

Advanced Training: Recent Features, Output Analysis and Current Best Practices Part 1

Bryan Tomsula
CPFD, LLC

2019 Barracuda Virtual Reactor Users Conference
June 27-28, 2019
Chicago, IL

Outline of Topics

Graphical Selection of ICs and BCs

Base Materials

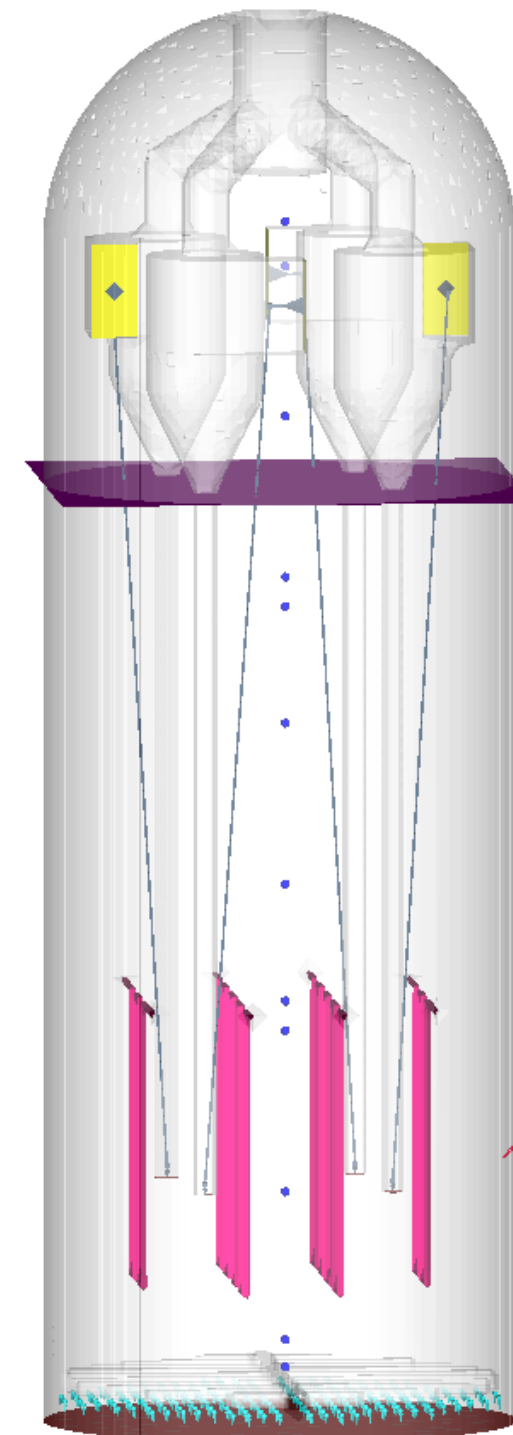
- Updated Base Materials Editor
- “Interpolated from SFF File”
- Material Property Limiters

Plotting Pressure Profiles

Initial Conditions from IC files

New GMV Output Options

Chemistry Reaction Rate Coefficient User-Defined Expression Parser



Example based on
Wednesday Gasifier from
Barracuda training class

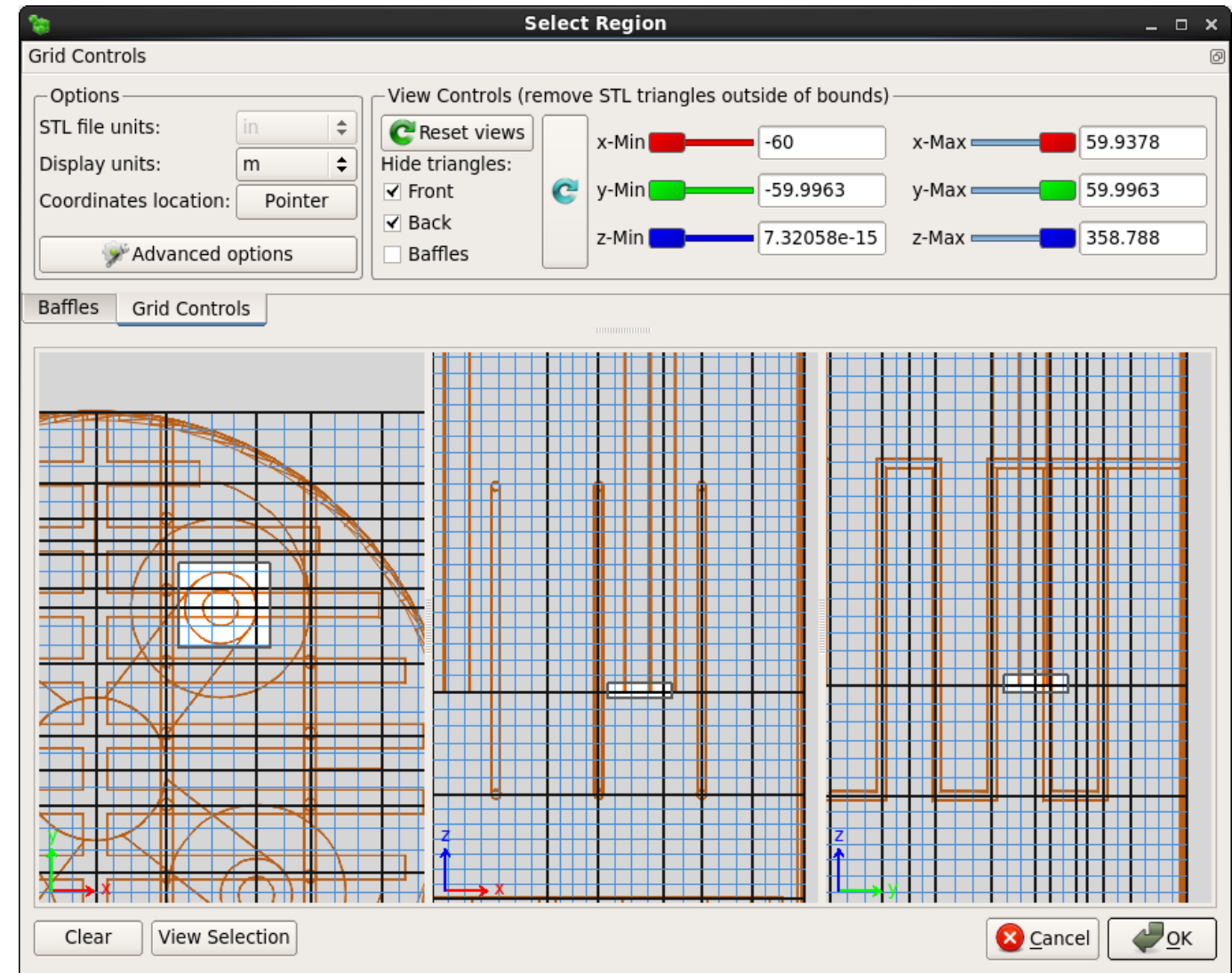
Graphical Selection of ICs and BCs

Added in release 17.3.0

No longer using IJK cell indices to locate ICs and BCs

Now using XYZ physical coordinates

- Will not change when grid resolution is increased or decreased
- Graphical point-and-click selection of IC and BC locations
- Quicker and easier to define



Selecting Pressure BC for Cyclone Inlets

Pressure BC Editor

Pressure boundary condition

Region

Direction: x

Select region (m)

X1 X2

Y1 Y2

Z1 Z2

Flux plane options

Name:

Gas species behavior:

Mass Fraction

Subdivide by radius Divisions 100

Output raw particle data

Comment

Fluid behavior at boundary

Pressure file: Edit

Specify values:

Area fraction 1

Pressure Pa

Temperature 0 K

K-factor 0

Properties

Fluid properties if inflow Interior cell values

Applied fluid species Define fluid species

Particle behavior at boundary

No particle exit

Particle out flow

Particle radius(m) range allowed to exit:

Min = 0 to Max = UNLIMITED

Particle feed (Slip and vol frac)

Particle feed (Slip and mass flux)

Particle feed (Slip and mass flow rate)

Edit particle feed

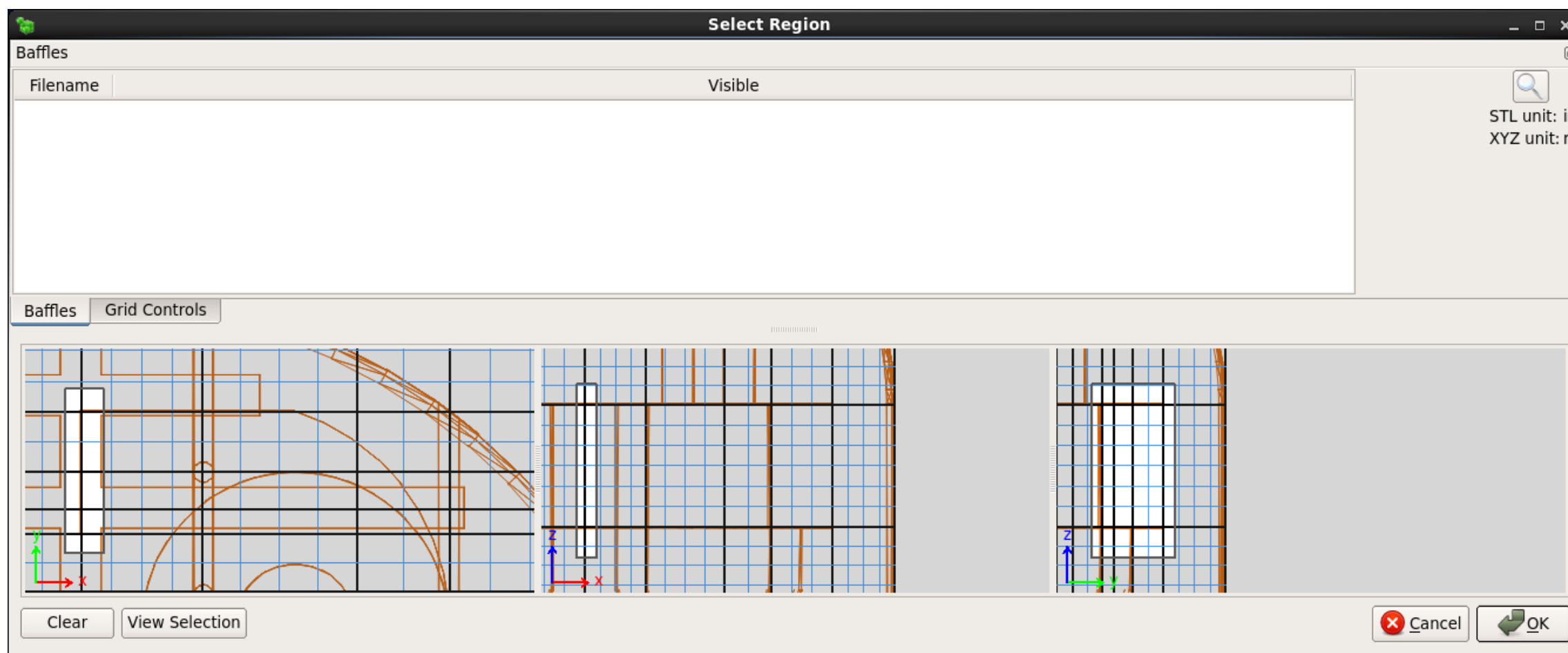
Particle Feed Control

Cancel OK

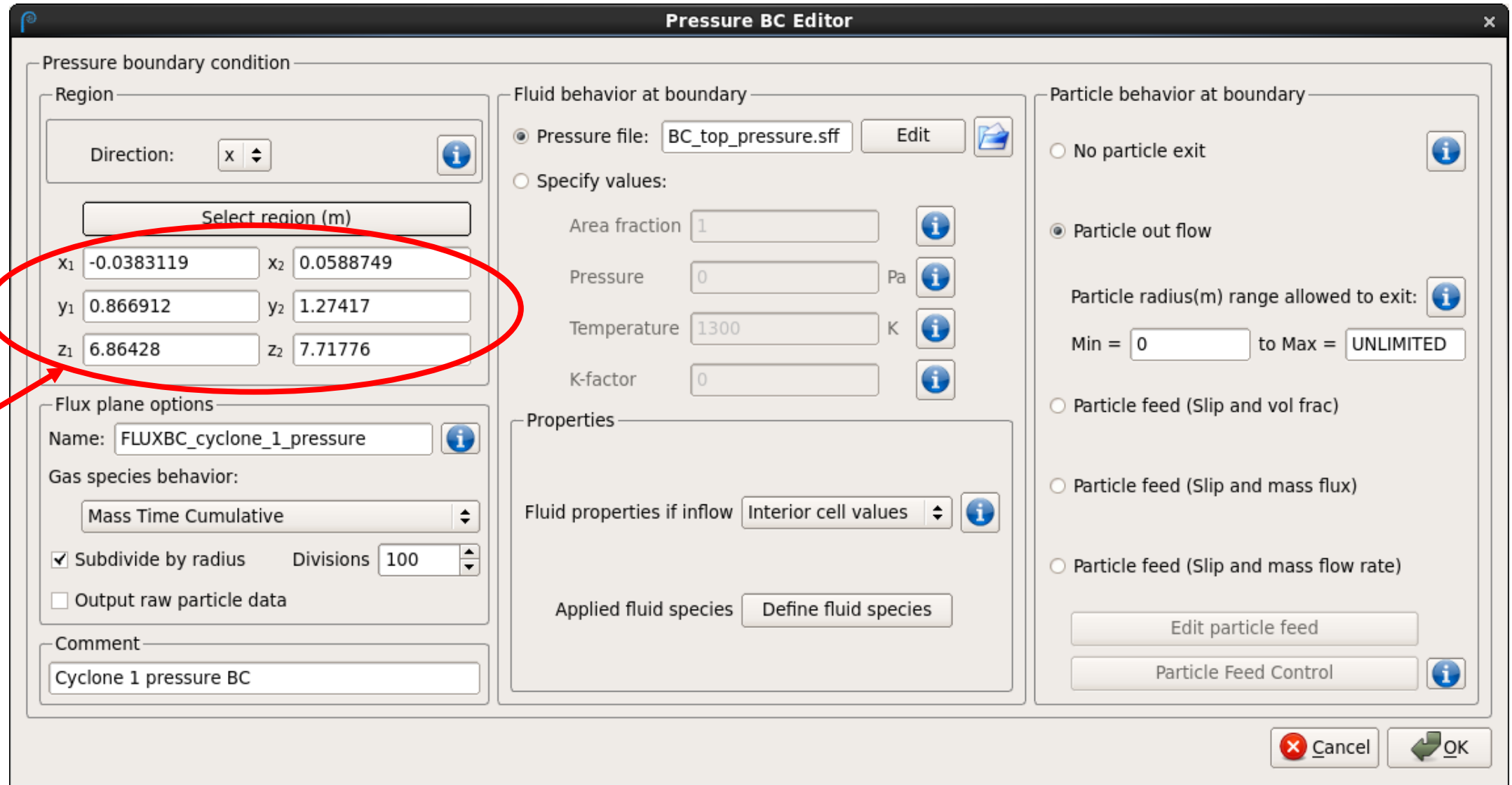
Click on *Select region*

Selecting Pressure BC for Cyclone Inlets

Click and drag to graphically select the cyclone inlet horn face



Selecting Pressure BC for Cyclone Inlets



x, y, z automatically populated from graphical selection

Updated Base Materials Editor

Added in release 17.3.0

More compact and logical arrangement of properties in the GUI

Color-coded labels

- **Green** = properties are valid across entire temperature range
- **Red** = properties are invalid at some temperatures

The screenshot shows the 'Base Materials Editor' window for N2. The 'Name' field contains 'N2' and the 'State' dropdown is set to 'Gas'. The 'Description' field contains 'N2 NITROGEN. REF ELEMENT'. Under the 'Properties' section, several input fields are visible: 'Molecular weight' (28.0134 g/mol), 'Heat of formation' (0 J/kg), 'Density' (0 kg/m³), and 'Critical temperature' (0 K). Below these are buttons for 'Viscosity', 'Heat Capacity', 'Mass Diffusivity', 'Thermal Conductivity', 'Vapor Pressure', and 'Enthalpy'. The 'Molecular weight', 'Heat Capacity', and 'Thermal Conductivity' labels are highlighted in green, indicating they are valid across the entire temperature range. The 'Cancel' and 'OK' buttons are at the bottom right.

Material Properties “Interpolated from SFF File”

Added in release 17.3.0

New functional form available for all material properties

- “Interpolated from SFF file”

“Input” is temperature in the user-selected units

“Output” is the current property in the user-selected units

Linear interpolation between rows

The image shows two overlapping software windows. The background window is the 'Material Property Editor' for material 'N2_1 (G)' and property 'Viscosity'. It shows units of Pa·s and K, and the 'Expression' field is set to 'Interpolated from SFF File' with the file 'n2_viscosity.sff'. The 'Temperature Limits' are set to Min. 200 and Max. 1000. The 'Verification' section shows 'T = 300 K' and 'Expression = 1.'. A message at the bottom says 'Expression is valid'. The foreground window is the 'Lookup Table Editor' for the file 'n2_viscosity.sff'. It contains a table with 10 rows of data:

	Input	Output
1	200	1.337e-05
2	300	1.763e-05
3	400	2.169e-05
4	500	2.555e-05
5	600	2.922e-05
6	700	3.269e-05
7	800	3.596e-05
8	900	3.903e-05
9	1000	4.190e-05
10		

The 'Lookup Table Editor' window has buttons for 'Add Row', 'Delete Row', 'Check Data', 'Graph', 'File: n2_viscosity.sff', 'Save', 'Save As', 'Close', 'Cancel', and 'OK'.

Material Property Temperature and Value Limits

Example: the 4th-order polynomial expression for the viscosity of N2 goes negative at about 4900 K

- Negative viscosity (or other material properties) can lead to invalid results and/or solver crashes

Temperature and/or Value Limits allow for bounds to be placed for each property

- If $T < \text{Min. value}$, property is evaluated at $T = \text{Min.}$
- If $T > \text{Max. value}$, property is evaluated at $T = \text{Max.}$
- Min. and Max. Value Limits can also be defined

Built-in database materials have limits pre-applied when available from the reference sources

Material Property Editor

Material: N2 (G)
Property: Viscosity
Property units: Pa·s
Temperature units: K

Expression
Polynomial (4th order)
 $4.26064e-06 + 4.7525e-08 \cdot T + -9.8826e-12 \cdot T^2 + 0 \cdot T^3 + 0 \cdot T^4$

Temperature Limits
Min. 150 Max. 2403

Value Limits
Min. 1e-07 Max. 0.001

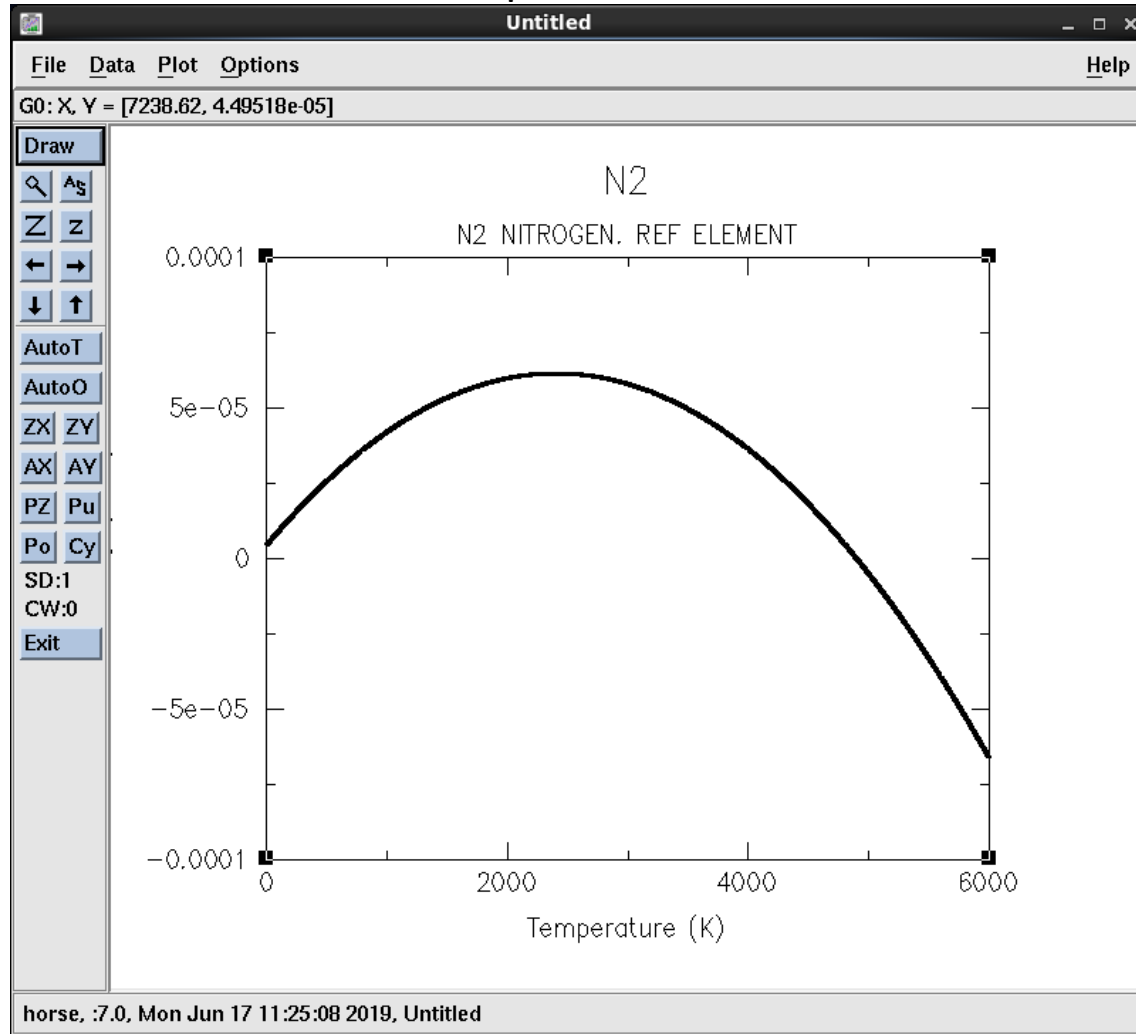
Verification
Display units as: Specified SI
T = 300 K Expression = 1.762871e-05 Pa·s
T_{min} = 0 K T_{max} = 6000 K ΔT = 1 K Plot

Messages
Expression is valid

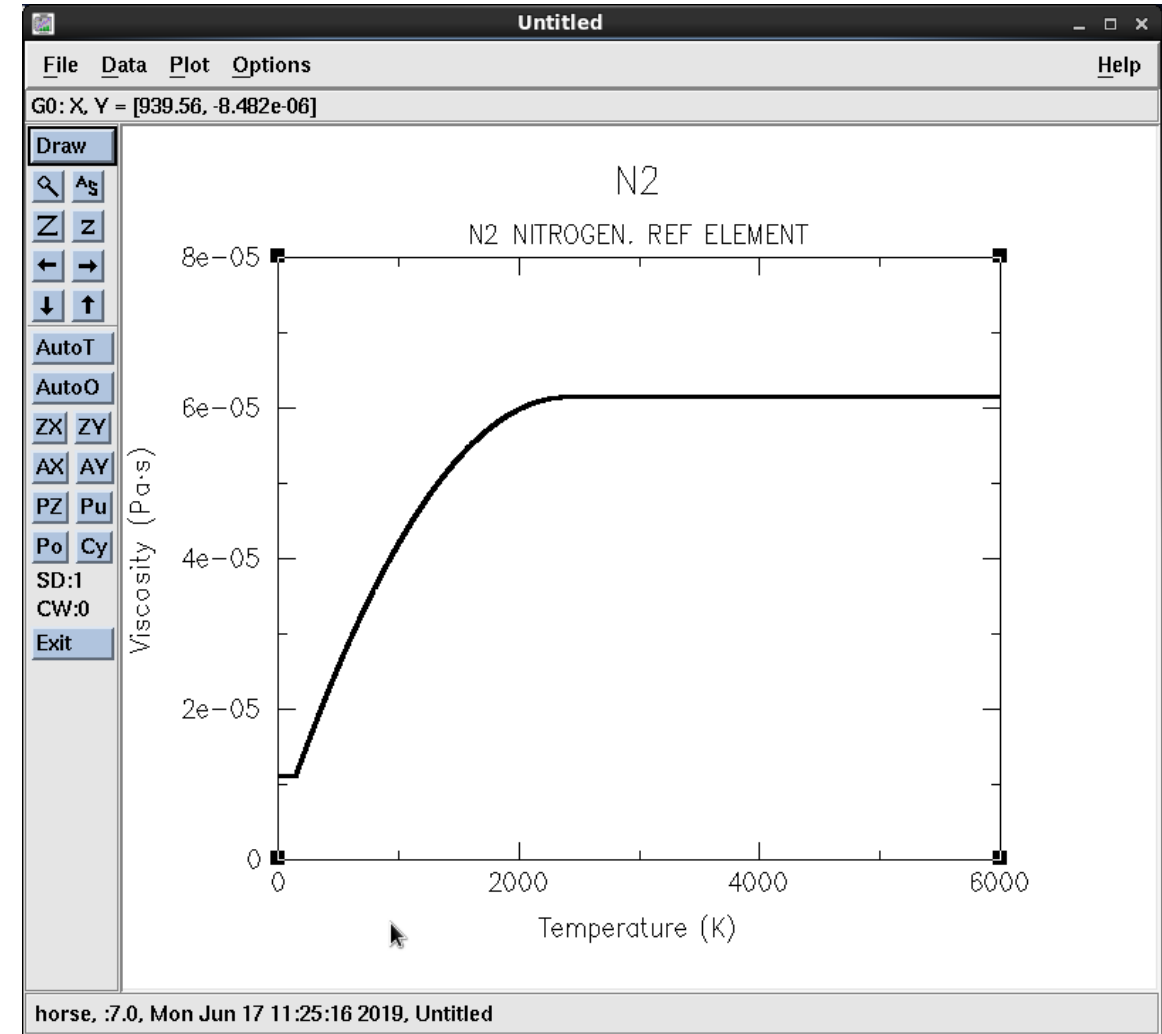
Cancel OK

Material Property Limiter Example: N2 Viscosity

No Temperature Limits



With Temperature Limits



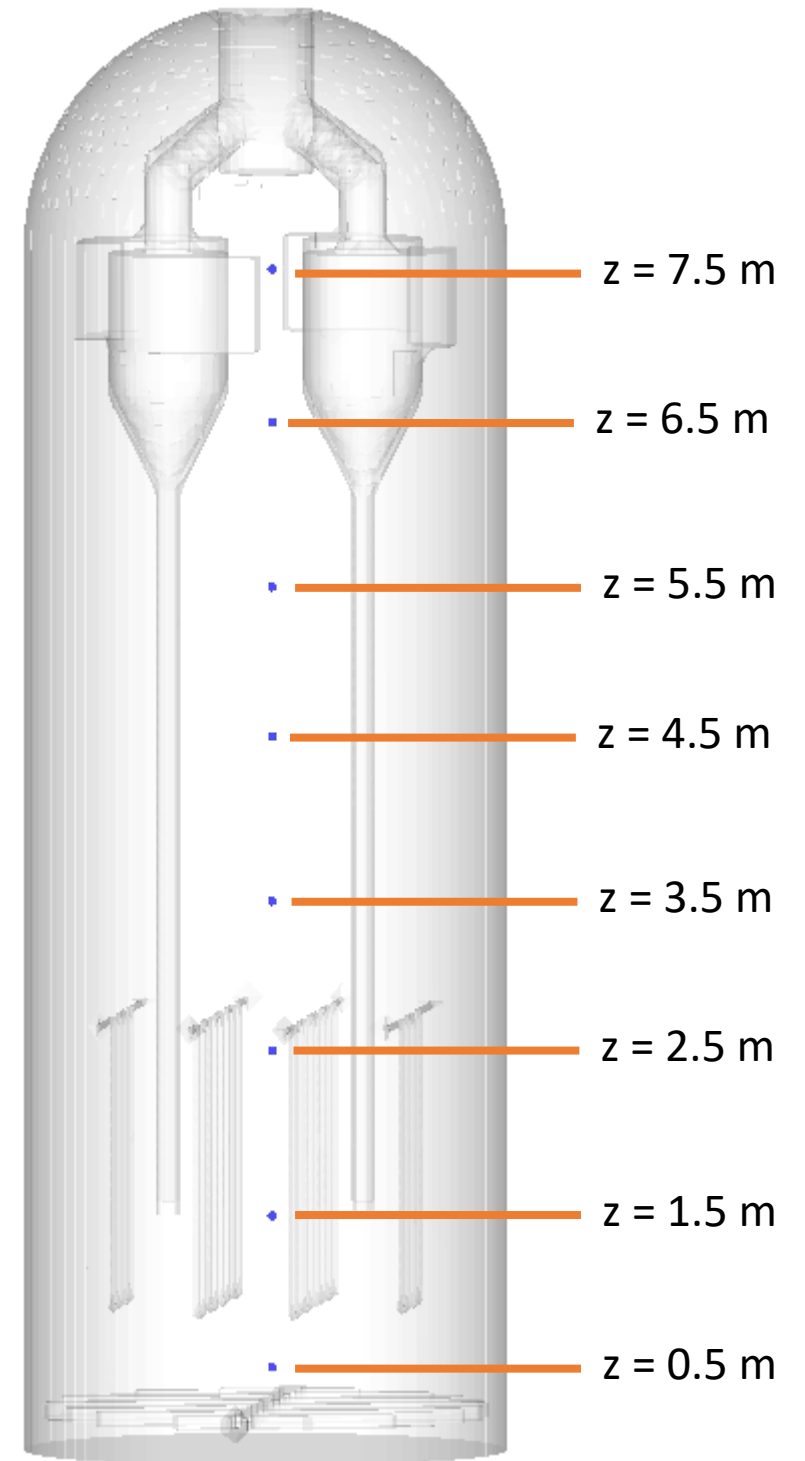
Plotting Pressure Profiles

Transient data points defined along the vertical axis of the vessel

- Record pressure every time-step
- Analogous to physical pressure taps

For convenience, space points evenly at 1-meter intervals

- This makes apparent bed density calculations very easy, since Δz is simply 1 m



Transient Pressure Data

Plot Manager can be used to plot pressure data recorded by the transient data points

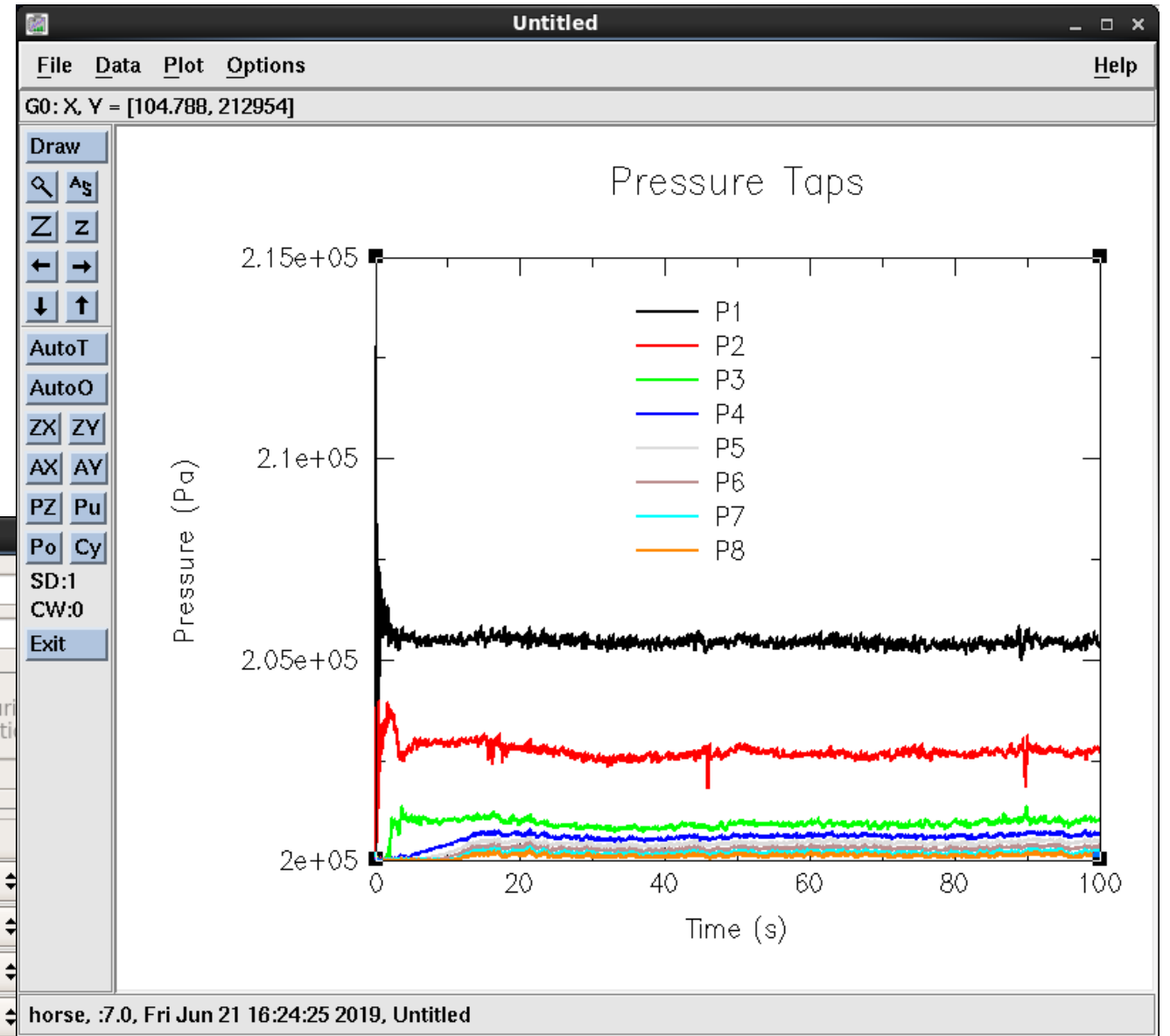
Plot Operations

Name: Pressure Taps
Title: Pressure Taps
Subtitle:
Legend: On Off

X-Axis Title: Time (s)
Y-Axis Title: Pressure (Pa)
 Make Image of Plot
Note: Image will be updated during "Make Image" and "Plot" operations
Image Name:
SD:1
CW:0

	File	Preview	X	Y	Line Name	Color	Style	Width
1	trans.data00		1	6	z=0.5 m	Black	Solid	Small-Medium
2	trans.data00		1	7	z=1.5 m	Red	Solid	Small-Medium
3	trans.data00		1	8	z=2.5 m	Green	Solid	Small-Medium
4	trans.data00		1	9	z=3.5 m	Blue	Solid	Small-Medium

Buttons: Add, Remove, Copy, Check Data, Make Image, Plot, Cancel, OK



Plotting the Pressure Profile at a Specific Time

Python can be used to perform data analysis on the transient pressure data

- Select data for a certain time-period: 90 to 100 s
- Perform a time-average for each location
- Plot the pressure profile vs height

Start a Jupyter Notebook in the simulation directory:

```
jupyter-notebook pressure_profile_example.ipynb
```

Start with a typical “import” cell in the Jupyter notebook:

```
In [1]: %matplotlib inline
        from __future__ import print_function
        from __future__ import division
        import matplotlib.pyplot as plt
        import numpy as np
```

Read from trans.data00 and Calculate Time-Average

```
In [2]: # Read data from transient data file
#       Variable name      Units  ijk          xyz(m)          Comment
#       -----
#@  1  "Time"              "s"
#@  ...
#@  6  "Pressure"          "Pa"   " 17 19  8" " 3.75307e-02 3.58722e-02 5.18390e-01" "P1"
#@  7  "Pressure"          "Pa"   " 17 19 18" " 3.75307e-02 3.58722e-02 1.47646e+00" "P2"
#@  8  "Pressure"          "Pa"   " 17 19 29" " 3.75307e-02 3.58722e-02 2.52164e+00" "P3"
#@  9  "Pressure"          "Pa"   " 17 19 39" " 3.75307e-02 3.58722e-02 3.46846e+00" "P4"
#@ 10  "Pressure"          "Pa"   " 17 19 50" " 3.75307e-02 3.58722e-02 4.50997e+00" "P5"
#@ 11  "Pressure"          "Pa"   " 17 19 60" " 3.75307e-02 3.58722e-02 5.45679e+00" "P6"
#@ 12  "Pressure"          "Pa"   " 17 19 71" " 3.75307e-02 3.58722e-02 6.49830e+00" "P7"
#@ 13  "Pressure"          "Pa"   " 17 19 81" " 3.75307e-02 3.58722e-02 7.46836e+00" "P8"

# List of data columns we want to read:
dataColumnList = [6, 7, 8, 9, 10, 11, 12, 13]

# Array of height values corresponding to each transient data point:
zArray = np.array([0.5, 1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5])

# Min and max time values we want to use for calculating average pressure:
tMin = 90 # (s)
tMax = 100 # (s)

# Array to hold calculated average pressure values:
avePressureArray = np.zeros_like(zArray)

for i, dataColumn in enumerate(dataColumnList):
    t, P = np.genfromtxt('trans.data00', usecols=(0,dataColumn-1), unpack=True)
    tFilter = (t > tMin) & (t <= tMax)
    avePressureArray[i] = np.mean(P[tFilter])
```

How to filter for a specific time period:

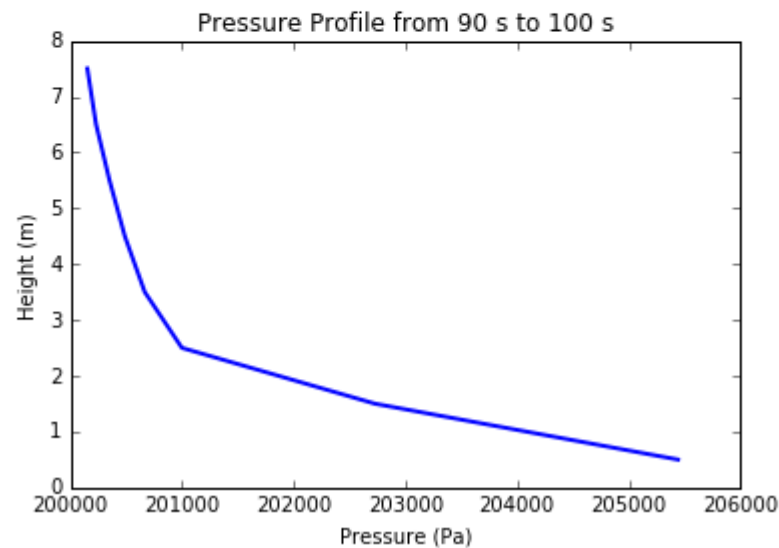
```
tFilter = (t > tMin) & (t <= tMax)
```

You can then use this as a filter on any other data array with matching dimensions:

```
myAverage = np.mean(P[tFilter])
```

Plot the Average Pressure Profile, 90 s to 100 s

```
In [3]: # Plot the pressure profile calculated above
fig, ax = plt.subplots()
ax.plot(avePressureArray, zArray, linewidth=2)
ax.set_title('Pressure Profile from 90 s to 100 s')
ax.set_xlabel('Pressure (Pa)')
ax.set_ylabel('Height (m)')
p = 'pressure_profile_90_s_to_100_s'
fig.savefig(p + '.png', format='png')
fig.savefig(p + '.pdf', format='pdf')
```



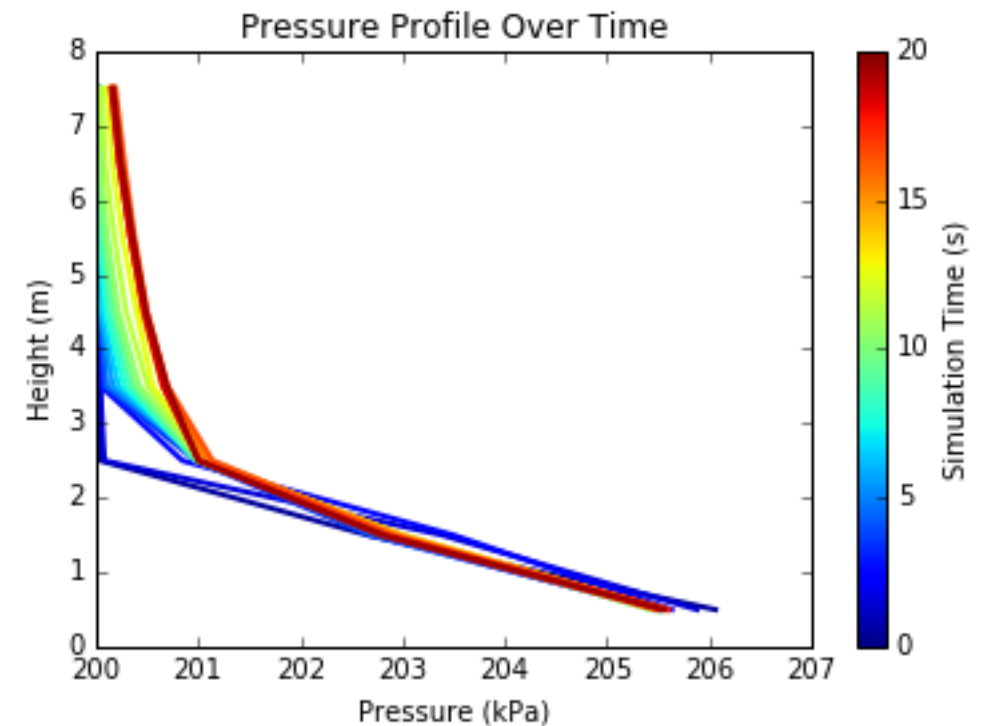
Plotting Pressure Profile vs Time

Useful to assess whether steady state has been reached

Lines are colored using a “colormap”

- Matplotlib includes many useful colormaps, see the web page [Colormap reference](#)

A color bar is placed on the right-hand side of the plot to indicate the time value of each line



Define Time Periods and Read trans.data00

```
In [1]: %matplotlib inline
from __future__ import print function
from __future__ import division
import matplotlib.pyplot as plt
import numpy as np
from matplotlib import cm
```

Import the cm module to use colormaps

...

```
In [4]: # Define minimum and maximum time, and number of lines you want to plot
tMin = 0
tMax = 20
N = 20

# For our plot, let's use "center" values for each time period
tArray = np.linspace(tMin, tMax, N+1)
tMinArray = tArray[:-1]
tMaxArray = tArray[1:]
tCenters = (tMinArray + tMaxArray) / 2
#print(tCenters)

# Read transient data into a 2-dimensional array *once* (don't want to read multiple times in for loops)
# Use unpack=True so that individual columns can be accessed easily
transData = np.genfromtxt('trans.data00', unpack=True)
```

transData is a 2-dimensional array

Create Plot for Pressure Profile Over Time

```
In [5]: fig, ax = plt.subplots()
my_cmap = cm.jet ← Set the desired colormap here

t = transData[0]

for tMin, tMax in zip(tMinArray, tMaxArray): ← Outer loop is for time values
    tFilter = (t >= tMin) & (t < tMax)

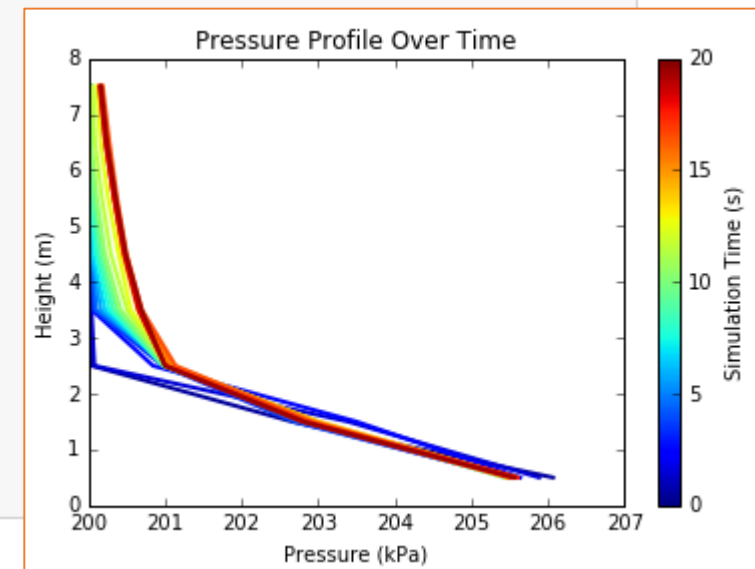
    avePressureArray = np.zeros_like(zArray)

    for i, dataColumn in enumerate(dataColumnList): ← Inner loop is for data columns
        P = transData[dataColumn-1]
        avePressureArray[i] = np.mean(P[tFilter])

    myTime = (tMin + tMax) / 2
    myColor = (myTime - min(tArray)) / (max(tArray) - min(tArray))
    ax.plot(avePressureArray / 1000, zArray, linewidth=2, color=my_cmap(myColor))

# Create a colorbar to put on the right side of the plot
my_norm = plt.Normalize(vmin=min(tArray), vmax=max(tArray))
sm = cm.ScalarMappable(norm=my_norm, cmap=my_cmap)
sm.set_array([])
cbar = fig.colorbar(sm, ticks=tArray[::5])
cbar.set_label('Simulation Time (s)')

# Set the usual axis labels
ax.set_title('Pressure Profile Over Time')
ax.set_xlabel('Pressure (kPa)')
ax.set_ylabel('Height (m)')
p = 'pressure_profile_0_s_to_20_s'
fig.savefig(p + '.png', format='png')
fig.savefig(p + '.pdf', format='pdf')
```



Fluid and Particle ICs: “Initialize from IC file”

Added in release 17.3.0

Use results from a previous Barracuda simulation as fluid and particle ICs for a new simulation (not a restart)

- Run a “base case” to steady state
- Start variations based on the base case steady state results
- All fluid and particle properties are used for the ICs

Current example using Wednesday Gasifier:

- Add additional Injection BC locations
- Add a new internal flux plane
- Select additional GMV output variables

Particle IC

Initial conditions

Initialize from IC file

IC file: IC_base_case_1.000e+02

Temperature: 1300 K

Reset particle residence time

Region

Select region (m)

x1: -1.524 x2: 1.52242

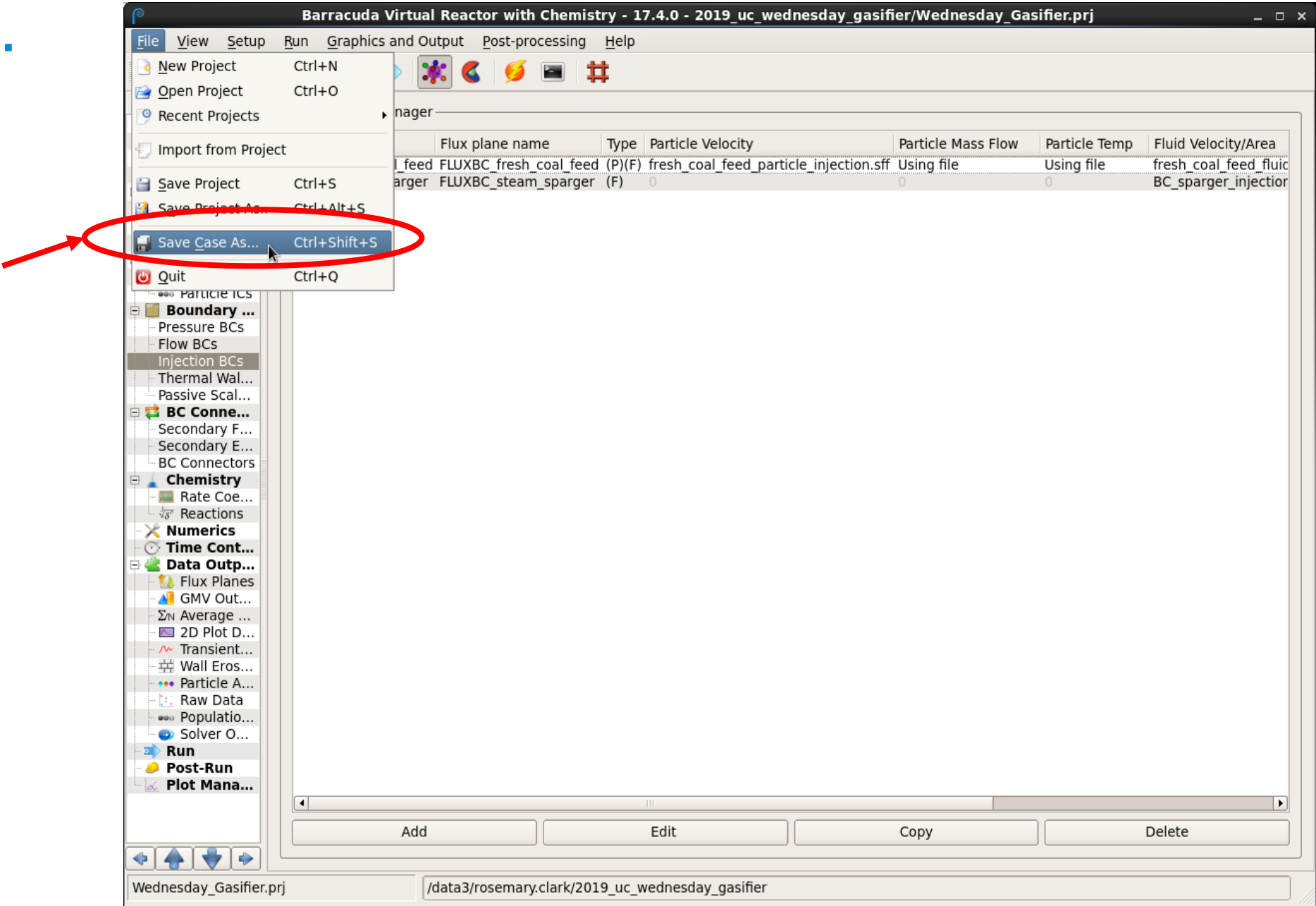
y1: -1.524 y2: 1.52361

z1: 1.85943e-16 z2: 9.11321

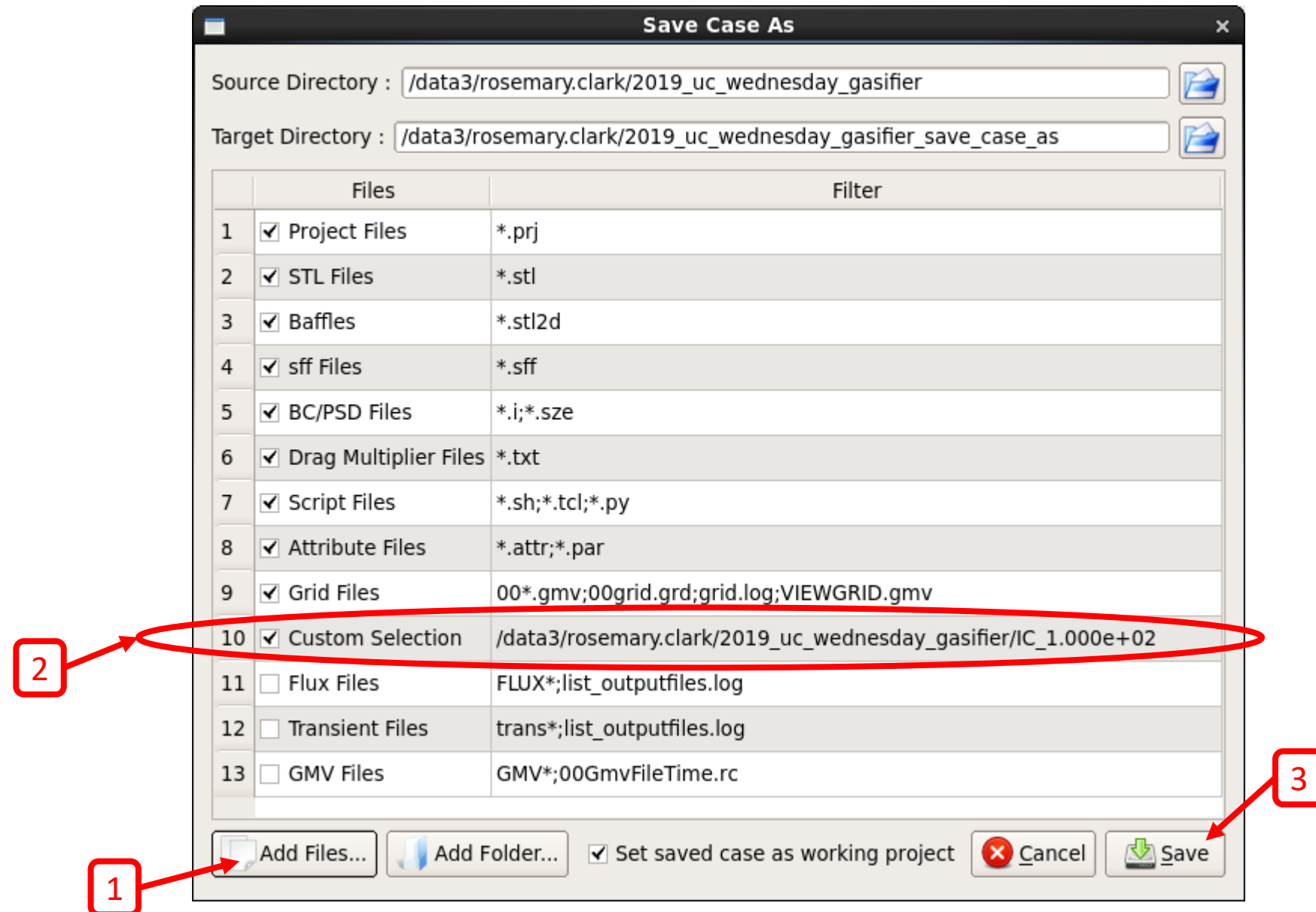
Comment

(Information icon)

Save Case As...



Remember to Select IC file



Edit Fluid IC

The screenshot displays the Barracuda Virtual Reactor software interface. The main window is titled "Barracuda Virtual Reactor with Chemistry - 17.4.0 - 2019_uc_wednesday_gasifier_save_case_as/Wednesday_Gasifier.prj". The "Project Tree" on the left shows the "Initial Conditions" folder expanded, with "Fluid ICs" selected. The "Fluid IC Manager" table shows a single entry with ID 000, checked "On", and coordinates x1: -1.524, x2: 1.52242.

The "Fluid IC Editor: 000" dialog box is open, showing the following settings:

- Initial Conditions
- Temperature: 1300 K
- Pressure: 200000 Pa
- Fluid species: Define fluids
- Velocity: 0 0 0 m/s
- Initial conditions from file
- IC File: [Empty field]
- Region: Select region (m) with coordinates x1: -1.524, x2: 1.52242, y1: -1.52361, y2: 1.52361, z1: 1.85943e-16, z2: 9.11321
- Comment: [Empty text area]

A red arrow points to the "Initial conditions from file" checkbox. A file selection dialog titled "Select an IC_file to use for initial conditions" is also open, showing the file "IC_1.000e+02" selected in the file list. The "File name" field contains "IC_1.000e+02" and the "Files of type" is set to "IC Files (IC_*)".

Edit Particle IC

The screenshot shows the 'Particle IC' dialog box in the Barracuda Virtual Reactor software. The 'Initial conditions' section is expanded, and 'Initialize from IC file' is selected and circled in red. The 'Region' section is also expanded, showing a 'Select region (m)' button and coordinate fields for x1, x2, y1, y2, z1, and z2. A red arrow points to the 'Select region (m)' button, with a text annotation: 'Select full domain, either with GUI click-and-drag or by using [and] shortcut keys in the Region text boxes'. The 'Cloud resolution' section has 'Use global resolution' selected. The 'Special settings' section has 'Random cloud initialization' checked and 'No particle momentum' unchecked. The 'Comment' section is empty. The 'Cancel' and 'OK' buttons are at the bottom right of the dialog box.

Add Injection BCs

Injection BC Editor

Injection name: Fresh_coal_feed
 Comment:

Particle/Tracer Injection
 Use BC Connector data
 Use file cle_injection.sff
 Use specified values

Injection type: Particle
 Species: 002 - fresh coal feed

Velocity: 0 m/s
 Mass flow: 0 kg/s
 Temperature: 0 K
 Number density: 125

Angle Expansion: θ_{E1} 15° θ_{E2} 15°
 Angle Orientation: α_{E1} 0°

Flux plane options:
 Name: FLUXBC_fresh_coal_feed
 Gas species behavior: No Output
 Subdivide by radius Divisions: 100
 Output raw particle data

Fluid Injection
 Use BC Connector data
 Use file coal_feed_fluid_injection.sff
 Use specified values

Fluid composition:
 Velocity: 0 m/s
 Mass flow: 0 kg/s
 Temperature: 0 K

Locations (4 of 4 injections are active)

Name	On/Off	X (m)	Y (m)	Z (m)	nx	ny	nz	Particle Mass Weight	Particle Temp Multiplier	Fluid Mass Weight	d Temp Multiplier
1 Feed	<input checked="" type="checkbox"/> On	1.5	0	1.75	-1	0	-1	1	1	1	1
2 Feed2	<input checked="" type="checkbox"/> On	-1.5	0	1.75	1	0	-1	1	1	1	1
3 Feed3	<input checked="" type="checkbox"/> On	0	1.5	1.75	0	-1	-1	1	1	1	1
4 Feed4	<input checked="" type="checkbox"/> On	0	-1.5	1.75	0	1	-1	1	1	1	1

Particle weight sum 4 Fluid weight sum 4

Add Flux Plane

The screenshot shows the Barracuda Virtual Reactor software interface. The main window title is "Barracuda Virtual Reactor with Chemistry - 17.4.0 - 2019_uc_wednesday_gasifier_save_case_as/Wednesday_Gasifier.prj". The Project Tree on the left shows a hierarchy of settings including Setup Grid, Global Settings, Base Materials, Particles, Initial Conditions, Boundary Conditions, BC Connections, Chemistry, Numerics, Time Controls, and Data Output. The Data Output section is expanded to show Flux Planes. The Flux Planes Manager table is visible, showing two existing flux planes: "000 FLUX_early..." and "001 FLUX_above...". The Flux Plane Editor dialog box is open, allowing the user to configure a new flux plane. The dialog includes fields for Surface plane direction (set to 'z'), Plane Location (xyz), and coordinates (x1, x2, y1, y2, z1, z2). It also has options for Name, Gas species behavior, and checkboxes for "Subdivide by radius", "Output raw particle data", and "Reset particle residence time". A comment field contains "Flux plane above heating coils". Buttons for Reference Grid, Cancel, and OK are at the bottom of the dialog. A note at the bottom of the Flux Planes Manager window reads: "Note: For location indices, 'min' or '[' denote the first possible value, while 'max' or ']' denote the last possible value." At the bottom of the Flux Planes Manager window, there are buttons for Add, Edit, Copy, and Delete.

ID	Filename	Direction	loc	Nod	i1	i2	j1	j2	k1	k2	Sub radius	Co
000	FLUX_early...											
001	FLUX_above...											

GMV Output Options

Barracuda Virtual Reactor with Chemistry - 17.4.0 - 2019_uc_wednesday_gasifier_save_case_as/Wednesday_Gasifier.prj

File View Setup Run Graphics and Output Post-processing Help

Project Tree

- Initial Conditions
 - Fluid ICs
 - Particle ICs
- Boundary Conditions
 - Pressure BCs
 - Flow BCs
 - Injection BCs
 - Thermal Wall BCs
 - Passive Scalar BCs
- BC Connections
 - Secondary Feeds
 - Secondary Exits
 - BC Connectors
- Chemistry
 - Rate Coefficients
 - Reactions
- Numerics
- Time Controls
- Data Output**
 - Flux Planes
 - GMV Output Options
 - Average Data
 - 2D Plot Data
 - Transient Data
 - Wall Erosion
 - Particle Attrition
 - Raw Data
 - Population Data
 - Solver Output Units
- Run
- Post-Run
- Plot Manager

General Mesh View Data Output Options
Please select Eulerian and Lagrangian data for export to the General Mesh Viewer (GMV). Only data selected here can be viewed during post-processing. Values inside parenthesis are the **field names** of the variables within GMV.

Output file interval
Plot interval: s Number of files produced using current end time of **100s**:

Eulerian (Cell) Output Data

<input checked="" type="checkbox"/> Particle volume fraction (p-volFra)	<input type="checkbox"/> Particle bulk density (p-dens)	<input type="checkbox"/> dp/dx (dp/dx)
<input checked="" type="checkbox"/> Fluid velocity (U, V, W)	<input type="checkbox"/> Turbulent viscosity (ViscTurb)	<input type="checkbox"/> dp/dy (dp/dy)
<input checked="" type="checkbox"/> Particle velocity (P_[xyz]Vel)	<input checked="" type="checkbox"/> CFL (CFL)	<input checked="" type="checkbox"/> dp/dz (dp/dz)
<input checked="" type="checkbox"/> Pressure (Pressure)	<input type="checkbox"/> Particle species (Species)	<input type="checkbox"/> Particle mass flux (P_[xyz]Mass)
<input type="checkbox"/> Dynamic pressure (DynPres)	<input checked="" type="checkbox"/> Fluid temperature (f-Temp)	<input type="checkbox"/> Fluid mass flux (F_[xyz]Mass)
<input checked="" type="checkbox"/> Fluid density (f-dens)	<input checked="" type="checkbox"/> Particle temperature (p-Temp)	<input checked="" type="checkbox"/> Wall heat transfer (wallHeat)
<input checked="" type="checkbox"/> Cell indices (i, j, k)	<input checked="" type="checkbox"/> Cell volume (cellVol)	

Lagrangian (Particle) Output Data

<input checked="" type="checkbox"/> Particle volume fraction (VolFrac)	<input checked="" type="checkbox"/> Particle material (Material)	<input checked="" type="checkbox"/> Velocity (vel[xyz])
<input checked="" type="checkbox"/> Particle speed (Speed)	<input type="checkbox"/> Particle density (Density)	<input checked="" type="checkbox"/> Residence time (ResTime)
<input checked="" type="checkbox"/> Particle radius in microns (rad)	<input checked="" type="checkbox"/> Particle species (Species)	<input type="checkbox"/> Residence time by species (ResTime##)
<input type="checkbox"/> Constant color (Particle)	<input checked="" type="checkbox"/> Unique particle ID (pid)	<input checked="" type="checkbox"/> Temperature (Temperat)
<input type="checkbox"/> Drag (drag)	<input type="checkbox"/> Liquid fraction total (liqFrac)	<input type="checkbox"/> Liquid mass total (liqMass)
<input checked="" type="checkbox"/> Cloud mass (cldMass)	<input checked="" type="checkbox"/> Particles per cloud (npCloud)	<input type="checkbox"/> Particle mass (mass)

Gas Species

<input type="radio"/> Mass fraction (<species>.mf)	<input checked="" type="radio"/> Mole fraction (<species>.nf)
<input type="radio"/> Mass concentration (<species>.mc)	<input type="radio"/> Mole concentration (<species>.nc)

Options

<input type="checkbox"/> Compress graphics output (not common)
<input type="checkbox"/> Generate predefined GMV attribute files

Wednesday_Gasifier.prj /data3/rosemary.clark/2019_uc_wednesday_gasifier_save_case_as

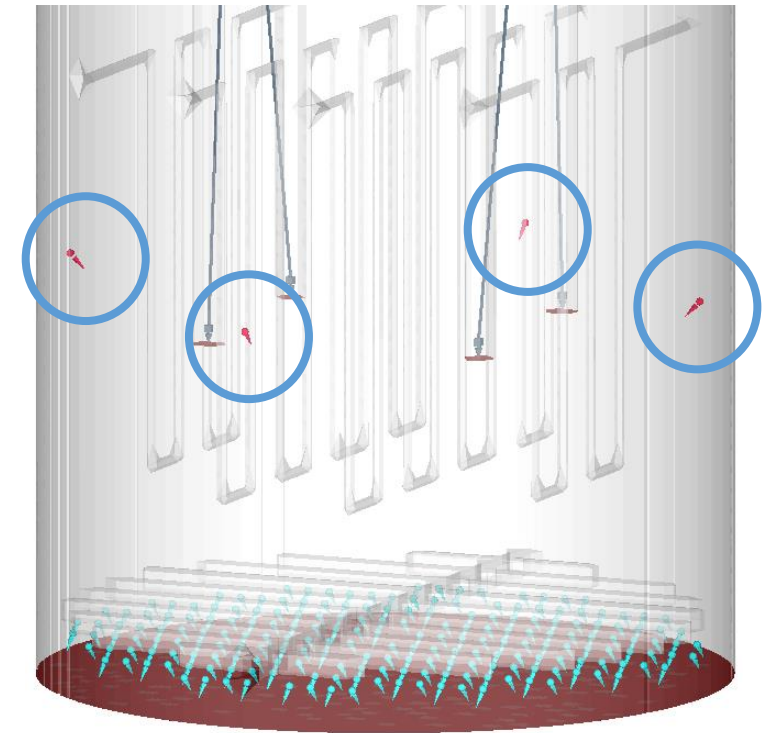
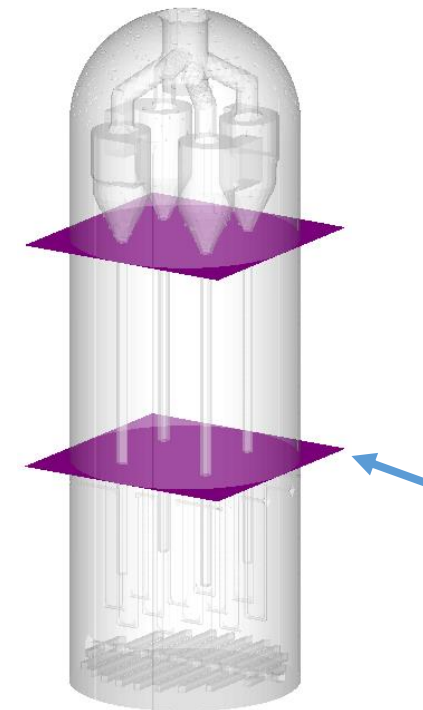
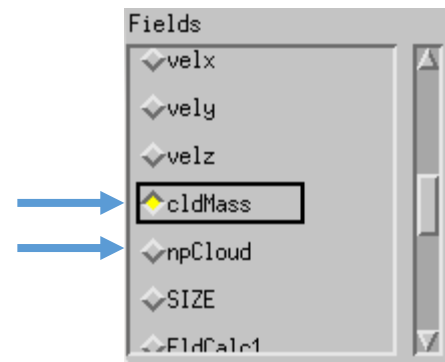
Run for a Single Time-Step to Verify Changes

New coal feed Injection BC locations have been added

- Original simulation had 1 Injection BC for coal feed
- New simulation has 4 Injection BCs for coal feed

New internal flux plane has been added

New GMV variables have been added

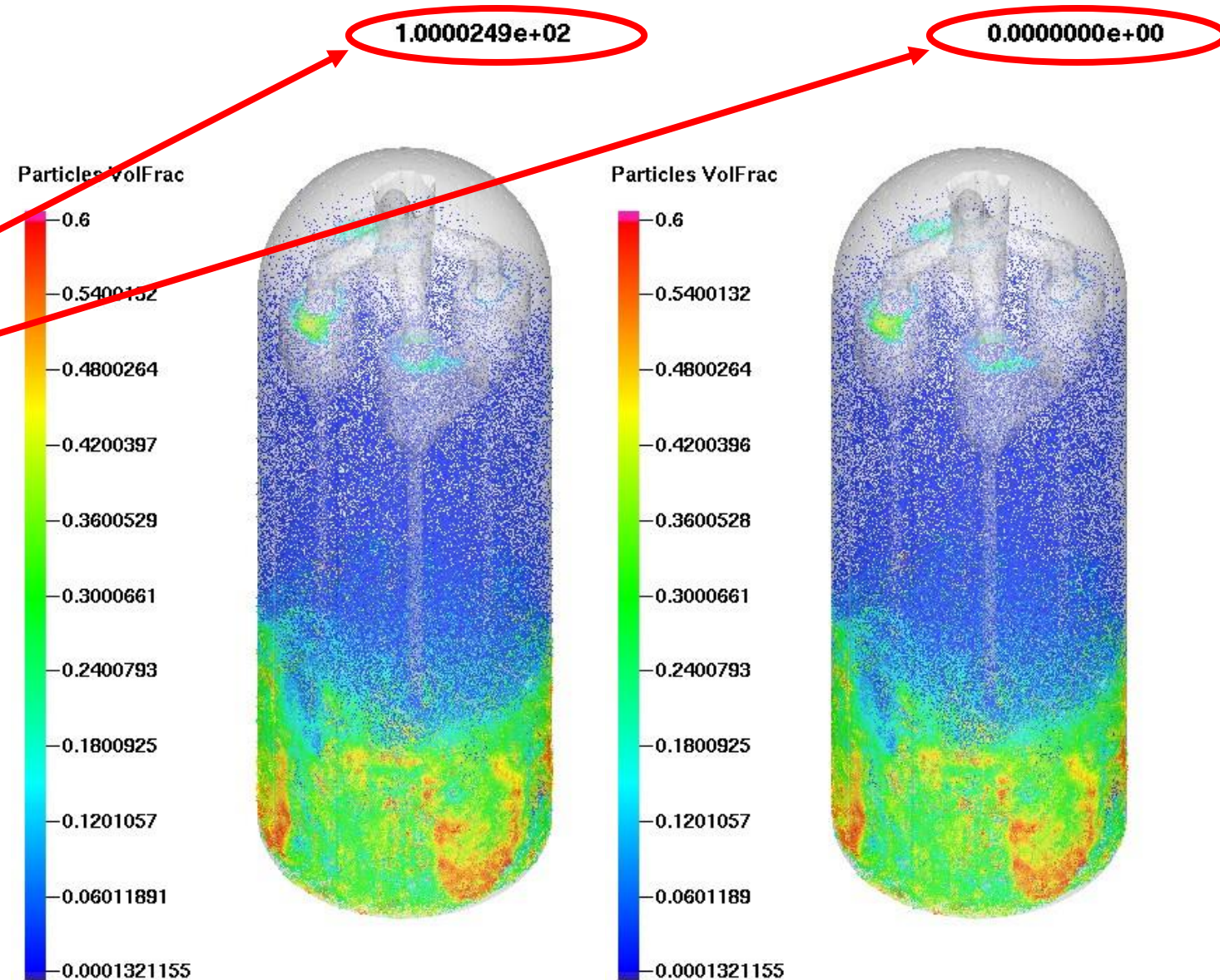


It's Not a Restart!

This is a new simulation, not a restart.

- Original simulation at $t = 100$ s
- Save Case As... simulation with changes at $t = 0$ s

The IC file from the original simulation was only used to define the initial fluid and particle fields in the new simulation



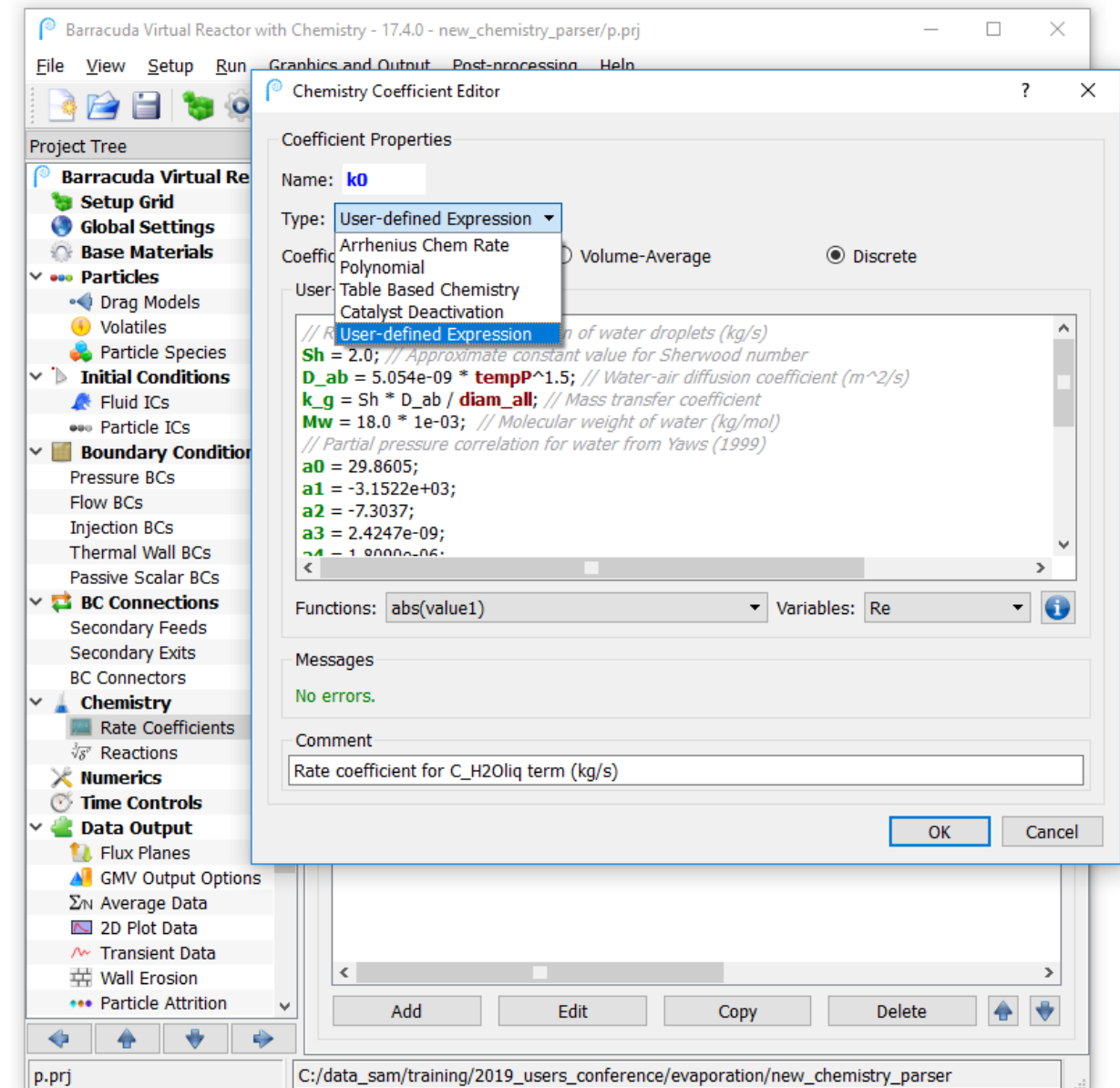
User-Defined Expressions for Reaction Rate Coefficients

New feature in 17.4.0

- Same parser interface as updated User-Defined Drag Model GUI

Interface is improved to allow user to simply and clearly define chemical reaction rate coefficients

- Subexpressions can be used
- Much more flexible and powerful than older constants-based input method



Example: Evaporation of Water Droplets

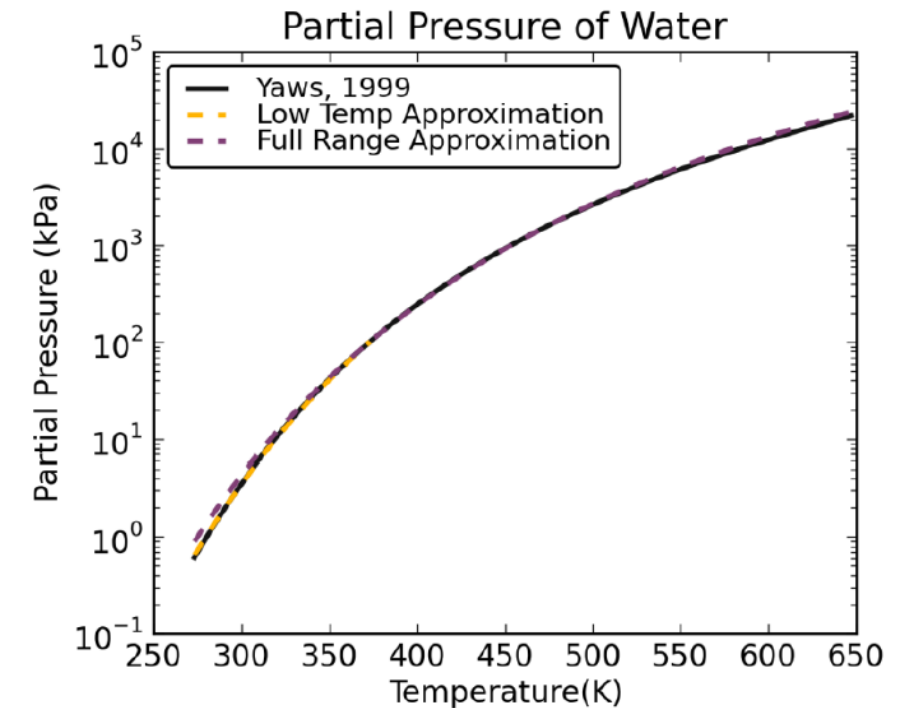
See [CPFD Support Site](#) post [Evaporation Model for Droplets and Solid Particle Drying \(Updated\)](#)

In previous versions of Barracuda:

- The expression for partial pressure of water (Yaws, 1999) could not be entered directly. A curve-fit approximation was needed.

$$\log_{10} P = 29.8605 - \frac{3.1522 \times 10^3}{T} - 7.3037 \log_{10} T + 2.4247 \times 10^{-9} T + 1.8090 \times 10^{-6} T^2$$

- Implementing the time-step stability term required multiple inter-related rate coefficient definitions.



$$\frac{dW}{dt} = \frac{\kappa C_{\text{H}_2\text{O,bulk}} - \kappa R^{-1} T_p^{-1} P_0 \exp(-\beta / T_p)}{1 + \tau_{sim} / \tau_{evap}}$$

Reaction Definitions Using pre-17.4.0 Input (Arrhenius)

The image shows two overlapping windows from the Barracuda Virtual Reactor software. The top window, titled 'Chemistry Rate Coefficients Manager', displays a table of reaction coefficients. The bottom window, titled 'Chemistry Reactions Manager', displays a table of reaction definitions.

Chemistry Rate Coefficients Manager Table:

ID	Name	Reaction Type	Coefficient Type	Expression	Comment
00	k2	Discrete	Arrhenius Chem Rate	$7298.74 T^{-1.5} e^{(-4822.21 / T)} m_{all}^{-1} d_{all}^1$	Liquid Droplet, version 3, CPFD Software
01	k3	Discrete	Arrhenius Chem Rate	$5.7223e-10 T^{1.5} d_{all}^1$	Liquid Droplet, version 3, CPFD Software
02	k4	Discrete	Arrhenius Chem Rate	$2.8566 T^{0.5} e^{(-4822.21 / T)} d_{all}^1$	Liquid Droplet, version 3, CPFD Software

Chemistry Reactions Manager Table:

ID	Reaction Type	Rate	Equation	Comment
00	Discrete	$d[H_2O(S)]/dt =$	$(-1k_4 + k_3[H_2O]) \cdot (1 + k_2)^{-1}$	Liquid Droplet, version 3, CPFD Software
00		$d[H_2O(G)]/dt =$	$-1 d[H_2O(S)]/dt$	

Reaction Definitions with User-Defined Expressions

The screenshot displays the Barracuda Virtual Reactor interface. The top window shows the 'Chemistry Rate Coefficients Manager' with a table of coefficients. A callout box highlights a user-defined expression for the reaction rate r .

$$r = \frac{dW}{dt} = -A_p k_g M_W (C_{H_2O,liq} - C_{H_2O,bulk})$$

The bottom window shows the 'Chemistry Reactions Manager' with a table of reactions. The first reaction is selected, showing its rate and equation.

ID	Name	Reaction Type	Coefficient Type	Expression	Comment
00	k0	Discrete	User-defined	area_all * k_g * Mw * C_H2Oliq / (1 + tau_sim / tau_evap)	Rate coefficient f
01	k1	Discrete	User-defined	area_all * k_g * Mw / (1 + tau_sim / tau_evap)	Rate coefficient f

ID	Reaction Type	Rate	Equation	Comment
00	Discrete	d[H2O(S)]/dt =	(-1k0)+(k1[H2O])	Evaporation of water
00		d[H2O(G)]/dt =	-1 d[H2O(S)]/dt	

User-defined expression is much closer to the way we naturally write the reaction rate with variables.

Syntax of User-Defined Expressions

C-like input

- End statements with ;
- Final return statement

Syntax highlighting

Comments are supported

- Use //

Real time error-checking

The screenshot shows the 'Chemistry Coefficient Editor' window. The 'Name' field is 'k0'. The 'Type' is 'User-defined Expression'. The 'Coefficient is for reaction type' is 'Discrete'. The 'User-defined Expression' text area contains the following code:

```
// Rate coefficient for evaporation of water droplets (kg/s)
Sh = 2.0; // Approximate constant value for Sherwood number
D_ab = 5.054e-09 * tempP^1.5; // Water-air diffusion coefficient (m^2/s)
k_g = Sh * D_ab / diam_all; // Mass transfer coefficient
Mw = 18.0 * 1e-03; // Molecular weight of water (kg/mol)
// Partial pressure correlation for water from Yaws (1999)
a0 = 29.8605;
a1 = -3.1522e+03;
a2 = -7.3037;
a3 = 2.4247e-09;
a4 = 1.8090e-06;
P_H2O = 10^(a0 + a1/tempP + a2*log10(tempP) + a3*tempP + a4*tempP^2); // P (mmHg), T (K)
R = 8.3145; // Universal gas constant (J/mol/K)
C_H2Oliq = P_H2O * 133.322 / (R*tempP); // Convert from mmHg to Pa by multiplier of 133.322 for units of (mol/m^3)
// Time-step based stabilization term
tau_sim = 1e-03; // Simulation time-step (s)
c_p = 4184; // Heat capacity (J/kg/K)
deltaH_v = 40000; // Heat of vaporization (J/mol)
P0 = 4.15040e+10; // P0 from partial pressure approximation (Pa) for range 273.15 K to 647.13 K
beta = 4822.21; // beta value in temperature exponent (K)
gamma = P0 * beta / tempP^2 * exp(-beta / tempP); // gamma for stabilization term
tau_evap = c_p * mass_all * R * tempP / (deltaH_v * gamma * k_g * area_all);
return area_all * k_g * Mw * C_H2Oliq / (1 + tau_sim / tau_evap);
```

The 'Functions' dropdown is set to 'abs(value1)' and the 'Variables' dropdown is set to 'Re'. The 'Messages' section shows 'No errors.' and the 'Comment' field contains 'Rate coefficient for C_H2Oliq term (kg/s)'. The 'OK' and 'Cancel' buttons are at the bottom right.