

GFM 4.0 Tips and Special Procedures

The information in this document is provided on diverse topics to help users effectively apply the Glass Furnace Model and resolve or avoid problems when using it. It includes procedures for calibrating the soot model and manually adjusting relaxation factors to improve simulation convergence or achieve convergence for numerically difficult problems.

1. Tips and Things to Know about GFM

Known Problems with Work-Arounds

- Case folders will not be automatically deleted. The *Delete Case* menu item will delete all the files in the case folder. However, an unresolved problem in Windows prevents the case folders from being deleted. GFM must be exited before a case folder can be manually deleted.
- The simulation run status displayed in the green boxes is now very limited. Some of the variables that were displayed before are no longer used or may have changed due to the major upgrades that have been done to the CFD code, especially for radiation. This interface with the GUI has not been reworked. The new RunPlot program is an excellent tool that can be used to monitor the simulation progress. The command prompt window also provides simulation progress information.

General Information

- Case folders and files must be in the proper directories.
- Case folders and files must be named properly. Renaming case folders using long file names with keywords can make saved cases easy to identify, however, to reprocess those cases in GFM, the case folders must be renamed with their original name.
- Files should not be modified manually. (An exception exists for the relaxfactor[m/c].txt files as explained below in the Adjust Relaxation Factors Procedure.)
- Some actions cause the furnace figure or grid plot to be moved or resized. Use the green box diagram handles to resize or reposition the diagram. The lower left corner handle is for moving the diagram. The upper right corner (where the 3 grid slices meet) handle is for resizing.
- Sometimes a new burner cannot be added to the furnace. Adding or deleting an object from a constructed furnace will cause the grid to be reconstructed and then

previous enhanced grid edits would be lost. Therefore, when an enhanced grid edit is made, the ***Protect Grid Edits*** option is automatically checked and objects will not be allowed to be changed. The ***Protect Grid Edits*** option can be turned off when the user really wants to modify the grid.

- The open file dialog box in RunPlot usually comes up in the directory of last use (as opposed to the one of the current case). The user may navigate to the directory of the file to open.
- Multiple instances of RunPlot may be run, and the RunPlot windows can be moved and resized to easily monitor the status of variables from different data files simultaneously.

Combustion Domain

- Air is assumed to be composed of nitrogen and oxygen only.
- Radiation must be turned on in the combustion space for cycling runs. In general, heat transfer under furnace operating conditions is dominated by radiation heat transfer. Therefore, although combustion simulations can be run without the radiation heat transfer computation included, the gas temperatures will likely be unrealistically high because heat loss from radiation has not been accounted for.
- The number of iterations in a combustion run may exceed the user specified value. Ten extra gas phase iterations will be added after a full radiation calculation has been done, even if the user set iteration count has been exceeded, so that the radiation information gets propagated into the gas flow field.

Melt Domain

- Configuring chargers and setting batch/cullet velocity to match observed, expected, or desired batch/cullet coverage on the melt surface is crucial to achieving stable, physically reasonable results in the melter model. The batch and cullet transport and melting models in GFM are by necessity simplifications of very complex physical processes. The batch/cullet melting process is modeled as a process that takes place on and just above the surface of the melt. Batch and cullet enter the melt through chargers at a user specified velocity. This velocity is assumed to be the mean value that solids move across the melt surface away from a charger. At present, the models do not account for any spreading out of the batch or cullet on the surface as it moves away from a charger. In the hot furnace environment, the surface of the solids reaches the melting point very rapidly. This melting point is also user specified, and is usually much cooler, by several hundred degrees, than the melt surface that is not covered by batch or cullet. As a consequence, the bulk of the radiation energy that enters the melt space will be transferred to the surface of the solids. The heat flux pattern at the melt surface is, therefore, highly dependent on the batch and cullet coverage. The following steps

are therefore recommended to obtain reasonably realistic melt surface conditions in the melt.

- The charger width should be adjusted from its actual width to a width that yields a batch coverage on the melt surface that matches coverage observed, expected, or desired in the judgment of the furnace engineer.
- Solids on the surface move away from a charger inlet in a straight line until heat transfer via radiation/convection from above and conduction from the melt below is sufficient to melt all the solids. If the solids are not completely melted before they reach the wall on the opposite side of the melter, the computation will not be numerically stable, and mass and energy balances will most likely not be achieved. This condition is not considered to be physically realistic or within the scope of problems that GFM can simulate in a steady state computation. In cases where solids do reach an opposite wall, the velocity of the solids specified by the user as the **charger inlet velocity should be lowered until the batch coverage that is computed in the simulation is close to what is observed, expected, or desired.**
- Side chargers may yield numerically unstable results of failure to converge with melt mass balance if the batch/cullet does not melt before reaching the opposite side of the melter. For opposed side chargers the same difficulties arise if the batch/cullet does not melt completely before reaching the center of the melter. The batch/cullet model does not have a sub-model to handle the physics of cases where opposing batch/cullet flows on the surface collide and either spread out or move off in another direction. **Currently, the best way to set up a simulation of a melter with side chargers is probably to move the chargers to the end on the long axis away from the exit and configure them to yield approximately the same batch/cullet surface coverage as is observed, expected, or desired from a side charger configuration.**
- Chargers must be positioned at the top of the melter. This position is required because the batch/cullet melting process is modeled as a surface process that takes place in and above the top layer of computational cells in the melt grid.
- Note that charger configuration and setting of batch/cullet velocity may require changes and adjustments after an initial simulation yields an unexpected or undesired batch/cullet coverage on the melt surface or a coverage that does not result in complete melting.
- The height of a charger just determines the height of the top layer of mesh cells in the melter computational grid. It does not need to correspond to the real charger height. Batch and cullet melting is modeled as a process

that takes place in and above the top layer of melter computational cells. The height of the batch and cullet above the surface does not need to be accounted for by the user in defining charger height. The software will feed in batch and cullet at the rate specified by the user even though the specified charger height does not appear large enough to accommodate the charging rate. This is because the top of the charger is really open for the purposes of the simulation and feeding solids into the melt. Entering a large charger height may cause it to be truncated to the maximum height allowed for a grid cell, which depends on the melter dimensions and coarseness. This is not a problem. A charger spanning two or more grid cells in height, however, is a problem because the melting model for batch and cullet is active only in the top layer of grid cells. If the user modifies the melter dimensions, changes grid density, or add/deletes grid lines, then the user should make sure the charger is only one cell high.

- To achieve mass balance in the melt (solids in = molten glass out), the force of gravity must be able to act to drain the melter. For this reason, the bottom of a charger should not be below the top of an outlet. This means that the “bottom gap” of a charger must be greater than or equal to the “bottom gap” plus the “height” of all outlets. In general, numerical stability and convergence with mass balance are enhanced when outlets are at larger depths below the melt surface because the static head due to gravitational force acting to move molten glass out the exit is greater.
- Refiners may lead to numerical instability and failure to converge for a number of reasons. Among them flow through the throat and in the refiner are difficult to resolve in the relatively coarse grids required to run on a single computer, there is some heat loss through the wall of a refiner, but currently no provision to maintain temperature in a refiner via small burners above the surface, and refiner outlets may be very close to the melt surface giving very little static head force to move molten glass out the exits. If simulation with a refiner does not converge well enough to give useful results for the overall furnace analysis, building the melter without a refiner and letting the melter throat be the melter outlet may provide converged results that are good enough for evaluating overall furnace efficiency and achieving the other goals of the analysis.
- In order to resolve the flow field at melter outlets, the grid will show a thick wall and exit tunnels. For the purposes of estimating heat losses through walls, the wall thickness specified by the user is used for all walls regardless of how they have been modified in the grid generation process.
- Melter outlet tunnel information (in the green boxes) applies to all the melter outlets. This information cannot be changed for individual outlets.
- Melter outlet pull rate fractions are automatically changed when an outlet is added or deleted. The user can change the pull rate fraction values individually.

Simulation

- Convergence is determined by looking at the `convgnnnn[m/c].plt` file via the RunPlot program. The graph will show the normalized mean and maximum mass residual as a logarithmic plot in either the melt [m] or combustion [c] domains. Considering a computational cell as a control volume, the absolute value of the log of the normalized mean mass residual indicates how many significant digits of mass balance have been achieved in the computational cells averaged over the entire grid. When the simulation has converged, the mean residual line should be at a level of -3 or less, although in difficult to converge cases, some useful information may be obtained when the mean mass residual is below -2. When the mass residual plot has gone asymptotic, further iteration is not likely to significantly improve convergence in terms of mass balance. However, other significant physical characteristics of the system, such as the mean temperature may still be evolving. Iteration should be continued until all major measures of system state have stabilized.
- Certain variables are extremely sensitive to various conditions that may cause the temperature to go way beyond a realistic value or prohibit the simulation from converging. When such a situation occurs, stop the run and do the following:
 - Look at the grid to be sure that it has been constructed properly. Make sure the inlets and exits are properly located and sized.
 - Carefully review all the properties, parameters, conditions, and options that can be set in the pre-processor to make sure that they are realistic and agree with the furnace design. Doing an approximate heat balance for the furnace operating conditions can be very useful in identifying unrealistic parameter values.
 - Examine the flow patterns within the furnace by entering the post processor and clicking on “vectors” in the grid control block. Change positions in the grid and relick “vectors” as needed to get a good idea of what the flow pattern is. Perhaps better resolution is needed near various objects to improve the flow. In that situation, create a new case with a refined grid that is denser near those objects.
 - Use the Soot Model Calibration Procedure given below.
 - Use the Adjust Relaxation Procedure given below.
- If the simulation display updates have been turned off during a run, then the old information that is not current will remain on the screen even after the run is done.
- There is more than one way to tell if a simulation is running.

1. There may be a long delay between GUI display updates to the status boxes in the main GFM application window, especially when there are many cells in the grid and during initialization (at startup and when changing between calculation models). First make sure that the menu item ***Options → Simulation Display Updates → Simulation Display Progress On*** is checked. This option defaults to being off because updating the colored plot display takes up extra CPU time.
 2. The GFM menu bar will have an item ***Stop Run*** while a simulation is running. This menu item will change to ***Done*** after the simulation stops.
 3. Look at the Windows task bar for the CFD program's command window icon. Click it open to see the simulation status. If the window cannot be found, then the simulation is not running.
 4. The Windows task manager can also be checked to see if the melt or combustion programs (melt.exe or comb.exe) are running as tasks.
- Remember that every new cycle and change of domain represents a new CFD program run and the option settings for each run are taken from the case settings for both domains before cycling was started, except for those that must be modified for cycling (such as “restart” versus “new start”). Before starting a cycling run, set the options as generally desired. Options can still be modified during each of the individual runs.
 - Sometimes a simulation restart fails. There are several things that can cause a restart to fail:
 1. If a grid has been modified, then the simulation must be started as a “new run.”
 2. In a series of parametric runs generated using ***Save Case As (Full Case)*** to create a “good” starting point from a previous case when a parameter is changed, the parameter change may be too large. Try changing the parameter in smaller increments or using a new start.
 3. When restarting a cycle run, do not open the case; just use the ***Simulate*** menu to start another cycle. If the case has been reopened, then verify that the run parameters are correctly specified for restarting before saving the case.
 - There are a couple strategies that can be used to speed up a simulation:
 1. Make sure the menu item ***Options → Simulation Display Updates → Simulation Display Progress Off*** is checked because updating the display takes extra processing time. Monitor run status by watching the command window or using RunPlot.
 2. Simplify the grid if possible. Make the grid coarser.
 3. It may be possible to increase computational relaxation factors by using the Adjust Relaxation Factors Procedure below. However, this is a trial and error strategy which may not work. The best situation for its use is to begin with a case that has an established flow field and is somewhat converged, as example

when doing parametric studies based from one converged case that is copied as a starting point for using a different value of the parameter being studied.

4. Reduce the number of iterations for radiation and/or minor species in the green boxes displayed when the menu item **Properties** → **Simulation Parameters** is clicked. This is also a trial and error strategy.
5. Reduce the radiosity solver loop limits via the menu item **Properties** → **Radiation Controls**. This is also a trial and error strategy.

2. Soot Model Calibration Procedure

In air-fuel furnace simulations, the radiation heat flux solution is very sensitive to soot production, distribution, and oxidation. To accommodate simulation with the resources of a PC, a reduced two step soot production and oxidation model is used that has four kinetic parameters. In air-fuel furnaces, the magnitude of the total heat flux to the melt is very sensitive to the kinetic model soot parameters, and these parameters may need adjusting for different types of burners, fuel, and furnace geometries.

GFM soot transport is assumed to be governed by the general equation for subspecies transport. Transport of soot as a mass fraction of the gas mixture is then:

$$\frac{\partial}{\partial x_i} \left(\rho u_i Y_s - \Gamma_s \frac{\partial Y_s}{\partial x_i} \right) = R_{sf} - R_{so} \quad 2.1$$

Where

$$R_{sf} = A_{sf} Y_f \exp \left(\frac{-E_{sf}}{R_g T} \right) \quad 2.2$$

and

$$R_{so} = A_{so} Y_s Y_o T^{1/2} \exp \left(\frac{-E_{so}}{R_g T} \right) \quad 2.3$$

Einstein index notation is assumed in Equation 2.1, and Y_s is the mass fraction of soot, Γ_s is the turbulent diffusivity of soot, and R_{sf} and R_{so} are the formation and oxidation rates of soot in kg/(kg_{mixture}) per unit volume, respectively. In Equation 2.2, A_{sf} is the pre-exponential constant in the soot formation rate equation, E_{sf} is the activation energy for soot formation, and Y_f is the mixture mass fraction of fuel. Similarly, in Equation 2.3 A_{so} is the pre-exponential constant in the soot oxidation rate equation, E_{so} is the activation energy for soot oxidation, and Y_o is the mixture mass fraction of oxygen. The model parameters are the pre-exponential constants and activation energies, A_{sf} , A_{so} , E_{sf} , and E_{so} . Their default values are given in Table 2.1.

Table 2.1 Soot Formation and Oxidation Model Kinetic Parameters

Parameter	Value	Units
A_{sf}	0.00353	kg _{soot} /(kg _{fuel} ·s)
A_{so}	30	kg _{mix} /(kg _{fuel} ·K ^{1/2} ·s)

E_{sf}	3325720	J/kmol
E_{so}	3325720	J/kmol

All of these model parameters can be changed by the user. The soot distribution and net radiation from soot appear to be most sensitive to the value of the pre-exponential constant for soot formation, A_{sf} . Therefore, the automated soot model calibration procedure finds a value of the soot formation kinetic constant, A_{sf} , based on operating condition data from an existing furnace or design point conditions for a new design combined with an expected efficiency at the design point. In new designs, the calibrated soot parameter provides a reasonable baseline from which to test design variations to see whether performance improves or degrades.

The soot calibration can also be done for oxy-fuel furnaces, however, soot kinetics are not nearly as significant in oxy-fuel furnaces because a large fraction of the radiation heat flux comes from the dominant combustion products, carbon dioxide and water vapor.

Step 1. Open or create a new case in the melt domain just as would be done for a regular (non-calibration mode) simulation. When setting the values for items in the green boxes (displayed after clicking on the *Properties* → *Simulation Parameters* menu item), make sure the second line in the left-most green box shows “new start” and the next line has an iteration count set high enough to get an established flow field and stabilized bulk melt and surface temperature in the melt, possibly as many as 1000 to 3000 iterations. The **RunPlot** program can be used to monitor how many iterations it takes for the mean temperature in the melt to begin to go asymptotic. In the right-most green box set the heat flux to “scaled”, “uniform value”, and provide an estimated value required for the furnace. This value is not significant when “scaled” is chosen because it will be scaled to the amount needed by the melt based on the batch inlet flow rate, material properties, and the glass outlet temperature. Save the case after the setup has been completed.

Step 2. Open or create a new case with the same case number in the combustion domain just as would be done for a regular (non-calibration mode) simulation. When setting the values for items in the green boxes (displayed after clicking on the *Properties* → *Simulation Parameters* menu item), make sure the second line in the left-most green box shows “new start” and the bottom line in the same box shows “calc radiation”. The top line in the right-most box should show “melt surface: calculated” and the “initial iterations” line should show an iteration count set high enough to get an established flow field, about 600 iterations is reasonable.

Step 3. Click on the *Properties* → *Soot Kinetics* menu item. A drop-down list will appear with lines for soot formation and oxidation kinetic constants and activation energy, followed by another line: “Calibrate Soot Kinetics”.

Step 4. Click on one of the first four lines in the drop down list to pop up a window showing the current value of that parameter. Specify a new value if desired. Repeat for each of the parameters, going back to the *Properties* → *Soot Kinetics* menu item each

time to get to the drop-down list. The algorithm used in the CFD program will adjust the soot formation kinetic constant while the soot oxidation kinetic constant is held constant. The adjustment is based on the user supplied fuel and air in flow rate and on the energy required by the melter. The correct fuel and air or oxygen flow rates can be determined by doing an energy balance on the furnace using a spread sheet. In addition, the output of the melt simulation in the “scaled” mode will provide detailed information on the energy requirements for the melt including heat losses from the melt through the walls. This information can be used to determine what fraction of the heating value of the fuel must be transferred to the melt, which determines a baseline efficiency. The soot calibration procedure will tune the soot production kinetic parameter to match these conditions.

Step 5. Get back to the drop-down list. Click on “Calibrate Soot Kinetics” to activate the calibration mode. (Go back to the drop-down list again to see that the “Calibrate Soot Kinetics” line is now checked.) Save the case and get back to the main menu.

Step 6. Click on the *Simulation → Cycle Domains (Melt First)* menu item and proceed as explained in the GFM 4.0 Automated Cycling Guide document. Set the number of cycles to about 20 to 30 cycles.

Step 7. Monitor the soot_calnnnnc.plt file by clicking on the *Options → Activate RunPlot* menu item. The file will only be changed once per cycle, after the wall radiation has completed in the combustion domain. If the graphed line for *Aform* flattens out and the “HT to melt” line merges with the “E needed by melt” line, then the calibration run can be stopped. When there is a wide difference between plot lines, it may be difficult to see enough detail via **RunPlot**. If so, then look at the soot_calnnnnc.plt file directly, via **Notepad** or another text file viewer.

Step 8. After the simulation has stopped, if the *Aform* value has not settled down, then if it is oscillating due to overshooting and undershooting the energy needed in the melt, more cycles should be run until the *Aform* parameter has stabilized to 2 or more significant digits. If the amount of energy going to the melt is consistently too high, increase the value of the soot oxidation kinetic constant (see step 4 above) and change the soot formation kinetic constant to the latest value of *Aform*. Save the case and go back to step 6.

Step 9. When the *Aform* value has settled down, then set the values of the soot oxidation kinetic constant (see step 4 above) and the soot formation kinetic constant to the latest values of *Aoxid* and “*Aform*” from the soot_calnnnnc.plt file. Also click on the “Calibrate Soot Kinetics” item in the drop down list to uncheck it and turn off the soot calibration mode. Save the case. The procedure is done.

3. Adjust Relaxation Factors Procedure

Relaxation factors are used to slow the rate of change of variables in the iterative solvers. During the iteration process dampening is usually required so that large changes in one variable do not cause other variables coupled to the value of that variable to be adversely affected, possibly slowly spiraling out of control and causing the computation to diverge or oscillate wildly. When a new value for a variable is calculated, that value is actually replaced with a fraction (the relaxation factor) of the new value plus the complimentary fraction of the old value, thus lowering the value of a relaxation factor will reduce how much a variable can change during an iteration (which also may increase the number of iterations needed for convergence) and increasing the value of a relaxation factor will increase how much a variable can change during an iteration (which also may decrease the number of iterations needed for convergence). There is a balance required in finding relaxation factors that allow for convergence and do not unduly prolong the number of iterations needed to reach convergence. The default values work well for most cases. However, in cases that are particularly sensitive numerically, a simulation that does not converge well with the default values may achieve adequate convergence if the relaxation factors are adjusted.

Default relaxation factor values are provided for the combustion domain in the file `relaxfactorc.txt` in the combustion directory and for the melt domain in the file `relaxfactorm.txt` in the melt directory. These files are read by the CFD codes and hence apply to all cases. The information in these files is mapped into the tables below: file `relaxfactorc.txt` corresponds to Table 3.1 and file `relaxfactorm.txt` corresponds to Table 3.2. These tables show the direct association of relaxation factors to data that can be plotted via RunPlot (when such an association exists). Notice that each table is divided into 3 sections. The first 2 sections are for the relaxation factors and the third section is for the number of equation solver sweeps done for particular variables.

The number of solver sweeps controls the amount of computational work done by the low level linear solver routines during each global iteration through all of the governing equations. Using a very large number wastes computational resources because the coefficients are often functions of other variables or of the independent variable (for non-linear equations) and are therefore only known to limited precision that normally increases as the global iteration proceeds. Doing too much work during any one global iteration on a single equation can slow convergence because most of the solution precision gained in the solution step may be lost when the other equations are resolved and the values of coupling and independent variables in the coefficients change. However, using too few solver sweeps can result in too little progress being made in each iteration resulting in numerical oscillation and a simulation that never converges.

The Adjust Relaxation Factors procedure is usually done to reduce relaxation factors when the simulation run is not converging. However, it may also be used to increase the relaxation factors. One situation in which increasing the factors might be done is when there is a fairly well converged case that is going to be used as a base for several other

cases (such as when doing parametric studies) and increasing the relaxation factors might produce optimum values that can be used for a set of parametric runs reducing the number of days or weeks required to complete the parametric study. Care must be exercised under these circumstances to verify that each case in the parametric study has adequately converged. This procedure description focuses on reduction of the relaxation factors to obtain convergence in cases that do not converge adequately with the default values. In this procedure, “modify” and “reduce” means to manually edit the relaxfactor[c/m].txt file with a text editor such as **Notepad**, replacing the default value(s) only and keeping the rest of the file exactly as it was.

Step 1. Determine how to manage the relaxfactor[c/m].txt files. Remember that when a relaxfactor[c/m].txt file is changed, the change impacts all cases that will be run subsequently in the associated domain. If other cases already exist which might need to be rerun or restarted in the future, then it is advisable to make a copy of the relaxfactor[c/m].txt file, uniquely identifying it with a new name (possibly including the other case number or date in the name). In the future when one of the other cases is going to be rerun, then save the current relaxfactor[c/m].txt file with a new unique name, and rename the old file back to relaxfactor[c/m].txt.

Just as a precaution, a copy of each default relaxfactor[c/m].txt file, named base_relaxfactor[c/m].txt is kept in the documents directory. These files are backups that can be copied if the default files are needed again.

Step 2. Stop the run if it is still active. If the data collection flags were not turned on, then set up the simulation again with all the data collection flags turned on and repeat the run. Stop the run at about the same number of iterations as before.

Step 3. Examine convergence data via **RunPlot**. The governing equation residuals provide the primary indicators of convergence. The residuals are normalized by the largest term for each computational cell and averaged over all cells in the grid. For most variables, the residuals should drop below 10^{-3} . The exceptions are some of the minor species. Mass fractions for some of these variables are very small and even negligible over most of the computational cells. The normalized mean for these variables, therefore, may reflect a value dominated by residuals from cells that have essentially none of the species present. Methane, oxygen, and NO, for example, typically do not have mean residuals below 10^{-3} in the minor species residuals plot. For most variables, however, if the mean residual plot is not dropping below 10^{-3} , then reducing the relaxation factors for the variables that do not meet these criteria may allow the computation to converge. Also try to determine if any specific variables are erratic or just not moving in the desired direction. If so, then reducing the relaxation factors for those variables may help. If specific variables cannot be identified as causing a problem, then reduce all the relaxation factors in the first section of the relaxation file may resolve the problem.

Relaxation factors can be safely cut in half. The penalty for making relaxation factors too small is slower convergence (more computation time for a simulation). If settings are found that achieve convergence for an otherwise unstable simulation, they can be

increased to speed up the computation for parametric runs as long as convergence is maintained.

Step 4. Setup and run the simulation again. If the simulation converges then exit this procedure.

Step 5. Try each of the alternatives below in turn. Setup and run the simulation again each time. If the simulation is converging then exit this procedure.

1. Reduce all the relaxation factors in the second section of the relaxfactor[c/m].txt file.
2. Reduce all relaxation factors a second time.
3. Increase the number of solver sweeps in the third section of the relaxfactor[c/m].txt file.

The number of solver sweeps can have a dramatic effect on convergence level as measured by the mean residual. It also has a dramatic effect on computation time. Because two or more orders of magnitude of convergence in driving down the residuals may be lost in the perturbations caused by changes in coupling variables over the course of a global iteration through the solver for all of the governing equations or in cycling between combustion and melt spaces, the computational cost of driving residuals down more than a few orders of magnitude may not produce any significant gains. Experimentation with the number of solver sweeps can help to find optimum values.

Step 6. Go through each of the relaxation factors one by one, reduce the factor by a significant amount and increase the solver sweeps by a significant amount, and redo the simulation. Restore the factor to its previous value if there is no change in the simulation results.

Table 3.1: Combustion Relaxation Factors		
Default Value	Item to Relax	Data to Check via RunPlot
Relaxation factors for equation loops		
0.7d0	general value	
0.7d0	equation for fuel mass fraction	Fuel in gresid_xtra####c.plt file
0.7d0	equation for oxygen	O2 in gresid_xtra####c.plt file
0.7d0	equation for enthalpy	Enthalpy gresid####c.plt file
0.7d0	equation for turbulent kinetic energy	k in gresid_xtra####c.plt file
0.7d0	equation for turbulent dissipation rate	Epsilon in gresid_xtra####c.plt file
0.7d0	equation for carbon dioxide	CO2 in gresid_xtra####c.plt file
0.7d0	equation for nitrogen	N2 in gresid_xtra####c.plt file
Relaxation factors for certain variables		
0.7d0	general value	
0.3d0	u momentum in x direction (pressure	
0.3d0	v momentum in y direction (pressure	
	w momentum in z direction (pressure	
0.3d0	correction)	
0.6d0	density	
0.6d0	reaction rate	
0.7d0	initial surface heat flux	
0.9d0	minor species	all but last heading in mresid####c.plt file
0.7d0	soot	Soot in mresid####c.plt file
Number of sweeps in equation solver for specified items		
2	general value	
2	u momentum in x direction	x-momentum in gresid####c.plt file
2	v momentum in y direction	y-momentum in gresid####c.plt file
2	w momentum in z direction	z-momentum in gresid####c.plt file
6	pressure	Pressure in gresid####c.plt file
5	pressure correction	
2	fuel	Fuel in gresid_xtra####c.plt file
3	co2	CO2 in gresid_xtra####c.plt file
10	enthalpy	Enthalpy in gresid####c.plt file
2	turbulent kinetic energy	k in gresid_xtra####c.plt file
2	epsilon	Epsilon in gresid_xtra####c.plt file
3	n2	N2 in gresid_xtra####c.plt file
3	o2	O2 in gresid_xtra####c.plt file
1	minor species	all but last heading in mresid####c.plt file

Table 3.2: Melt Relaxation Factors		
Default Value	Item to Relax	Data to Check via RunPlot
Relaxation factors for equation loops		
0.4d0	default value	
0.4d0	u momentum in x direction	x-momentum in gresid####m.plt file
0.4d0	v momentum in y direction	y-momentum in gresid####m.plt file
0.4d0	w momentum in z direction	z-momentum in gresid####m.plt file
0.4d0	enthalpy	Enthalpy in gresid####m.plt file
1.0d0	electric potential energy	
0.5d0	melt rate	
Relaxation factors for certain variables		
0.3d0	default value	
0.3d0	u momentum in x direction (pressure correction)	
0.3d0	v momentum in y direction (pressure correction)	
0.3d0	w momentum in z direction (pressure correction)	
0.0d0	place holder	
0.0d0	place holder	
0.3d0	particle number density	
0.3d0	particle temperature	
0.3d0	density	
0.3d0	melt rate	
0.2d0	relaxation factor for surface temperature	
0.3d0	down scaling factor for heat transfer from melt surf back into combustion space	
Number of sweeps in equation solver for glass		
2	default value	
2	u momentum in x direction	x-momentum in gresid####m.plt file
2	v momentum in y direction	y-momentum in gresid####m.plt file
2	w momentum in z direction	z-momentum in gresid####m.plt file
2	pressure	Pressure in gresid####m.plt file
4	pressure correction	
2	enthalpy	Enthalpy in gresid####m.plt file
2	electric potential energy	