



Spect3D

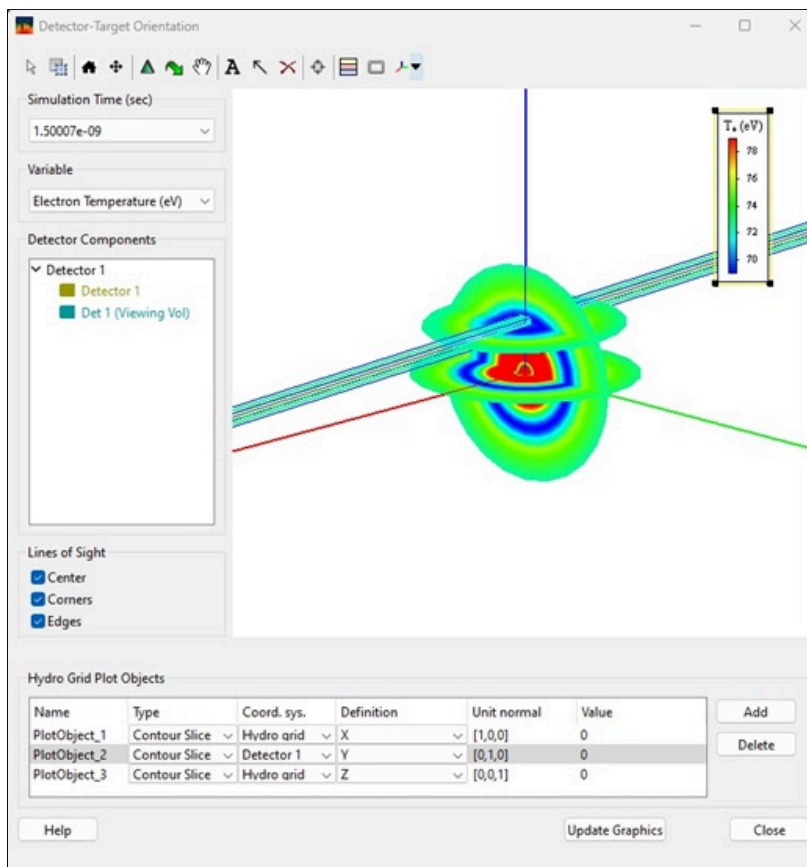
user's guide

Revisions for SPECT3D 20.5.0

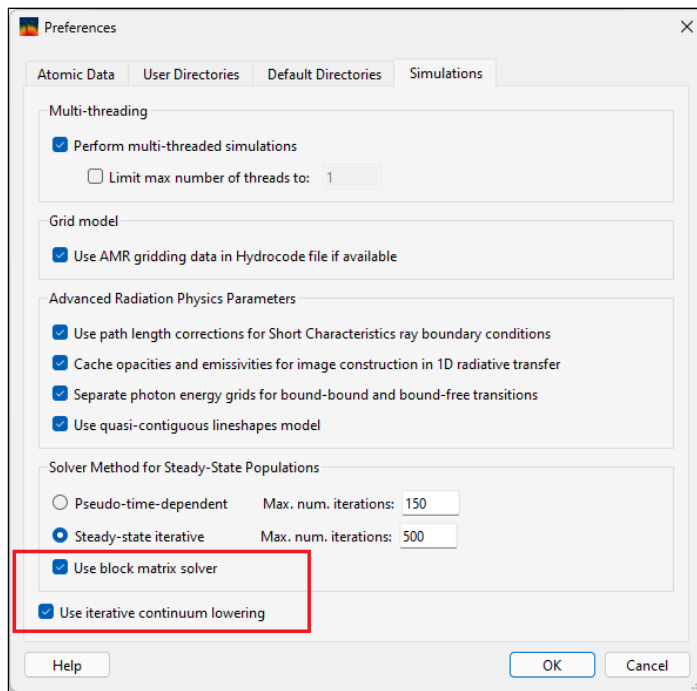
SPECT3D:

- Improved continuum lowering model where the electron density and the amount of IPD are determined self consistently with a combination of iteration and bisection techniques was implemented. This model can significantly affect ionization balance, especially for low temperature plasma. More details on the model can be found in an appendix of the main documentation.
- Improved consistency in modeling IPD for mixtures of elements.
- New block matrix population solver for large atomic systems has been implemented. For atomic models containing over 20000 levels, the new solver is expected to use significantly less memory, and be faster by at least a factor of few. More details on the model can be found in an appendix of the main documentation.
- Improved models for ion contribution for Stark broadening. Previously, the effect of ion contribution could have been significantly overestimated at lower densities.
- Added support for new distribution of FAC atomic data (v. 1.5.0U).
- Updates were made to better support graphics on high-resolution monitors.
- Several additional significant changes in physics modules have been implemented. For more information on the models please refer to an appendix in the documentation titled "Physics Models Improvements April 2024"
 - Photon energy grid refinement for both photo-absorption and spectral modeling.
 - Continuum lowering model avoids destroying transitions just below the ionization threshold.
 - Including more line wings in bound-bound emissivities.
 - Extrapolation of photo-ionization cross sections below depressed ionization potential.
 - Automatic adjustment of time steps for non-local time dependent population solver.
- The modeling for *Logically Rectangular* grids has been updated to provide improved computational speed and memory use, and extended to additional geometries.
 - To use the new modeling:
 - parameters specifying the number of volume elements in each dimension need to be provided in the hydro data file. (Contact Prism for information on required parameters.)
 - Because of the significant revisions to the modeling, old workspaces that specified using the *Logically Rectangular Grid* model for the *LOS-Grid Intersection Model* (see *Simulation Type* widget) are switched over to *Evaluate for Individual Elements* model (which supports unstructured grids).
 - Supported grid geometries are (previously only 2D cylindrical *r-z* geometry was supported):
 - 2D Cartesian grids [with nodes at $x(i,j), y(i,j)$]
 - 2D Cylindrical grids [with nodes at $r(i,j), z(i,j)$]
 - 3D Cartesian grids [with nodes at $x(i,j,k), y(i,j,k), z(i,j,k)$]
 - 3D Cylindrical grids [with nodes at $r(i,j,k), z(i,j,k), \phi(i,j,k)$]
 - 3D Spherical grids [with nodes at $r(i,j,k), \theta(i,j,k), \phi(i,j,k)$]
 - The modeling supports *SPECT3D* simulations with photoionization/photoexcitation processes.
- The modeling for *Physically Rectangular* grids has been added for 3D hydro grids.
 - To use the new modeling, parameters specifying the number of volume elements in each dimension need to be provided in the hydro data file. (Contact Prism for information on required parameters.)
 - Supported grid geometries are:
 - 2D Cartesian grids [with nodes at $x(i), y(j)$]
 - 2D Cylindrical grids [with nodes at $r(i), z(j)$]
 - 3D Cartesian grids [with nodes at $x(i), y(j), z(k)$]
 - 3D Cylindrical grids [with nodes at $r(i), z(j), \phi(k)$]
 - 3D Spherical grids [with nodes at $r(i), \theta(j), \phi(k)$]
 - The modeling supports *SPECT3D* simulations with photoionization/photoexcitation processes.
 - Testing has been performed on grids with up to 90 million volume elements on desktop PCs.
- The efficiency of calculations have been improved for the case when *Evaluate for Individual Elements* is selected for the *LOS-Grid Intersection Model*.
- A number of updates have been made related to the *Detector-Target Viewer*:
 - A toolbutton () to open the viewer has been added to *SPECT3D Main Window*.
 - The *Detector Components* list has been updated to organize the components (e.g., viewing volume, backlighter, x-ray scattering source) under each detector.
 - When displaying axes () in addition to the hydro grid coordinate systems, axes can now be displayed in the coordinate system of one of the detectors.
 - When generating *Hydro Grid Plot Objects*, the contour slice plane can now also be specified in the coordinate system of one of the detectors.
 - The unit normal for contour slices are no longer restricted to the *xy*-, *yz*, or *xz*-planes. An arbitrary unit normal can be specified.

- Properties of the *Detector Components* and *Hydro Grid Plot Objects* (e.g., color, visibility) are now saved the clicking on the *Update Graphics* button.



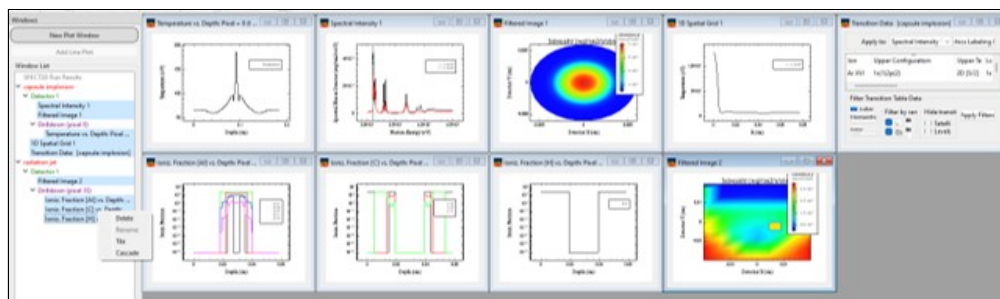
- Updates were made in 3D graphics displays of hydro data to reduce memory requirements.
- Updates were made that significantly improve efficiency of reading large data files, and in viewing data tables in plot widgets.
- Two improved models are now implemented, which can be turned off in Preferences if desired:
 - *Block matrix solver*: applies only if "Steady-state iterative" is selected for the populations solver. This is a custom designed linear equations solver grouping a larger number of atomic levels in a smaller number of blocks. Each block typically consists of one ionization stage. The solution to the original system of linear equations is obtained iteratively by solving the populations of blocks. Once the block populations are known, the smaller system of equations for individual levels in each block are solved one block at a time. The iteration continues until the solution of all levels and blocks reaches self-consistency.
 - *Iterative continuum lowering*: continuum lowering and level population are inter-dependent. Previous versions compute the continuum lowering parameters, such as occupation probabilities and ionization potential depression, using the LTE population assuming no continuum lowering. The Iterative procedure makes the continuum lowering and level populations self consistent.



- Plot axes now factor out common factors from tick labels and place said common factors at the end of their respective axes.
 - This common factor feature is on by default, but can be turned off by switching the tick label type notation option to decimal or scientific
- Bug fixes:
 - Fixed crash that occurred when displaying hydro data in 3D graphics. The crash could occur after selecting a volume element to display its hydro data.
 - In *Hydro Data Viewer* and *Detector-Target Viewer*, updates were made to fix issues for displaying grids utilizing supported *Symmetry Types*.
 - *Detector-Target Viewer*: fixed cases when node position units (specified on *Plasma* tab) are not in *cm*.

SpectraPLOT:

- Support for displaying pixel-dependent spectra has been added. When adding a plot widget for *Space-Resolved Spectrum*, an option to generate a contour plot or line plots (one for each selected pixel) is now available. For line plots, a list of pixel positions is provided. For each pixel selected, a line plot for the corresponding spectrum is displayed.
- The *Windows List* on the left side of the *Main Window* has been updated:
 - windows in the list are now grouped by *SPECT3D Run Name*, the *Detector*, and for *Drilldown Plots*, the pixel index.
 - the capability to arrange windows by tiling and cascading has been added (to do this, select windows in the list, and right-click to see options for tiling and cascading).



- Improvements were made to significantly reduce CPU time needed to generate *Space-Resolved Spectra*.
- When displaying transition gridlines and labels in a *Spectrum* plot, the graph widget is immediately updated when changes are made in the *Transitions Table* and the *Transition String Properties* dialog.
- Bug fixes:
 - Fixed bug getting the correct pixel index for drilldown when multiple detectors are used.
 - When changing the *Run Name* in the *SPECT3D Run Results List*:
 - a check is made to make sure the new run name is unique
 - the plot widget items and transition data widgets that were previously set up are updated with the modified run name.