MCAT Institute
Annual Report 92-008

USER INTERFACE USER'S GUIDE FOR HYPGEN

Dr. Ing-Tsau Chiu

April 1992

NCC2-692

MCAT Institute
3933 Blue Gum Drive
San Jose, CA 95127

User Interface User's Guide for HYPGEN

Ing-Tsau Chiu

Abstract

The user interface (**UI**) of **HYPGEN** (Chan and Steger, 1991) is developed using Panel Library (Tristram et al., 1990) to shorten the learning curve for new users and provide easier ways to run **HYPGEN** for casual users as well as for advanced users. Menus, buttons, sliders and type-in fields are used extensively in **UI** to allow users to point and click with a mouse to choose various available options or to change values of parameters. On-line help is provided to give users information on using **UI** without consulting the manual. Default values are set for most parameters and boundary conditions are determined by **UI** to further reduce the effort needed to run **HYPGEN**; however, users are free to make any changes and save it in a file for later use. A hook to **PLOT3D** (Buning and Steger, 1985) is built-in to allow graphics manipulation. The viewpoint and min/max box for **PLOT3D** windows are computed by **UI** and saved in a **PLOT3D** journal file. This makes it easy for users to view their grids. For large grids which takes a long time to generate on workstations, the grid generator (i.e. **HYPGEN**) can be run on fast computers such as Cray, while **UI** stays at workstations. Extensive warning messages, such as incompatible boundary condition specifications, bad grid cells, etc., are used to give users clues about what is happening in their grids. In case of bad cells, the **PLOT3D** journal file created by **UI** contains **PLOT3D** commands to display locations of bad cells. Other features like displaying minimum and maximum stretching ratio of grid spacings and initial and end spacings before running **HYPGEN** are aimed at shortening the iterative process in obtaining grids. In short, the objective of **UI** is to make grid generation using **HYPGEN** an easier and faster process.

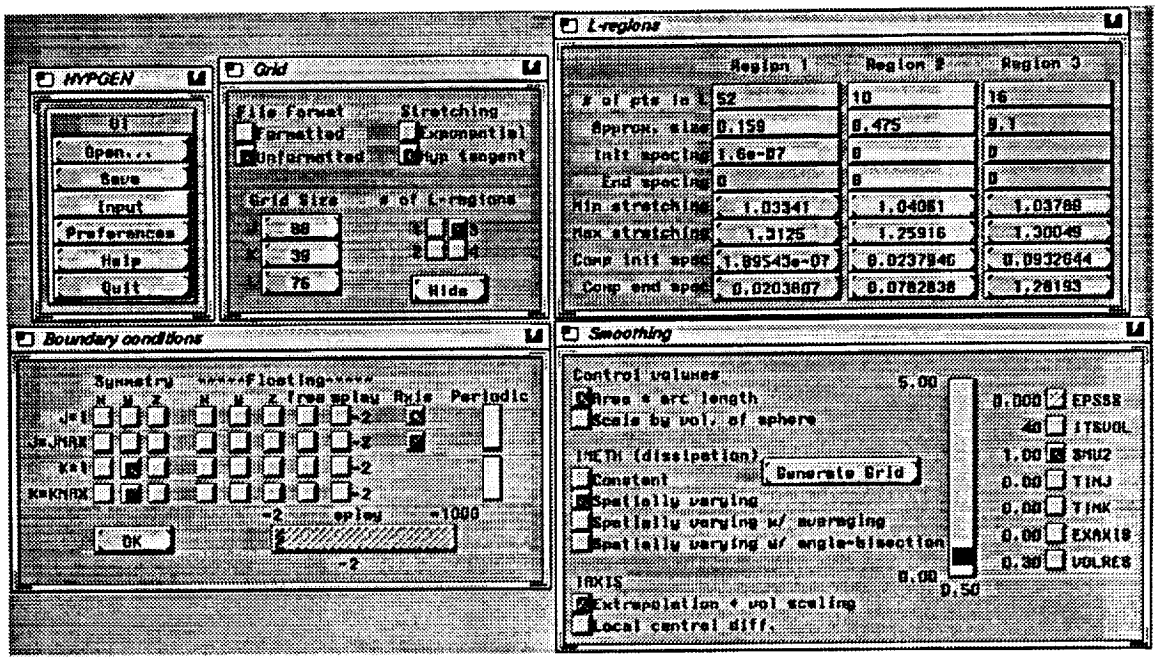


Figure 1: User interface for HYPGEN

UI User's Guide

Version 1.2

Ing-Tsau Chiu
MCAT Institute
Mail Stop T045-2
NASA Ames Research Center
Moffett Field, CA 94035
Mar. 31, 1992

About UI

UI (stands for user interface) is a graphics front end for the hyperbolic grid generator (HYPGEN) developed by Chan and Steger (1991). It provides menus, buttons, sliders and typein fields (as shown in Fig. 1) for users to enter parameters needed to run HYPGEN. A hook to PLOT3D is built-in to allow viewing of 2-D surface or 3-D volume grids; however, users still need to type in filename of the PLOT3D "com" file written by UI to display grids. Current version of UI allows

hyperbolic grid generator to run on a remote machine through the use of shell scripts on unix operating systems. The volume grid generated is then copied back to local machine for graphics manipulation. Following is a brief introduction of UI.

Installation

The installation instruction below is particular to Iris workstation and Cray YMP combination. However, the installation procedure for Cray YMP can be applied to any remote computer at your site as long as programs are available to convert Iris unformatted to/from unformatted files for your remote computer.

- On Iris workstation:
    1. `zcat ui.tar.Z | tar xfvo -`
    2. `cd ui/bin` (change directory to ui/bin)
    3. `cp hypgen.4d ui your_bin_directory`
    4. Compile hypgen (source code in `ui/hypgen`) and move the executable to the same directory as `ui` and `hypgen.4d`.
    5. Make sure that `hypgen.4d`, `ui` and `hypgen` are in a directory specified in the environment variable "`path`" for csh or "`PATH`" for sh.
- On Cray YMP (skip this part if you just want to generate grids on Iris workstation)
    1. Compile `hypgen` (source code in `ui/hypgen`).
    2. Move `hypgen` and `hypgen.ymp` (in `ui/bin`) to a directory along the command searching path specified by the environment variable "`path`" for csh or "`PATH`" for sh.
    3. Check if `4d2cray` and `4Dconv` (programs used by shell script `hypgen.ymp` to convert Iris unformatted to/from Cray unformatted files) are available or in a directory specified in the environment variable "`path`" for csh or "`PATH`" for sh. If you plan to run `hypgen` on a different remote computer, replace `4d2cray` and `4Dconv` with your own programs that do the file format conversion between your workstation and your remote computer and modify the shell script file, `hypgen.ymp`, to reflect this change.
    4. For csh users, add/modify "`path`" environment variable in .cshrc to include the directory where you put `hypgen.ymp` and `hypgen`. Usually, "`path`" is set in .login which is not "sourced" by remote shell (rsh). Thus "path" has to be set in .cshrc; otherwise, the absolute path, e.g.

```
/u/ne/chiu/bin/hypgen.ymp,
```

has to be used instead of `hypgen.ymp`. Following shows an example of the syntax for setting "path" in .cshrc.

```
set path =( . ~/bin /usr/bin /bin /usr/local/bin \
/usr/unsupported/bin )
```

For sh users, add/modify "PATH" environment variable in .profile accordingly. The syntax is like

```
PATH=.:$HOME/bin:/usr/bin:/bin:/usr/local/bin:
```

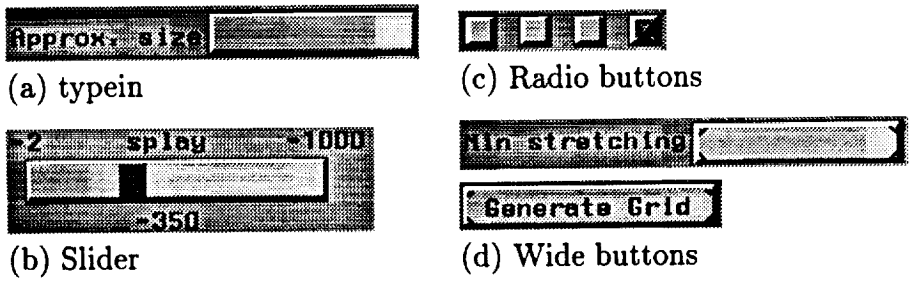
5. Edit .rhosts file on your workstation and the remote computer to include entries for both your workstation and the remote computer. The syntax of the entry in .rhosts file is best explained by an example:

```
wk211.nas.nasa.gov
```

If you set up UI following the procedures above and still encounter problems in running HYPGEN on a remote machine, read the "Frequently Asked Questions" section at the end of this manual. It might contain answers to your problems.

Changes

- Version 1.2
  1. Made UI compatible with HYPGEN v.1.2.
  2. Get rid of format conversion (plane format to whole format) as HYPGEN v.1.2 now generates whole format grids.
- Version 1.1
  1. Correctly sets the boundary conditions for 2-D grid generation.
  2. Uses dynamic memory allocation.
  3. Allows basename to be optionally specified at startup.
  4. The grid generated by HYPGEN is converted from PLOT3D plane format to whole format.
  5. User's `.login`, in addition to `.cshrc`, is also checked for environment settings for the default window shell to launch PLOT3D.
  6. Uses only subsets of the generated grid for tetrahedron decomposition cell volume check to avoid unnecessary calculations when bad cells are found.



(a) typein (c) Radio buttons

(b) Slider (d) Wide buttons

Figure 2: Objects of user interface

7. Outputs proper PLOT3D script file for 2-D and 3-D grids.
8. Expanded instructions on setting up UI and HYPGEN to run on different machines.
9. Added 2-D examples.
10. Fixed a bug in reading formatted grid files.

- Version 1.0
  1. The default hyperbolic grid generator is changed from **hyg3d** to **hypgen**.
  2. File format of the input surface grid is determined by UI.
  3. Allows hyperbolic grid generator and UI to run on different computers.
  4. Added preferences panel to allow customization of UI environment.
  5. Warns about bad grids.
  6. Fixed a bug in writing PLOT3D script file for formatted grid files.

How to start UI

Two ways can be used to start UI:

- `ui`: start UI with the "*Filename*" panel.
- `ui basename`: bypass "*Filename*" panel at startup.

Mouse, typein, slider, button and menu

Using the mouse Only the left mouse button is used with UI to operate on menus, buttons, sliders, and typeins. The following explains the terms used for mouse operations in this manual.

Click Press the left mouse button.

Drag Move the mouse while holding down the left mouse button.

Typein Accepts input from users. Looks similar to wide buttons used in UI, except that the wide buttons have beveled edges. In UI, to make changes take effect immediately for typein fields, the changes have to be entered followed by a carriage return.

Slider The current value of the slider is controlled and reflected by the position of a slider bar within a bounding rectangular region. In UI, the min/max values of a slider are shown along the side of the slider. To change the value of the slider, click within the rectangular region. The slider bar jumps to the mouse location and then follows subsequent mouse motion until the mouse button is released. One can also click on the slider bar then drag it to change the value of the slider. To enter slider fine control mode, press the control key on the keyboard while holding down the left mouse button. This is useful when the difference of the min/max values is large.

Radio button Usually several radio buttons form a group within which they interact with each other. When a radio button is selected (i.e. clicked with left mouse button), the rest of the radio buttons in the same group are deselected.

Wide button In UI, wide button when clicked is used to perform function as labeled on the button or to display related information. Wide button will stay highlighted through out the entire period of the action requested.

Menu Offers ways to access different parts of UI. Submenus become active by dragging (i.e. moving with left mouse button down) the mouse over them.

Default file extension

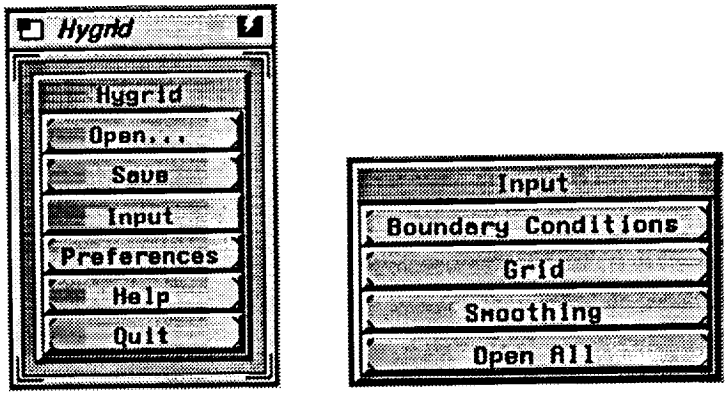
.i : HYPGEN input file

.out : HYPGEN output file

2d.dat : 2-D surface grid

3d.dat : 3-D surface grid

.com : PLOT3D com file



(a) Main menus (b) Input submenus

Figure 3: Menus

Main menus

Main menu provides entry points to different parts of UI.

Open : Pops *"Filename"* panel for file I/O.

Save : Saves HYPGEN input data to a file.

Input : Provides submenus for choosing panels to write input data.

Preferences : Pops preferences panel to allow choice of hyperbolic grid generator or shell script to run hyperbolic grid generator and customizations of PLOT3D popup window and the range of allowable values for smoothing parameters.

Help : Provides help information on menus.

Quit : Exit UI.

When clicking on *"Open"* or *"Save"* menus mentioned above, the *"Filename"* panel as shown in Fig. 4 pops front providing typein fields to enter filenames for files needed to run HYPGEN and UI. The *"Help"* button in the panel gives help information about *"Filename"* panel, the *"Cancel"* button discards whatever filenames entered, and the *"OK"* button accepts filenames entered, and does certain functions depending on whether *"Open"* or *"Save"* menu was clicked previously. When *"Save"* menu was clicked previously, the HYPGEN input data is simply saved to a file specified in *"Input File"* typein field. Whereas in the case of *"Open"* menu clicked previously, several functions will be performed. The min/max for coordinates of the 2-D surface grid will be computed and saved to a PLOT3D com file and PLOT3D

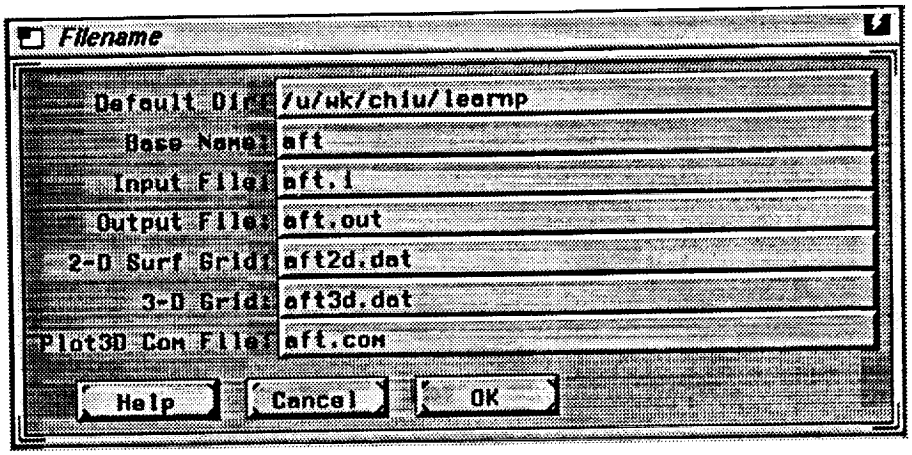


Figure 4: Filename panel

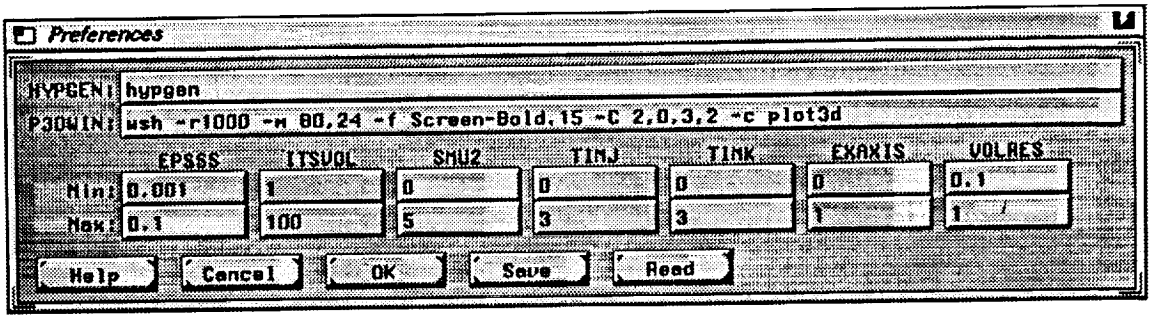


Figure 5: Preferences panel

will be launched in a separate window if it is not already running. Then the panels for entering/modifying HYPGEN input data pops up in place of the *"Filename"* panel if they haven't been launched. To view the 2-D surface grid, just type the filename entered in *"Plot3D Com File"* typein field in the PLOT3D window. The *"Default Dir"* typein field shows the current working directory and can be changed. However, if a directory specified can not be found, the directory is not changed and a warning panel showing the error message will pop up. Whenever a file can not be found by UI, you'll get similar warning message. The *"Base Name"* typein field allows easy entry for filenames of various files needed to run HYPGEN and UI. As shown in Fig. 4, UI appends proper file extensions to whatever is entered in *"Base Name"* typein field. However, further modification on individual filename is possible by manually changing whatever is shown in any of the typein fields.

Preferences

"Preferences" panel (as shown in Fig. 5) provides typein fields for dynamically changing the hyperbolic grid generator used in generating volume grids and customization of the window shell to launch PLOT3D. There are also buttons for saving and reading preferences to/from a file, .uirc, in the current working directory. At startup, UI searches for .uirc in the current working directory first and if .uirc cannot be found there, it then looks into user's home directory for the file. If there is no .uirc in the current working directory and user's home directory, UI then checks the environment variables, HYPGEN and P3DWIN, for the preference settings. If none of them have been set, `hypgen` and `wsh -r1000 -m 80,24 -f Screen-Bold.15 -C 2,0,3,2 -c csh -c "source ~/.login; plot3d"` are used as the default hyperbolic grid generator and window shell by UI. Following shows the syntax for setting the environment variables, HYPGEN and P3DWIN. The P3DWIN shown below uses xterm as the window shell for launching PLOT3D with courier-bold.14 as the character font, black as the background and green the foreground.

```
setenv HYPGEN hypgen
setenv P3DWIN "xterm -fn courier-bold.14 -fg green -bg black -e plot3d"
```

Following shows what is in a typical .uirc file; the first line specifies the filename of the hyperbolic grid generator, and the second line the user's window shell. If .uirc contains only one line (i.e. the window shell is not specified), the default window shell will be used to launch PLOT3D.

```
hypgen
wsh -r1000 -m 80,24 -f Screen-Bold.15 -C 2,0,3,2 -c csh -c "source
~/.login; plot3d"
```

The environment variable, HYPGEN, can also be set to a shell script with all the necessary arguments (see the example below) to run the hyperbolic grid generator at a remote machine (usually a faster one). The "Installation" section gives the details for setting this up.

```
setenv HYPGEN "hypgen.4d reynolds hypgen.ymp /scr8/chiu/geom"
```

Similarly, the first line of .uirc file can also be changed accordingly to run a shell script.

The *"Min/Max"* typein fields are provided to set minimum and maximum allowable values for smoothing parameters in *"Smoothing"* panel. The *"Help"* button

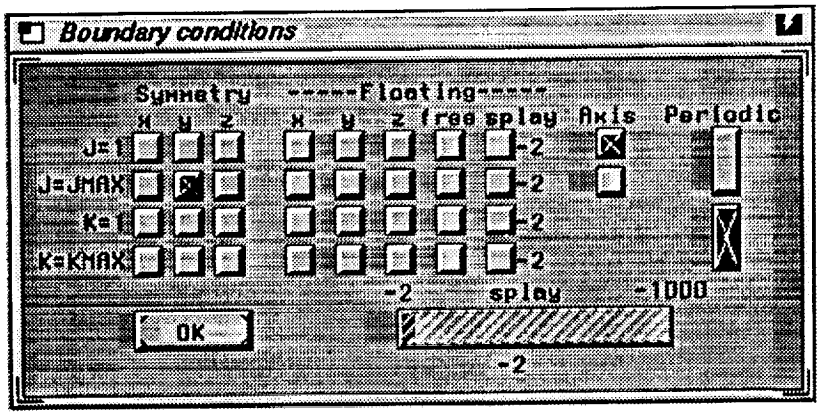


Figure 6: Boundary conditions panel

underneath the *"Min/Max"* typein fields provides relevant information about customization of user preferences, the *"Cancel"* button discards all changes made not followed by a carriage return, and the *"OK"* button accepts all changes shown. The *"Save"* button saves preferences in `.uirc` file in the current working directory, and the *"Read"* button reads in preferences from `.uirc` in the current working directory or user's home directory.

Input submenus

Input submenus can be accessed by first clicking on *"Input"* menu from the main menu and then dragging over the desired selection.

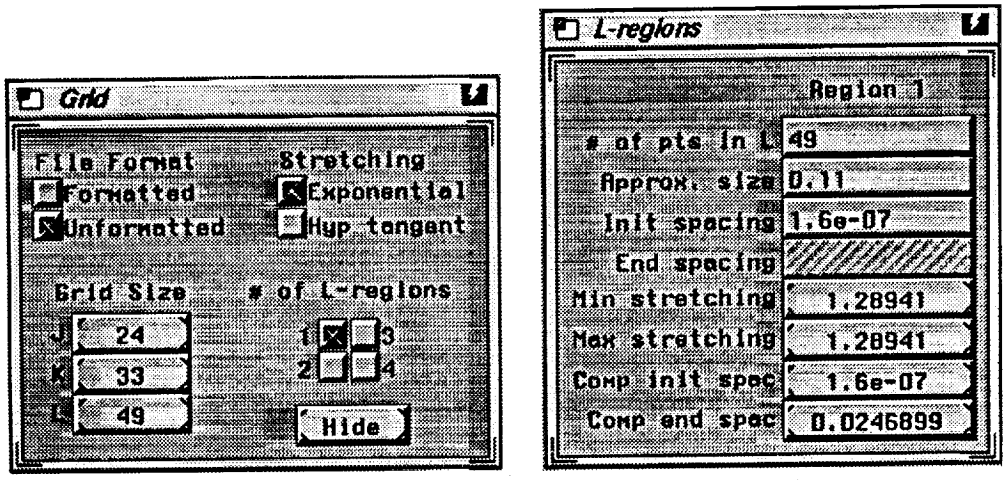
Boundary Conditions : Pops *"Boundary conditions"* panel (see Fig. 6) for entering boundary conditions.

Grid : Provides two panels (as shown in Fig. 7) for entering grid related information.

Smoothing : Provides a panel (see Fig. 8) for setting smoothing parameters.

Boundary conditions panel

Four different types of boundary conditions, symmetry, floating, axis and periodic, are provided by HYPGEN. They are determined by UI from the specified 2-D surface grid; however, they can be changed by clicking on the button for the specific boundary condition. The *"OK"* button hides the *"Boundary conditions"* panel when clicked. It can be made visible again by clicking the *"Input"* submenu in the main menu and drag over the *"Boundary conditions"* button. The slider (for setting values



(a) Grid panel (b) L-regions panel

Figure 7:

of splay boundary condition) shown with hash pattern in Fig. 6 means that it is not selectable and will become active only when splay boundary condition is chosen.

Grid panels

Fig. 7 shows the two panels - *"Grid"* and *"L-regions"*, for entering grid related information for HYPGEN. In *"Grid"* panel, file format for the 2-D surface grid, method of stretching, and number of regions in ζ direction can be set by clicking on the corresponding button. In the current version, the file format is determined by UI; thus, it is not necessary for user to enter the correct file format. The *"Grid Size"* buttons (in Fig. 7a) are used to show the grid size only and can not be altered like typein field. The *"Min/Max stretching"* and *"Computed init/end spacing"* buttons in Fig. 7b are also for showing the corresponding information only. In *"L-regions"* panel, four typein fields are provided to enter grid related information. Any changes to any of the typein fields will affect the min/max stretching ratios and the computed initial/end spacings for all regions in ζ direction. The *"End spacing"* typein field in Fig. 7b is shown with hash pattern means that the typein field is not selectable since exponential stretching is chosen (see Fig. 7a) and end spacing is not required. The *"End spacing"* typein field will become active when hyperbolic tangent is chosen as the method of stretching.

Smoothing panel

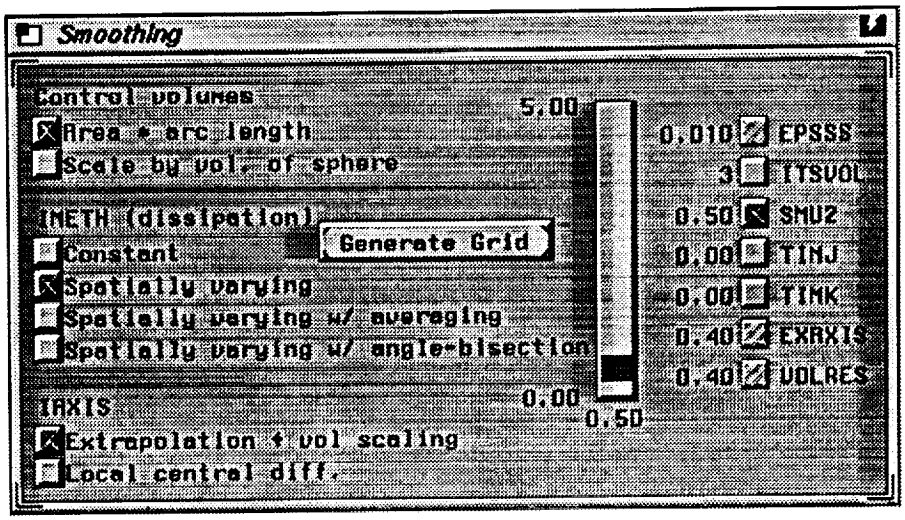


Figure 8: Smoothing panel

The *"Smoothing"* panel (see Fig. 8) provides buttons and sliders for choosing options and setting values for smoothing parameters. For explanations of the available options and smoothing parameters, the user is referred to the documentation of HYPGEN. To set values of each parameter, say SMU2, click first on *"SMU2"* button, then drag the slider bar up or down till the value shown at the bottom of the slider meets your needs, then release the mouse button; the label of *"SMU2"* button will show the value just set. The *"Generate Grid"* button spawns a shell to run the hyperbolic grid generator specified in *"Preferences"* panel.

Running grid generator at a remote machine

A pair of shell scripts, hypgen.4d and hypgen.ymp, that allows running hyperbolic grid generator on Cray YMP or Cray 2 running Cray COS 6.0 and UI on an Iris-4D/VGX running IRIX 3.3 were tested. However, the shell script, hypgen.ymp, can be modified to run on any machines running unix operating systems. Listings of hypgen.4d and hypgen.ymp are included at the end of this user guide. Users are advised to read the comment statements before making modifications. For details of setting UI to run HYPGEN on Cray YMP or Cray 2, please read the "Installation" section at the beginning of this user guide.

UI provides several ways to set up the hyperbolic grid generator to run on a remote computer. Following shows all the possible methods; the users are free to pick whatever is convenient to them. In the example below, **hypgen.4d** is the script to be run on Iris workstation, "**reynolds**" is the name the remote computer,

`hypgen.ymp` is the script to be run on the remote machine (in this case, `reynolds`) and `/scr8/chiu/geom` is the directory on the remote machine that HYPGEN looks for input/output files.

- Method 1: Set the environment variable, HYPGEN; e.g.

```
setenv HYPGEN "hypgen.4d reynolds hypgen.ymp /scr8/chiu/geom"
```

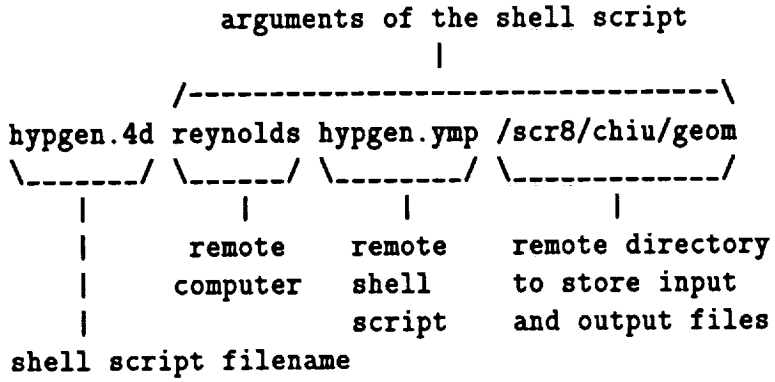
- Method 2: Put what you would set for the environment variable, HYPGEN, to the first line of .uirc; e.g.

```
hypgen.4d reynolds hypgen.ymp /scr8/chiu/geom
```

- Method 3: Choose *"Preferences"* from the main menu of UI and type in what you would give to the environment variable, HYPGEN, in the *"HYPGEN"* typein field. Then click on *"OK"* button to confirm the setting. Optionally, you can save it to .uirc in the current working directory for later use.
- Method 4: Choose *"Preferences"* from the main menu of UI and click on *"Read"* button to read from a previously saved .uirc file in the current directory.

Combinations of the above methods can be used; for example, the users can set the environment variable, HYPGEN, before running UI and later read from a .uirc file by clicking on the *"Read"* button in the *"Preferences"* panel or type in the new preference in the *"HYPGEN"* typein field.

The examples shown in Methods 1 and 2 are particular to the shell scripts, hypgen.4d and hypgen.ymp, provided along with the UI source code. If you use your own shell scripts, the change has to be made accordingly. Following explains again the meaning of each term in the above example:



Getting UI

The UI source code and documentation can be found in `reynolds:~chiu/ui`. The program was only tested on Iris-4D/VGX. To compile UI, "Panel Library" is needed. To obtain the Panel Library, send e-mail to

```
panel-request@nas.nasa.gov
```

Or send regular mail to:

```
NAS Applied Research Office
ATTN: PANEL LIBRARY REQUEST
M/S T045-1
NASA Ames Research Center
Moffett Field, CA 94035
```

Reference

Buning, P. G. and Steger, J. L. (1985). "Graphics and Flow Visualization in Computational Fluid Dynamics." AIAA Paper 85-1507.

Chan, W. M. and Steger, J. L. (1991). "A Generalized Scheme for Three-Dimensional Hyperbolic Grid Generation." AIAA Paper 91-1588.

Tristram, D. A., Walatka, P. P. and Raible, E. L. (1990). "Panel Library Programmer's Manual." NASA ARC Report RNR-90-006.

If you have any questions, suggestions, bug reports or comments, please contact

```
Ing-Tsau Chiu
MCAT Institute
Mail Stop T045-2
NASA Ames Research Center
Moffett Field, CA 94035
(415)604-3857

E-mail address:
chiu@nas.nasa.gov
chiu@wk211.nas.nasa.gov
```

Frequently Asked Questions

Q. I had set the environment variable "path" to include the directory where hypgen and hypgen.ymp are located in my .cshrc, UI still won't run hypgen and hypgen.ymp remotely on the remote machine.

A. You probably had the following statement in your .cshrc preceding the statements that set the path,

```
if (! $?prompt) exit
```

The above statement would cause rsh to exit immediately w/o proceeding to set the path for rsh. Thus, the statements that set the path in .cshrc have to be moved before the above statement. One way to check if your rsh has the proper path set to run hypgen remotely is to run the following from your Iris workstation,

```
rsh your_remote_machine env          (AT&T SYSV unix)
```

or

```
rsh your_remote_machine printenv     (BSD unix)
```

and check the "PATH" variable returned from the above command.

Q. I had setup UI properly. My rsh returned the right path from the remote machine and UI still won't run hypgen.ymp and hypgen remotely.

A. It is possible that your remote machine does not know the existence of your workstation. Try the following command from your workstation to see if this is the case.

```
rsh your_remote_machine rsh your_workstation
```

or if your remote machine is a Cray running UNICOS

```
rsh your_remote_machine remsh your_workstation
```

If you get a message like "Unknown host", then the remote machine does not know the existence of your workstation. Assign the "`Local_Domain`" variable in hypgen.4d with the domain name of your workstation. The following example shows what the hostname and the local domain name mean in hypgen.4d:

```
wk211.nas.nasa.gov
```

hostname is "`wk211`" and local domain name is "`.nas.nasa.gov`".

Usually, if your workstation is not on the same domain as the remote machine, e.g. eagle and NAS workstations, then you need to edit the "`Local_Domain`" variable in hypgen.4d.

Q. My path set up on the remote machine is O.K. and the remote machine communicates properly with my workstation, but UI still does not run hypgen and hypgen.ymp properly on the remote machine.

A. Check the dimensions in hypgen. You might have large dimensions declared in hypgen that requires huge run time memory which exceeds the memory limit for interactive jobs on the remote machine.

Shell script hypgen.4d

```
#!/bin/sh
# To use this with UI, set the environment variable HYPGEN like
# the example shown below:
#     setenv HYPGEN "hypgen.4d reynolds hypgen.ymp /scr8/chiu/geom"
#                    \_______/ \_________________________________/
#                        |                     |
#                        |        arguments of the shell script
#                        |
#                   filename of this shell script
# or put
#        hypgen.4d reynolds hypgen.ymp /scr8/chiu/geom
# in the HYPGEN typein field.  You can optionally save this to .uirc
# in the current working directory for later use.
#
#   Note:  Absolute path has to be given for hypgen.ymp, e.g.
#               /u/ne/chiu/bin/hypgen.ymp
#          if path is not set in .cshrc on reynolds.
#
#   Note:  Only the first three arguments are needed for this
#          shell script since the BaseName will be given by the
#          user interface and LocalHost and LocalDir will be
#          determined by the programs hostname and pwd respectively.
#
# RemoteHost: the remote machine where the hyperbolic grid generator
#             is run.
# HypgenSh:   the shell script run at the remote machine that invokes
#             rcp to transfer files and hyperbolic grid generator to
#             generate volume grids.
# HygDir:     where hyperbolic grid generator looks for input, 2-D
#             surface grid files and saves output and volume grid
#             files at the remote machine.  If HygDir does not exist
#             on the remote machine, it will be created (see
#             hypgen.ymp for details).
# BaseName:   basename used to assign filenames for the files needed
#             to run hyperbolic grid generator and UI (Note: BaseName
#             will be given by the user interface).
# Local_Domain: domain name of the local machine (usually not required
#               if remote host is able to resolve the hostname of the
```

```
#                 local machine)
# LocalHost:  local machine where you run UI (determined by hostname).
# LocalDir:   local directory where you store the input data file and
#             2-D surface grid for the hyperbolic grid generator
#             (determined by pwd).
#
RemoteHost=$1
HypgenSh=$2
HygDir=$3
BaseName=$4
#Local_Domain=".nas.nasa.gov"
Local_Domain=""
LocalHost=`hostname`$Local_Domain
LocalDir=`pwd`
rsh $RemoteHost $HypgenSh $HygDir $BaseName $LocalHost $LocalDir
```

Shell script hypgen.ymp

```
#!/bin/sh
# HygDir: where the 2-D surface grid is
# BaseName: basename used to assign filenames for the files needed
#           to run hyperbolic grid generator and UI.
# RemoteHost: remote machine on which the user interface is run.
# RemoteDir:  remote directory from which the input data and 2-D
#             surface grid are to be taken and to which the volume
#             grid will be sent.
# Grdgen:     name of the hyperbolic grid generator (need to be
#             changed if it is not hypgen).
#
HygDir=$1
BaseName=$2
RemoteHost=$3
RemoteDir=$4
Grdgen="hypgen"
#
if test ! -d $HygDir
then
   echo "Directory "$HygDir" does not exist"
   echo "Creating directory... "$HygDir
   mkdir $HygDir
else
   echo "Found "$HygDir
fi
cd $HygDir
#
#  rcp 2-D surface grid and hyperbolic grid generator input file from
#  workstation
#
rcp $RemoteHost":"$RemoteDir"/"$BaseName"{2d.dat,.i}" .
#
#  Delete surf.i if it exists
#
if test -f surf.i
then
   echo surf.i exists and is deleted
   /bin/rm surf.i
```

```
fi
#
#  Convert Iris-4D PLOT3D file to Cray YMP format
#  Change the following two lines if you use your own format
#  conversion program.
#
echo "Converting 2-D surface grid to Cray YMP format"
4d2cray -ux $BaseName"2d.dat" surf.i
#
#  Run hyperbolic grid generator to generate volume grid
#
if test -f $BaseName".out"
then
   echo $BaseName".out exists and is deleted"
   /bin/rm $BaseName".out"
fi
echo "Generating grid..."
time $Grdgen < $BaseName".i" > $BaseName".out"
echo "Grid generated"
#
if test -f $BaseName"3d.dat"
then
   echo $BaseName"3d.dat exists and is deleted"
   /bin/rm $BaseName"3d.dat"
fi
mv plot3d.dat $BaseName"3d.dat"
#
#  Convert Cray YMP PLOT3D file to Iris-4D format
#  Change the following two lines if you use your own format
#  conversion program.
#
echo "Converting volume grid to Iris-4D format"
4Dconv `echo $BaseName"3d.dat"` plot3d.dat
#
# Transfer volume grid back to workstation
#
echo rcp plot3d.dat $BaseName".out" to $RemoteHost
rcp plot3d.dat $RemoteHost":"$RemoteDir
rcp $BaseName".out" $RemoteHost":"$RemoteDir
```