*) or common flow (*.cfl*) file, using a procedure similar to the one used with the SEQUENCE keyword in Wind-US.

Note — When used with .cfl files, currently this utility will only work for files created using Wind-US 1.0, and earlier. When used with .cfl files created using Wind-US 2.0, it will abort with a "Requested variable does not exist" error.

Input to *cfsequence* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and parameters may be specified.

`CHECK`	Checks the input file for errors without performing any operations.
`FILE` *file_in*	The input *.cgd* or *.cfl* file.
`OUTPUT` *file_out*	The output common file, for the sequenced grid. For *.cgd* files, any boundary conditions specified in the original (fine) grid are retained in the new (coarse) grid, and zone coupling data are regenerated for the new grid. In the output file, zones with holes will not have a complete fringe boundary; GMAN must be used to recreate it.
`SEQUENCE` *nsi nsj nsk* [*izone*] `SEQUENCE` *nsi nsj nsk* [*izone*] ...	The `SEQUENCE` commands tell *cfsequence* how to sequence the zone. The parameters *nsi*, *nsj*, and *nsk* are the number of sequencing levels in the i, j, and k directions, and *izone* is the zone number. If *izone* is zero or omitted, the specified sequencing will be applied to all zones. The number of grid points in the sequencing direction must be equal to $2^n m + 1$, where m is an integer and n is the number of sequencing levels. Thus for 1 level of sequencing, the grid must have an odd number of points in the sequencing direction. For two levels there must be $4m + 1$ points, etc.

Example

Suppose we have the simple three-zone configuration shown below, with grid sizes 17×6, 33×11, and 17×11 in zones 1, 2, and 3, respectively.

The following *cfsequence* input file, named *cfseq.inp*, will create a new *.cgd* file, with a coarser grid in zone 2.

```
/ Input grid file
/
FILE test6.cgd
/
/ Output grid file
/
OUTPUT test6_seq.cgd
```

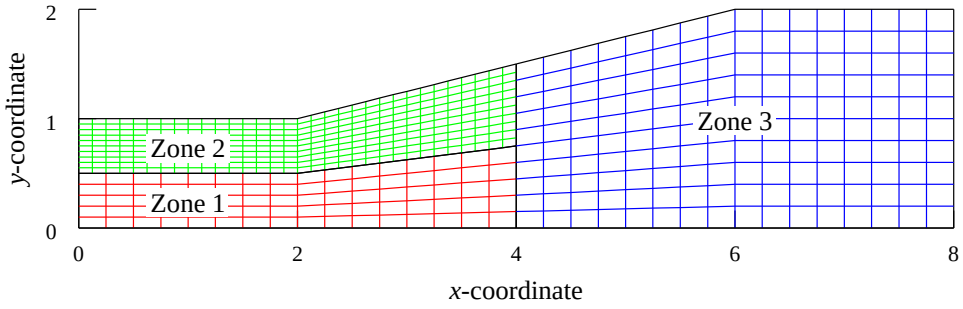

Figure 10: Input grid for *cfsequence*

```
/
/ Sequence grid in i and j directions, zone 2
/
SEQUENCE 1 1 0 2
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfsequence

                          ***** cfsequence *****

          Select the desired version from the following list.

0) END
1) cfsequence

Single program automatically selected.

Omit the .inp extension!
Enter cfsequence INPUT FILE ............. (<CR>=cfsequence.inp) : cfseq
 cfsequence - Version 1.6 (last changed 2007/02/14 00:07:27)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  SEQUENCING ZONE   1
  SEQUENCING ZONE   2
  SEQUENCING ZONE   3
  SEQUENCING BOUNDARIES OF ZONE   1
  SEQUENCING BOUNDARIES OF ZONE   2
  SEQUENCING BOUNDARIES OF ZONE   3
  RESETTING FACTORS COUPLED TO ZONE   1
  RESETTING FACTORS COUPLED TO ZONE   2
  RESETTING FACTORS COUPLED TO ZONE   3
```

The resulting grid is shown in Figure 11. With one level of sequencing in both the i and j

directions, the resulting grid size in zone 2 is 17×6.

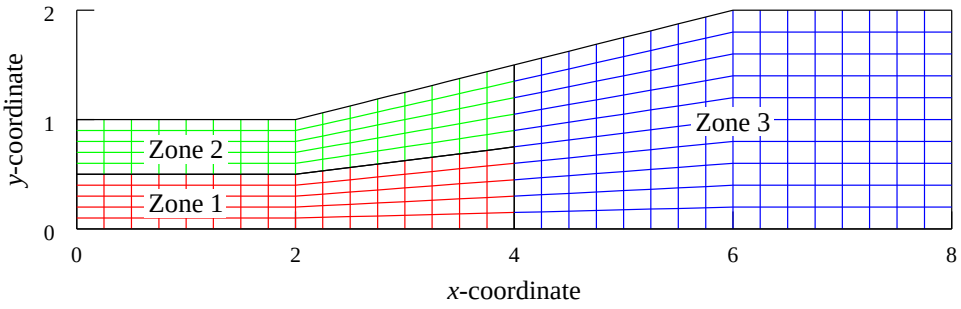

Figure 11: Output sequenced grid from *cfsequence*

4.5 cfsubset

The cfsubset utility is now obsolete and no longer being maintained.

cfsubset may be used to remove specified grid points from a structured common grid (*.cgd*) or common flow (*.cfl*) file. This may be useful, for example, in removing densely-packed grid points from a grid developed for a fully-viscous problem, for use in an inviscid calculation or a calculation using wall functions.

If *cfsubset* is used with *.cgd* files containing zones with holes, the output *.cgd* file will not have a complete fringe boundary. GMAN should be used to fix the new file.

Input to *cfsubset* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and parameters may be specified.

`CHECK` Checks the input file for errors without performing any operations.

`FILE` *file_in* The input *.cgd* or *.cfl* file, containing zones with points to be removed.

`OUTPUT` *file_out* The output common file, with the "less dense" zones. If the file is a *.cgd* file, coupling data is also generated and included in the output file.

`SUBSET ZONE` *nzone* `I` *irange* `J` *jrange* `K` *krange*
The `SUBSET` keyword tells *cfsubset* which points to *keep* from the original file. The parameter *nzone* is a zone number, or zone number range. Any zones not specified will be copied as is into the output file.

The *irange*, *jrange*, and *krange* parameters specify the points to keep in zone(s) *nzone*. These parameters are each a series of single numbers, or ranges of numbers separated by a "`-`". In ranges, no white space is allowed between the numbers and the "`-`", and an optional increment may be specified by using a "`;`" to separate it from the second range number. Note that when an increment is specified, no checking is done to assure that the endpoint is included in the resulting grid. The special values "`ALL`" and "`LAST`" may be used as appropriate.

Example 1

Suppose we have the simple three-zone configuration shown below, with grid sizes 17×6, 33×11, and 17×11 in zones 1, 2, and 3, respectively.

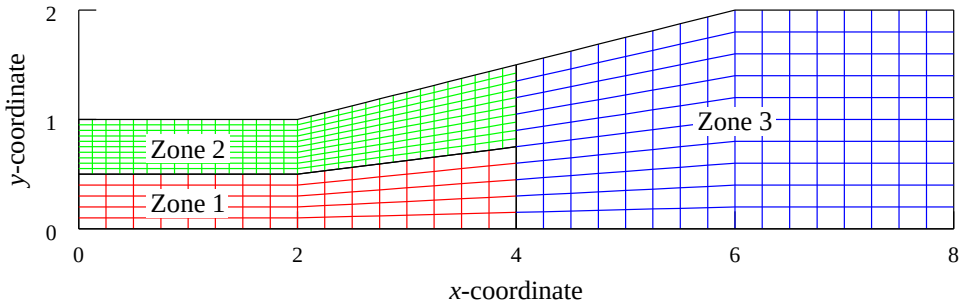

Figure 12: Input grid for *cfsubset*

The following input file for *cfsubset*, named *cfsubset1.inp* will:

- In zone 1, keep all the grid points in both directions
- In zone 2, keep every other grid point in the I direction from I = 1 to 17, all the grid points in the I direction from I = 18 to 33, and every other grid point in the J direction
- In zone 3, remove the grid points at I = 2–4, and 6–7, and keep all grid points in the J direction

```
/ Input grid file
/
FILE testa.cgd
/
/ Output grid file
/
OUTPUT testb.cgd
/
/ Define zone subsets
/
   SUBSET ZONE 2  I 1-17;2 18-LAST  J 1-LAST;2  K ALL
   SUBSET ZONE 3  I 1 5 8-LAST         J ALL       K ALL
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfsubset

                          ***** cfsubset *****

          Select the desired version from the following list.

0) END
1) cfsubset

Single program automatically selected.

Omit the .inp extension!
Enter cfsubset INPUT FILE ............. (<CR>=cfsubset.inp) : cfsubset1
 cfsubset - Version 1.7 (last changed 2007/02/14 00:34:38)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  SUBSETTING ZONE   1
  SUBSETTING ZONE   2
  SUBSETTING ZONE   3
  SUBSETTING BOUNDARIES OF ZONE   1
  SUBSETTING BOUNDARIES OF ZONE   2
  SUBSETTING BOUNDARIES OF ZONE   3
  RESETTING FACTORS COUPLED TO ZONE   1
  RESETTING FACTORS COUPLED TO ZONE   2
  RESETTING FACTORS COUPLED TO ZONE   3
```

The resulting grid is shown below. After running *cfsubset*, the grid sizes are 17×6, 25×6, and

12×11 in zones 1, 2, and 3, respectively.

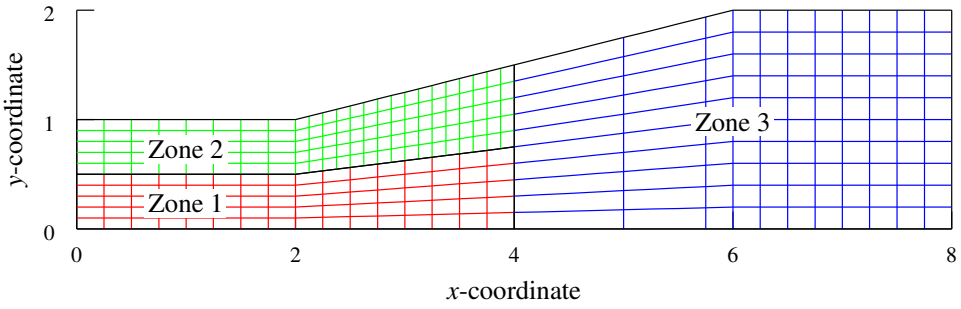

Figure 13: Output grid from *cfsubset*

Example 2

The following input file for *cfsubset* will:

- Keep all grid points in the `I` direction
- Remove the grid points at `J` = 2–16, and `K` = 2–20 in zones 1 and 3–22
- In zone 1, keep only every other grid point in the `J` direction, starting at `J` = 17
- In zones 3–22, keep only every third grid point in the `K` direction, starting at `K` = 21
- Keep all the grid points in zone 2 (and zones 23 and higher)

```
/ Input grid file
/
FILE testc.cgd
/
/ Output grid file
/
OUTPUT testd.cgd
/
/ Define zone subsets
/
SUBSET ZONE 1     I ALL  J 1 17-LAST;2  K 1 21-LAST
SUBSET ZONE 3-22  I ALL  J 1 17-LAST    K 1 21-LAST;3
```

4.6 cfunsequence

The cfunsequence utility is now obsolete and no longer being maintained.

Note — Currently this utility will only work for .cfl files created using Wind-US 1.0, and earlier. When used with .cfl files created using Wind-US 2.0, it will abort with a "Requested variable does not exist" error.

cfunsequence may be used to add grid points to specified zones in a structured common flow (*.cfl*) file.

Input to *cfunsequence* is specified in a keyword input file, with the three-letter extension *.inp*. Lines in the input file starting with a "/" are comments. The following keywords and parameters may be specified.

`CHECK`	Checks the input file for errors without performing any operations.
`FILE` *file_in*	The input *.cfl* file.
`OUTPUT` *file_out*	The output *.cfl* file.
`SEQUENCE` *nsi nsj nsk* [*izone*] `SEQUENCE` *nsi nsj nsk* [*izone*] ...	The `SEQUENCE` commands tell *cfunsequence* how to add grid points to the zone. Grid points will be added to "reverse" the effect of grid sequencing, assuming the input *.cfl* file was created using the *cfsequence* utility with the specified `SEQUENCE` commands. The parameters *nsi*, *nsj*, and *nsk* are the number of sequencing levels that were used in the i j, and k directions, and *izone* is the zone number. If *izone* is zero or omitted, it is assumed that the specified sequencing was applied to all zones. Thus, for a sequencing level of one in a given direction, new grid points will be added between each original grid point in that direction. For a sequencing level of two, the process is repeated, resulting in three grid points being added between each original grid point.

Flow field values at the added grid point locations are set equal to the average of the values at the eight surrounding points (or four in two dimensions) from the original input file. Note that, for more than one level of sequencing, there will be multiple new points inside a surrounding "box" of original grid points, and that in the output *.cfl* file the flow field values at all of these new points will be the same.

Note that *cfunsequence* cannot be used successfully with an input common grid (*.cgd*) file to create a denser grid. *cfunsequence* will run, but the averaging method used to compute grid coordinates for the output file will result in an invalid grid.

Example

The following *cfunsequence* input file, named *cfunseq.inp*, will "unsequence" the solution in the flow file *case_seq.cfl*, writing the results into the file *case_unseq.cfl*. One level of unsequencing is specified in the i and j directions in zone 2, causing solution points to be added between each between each original grid point in those directions. The solution in other zones is unchanged.

```
/ Input flow file
/
FILE case_seq.cfl
/
/ Output flow file
/
OUTPUT case_unseq.cfl
/
/ Input file was sequenced one level in i and j directions, zone 2
/
SEQUENCE 1 1 0 2
```

The terminal session is shown below. Lines in a slanted font are typed by the user.

```
% cfunsequence

                                ***** cfunsequence *****

          Select the desired version from the following list.

0) END
1) cfunsequence

Single program automatically selected.

Omit the .inp extension!
Enter cfunsequence INPUT FILE ............. (<CR>=cfunsequence.inp) : cfunseq
 cfunsequence - Version 1.6 (last changed 2007/02/14 00:49:24)
 ************************************************************************
 * Warning: This software contains technical data whose export is       *
 * restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751,   *
 * et seq.) or Executive Order 12470. Violation of these export-control     *
 * laws is subject to severe criminal penalties. Dissemination of this      *
 * software is controlled under DoD Directive 5230.25 and AFI 61-204.       *
 ************************************************************************
  UNSEQUENCING ZONE   1
  UNSEQUENCING ZONE   2
  UNSEQUENCING ZONE   3
```

4.7 cfview

The cfview utility is now obsolete and no longer being maintained. The CGNSview utility available as part of the CGNS (CFD General Notation System) project (http://cgns.github.io/) is recommended for examining the contents of a common file. This tool provides a graphical user interface that makes navigating the file structure much easier.

cfview may be used to view (as text) the structure and contents of a common file, such as a common grid (*.cgd*) or common flow (*.cfl*) file. It arose out of a need to see the very guts of a common file without imposing any assumptions about "zones" or "boundaries", etc.

Input to *cfview* is specified through commands, described below. In the following list, `|` separates multiple choices, `[]` are delimiters surrounding optional entry(s), and `{}` are delimiters surrounding multiple entries when exactly one of them is required.

`FILE` *cfname* — Open the common file *cfname*.

`{EXIT | BYE}` — Exit *cfview*.

`STATUS` — Display current node and list of subnodes and variables.

`NODE "`*node_name*`"` — Select node *node_name* as current (quotes required).

`ROOT` — Select root node as current node.

`BACK` — Go back one node in the hierarchy.

`{DUMP | LIST} NODE` — Print node header data.

`{DUMP | LIST} VARIABLE` *vname* `[START` *beg*`] [END` *end*`] [INCREMENT` *inc*`]` — Print data in variable *vname* in linear fashion, starting at *beg* (default is 1), ending at *end* (default is end of array), with increment *inc* (default is 1).

`TREE [OUTPUT` *outfile*`]` — Print the structure of the common file, indicating the nodes and subnodes, and the variable names, sizes, and types.

4.8 chmgr

The chmgr utility is now obsolete and no longer being maintained.

4.8.1 Introduction

CHMGR (CHemistry ManaGeR)[1] is a utility that assists users in selecting, assembling, checking, and formatting chemistry inputs to Wind-US. The chemistry information required by Wind-US is stored in a *.chm* file, which CHMGR will create. CHMGR manipulates the three types of information used by Wind-US to perform reacting flow analyses:

Thermodynamic data	Data describing the thermodynamic properties of a constituent including heat of formation, specific heat, and molecular weight
Transport properties	Data describing the transport properties of a constituent including viscosity and conductivity
Finite rate coefficients	Data describing the reaction rates at which the various constituents react to form new species

It should be noted that not all components of the *.chm* file are required to run Wind-US. A frozen chemistry case which is run either inviscidly or using air transport properties requires only that the thermo data be specified. To this can be added either the finite rate coefficients or transport properties. Of course, the finite rate coefficients must be specified to permit execution of a finite rate chemistry run, and the transport properties must be specified to permit accurate computation of the viscosity and conductivity of the mixture.

The program may be run using a simple text menu driver (see Section 4.8.7) or through the user-friendly *Tcl/Tk* based Graphical User Interface (GUI). The GUI simply generates the correct text menu commands and user inputs, and directs them to the input stream for the CHMGR main executable. CHMGR is programmed in Fortran 90 taking advantage of data structures for ease of manipulating data sets. Modular construction and a simple menu driver allow flexibility in dealing with different data sources and ease in adding new capabilities.

Each execution of CHMGR writes a session log of all user inputs along with a brief annotation to journal file *chmgr.jou*. The annotations make the file simple to read and edit. This journal file can be read as a script file, to control the automatic execution of the program.

The functions of the program are focused on the construction of a Wind-US chemistry model. This model has three components: thermodynamic data, transport properties, and finite rate coefficients. There is an option to load an existing Wind-US *.chm* file as a baseline in the construction of the desired chemistry model. This baseline may then be modified by the deletion or addition of components. In general, the desired data is assembled from source files, either adding to a baseline or starting from scratch. To accomplish this for a given data type (thermodynamic, transport, or reaction rate), the source file for that data is opened. The contents are then displayed in the GUI, or can be listed with the appropriate command in the text menu. The desired species or reaction rates are then selected from the source file, thus adding them to the model. Once added to the model, the species or reaction rates can be deleted or reordered. Multiple source files may be opened in sequence and selected data added to the model. When the model is completed, it is checked for internal consistency prior to output into the *.chm* file. In this process, flaws which can be repaired are automatically corrected.

[1] The material in this section was originally written by R. S. Dyer and G. P. Finfrock of Boeing, as document CM-00-04.

4.8.2 The Main GUI Window

The main GUI control is shown in Figure 14.

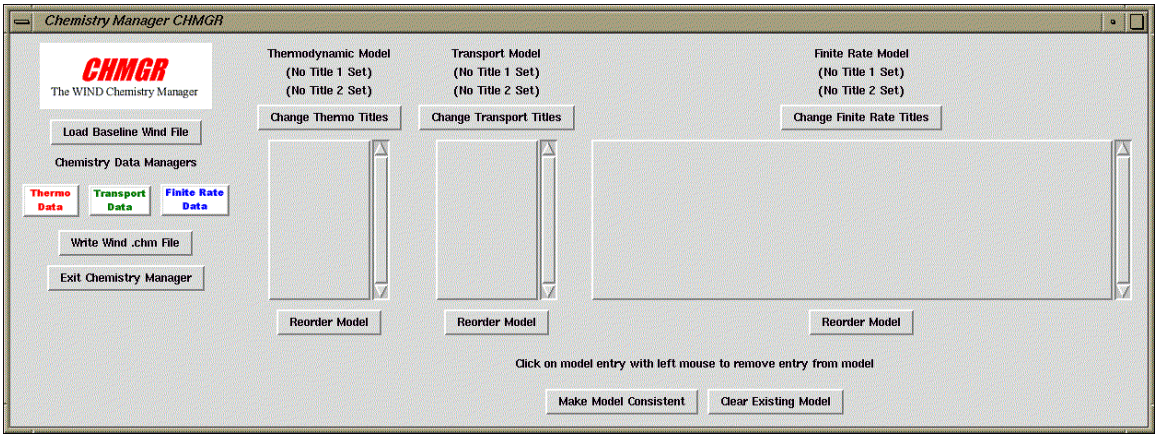

Figure 14: CHMGR Main Window

This window consists primarily of list boxes displaying the thermodynamic, transport, and finite rate data models being created. Initially, these boxes are empty as shown in Figure 14. The buttons to the left of the window control the transfer of data to the model. If a baseline Wind-US file is to be used as a starting point, the "Load Baseline Wind File" button should be selected, and a list of existing *.chm* files will be presented to select from. If the file */applusr/cfd/bin/chemistry* exists, the standard list of Wind-US chemistry files is listed from that directory. Otherwise, the directory CHMGR is being run from is used as the default location for the list of existing *.chm* files. The user may enter a different directory path as desired. An example of this window is shown in Figure 15.

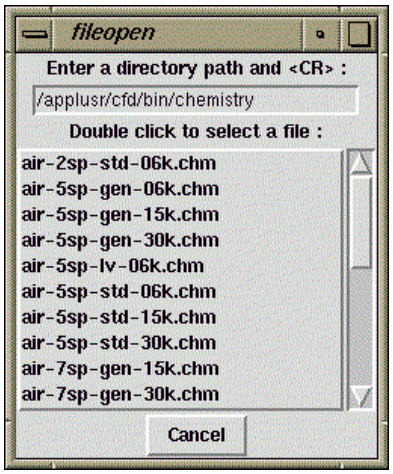

Figure 15: File Open

Note that whenever a new GUI window is opened, all other open windows become inactive until the currently presented window is acted upon.

As an example, when the *air-7sp-gen-30k.chm* standard Wind-US file is selected, the main GUI window is modified to reflect the contents of the file, as shown in Figure 16.

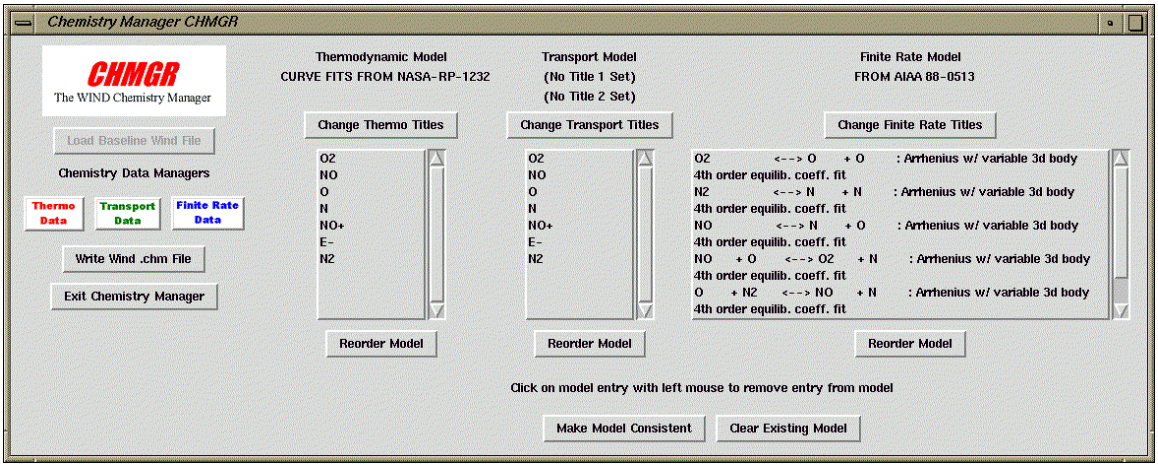

Figure 16: Main Window After Loading Baseline Wind-US File

For each section of the *.chm* file, the corresponding list boxes have been filled with data from the file. If title lines were included in the file, these have also been set at the top of the window section. At this point, the user may begin modifying the baseline file as desired. If new titles are to be set, the appropriate "Change Title" button can be selected, and a window similar to the one in Figure 17 will be displayed. The current titles are displayed in this window.

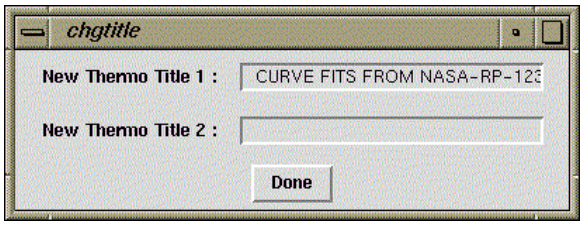

Figure 17: Change Title

The new titles entered will be reflected in the main GUI window when the "Done" button is selected.

At any time control is held by the main GUI window, the order of entries in any of the models may be modified by selecting the "Reorder Model" button associated with the model. This will cause the list box containing the specified model to be cleared, and a new window to appear containing the list of model entries, as shown in Figure 18.

The entries in the model should simply be reselected from the list in the window in the order in which they should now appear in the model being created.

As the main GUI window directs, an entry can be removed from any model when the main GUI screen is active by simply clicking on the entry name with the left mouse button. This causes the selected name to be removed from the listbox and the associated model.

To completely clear all models, the "Clear Existing Model" button can be selected any time the main GUI screen is active. This will erase all entries in all models, resulting in a "fresh start" to the CHMGR process.

To make additions to the individual models, whether or not a baseline file has been used, the

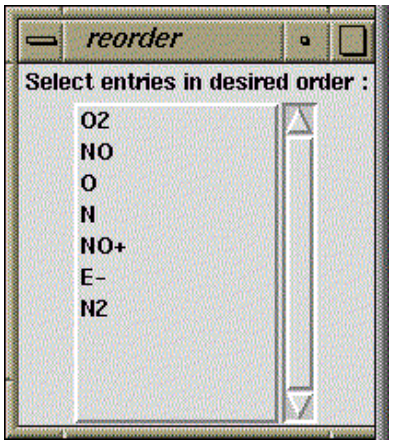

Figure 18: Reorder Entries

three buttons labeled "Thermo Data", "Transport Data", and "Finite Rate Data" on the left side of the window are used to access the various CHMGR data managers. These are described in more detail in the following sections.

4.8.3 Thermodynamic Data

Functions in the Thermodynamic Data Manager permit the user to review the contents of a file containing thermodynamic data curve fits in either the Wind-US or the NASA Glenn Research Center (GRC) format and select the desired species. When selected, the corresponding thermodynamic curve fit data for these species are then added to the thermodynamic model. Multiple input files may be opened in turn and selected species added to the model. It is also possible to output the data into another file in either Wind-US or GRC format. This might be done in order to maintain a library of curve-fit data. The format of the thermodynamic data in the *.chm* file is described in the "Files" section of the *Wind-US User's Guide*. It is similar to the NASA GRC curve fit developed by Gordon and McBride (1976). It should be noted that the C_p/R factor is used to compute a perfect gas ratio of specific heats (γ) for temperatures below the lowest curve fit temperature interval.

When the GUI is used, selection of the "Thermo Data" button results in a window containing a new set of buttons to be displayed, as shown in Figure 19.

The first two buttons in this window allow the user to open files to access thermodynamic data in either of the two supported formats. When either of these buttons is selected, a "file open" window is presented as shown in Figure 15. When a file is selected from this menu, rather than adding the full set of thermodynamic data to the new model being created, the contents of the thermodynamic data for the selected file are displayed in a new window as shown in Figure 20.

The species from the opened file that are desired to be added to the main model should now be selected from the list, or the name entered in the provided entry box. A button is also provided to add all components from the file to the main thermodynamic model being created. When all desired components have been added, the "Done" button should be selected to return the user to the main Thermodynamic Data Manager GUI shown in Figure 19.

The "Cp Curve Fit Manager" button in the Thermodynamic Data Manager GUI provides access to CHMGR's method for deriving a new thermodynamic curve-fit based on input C_p data gleaned, for

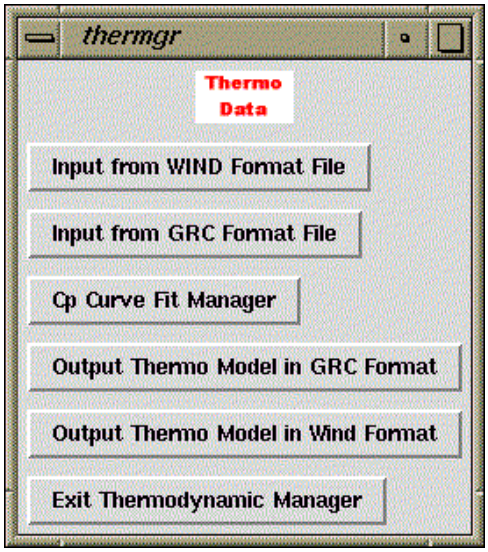

Figure 19: Thermodynamic Data Manager

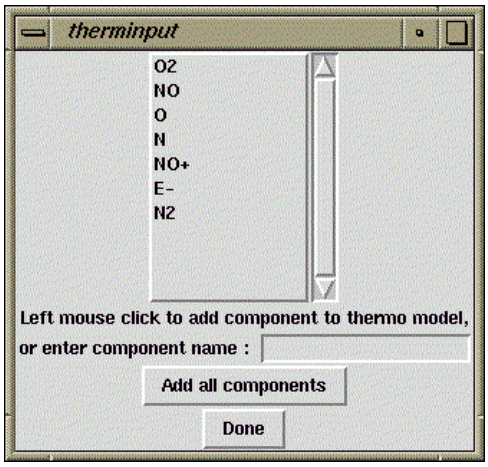

Figure 20: Thermodynamic Data Input

example, from the JANNAF tables. The resulting curve-fit data is added to the main thermodynamic model. The format and required information for these curve fits are given in Table 2.

The first three temperature-C_p/R pairs set the first interval and each additional pair of points adds another interval. For the example given in Table 2, the first interval runs from 300K to 1000K, and the second from 1000K to 5000K. It should be noted that the fifth order polynomial used in the curve-fit can produce unphysical overshoots, so the control points (or midpoints in the three-point temperature intervals; 900K and 2000K in the example) must be chosen with care.

The GUI for the C_p Curve Fit Manager is shown in Figure 21.

The species name being defined is entered in the entry box at the top of the window. Atom names and numbers as defined in Table 2 are entered in the second list box. T and C_p data as defined in Table 2 are entered in the third list box. Finally, entry boxes are provided for the molecular weight and heat of formation for the species. Also, a conversion factor is provided for the user's convenience

Table 2: C_p Curve Fit Data Format

`C4H3`	Name of molecule
`C`	Name of first atom
`4`	Number of first atoms
`H`	Name of second atom
`3`	Number of second atoms (repeat to max of four atoms)
`Q`	Q
`51.0677, 0.06338071e+06`	Molecular weight, (heat of formation)/R
`3.0000E+02 1.4770E+03`	Temperature, C_p/R
`9.0000E+02 2.4290E+03`	Temperature, C_p/R
`1.0000E+03 2.4963E+03`	Temperature, C_p/R
`2.0000E+03 2.8942E+03`	Temperature, C_p/R
`5.0000E+03 3.0545E+03`	Temperature, C_p/R

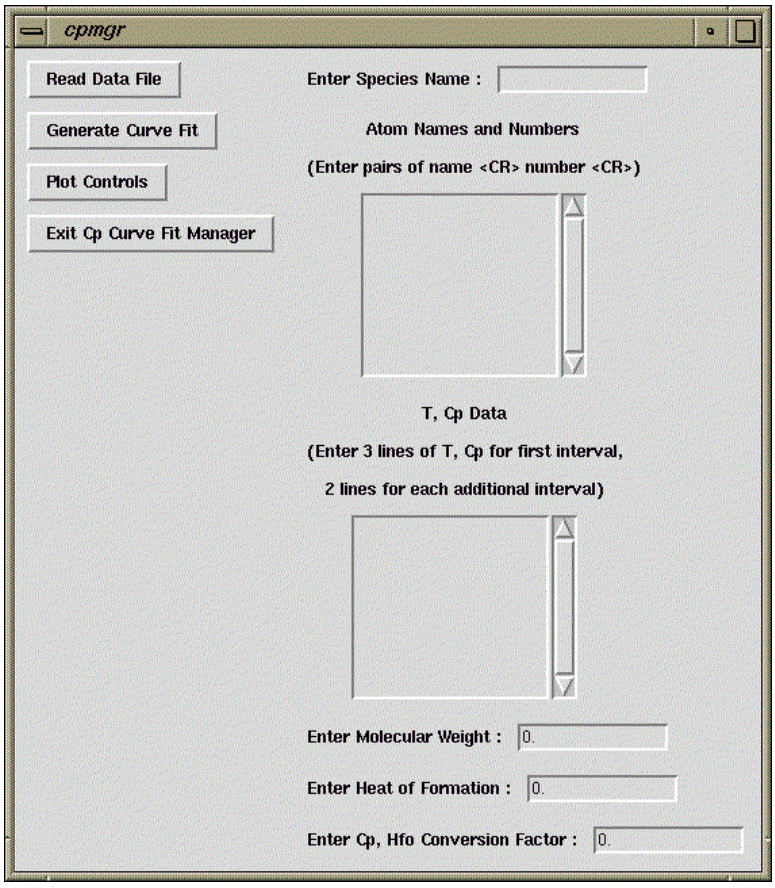

Figure 21: C_p Curve Fit Manager

and simply scales the input C_p/Rs in the case where a unit conversion is required. This should be set to unity for the case where no conversion is required. If the required curve fit data has been previously defined in a text file, this file can be read by using the "Read Data File" button on this

window. In this case the "file open" GUI from Figure 15 is presented and the corresponding data loaded to the window.

When the species has been defined, the "Generate Curve Fit" button is used to actually generate the fit. The results of the curve-fit operation can be monitored using the plotting feature of CHMGR. This feature produces a GENPLOT formatted plotting file that can be viewed using the CFPOST utility, and can be accessed from the "Plot Controls" button on this window. This button causes the window in Figure 22 to be displayed.

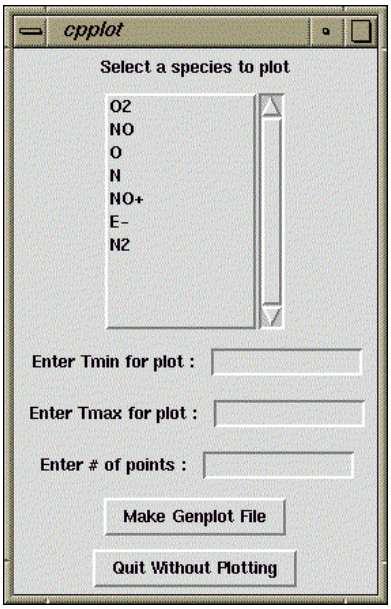

Figure 22: Thermodynamic Data Plot Control

This window displays all of the species in the main thermodynamic model being created. To obtain a plot of T vs. C_p for any species in the model, the desired species should be selected from the list, and a range of temperature to plot entered into the entry boxes. The number of points desired for the plot will be equally spread between the input temperature extremes. When the "Make Genplot File" button is selected, a *.gen* file ready for viewing in CFPOST will be created.

When the "Exit Cp Curve Fit Manager" button in Figure 21 is selected, the user is returned to the main Thermodynamic Data Manager of Figure 19. The remaining buttons on this window allow the user to output only the thermodynamic data in either of the two supported formats. The user will be prompted for a file name as shown in Figure 23.

Figure 23: Output File Name Prompt

Remember that files generated at this level contain only a section of the complete chemistry model (the thermodynamic data in this section), and cannot be used to initiate a Wind-US analysis.

The Thermodynamic Data Manager is exited by selecting the "Exit" button from the GUI.

4.8.4 Transport Data

Transport properties (viscosity and conductivity) in Wind-US are in a Sutherland's Law form:

$$\mu = A\frac{T}{B}^{3/2}\frac{B+C}{T+C}$$

which is equivalent to a form having two independent variables:

$$\mu = T^{3/2}\frac{A'}{T+C}$$

This stands in contrast to the polynomial curve fits available from the NASA GRC transport properties curve fits of Gordon 1982 which have four coefficients and are of the form:

$$\ln\mu = a\ln T + \frac{b}{T} + \frac{c}{T} + d$$

CHMGR permits the user to read in the curve fits in the NASA form and then converts them to the Wind-US form. Because the Wind-US format effectively has only two coefficients, only two values of the viscosity or conductivity can be used. The two selected are those at the endpoints of each temperature interval in the NASA data. This ensures continuity between temperature intervals. Figure 24 shows a sample comparison between the Wind-US curve-fit and the original NASA curve-fit for N_2. There is currently no capability to derive the NASA curve-fit form from Wind-US curve-fits.

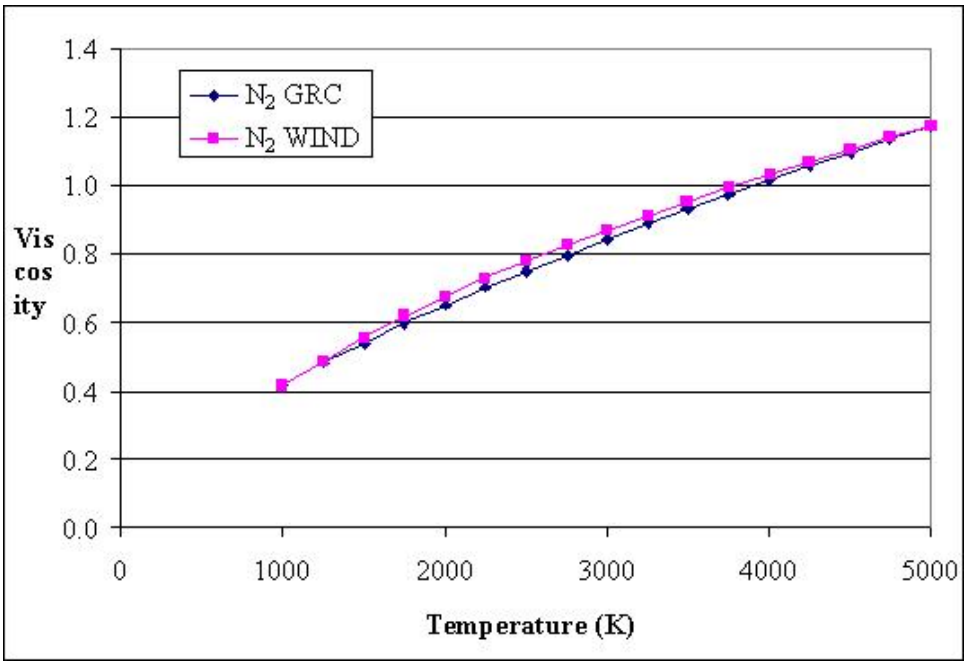

Figure 24: Comparison of Viscosity Curve-Fits for N_2

The Transport Data Manager GUI is shown in Figure 25.

The first two buttons on this window allow the user to load just the transport data from either of the two supported file formats. As with the thermodynamic data, selection of either button causes

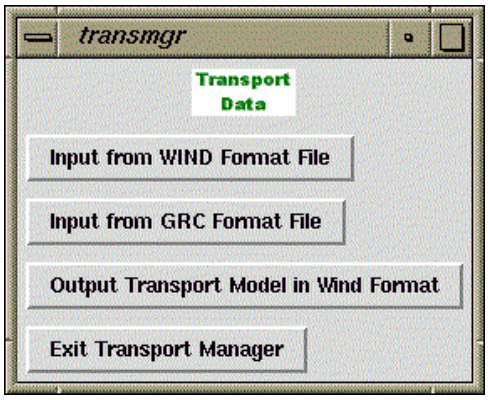

Figure 25: Transport Data Manager

the file select window from Figure 15 to be presented. When a file has been selected, a window similar to that in Figure 20 is presented, but for the transport species instead of the thermodynamic components. When a species is selected from the list, it is added to the main chemistry model in the main CHMGR window. All species can be added at once using the provided button.

As discussed above, the transport data model input from either file format can be output in Wind-US *.chm* file format, but not in the NASA format. For this reason, a single output button is provided in the Transport Data Manager GUI, labeled "Output Transport Model in WIND Format". Selection of this button will cause only the transport data from the main model to be written to a Wind-US file. A new file name is prompted for as in Figure 23, and the file is written. As in the Thermodynamic Data Manager, the file created at this level cannot be run in Wind-US, as it contains only the transport data and not the full chemistry model.

4.8.5 Finite Rate Data

The `ISPEC` flag at the top of the *.chm* file specifies the reaction rate type. Table 3 gives the current available combinations of reaction type. While only one reaction rate type is permitted in a Wind-US *.chm* file, CHMGR permits the user to manipulate many types at once. This might be desirable if constructing a library of reaction rates.

The "Minimum Reaction Temperature" sets the temperature threshold below which Wind-US will not compute the reaction rates. This prevents the execution of the reaction rate computations at temperatures which are too low to initiate a chemical reaction, conserving computer resources.

There are several obsolete forms of the reaction data in the *.chm* file which are hard-coded for certain chemical systems, rather than general reaction types. These obsolete forms are not supported by CHMGR.

The GUI for the Finite Rate Data Manager is shown in Figure 26.

Selection of the first button, "Set Minimum Reaction Temperature" causes the window shown in Figure 27 to be displayed. The user simply enters the new minimum reaction temperature in degrees Kelvin in the entry box, as described above.

The new value will be written to the *.chm* file when the finite rate model is output. The currently set value is always output at the top of the main Finite Rate GUI as seen in Figure 26.

Table 3: Available Reaction Rate Types

ISPEC	Wind-US Routine	Forward Reaction Type	Reverse Reaction Type
100	`rates`	Arrhenius; variable 3rd-body efficiency	Equilibrium coefficient curve fit
110	`ratesa`	Arrhenius; average 3rd-body efficiency	Equilibrium coefficient curve fit
115	`ratesadl`	Arrhenius; average 3rd-body efficiency	Alternate equilibrium coefficient curve fit
120	`ratesf`	Westbrook-Dreyer 1-step global reaction	—
130	`ratesb`	Arrhenius; average 3rd-body efficiency	Arrhenius; average 3rd-body efficiency

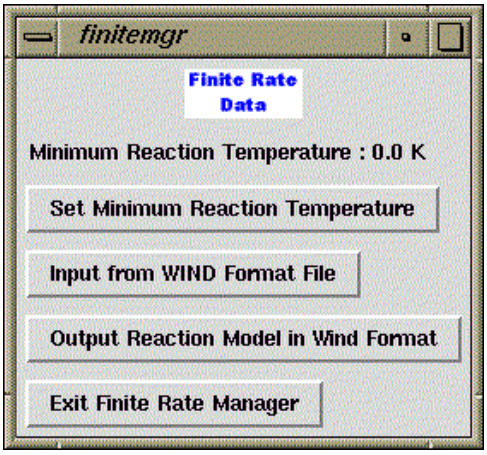

Figure 26: Finite Rate Data Manager

Figure 27: Minimum Reaction Temperature Input

To add finite rate data from a Wind-US model, the "Input from WIND Format File" button is selected, and the file open window from Figure 15 is displayed. The finite rate data from the selected file is output to a list similar to those for transport data or thermodynamic data, as in Figure 20. Each finite rate equation takes up two lines in the finite rate data list box, however. Either line may be selected to add the reaction to the main chemistry model being generated. As with the transport and thermodynamic sections, a button is provided to add all the data from the file to the main

model, and a "Done" button returns the user to the main Finite Rate Data Manager GUI.

To output only the finite rate data to a *.chm* file, the "Output Reaction Model in Wind Format" button is used. This causes the currently defined finite rate section of the main chemistry model to be output to a *.chm* file. A file written at this level cannot be run in Wind-US, as it contains only finite rate data.

When the Finite Rate model has been defined, the "Exit Finite Rate Data Manager" button can be used to return the user to the main CHMGR GUI.

4.8.6 Final Chemistry Model Output

At this point, the CHMGR system has allowed the user to create a new chemistry model based on thermodynamic and transport data from either Wind-US or GRC formatted files and finite rate data from Wind-US files. Optionally, a baseline file may have been loaded as a starting point. The various entries in the model sections can be reordered as desired, or interactively deleted from the model being built. When a final model has been assembled, the "Make Model Consistent" button on the main GUI should be selected. This will check for consistencies required by the Wind-US chemistry routines, and correct problems automatically if it can.

For example, Wind-US requires that the species in the thermodynamic and transport models be listed in the *.chm* file in the same order, and that if variable third body efficiency is used in the finite rate model, that the third bodies also match this order. If CHMGR detects that the same set of species exist in the model being defined, but in a different order, the transport and finite rate models are reordered to match the thermodynamic data. If completely different species are defined, however, the model cannot be fixed automatically, and a *.chm* file cannot be output.

Similarly, the temperature intervals defined for all species in the transport model must be the same. Otherwise, the *.chm* file cannot be generated because the problem cannot be fixed automatically. The user needs to redefine the transport data such that the temperature intervals are the same for all species.

Thus, several messages may be generated letting the user know whether the model is consistent. If it is, the user may continue with output of the final Wind-US file by selecting the "Write Wind .chm File" button. This will actually cause the consistency check to be run again in case the user neglected to do so, or attempts to output a file even if the model is inconsistent. If all consistency tests are passed, the user will be prompted for a file name as in Figure 23, and the final Wind-US *.chm* file will be written.

The "Exit Chemistry Manager" button on the main GUI ends program execution. A confirmation window is presented in case the user accidentally selects the "Exit" button. A "Yes" response to the confirmation request terminates the program.

4.8.7 Text Menu Structure and Commands

`ex` **Exit CHMGR**
`rs` **Run A Script File**
`io` **WIND Model Input/Output**
- `q` Quit This Menu
- `cl` Clear Existing Model
- `iw` Load Wind .chm File
- `th` Set Thermo Section Titles

- `tf` Set Finite Rate Section Titles
- `tr` Set Transport Section Titles
- `st` Set the Minimum Reaction Temperature
- `mc` Make Model Consistent
- `ow` Write Wind .chm File

`th` **Thermodynamic Data Manager**

- `q` Quit This Menu
- `il` Select File for Input in GRC Format
- `iw` Select File for Input in WIND Format
- `lf` List Species Available in File
- `as` Add Species to Model
- `ds` Delete Species from Model
- `ls` List Species in Model
- `rs` Reorder Species in Model
- `ow` Output Thermo Model in WIND Format

`cp` **Cp Curve Fit Generation**

- `q` Quit This Menu
- `rd` Read Data File
- `uc` Set The Units Conversion Factor
- `sn` Enter The Species Name & Components
- `mw` Enter The Molecular Weight & Hfo
- `in` Prompt For Cp Data
- `gc` Generate Curve Fit
- `sp` Select Species to Plot
- `pi` Set Plot Interval & Increment
- `op` Output Test Plot

`tr` **Transport Data Manager**

- `q` Quit This Menu
- `il` Select File for Input in GRC Format
- `iw` Select File for Input in WIND Format
- `lf` List Species Available in File
- `as` Add Species to Model
- `ds` Delete Species from Model
- `ls` List Species in Model
- `rs` Reorder Species in Model
- `ol` Output Thermo Model in GRC Format
- `ow` Output Thermo Model in WIND Format

`fr` **Finite Rate Data Manager**

- `q` Quit This Menu
- `iw` Select WIND File for Input
- `ig` Select Generalized File for Input
- `la` List Reactions in File/Add to Model
- `dr` Delete Reactions from Model
- `rr` Reorder Reactions in Model
- `ow` Output Reaction Model in WIND Format

4.9 timplt

The timplt utility is now obsolete and no longer being maintained.

timplt may be used to extract information stored in a time history (*.cth*) file, and create a GENPLOT file for post-processing. (See the *CFPOST User's Guide* for a description of the format of GENPLOT files.) Time history files are created during a Wind-US run by using the `HISTORY` keyword This is useful in tracking the values of certain specified parameters over time in an unsteady flow.

The timplt utility is used with .cth files created using Wind versions prior to alpha 5.52. It will not work with .cth files created using newer versions of Wind; for those, use the new thplt utility.

Example

Suppose the `HISTORY` keyword has been used in the form

```
History variable pressure frequency 5
   History 1 17 17  1  1 1 1
   History 2 17 17 11 11 5 5
   History 3  9  9 11 11 7 7
```

to create a time history file named *pressure.cth* containing the static pressure at the points (17,1,1), (17,11,5), and (9,11,7) in zones 1, 2, and 3, respectively. The *timplt* utility would be used as follows to create a GENPLOT file containing the unsteady pressure data. Lines in slanted type are typed by the user.

```
% timplt
 Enter name of time history file: pressure.cth


          1  Select grid point
          2  Select grid range
          3  Select variable
          6  Open another time history file
          7  End

 Enter menu selection: 1

Enter zone number of point to track: 1
Enter I, J, K of point to track: 17,1,1

 Currently selected point = (17,1,1)

          1  Select grid point
          2  Select grid range
          3  Select variable
          6  Open another time history file
          7  End

 Enter menu selection: 3
     1  Mach number
     2  Static pressure
     3  Static temperature
```

```
     4  U velocity
     5  V velocity
     6  W velocity

  You may select up to            5 variables.

 Enter selection: 2

 Currently selected point = (17,1,1)
 Currently selected variables are
Static pressure

          1  Select grid point
          2  Select grid range
          3  Select variable
          4  Create GENPLOT file
          5  Analyze data
          6  Open another time history file
          7  End

 Enter menu selection: 4
Enter name of genplot file: press1.gen
 Reading data...
 complete.

 Currently selected point = (17,1,1)
 Currently selected variables are
Static pressure

          1  Select grid point
          2  Select grid range
          3  Select variable
          4  Create GENPLOT file
          5  Analyze data
          6  Open another time history file
          7  End
```

[*Repeat "Select grid point," "Select variable," and "Create GENPLOT file" steps for the points at (17,11,5) and (9,11,7)*]

```
  Enter menu selection: 7
```

4.10 usplit-hybrid

> *The usplit-hybrid utility is obsolete and no longer being maintained. The cfpart utility is recommended for partitioning an unstructured grid.*

usplit-hybrid is used to partition a single-zone unstructured grid into multiple zones for parallel processing. The user is prompted for the name of the input common grid (*.cgd*) file containing the single-zone unstructured grid, the name to be used for the new *.cgd* file containing the multi-zone unstructured grid, and the desired number of zones. The file names are limited to a maximum of 32 characters.

In the output multi-zone grid, each zone will have (roughly) the same number of cells. Thus for maximum efficiency, the number of zones should match the number of available processors (except for one processor left free for use as the master), and each processor should be equal in computing power.

Boundary conditions and surfaces defined in the original file are preserved in the partitioned file. Additional "crinkly" surfaces will appear at the boundaries between the newly-created zones, with negative surface ID numbers. The appropriate coupled boundary conditions are automatically set for these zonal boundaries.

Example

Figure 28 shows the single-zone unstructured grid used for the M2129 s-duct tutorial, available at the NPARC Alliance CFD Verification and Validation web site.

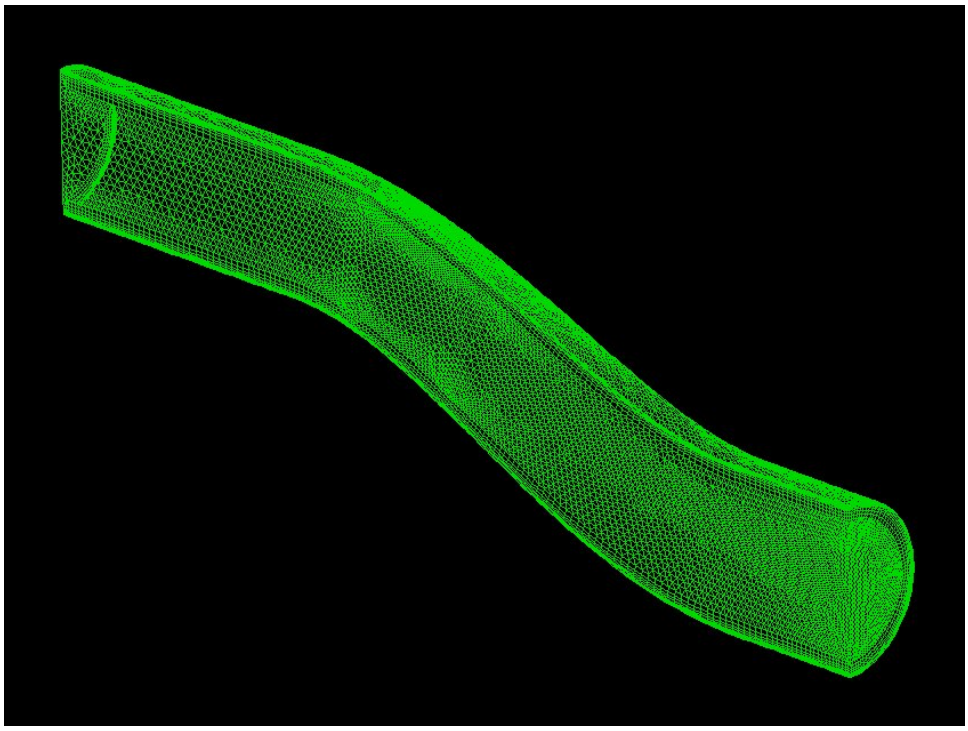

Figure 28: Single-zone grid for M2129 s-duct

The following terminal session shows how *usplit-hybrid* would be used to split this grid into five

zones, suitable for parallel processing on five CPUs. Input data specified by the user is in slanted type.

```
% usplit-hybrid

                          ***** usplit-hybrid *****

          Select the desired version from the following list.

0) END
1) usplit-hybrid

Single program automatically selected.

Enter single zone unstructured grid common file ....... : M2129_1zone.cgd
Enter name for partitioned common file (< 32 characters) ....... : M2129_5zone.cgd
Enter number of partitions ....... : 5
**********************************************************************
      usplit-hybrid   - Version 2.6 (Last changed 2007/09/25 13:19:00)
**********************************************************************

...
[Many lines describing usplit-hybrid progress and zonal properties]
...

 ******************************************
successfully wrote preprocessed data in file 5zone.cgd
 ******************************************

...
```

The resulting five-zone grid is shown in Figure 29.

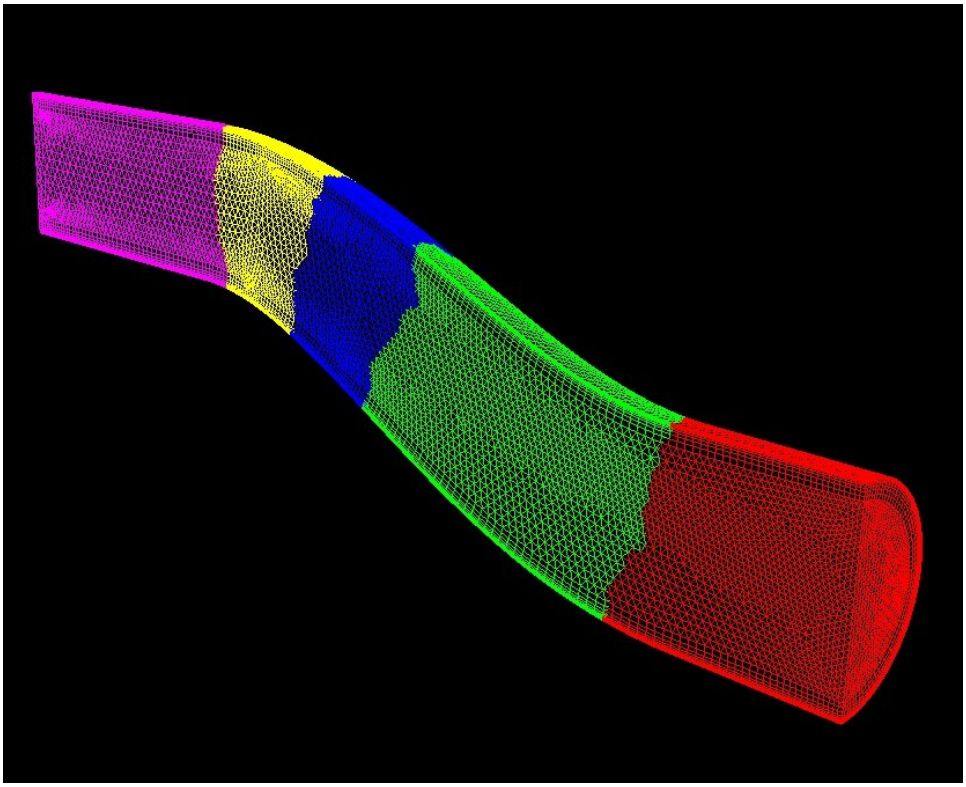

Figure 29: Five-zone grid for M2129 s-duct

References

Gordon, S., and McBride, B. J.. (1976) "Computer Program for Calculation of Complex Chemical Equilibrium Composition, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouguet Detonations," NASA SP-273.

Gordon, S. (1982) "Thermodynamic and Transport Combustion Properties of Hydrocarbons with Air, I-Properties in SI Units," NASA TP-1906.