MaxLLG User Guide

Ian Wellaway/Feodor Ogrin/Conor McKeever/Sam Morrell
Version 6.6.2
Last updated 3 Nov 2024

Contents

Summary

This document provides guidance on how to use the MaxLLG GUI and Solver.

To run the Solver on an object, a matrix file is required. This file is a collection of values that describe the three-dimensional structure of the object.

To create the matrix file there are two different approaches to using the GUI:

1. Using two-dimensional .png images created in photo editing software (E.g. Photoshop or Gimp)
   This approach is suitable for objects which have a linear third axis, such as cables.

2. Using .step files created in a CAD software application from three-dimensional objects (E.g. AutoCAD or FreeCAD)
   This method is suited for more complicated geometrical parts.

The Solver also relies on a host of other parameters which are passed to the solver in a separate file. These parameters can be amended on the Meta Script screen.

Once the matrix file has been created and the meta script values set, the Solver can be run. During its run the Solver creates 'out_' files that can be viewed as heatmaps in the output section.

The GUI uses locally stored stylesheets and jquery files. It also creates a sqlite database to store user information.

Installation

Prior to installation, please ensure that you have the following installed:
- Python 3
- Freecad & Python2
- Pip (E.g. apt install python3-pip)
- Venv (E.g. apt install python3.9-venv)

If you are intending to use the 3D CAD approach to creating matrix files, then the FreeCAD application must be installed. Get it from www.freecadweb.org.

FreeCAD requires Python version 2. This also needs to be installed.

Running setup.sh

Once the prerequisite packages have been installed, run the setup script to ready your system:

```
./setup.sh
```

Once run, the script will begin to setup the app on your system.

Specifying the Port number

The app requires a port to run locally and defaults to 5000. However, if this port is already in use, an alternative port number can be specified as an argument. E.g.:

```
./setup.sh 5001
```

User Input

As the setup script runs, there will be three requests for user input.

User Input 1

The first, almost immediately, is simply to 'Press any key to continue'

Python

The setup script then attempts to determine the most up to date python version, and the accompanying command.

User Input 2

The user is then asked for the python version for use with the FreeCad package. This is usually **python2**.

Virtual Environment

The setup file then creates a virtual environment within which to run the app.
The required packages are then installed into this environment.

FreeCAD

User Input 3

Once the python commands have been identified, setup will ask for the location of the freecad lib directory. Please supply this.
E.g. /usr/lib64/freecad/lib

Troubleshooting

If the port is already in use, the setup will fail at the very end. If this happens, simply use the start.sh script to restart the app as detailed in the ‘Restarting the GUI’ section below.

Restarting the GUI

Once the setup file has been run, there is no need to run it again. Instead, if the Flask server needs to be restarted, use the *start* script:

```
./start.sh
```

Again, the port can be specified as an argument:

```
./start.sh python 5001
```

NB. The main python command must also be sent as an argument (to start the GUI process). E.g. python or python3

The Home Page

The first page loaded by the app gives obvious links to the start the solving process:

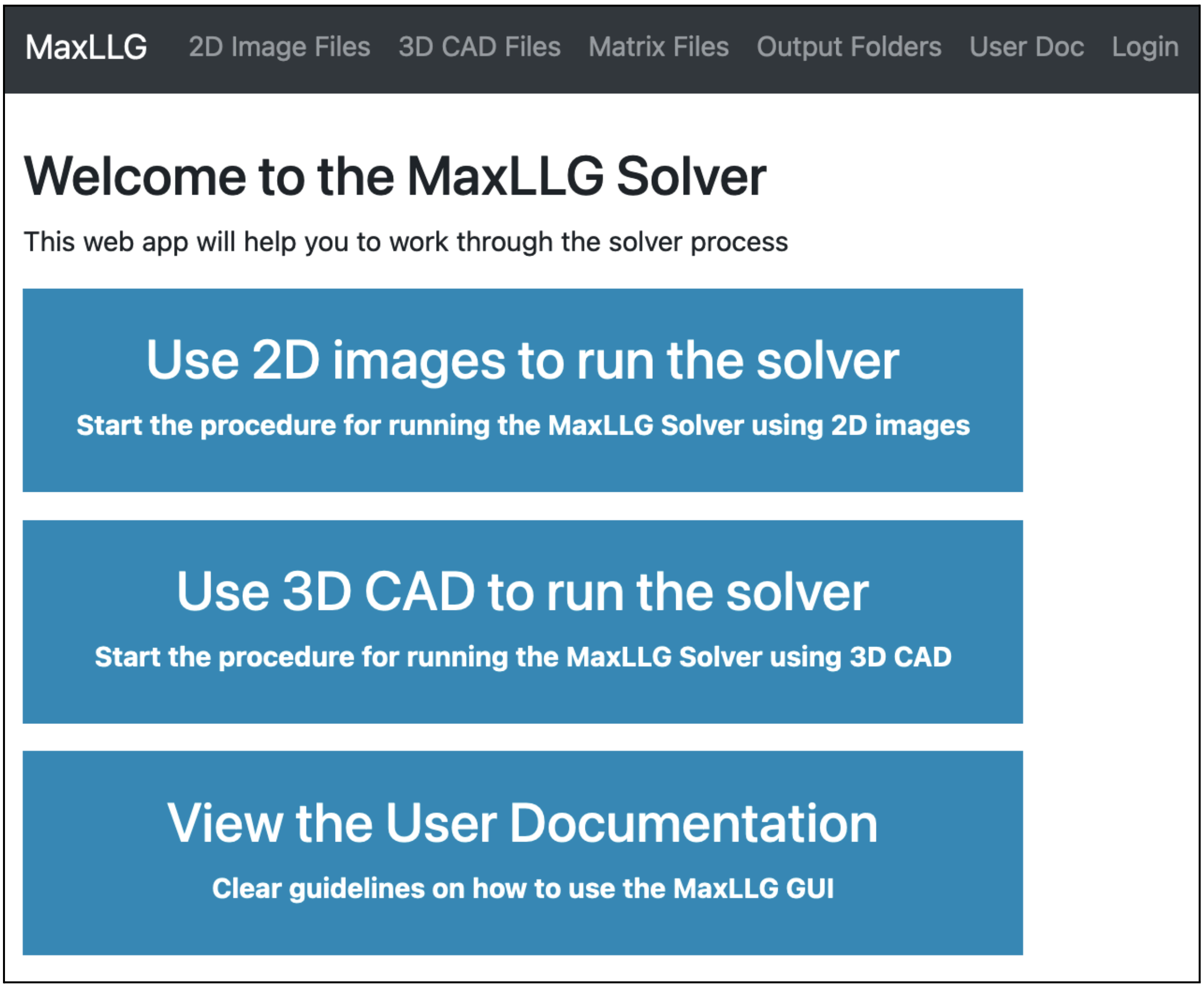


Figure 1: Home page

Logging in

When clicking on any other page of the GUI you will need to login. A user is automatically created during setup and can be used to proceed:

Username: admin
Password: ****Please ask****

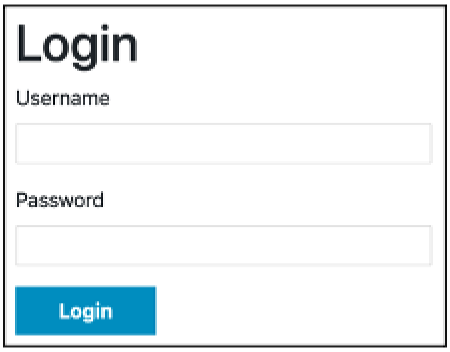


Figure 2: Logging in

Once logged in you will see ‘Logged in as <<*username*>>’ in the top navigation bar:

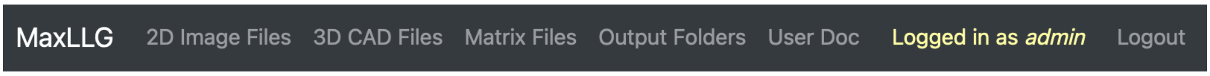


Figure 3: The nav bar showing the logged in user

Adding more users

To add further users, run the add_user.py file with the new username and password as parameters:

```
python add_user.py <<username>> <<password>>
```

Creating the matrix file

Using 2D images

Create an Image

Create an image that resembles the cross section of your part. The image below uses a coaxial cable as an example:

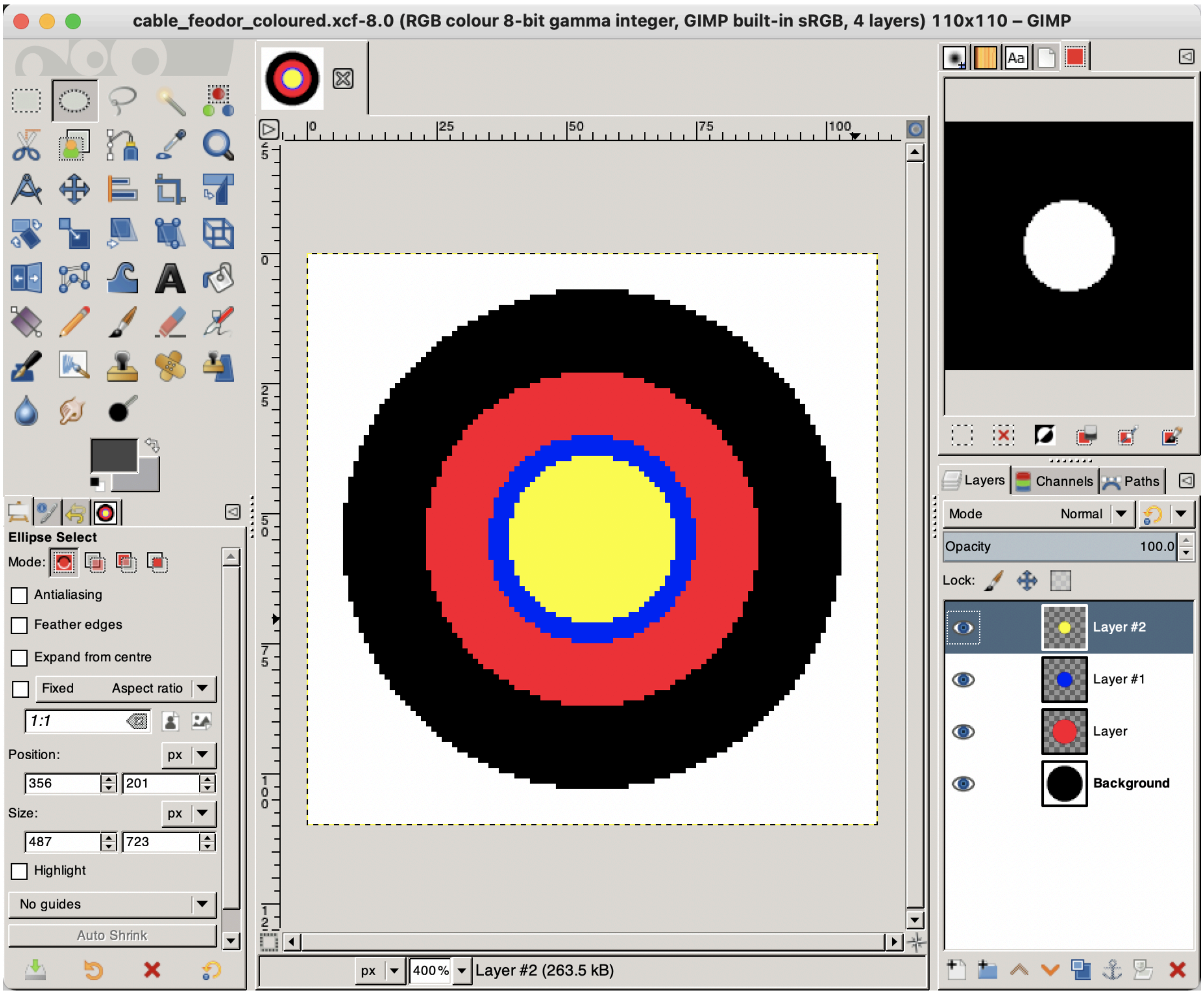


Figure 4: Creating a 2D image in GIMP

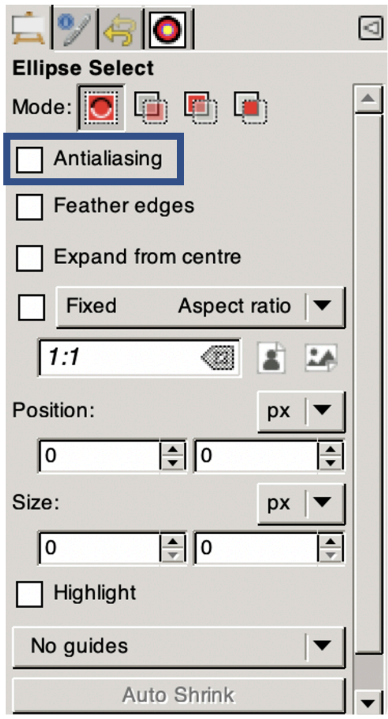


Figure 5: Untick the 'Antialiasing' checkbox in the 'Tool Options' menu

NB. Importantly, ensure the following:

- **Antialiasing is TURNED OFF** so the colours do not merge into one another.
- The area around the part is coloured white. This is so a 3D image can be rendered later.

The drawing above uses a canvas of 110x110 pixels.

On the '2D Image Files' page, load the image into the GUI by clicking on the 'Browse' button.

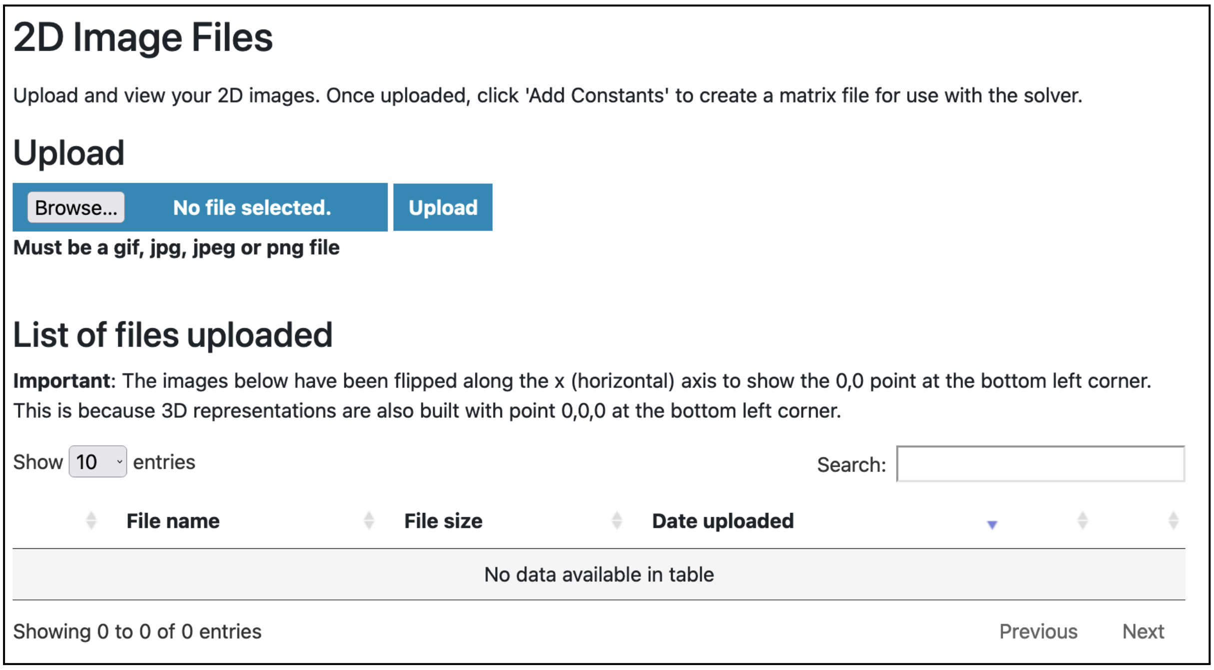


Figure 6: The empty '2D Image Files' screen

Navigate to your image and click 'Open'.

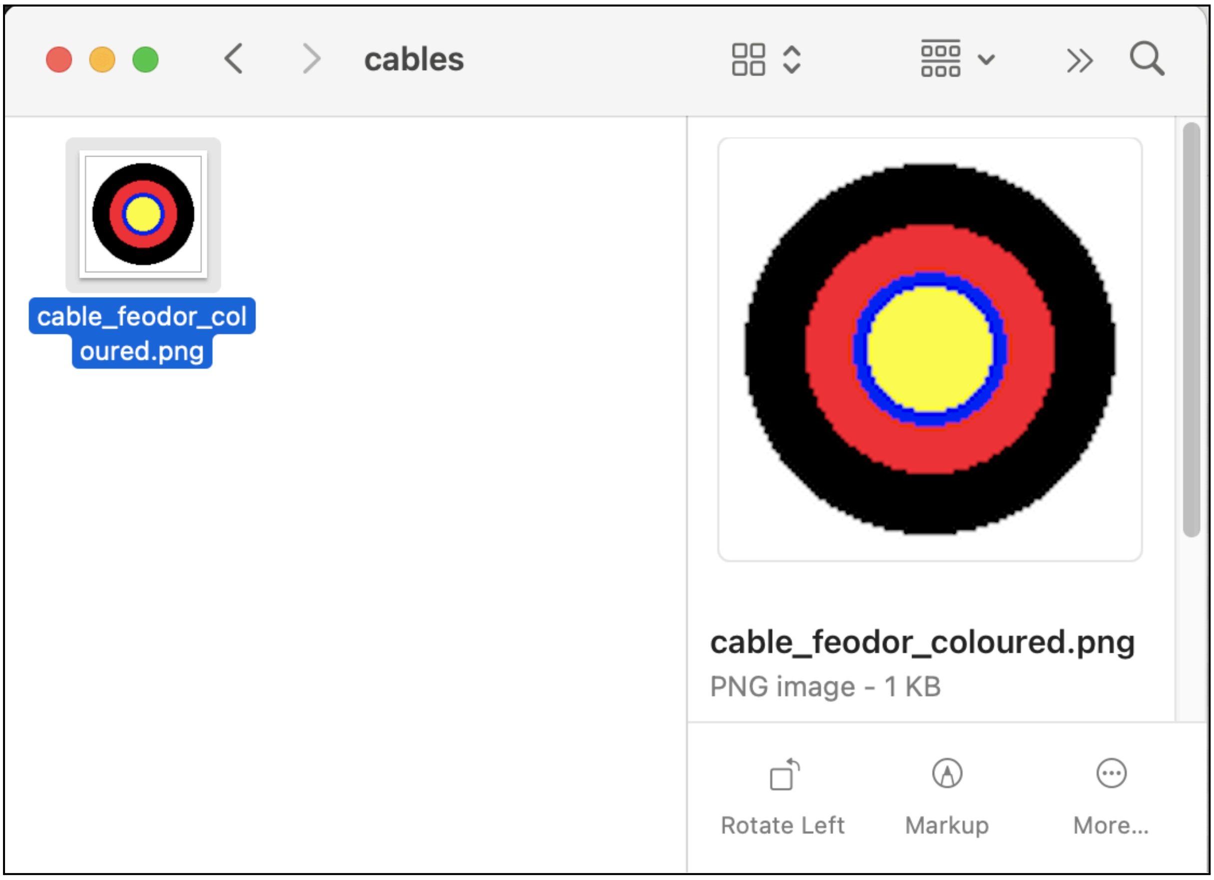


Figure 7: The 'File Upload' dialog box

On the original ‘2D Image Files’ screen, the path to your file will now show next to the ‘Browse’.

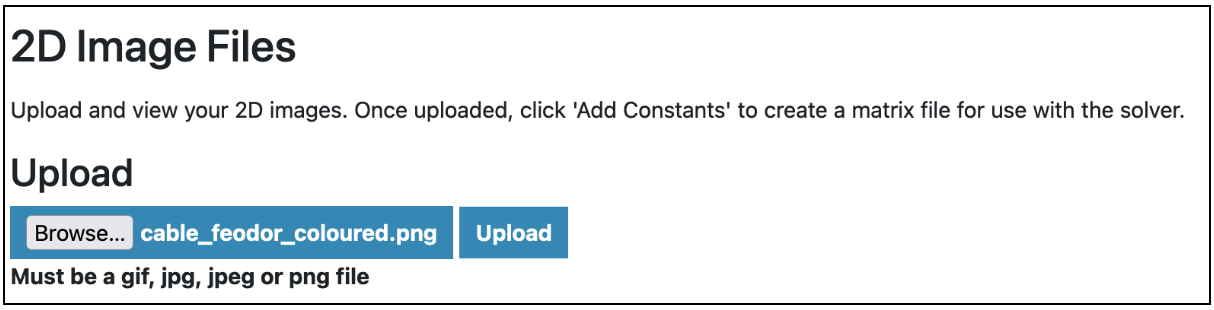


Figure 8: Upload message

Click the ‘Upload’ button. This will load your image into the GUI:

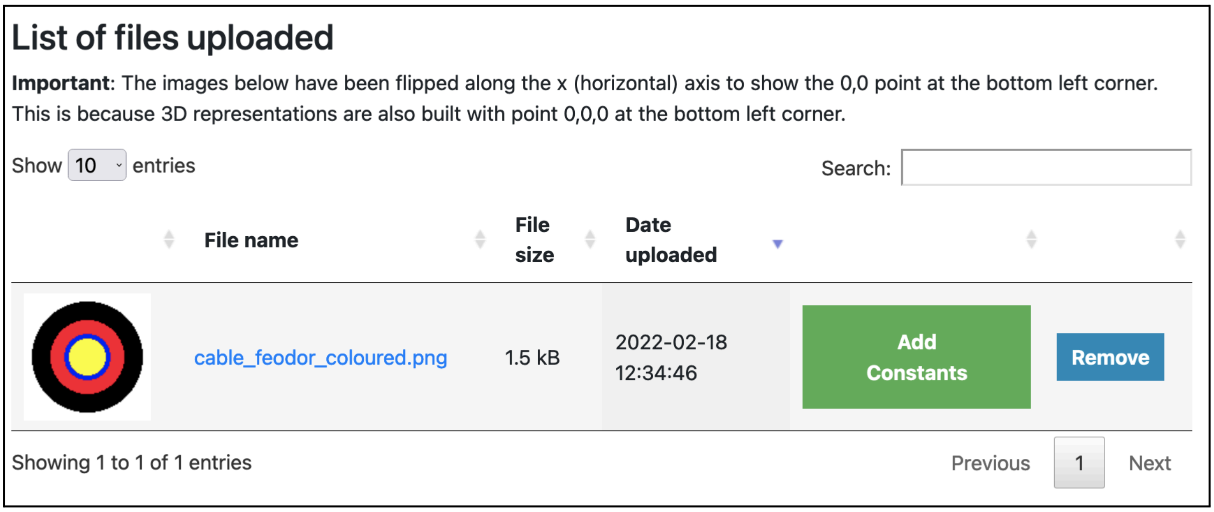


Figure 9: Uploaded 2D image file

Add Parameters

To create your matrix file, click the ‘Add Constants’ link:

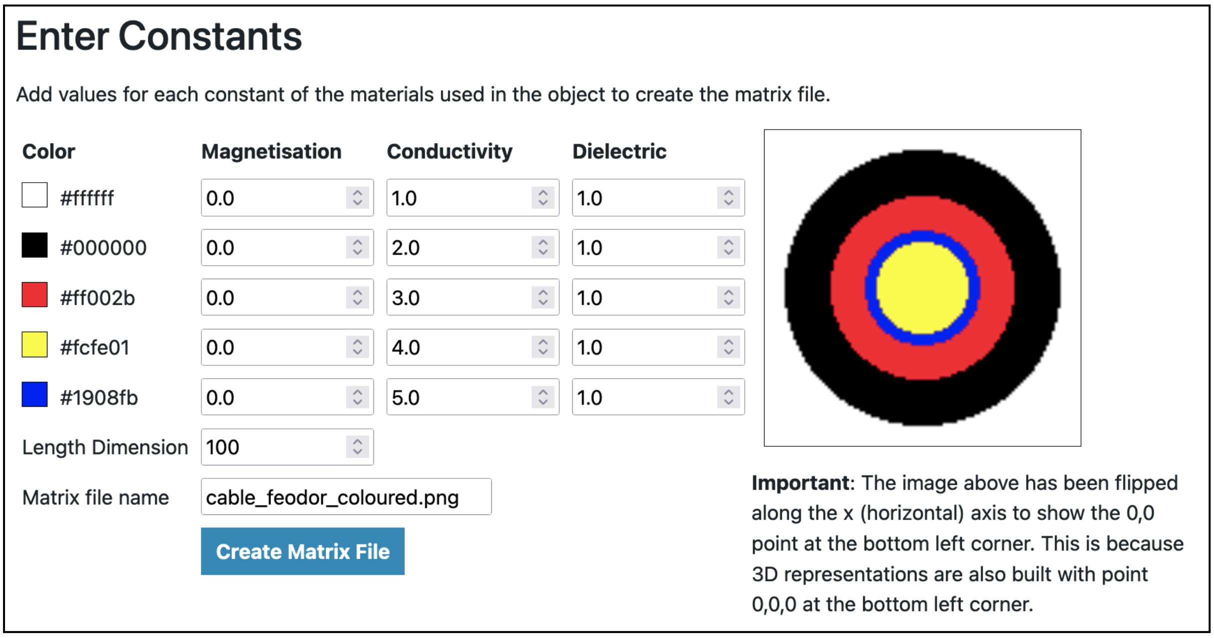


Figure 10: Add parameters

Choose values for Magnetisation, Conductivity and Dielectric.
The screen will remember any values entered previously *for this image*.

Length Dimension

The 'Length Dimension' is the depth of the part. For a cable, this would be the length of the cable.
This value essentially gives the 2D drawing a third dimension and allows the GUI to create a 3D representation of the object.

Matrix file name

The name of the matrix file that will be created can also be set here. It can be different than the actual image filename.

When ready, click the 'Create Matrix File' button.

This will combine the data in the image and the parameters to create the matrix file for the Solver. The browser will be redirected to the 'Matrix Files' page. This should only take a few seconds.

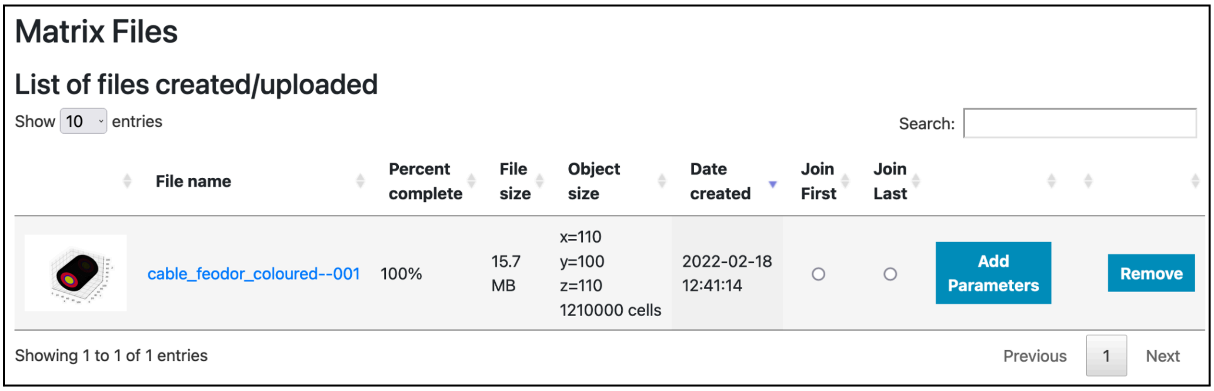


Figure 11: The 'Matrix Files' page

Naming Convention

Since the same png can be used to create multiple matrix files, a three-digit number is appended to the end of the filename:
E.g. cable_feodor_coloured--001 (see above)

Dimensions

The dimensions of each matrix file are shown to give the user a clear indication of the size of the object.

3D representation of png files

Once the matrix file has been created, the GUI will display the thumbnail of the original png image. However, the GUI will also begin to construct a graphical 3D representation of the part using the x and y dimensions of the png and the length dimension supplied on the 'Add parameters' page. This may take a few minutes depending on the size of the matrix file. When finished, the GUI will display the 3D version instead of the 2D png (as in the Figure above).

The thumbnail can be clicked on to open a new window showing a larger version of the 3D image and 2D views from each different axis:

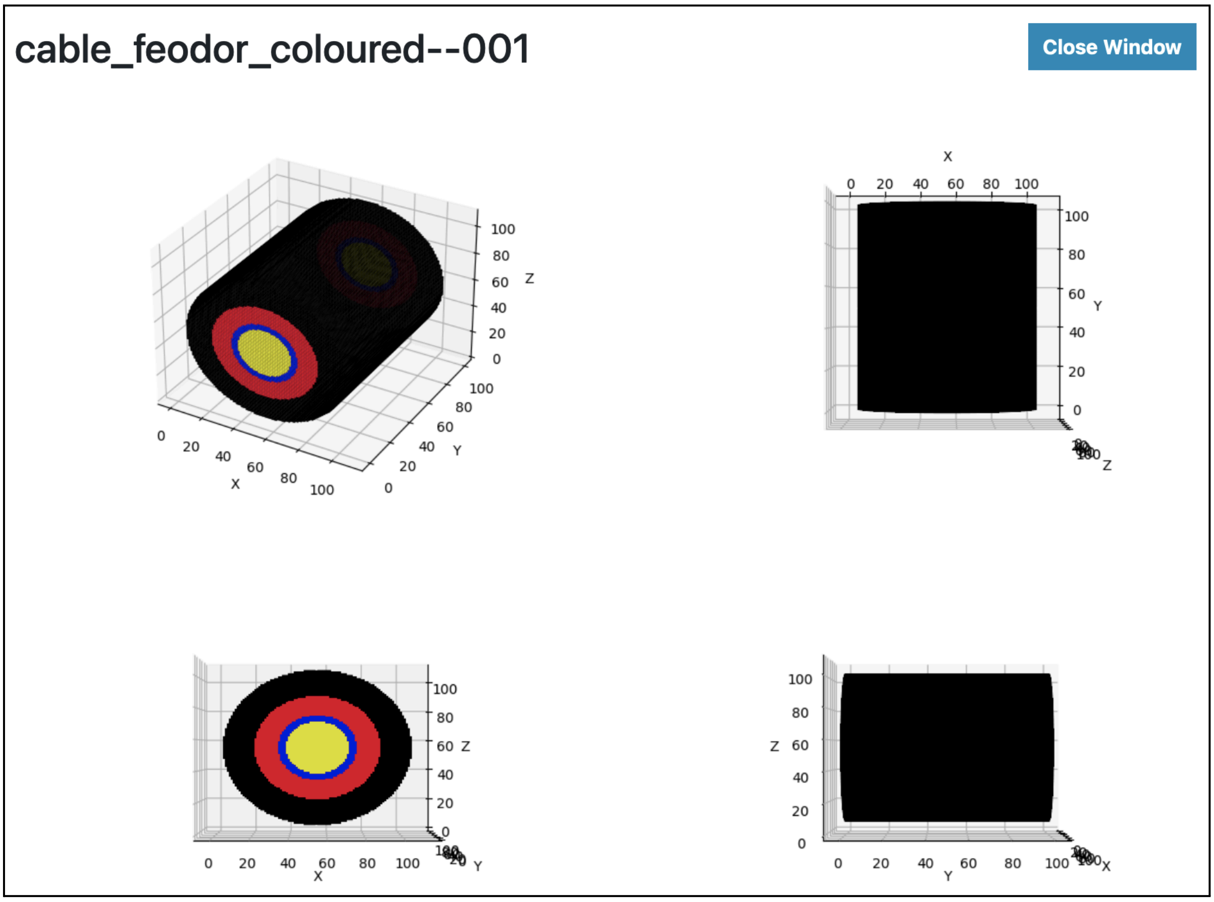


Figure 12: 3D representation of the 2D png file

NB. These images are created in sequence so there may be a small time lag between each displaying.

To create the image, the GUI uses the values for Magnetivity, Conductivity and Dielectric and assigns the same colour to each unique set, based on the colours used in the original png image.

NB. In the image above, some of the layers of the coaxial cable were assigned the same parameters so share the same colour.

2D matrix file creation is now complete.

Using 3D CAD files

The MaxLLG GUI uses the power of the FreeCAD application to convert 3D CAD images into input_files for use by the Solver.

License information for FreeCAD can be found here.

Create the 3D CAD image

To create the 3D image (also known as a 'part') in FreeCAD, or another CAD application. Any application can be used but the part must be exported as a .step file.

Also, if the part is made up of multiple sub-parts, then each part must be exported as a separate .step file.

The figure below shows a spherical part in FreeCAD with a radius of 50mm.

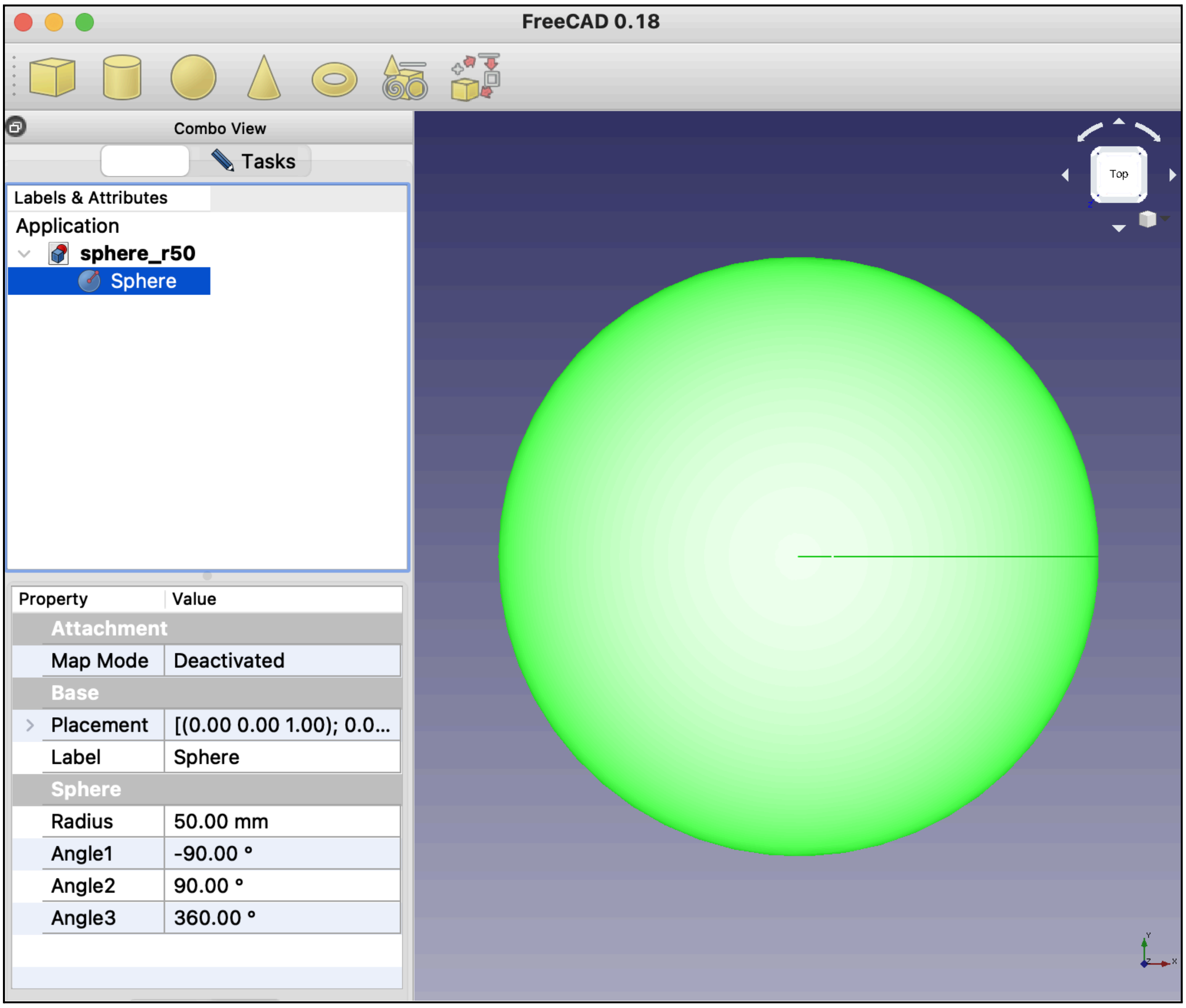


Figure 13: Creating a sphere in FreeCAD

This spherical part can be created in FreeCAD in the following way:

I. In the 'View' menu, open the 'Workbench' section and choose the 'part' workbench
II. Also in the 'View' menu, choose the 'Toolbars' section and open the 'Solids' toolbar.
III. This will display the different shapes that can be easily created in FreeCAD (see the top left section of the Figure above).
IV. Click on the sphere icon.
V. Draw a sphere on the main canvas area of the FreeCAD app (where the sphere is in the Figure above).
VI. Once drawn, many parameters for the part can be set by clicking of the word 'sphere' in the left-hand section and choosing 'Data' in the lower 'Property View' section. In particular, the radius for the sphere can be set here.
VII. Once any parameters have been set, the part can be exported in the 'File > Export' menu:

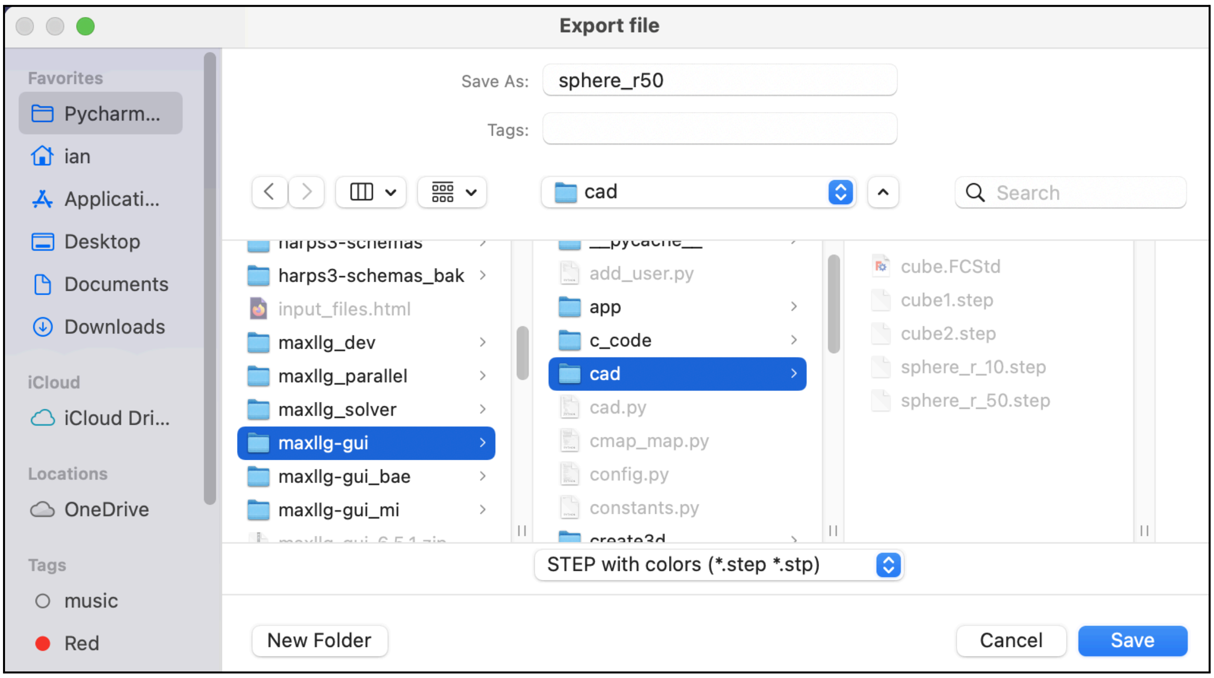


Figure 14: FreeCAD dialog box to save part as .step file

For a part made up of multiple sub parts (or different materials), select each part in turn in the left-hand menu and use the 'File > Export' option.

The 3D CAD File

The GUI needs to know the location of the step files and the values for Magnetivity, Conductivity and Dielectric (as in the 2D process).

Once all parts have been exported, open the GUI and click on the '3D CAD Files' menu option:

Figure 15: The ‘3D CAD files’ page showing the list of data files

Create a new data file by clicking the ‘Create new’ button beneath the header.

The dialog box opens as below:

Matrix File

Enter the name of the matrix file which will be created from the step files.

Matrix file name: input_sphere_50r

NB. If editing the file name, changing the name will create a new CAD file

Fast method? ☑

Tick this box if your CAD image has been split into separate step files.
This will make the matrix file generation run faster.

Untick this box if you have created a single step file containing multiple objects.
This approach requires further calculations to create an accurate matrix file.

Step Files and Constants

Step File 1 Details

Enter the location of the first step file and the constants.

Filename: [ ] Clear

Choose new: Browse... sphere_r_50.step Clear

NB. File extension must be .step

Magnetisation: 0.0

Conductivity: 5000000.0

Dielectric: 0.0

Default Constants

These constants are for the area around the part(s).
E.g. trunking or just air.

Magnetisation:

Conductivity:

Dielectric:

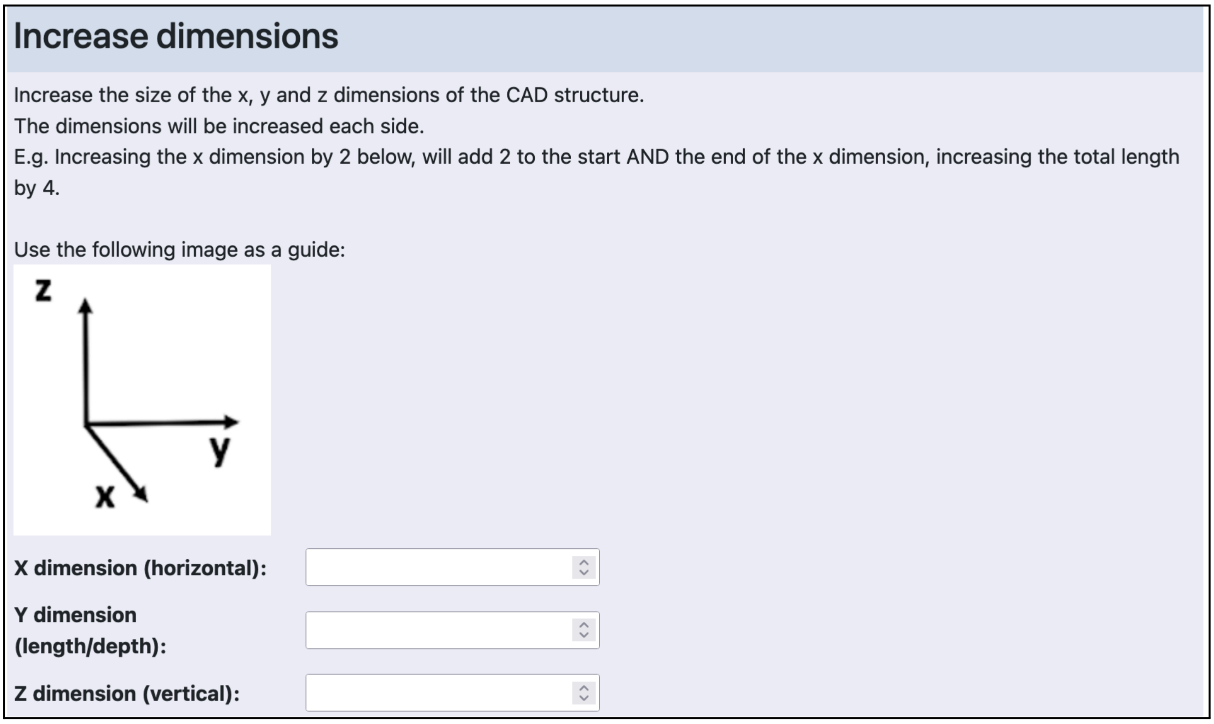


Figure 16: The Create/edit 3D CAD file dialog box

Complete the required parameters:

Matrix file name (shown above)

Choose a suitable name for your matrix file. This will also be the name of the output directory that the files created by the solver will be stored in.

Fast method? (shown above):

Ticking this box (which is ticked by default) greatly increases the speed of the matrix file creation.

If a single object is being used (E.g. a sphere) or multiple objects that have been split into separate step files (E.g. three spheres saved as separate step files) then keep this box ticked.

If multiple step files are saved as a single step file then this needs further computation so untick this box.

Step file 1 details (shown above):

Filename:

Click ‘Browse’ and navigate to your step file.

Magnetisation:

Enter the value of magnetisation of the object in Tesla (e.g. 0.1 will correspond to 0.1 Tesla or 1KG)

Conductivity

Value of the conductivity in [siemens/m] (e.g. ~1e7 for Iron).

Dielectric:
Relative dielectric constant (1 for vacuum). **NOTE** this should not be zero, unless the conductivity is high.

Default constants (shown above):
Constants for any section of the CAD image which does not have any material.
E.g. the area outside of the sphere. A default (vacuum) setting would be 0.0 0.0 1.0.

Increase Dimensions (shown above):
The size of the 3D structure can be increased along the x, y and z axis separately. Each dimension will be extended at both ends.
E.g. Entering 2 in the X axis box will increase the x dimension by 2 cells at each end, making a total extension of 4. When PML or PBC boundary conditions are used, this extension should at least be of the same or larger size.

Step file 2/3/4/5 details (not shown above):
For 3D parts with multiple sub parts. These boxes should be completed as the step file 1 boxes.

Once all of the parameters have been entered, click 'Save'

Create the matrix file

Your new data file will appear at the top of the list on the '3D Image Files' page.

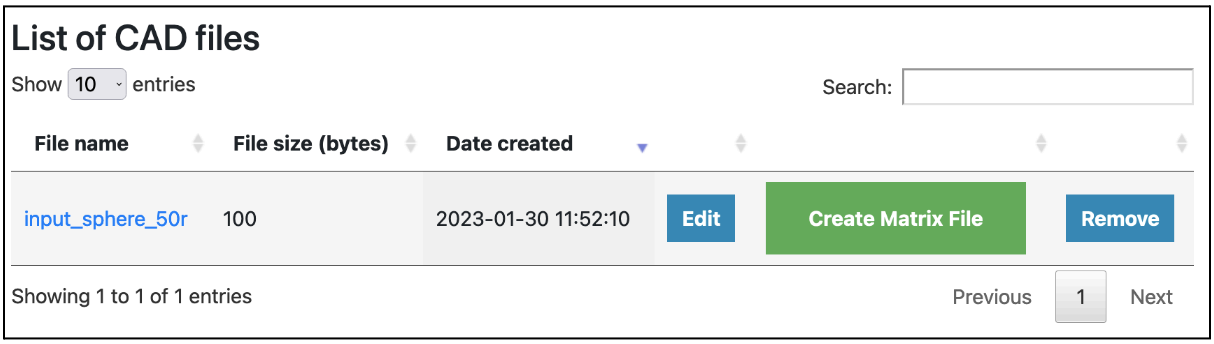


Figure 17: Showing the new file data has been created

To create the matrix file, click the 'Make matrix file' link.

Matrix file creation and progress

The GUI will be redirected to the 'MatrixFiles' page where a message will inform you that the matrix file creation has begun:

Matrix Files

- **Input file creation started - refresh the screen to see updates in the list below.**

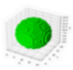
Figure 18: Message on the Matrix Files' screen once an matrix file creation has begun

The new input_file will begin to be created. The progress can be seen in the file list:

Matrix Files

List of files created/uploaded

Show 10 entries Search:

	File name	Percent complete	File size	Object size	Date created	Join First	Join Last		
	input_sphere_50r--001	19%			2023-01-30 12:26:28				Stop
	input_sphere_10r--001	100%	121.0 kB	x=20 y=20 z=20 8000 cells	2023-01-30 12:15:31	○	○	Add Parameters	Remove

Showing 1 to 2 of 2 entries Previous 1 Next

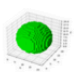
Figure 19: The Matrix File list shows the progress of the new matrix file creation. Here it is currently at 19%

Creating a matrix file from a large step file (E.g. a grid size of 100x100x100 pixels will take several minutes. During this time it is possible to stop the process using the red 'Stop' button.

Once created, the completed matrix file will show at the top of the list:

List of files created/uploaded

Show 10 entries Search:

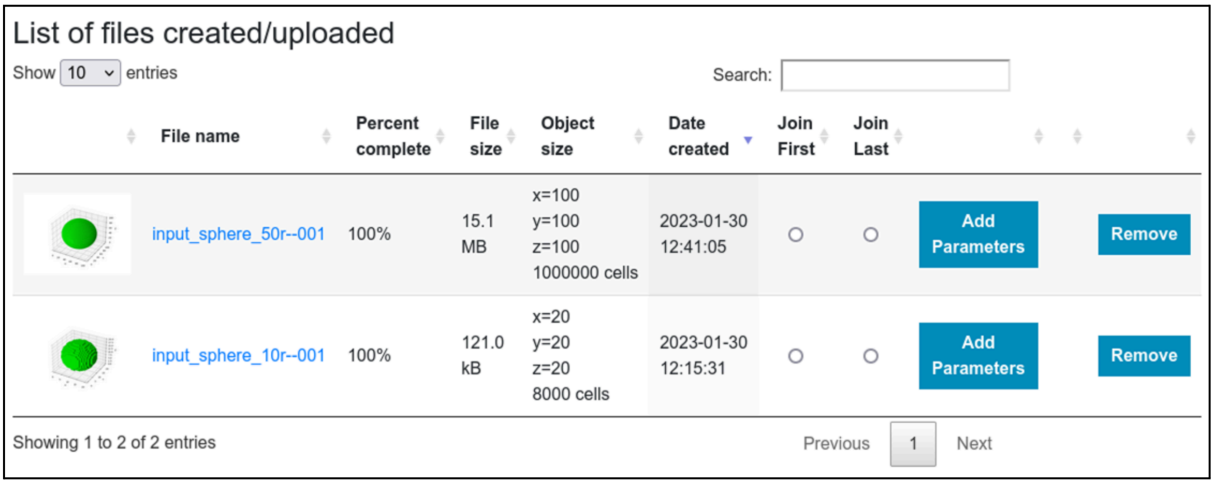
	File name	Percent complete	File size	Object size	Date created	Join First	Join Last		
Image being created Stop	input_sphere_50r--001	100%	15.1 MB	x=100 y=100 z=100 1000000 cells	2023-01-30 12:41:05	○	○	Add Parameters	Remove
	input_sphere_10r--001	100%	121.0 kB	x=20 y=20 z=20 8000 cells	2023-01-30 12:15:31	○	○	Add Parameters	Remove

Showing 1 to 2 of 2 entries Previous 1 Next

Figure 20: The completed matrix file is shown

Once the matrix file has been created, the GUI will build a 3D representation. For large files, this can take several minutes but can be stopped by clicking on the ‘Stop’ link.

Once the 3D representation has been built, it will display as a thumbnail next to the matrix file:



List of files created/uploaded

Show 10 entries   Search:

	File name	Percent complete	File size	Object size	Date created	Join First	Join Last		
	input_sphere_50r--001	100%	15.1 MB	x=100 y=100 z=100 1000000 cells	2023-01-30 12:41:05	○	○	Add Parameters	Remove
	input_sphere_10r--001	100%	121.0 kB	x=20 y=20 z=20 8000 cells	2023-01-30 12:15:31	○	○	Add Parameters	Remove

Showing 1 to 2 of 2 entries   Previous 1 Next

Figure 21: The completed matrix file is shown, along with the 3D representation

Click on the thumbnail image to view the 3D representation of the matrix file:

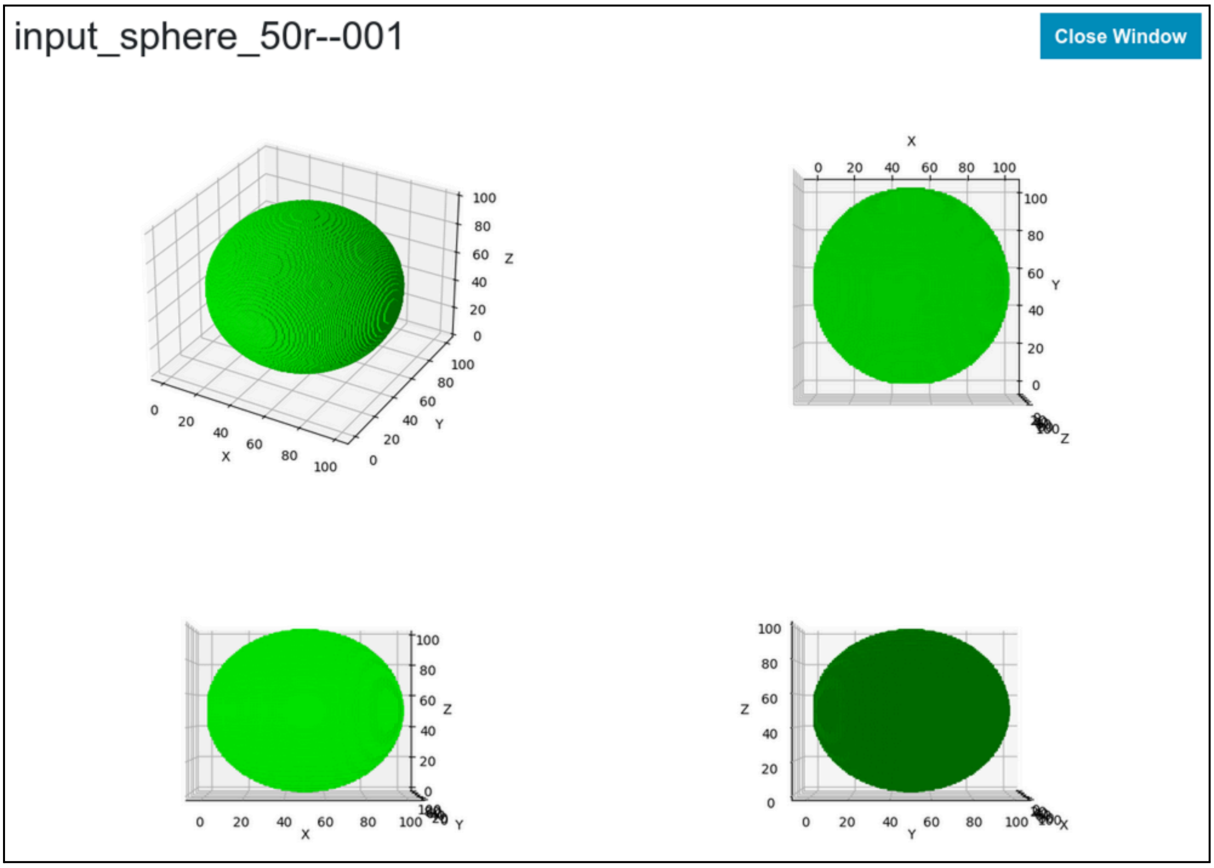


Figure 22: 3D representation of the .step file

3D Matrix file creation is now complete.

Joining matrix files

Any two matrix files can be joined to create a larger file, essentially increasing the size of the 3D part it is based upon. This is particularly useful when using 2D images as they can be joined together to create a non-linear 3D structure.

To join two matrix files, the files must have the same length x and z axis. For example, an image of 50x40x50 has the same x and z axis as a second image of size 50x20x50. The y axis can be any length as it represents the depth of the 3D image which is the axis that the two images are joined on.

E.g.:

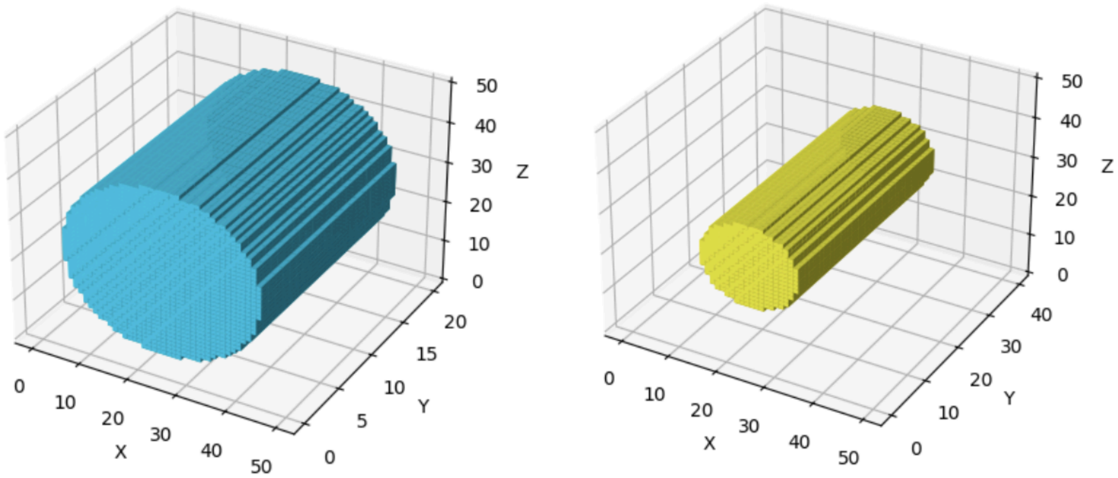


Figure 23: 3D representations of two matrix files with the same length x and z axis that can be joined.

NB. In each image, the area surrounding the part must have the same parameters.
E.g. When creating the matrix files for the two parts above, the white area surrounding the actual part is given the same parameters in each:

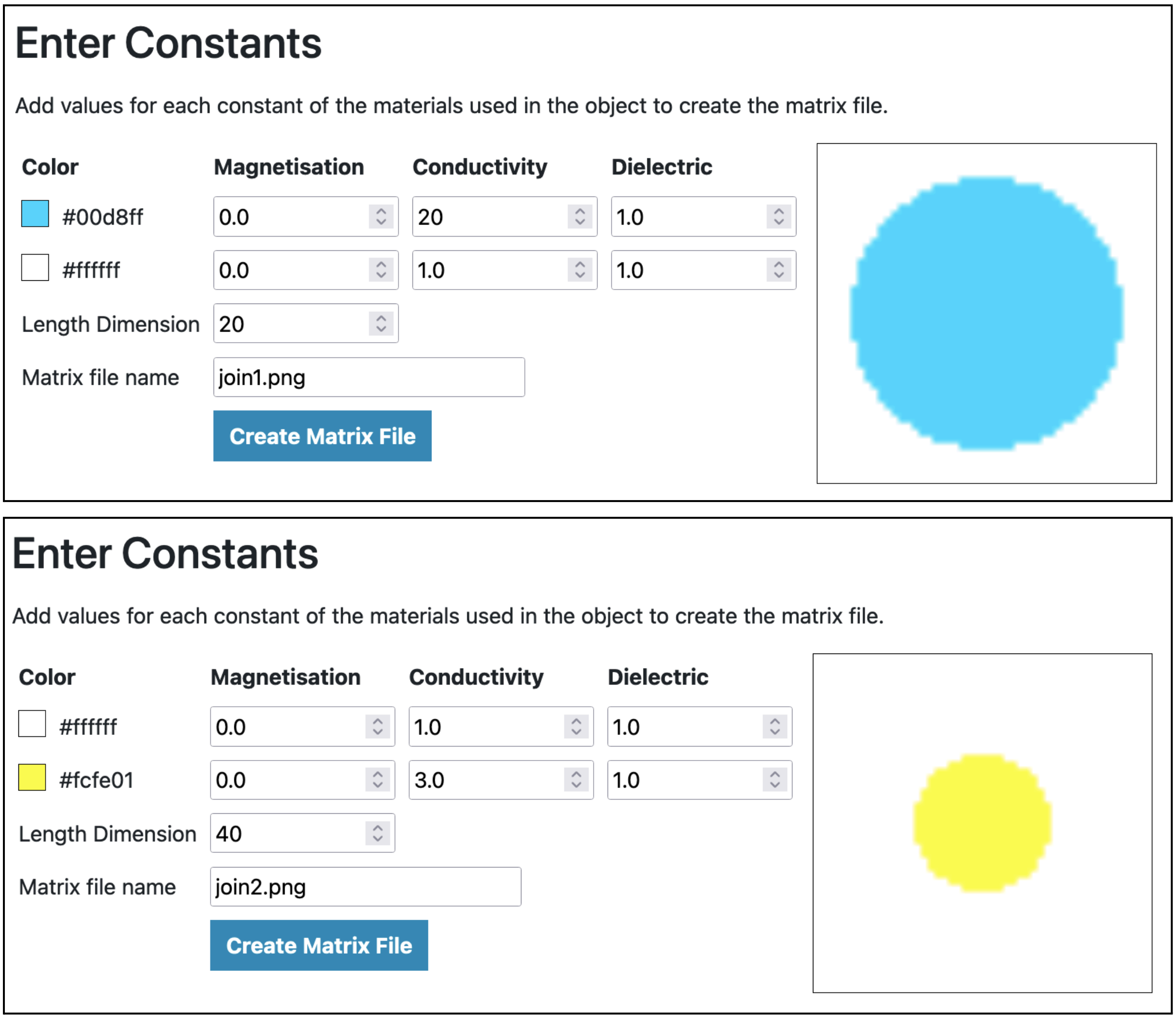


Figure 24: Set the same parameters for the white surround before joining

To join two images, select the ‘Join First’ and ‘Join Second’ radio buttons next to the desired images on the ‘Matrix Files’ page:

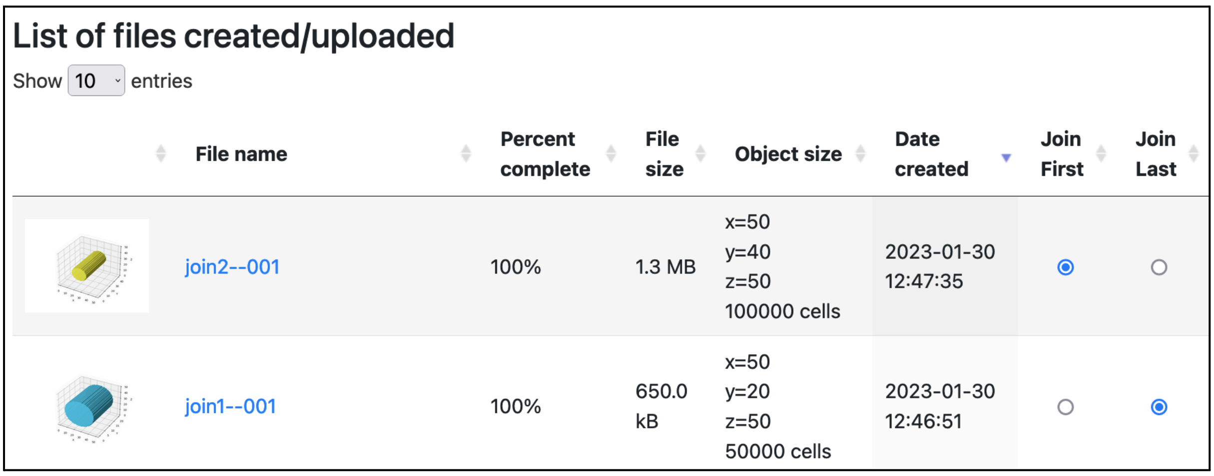


Figure 25: Selecting the matrix files to join using the ‘Join First’ and ‘Join Second’ radio buttons

Finally, click the ‘Join Files’ button at the bottom of the page:

NB. A name for the new file must be chosen.

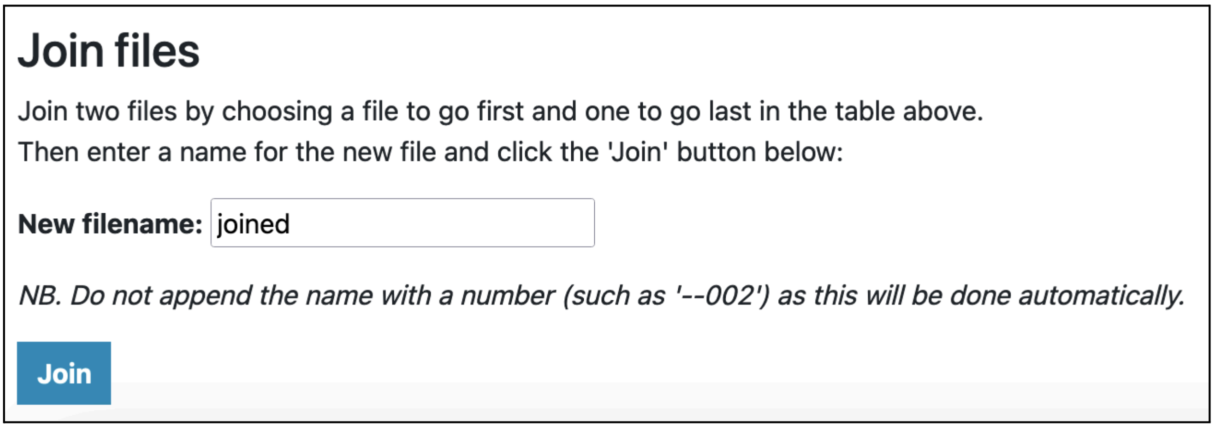
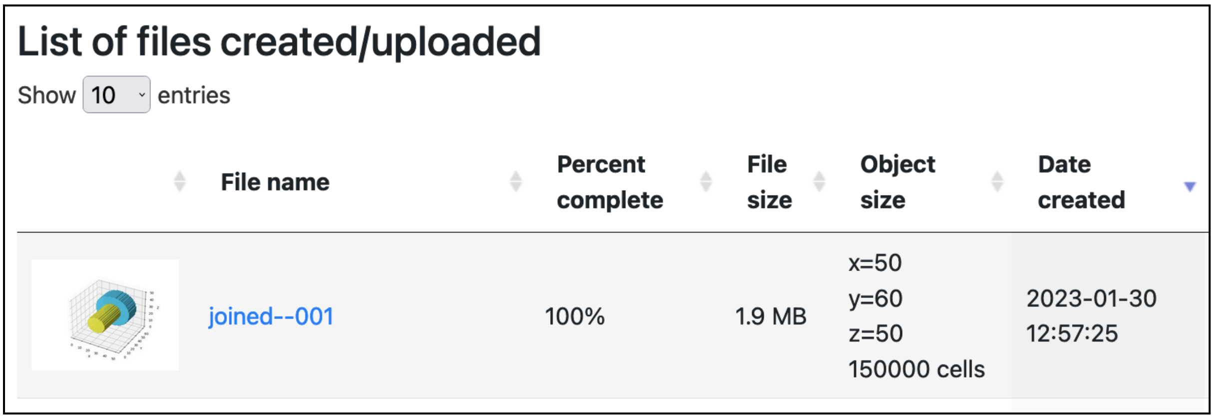


Figure 26: The ‘Join files’ section

A new matrix file named ‘joined—001’ will be created (subsequent joined files will increment E.g. joined—002, joined—003, etc):



List of files created/uploaded

Show 10 entries

	File name	Percent complete	File size	Object size	Date created
	joined--001	100%	1.9 MB	x=50 y=60 z=50 150000 cells	2023-01-30 12:57:25

Figure 27: The new joined object on the ‘Matrix Files’ page

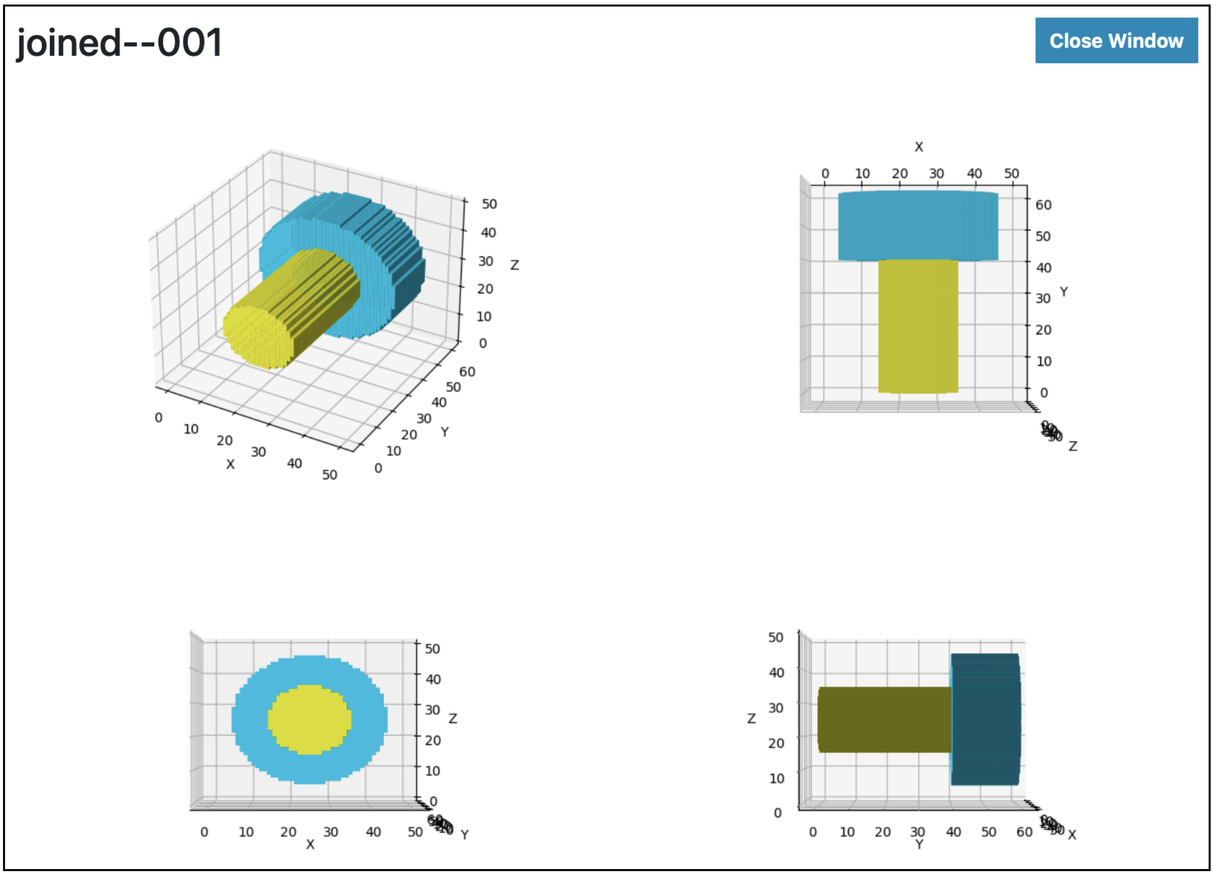


Figure 28: A 3D representation of the new joined object

As with other 3D representations, this image may take a few seconds/minutes to build. It will appear on the matrix files page when finished (the page will need to be refreshed).

Setting the metascript parameters

The Solver also requires a number of extra parameters in order to run. These parameters are stored in a file (also referred to as the ‘metascript’ file) and passed to the Solver with the matrix file.

The GUI is packaged with the metascript parameters already set to default values. However, they can be changed on the ‘Parameters’ page:

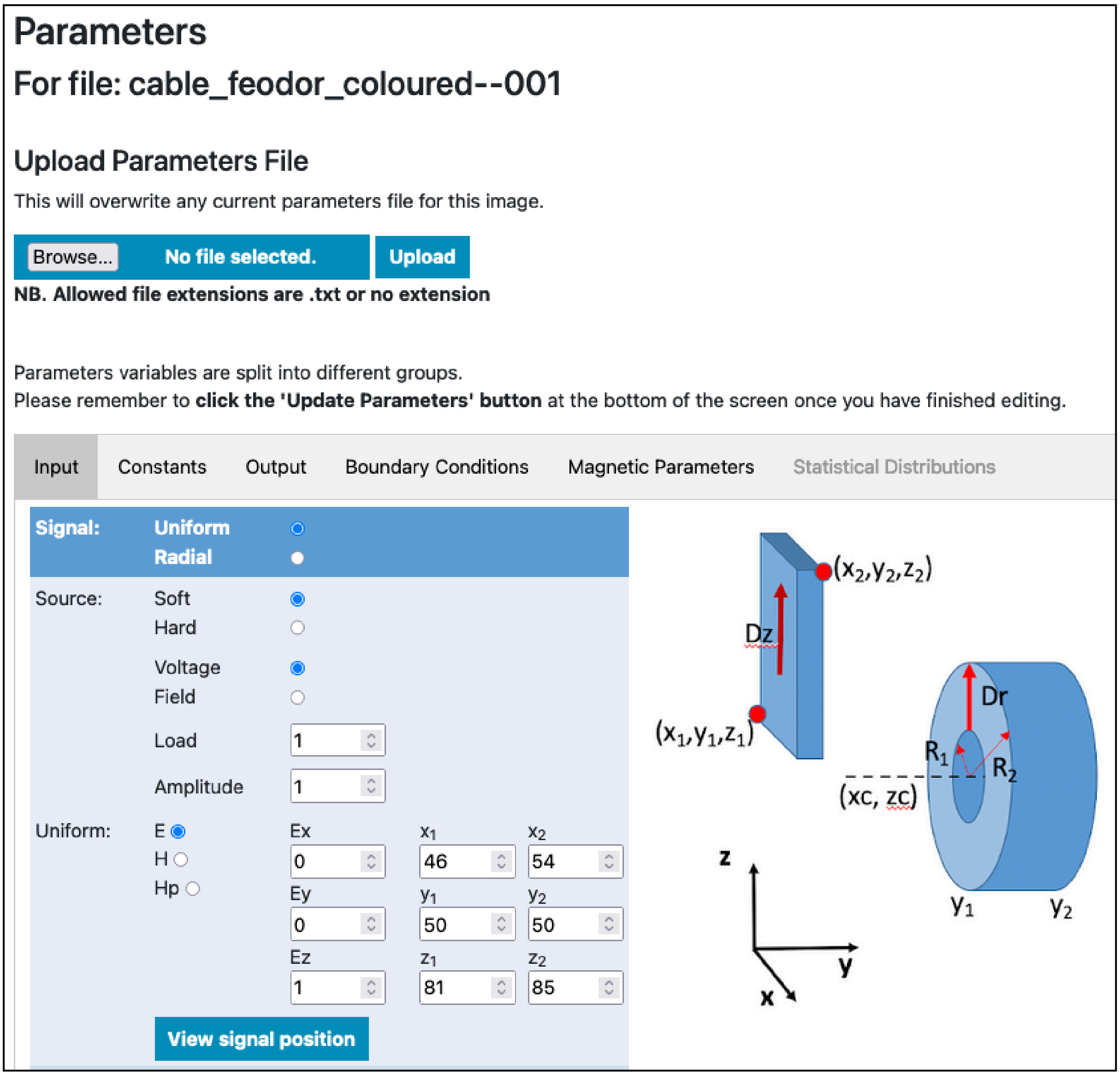


Figure 29: The ‘Parameters’ page showing the ‘Uniform Signal’ section.

NB. The parameters file must be created for a matrix file before the solver is run. The ‘Solve’ link on the matrix files page will only be displayed once the parameters file has been created for the matrix file.

Each matrix file has its own parameters file.

Uploading a parameters file

A pre-existing parameters file, perhaps that has been created before for another matrix file, can be uploaded using the section at the top of the page:

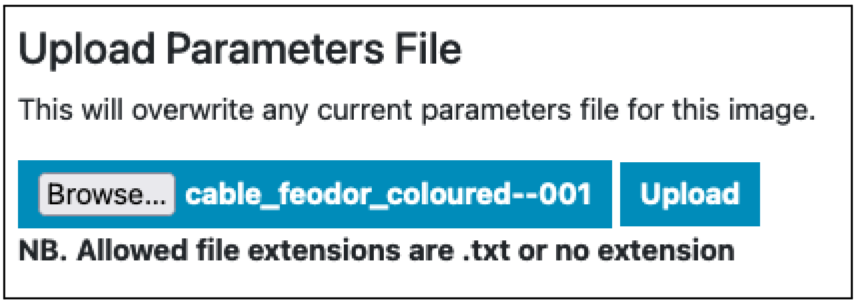


Figure 30: Uploading a parameters file

The parameters page is split into several separate tabs:

- Input
- Constants
- Output
- Boundary Conditions
- Magnetic Parameters
- Statistical Distributions

IMPORTANT!
The variables in the final ‘Statistical Distributions’ pane are not currently used by the solver.
However, they are described below to be made ready for implementation in a later versions.

Input

The input page is dynamic and offers different options based on the signal selection and signal shape selection. The image on the right of the page gives some guidance to these parameters.

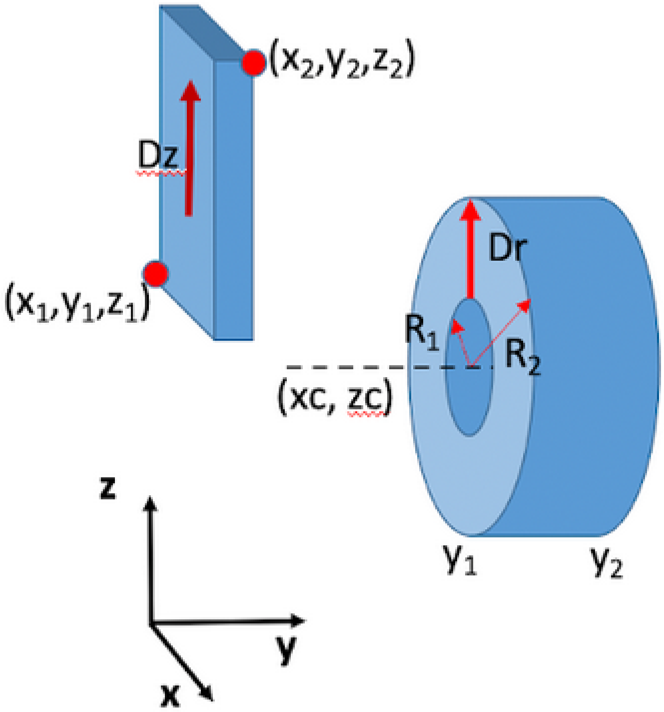


Figure 31: The Input pane: Visual explanation of the Uniform and Radial Signal

They are also detailed below, including their effect on the Solver:

Signal

Signal input can be chosen as either a uniform or radial (as shown in the Figure above).

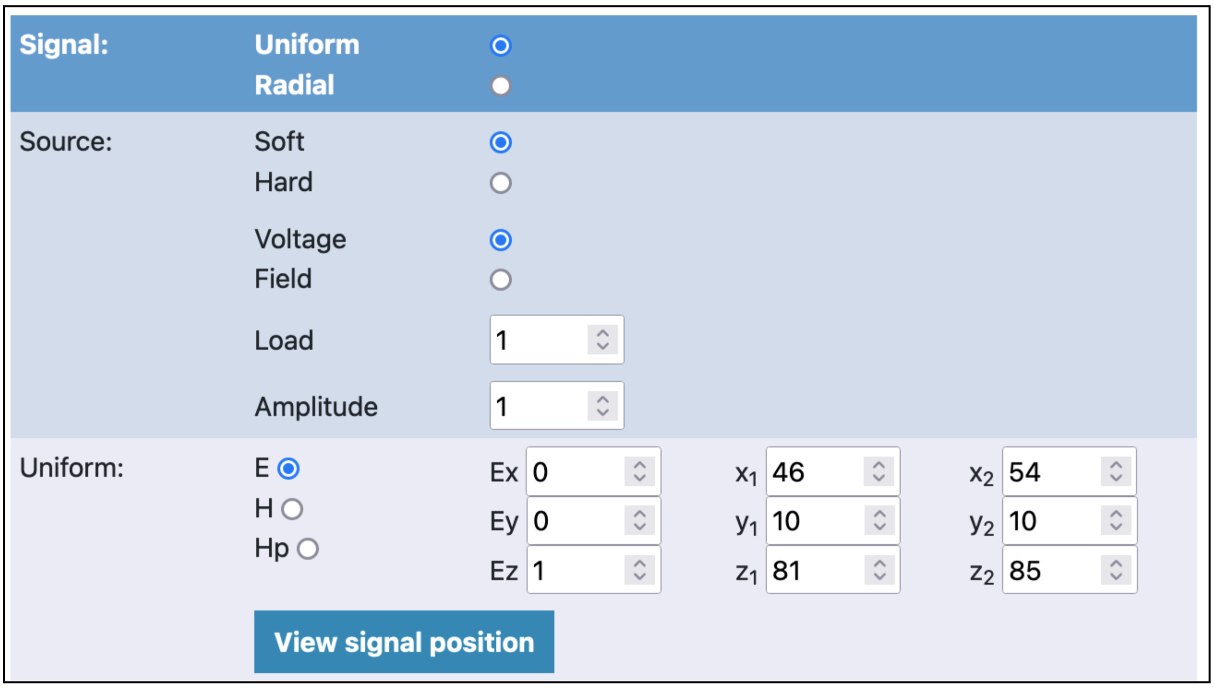


Figure 32: The Input pane: Uniform section

Uniform

Uniform signal with the same values of the chosen field (E,H or Hp) across the cuboid defined between the diagonal coordinates (x_1,y_1,z_1 and x_2,y_2,z_2).

Soft/Hard
Hard source imposes the signal by overwriting the present number with that in the signal. Soft source adds the value in the source to that currently present in the cell. NOTE! If hard source is chosen, the source cells will be hard reflectors for any waves propagating towards the source.

Voltage/Field
Choose Field for sourcing the fields (E,H,Hp) within the source volume. The units are respectively, V/m and A/m. The Voltage selection implies the use of electric field, but with a particular difference of potentials. This option is normally used for transmission lines or antennas, where voltage is supplied to metallic conductors.

Load
Is used with the Voltage source, providing the resistance of the internal voltage source (e.g. 50 Ohm).

Amplitude
Numerical value of the sourcing field (or voltage). The units will be used according to the chosen field.

E, H & Hp (radio buttons)
Choice of the fields

Ex, Ey & Ez
Directional cosines of the field E (e.g. Ex=cos(θ), where θ is the angle between the vector E and x-axis. The magnitude of the vector is defined by the value in the **Amplitude**. $Ex^2 + Ey^2 + Ez^2 = 1$)

Hx, Hy & Hz
Directional cosines of the field H

Hpx, Hpy & Hpz
Directional cosines of the field Hp

View Signal Position button
To view the positioning of the signal set by the x, y and z parameters, click this button:

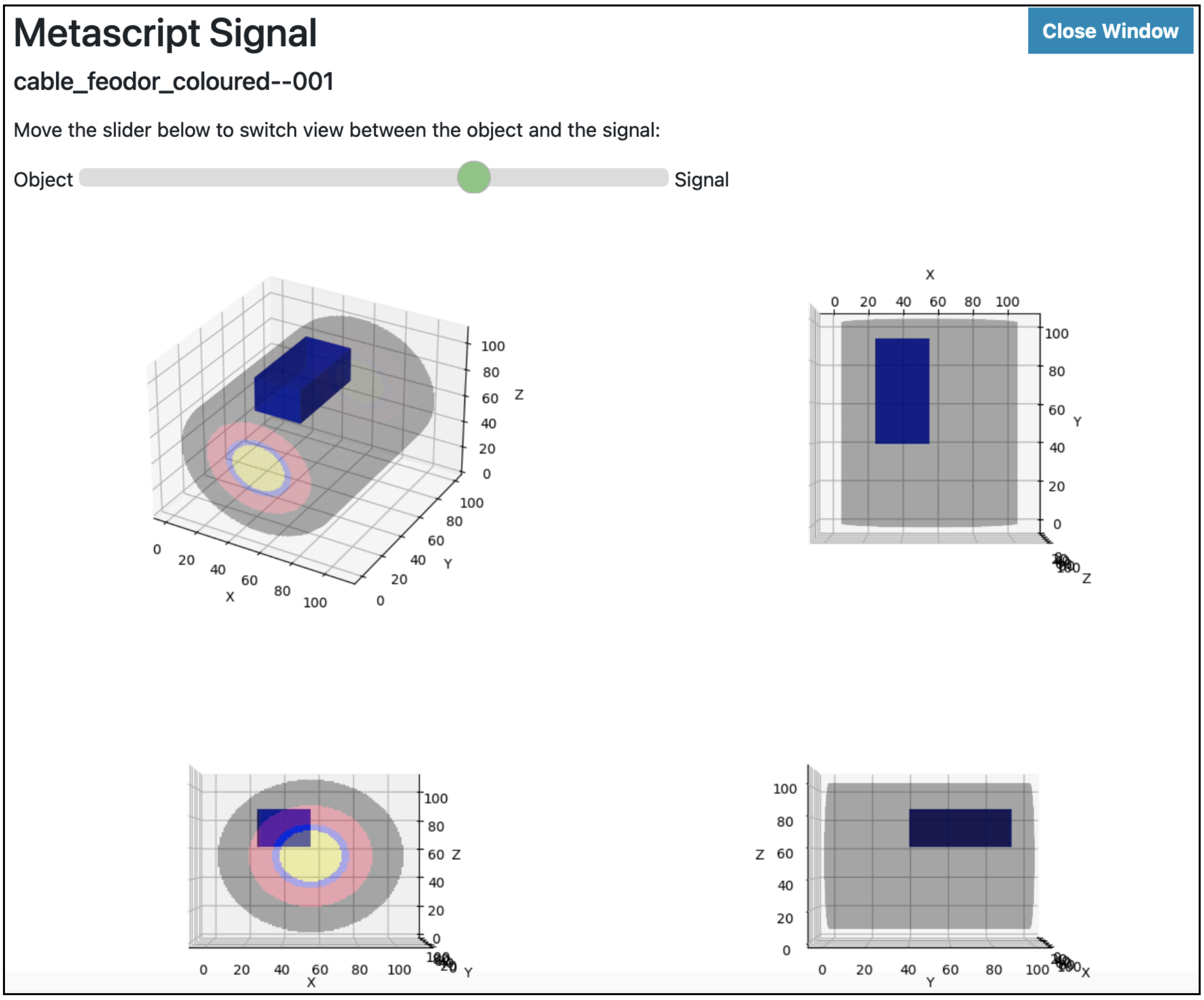


Figure 33: The positioning of the Uniform signal

The slider changes the opacity of each image so it is possible to see more clearly the position of the signal against the object.

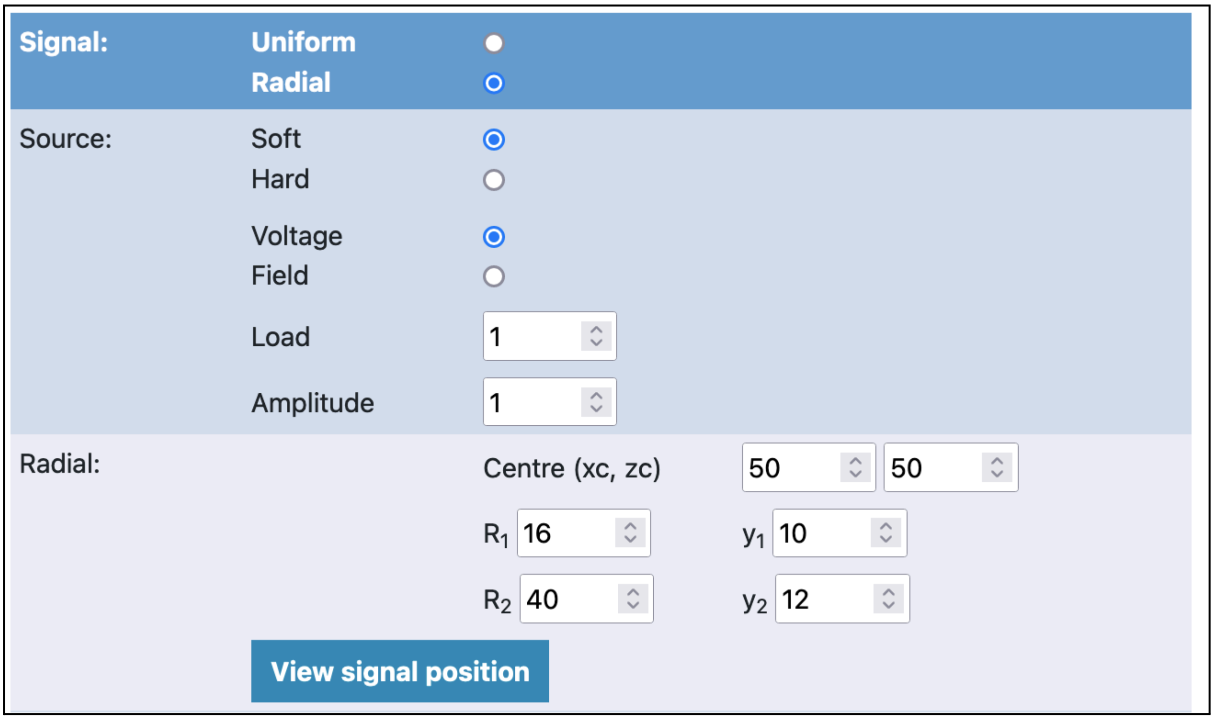


Figure 34: The Input pane: Radial section

Radial
Input signal for a radial input pulse.

NB. The Radial input also contains several fields visible in the 'Uniform' section.

Dr
Amplitude of radial pulse.

Centre (xc, zc)
Central position for radial pulse in xz plane.

R_1
Value of inner pulse radius.

y_1
Pulse position range along the y direction beginning with y1.

R_2
Value of outer pulse radius.

y_2
Pulse position range along the y direction ending with y2.

View Signal Position button
To view the positioning of the signal set by the x, y and z parameters, click this button:

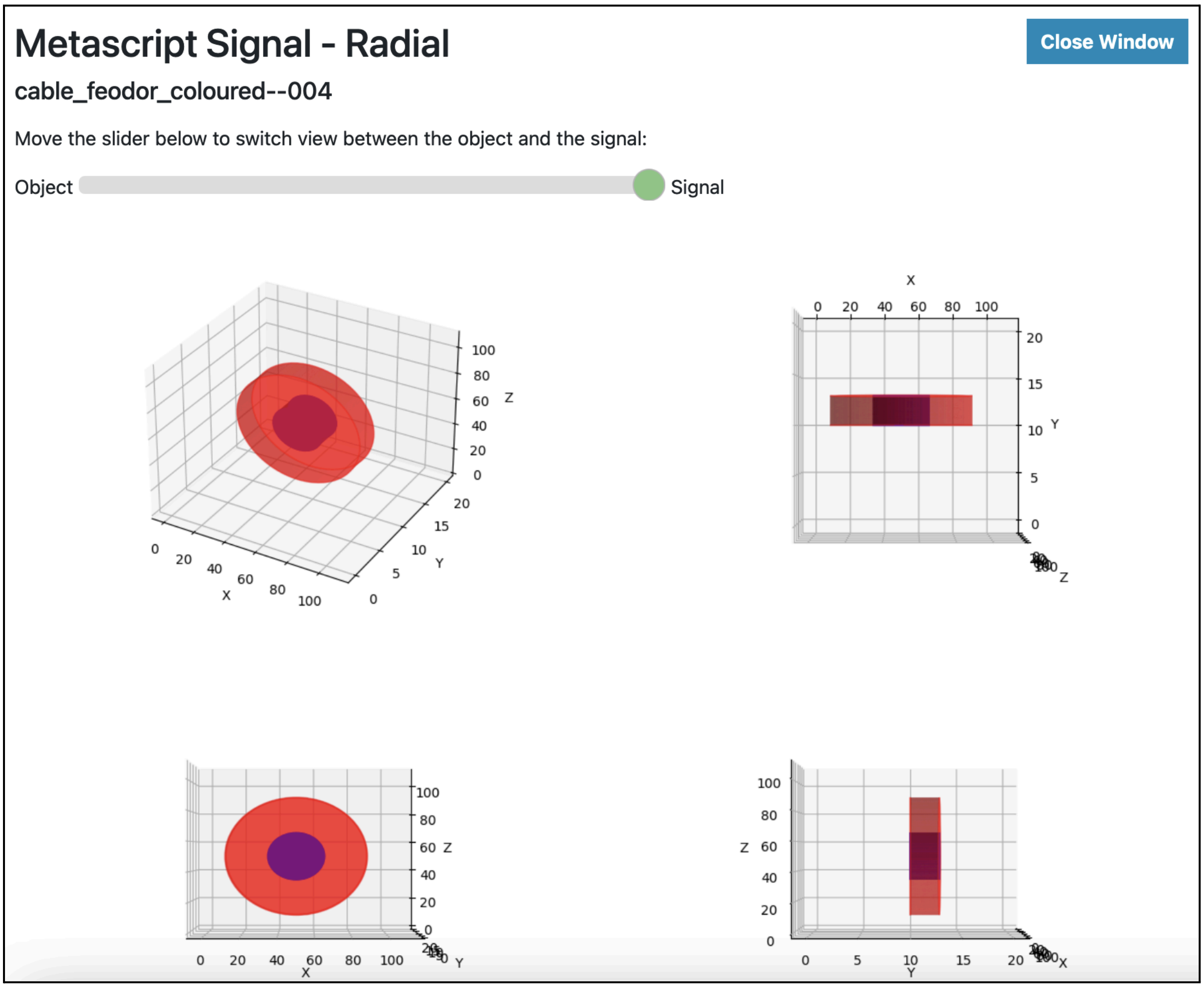


Figure 35: The positioning of the Radial signal

Signal Shape:

Signal shape can be either Pulsed or Continuous

Pulsed

Select dropdown menu for pulse excitation.
Choose between a Gaussian "Top Hat" and Sinc for the pulse excitation.

Gaussian

Selects Gaussian function as input pulse.

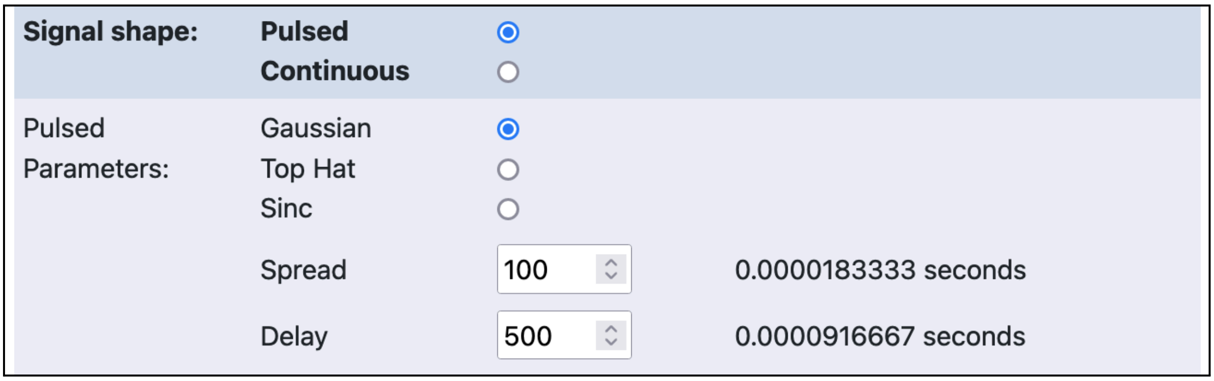


Figure 36: Pulsed Gaussian parameters

Spread

Spread of Gaussian function. Shows conversion to seconds.

Delay

Delay of Gaussian function. Shows conversion to seconds.

Top Hat

Selects "Top Hat" function as input pulse.

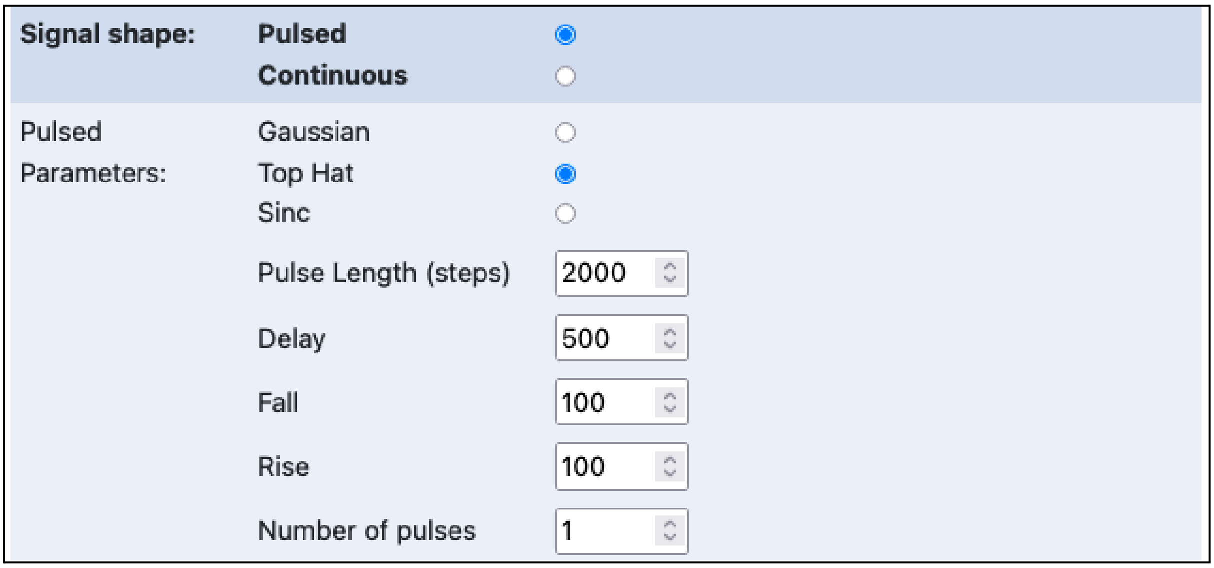


Figure 37: Pulsed "Top Hat" parameters

Pulse length (steps)

Number of steps that pulse excitation remains active.

Delay

Sets the delay before the start of the pulse

Fall

Number of steps in pulse fall.

Rise

Number of steps in pulse rise.

Number of pulses

Number of pulses that are applied in the simulation.

Sinc

Selects Sinc function as input pulse.

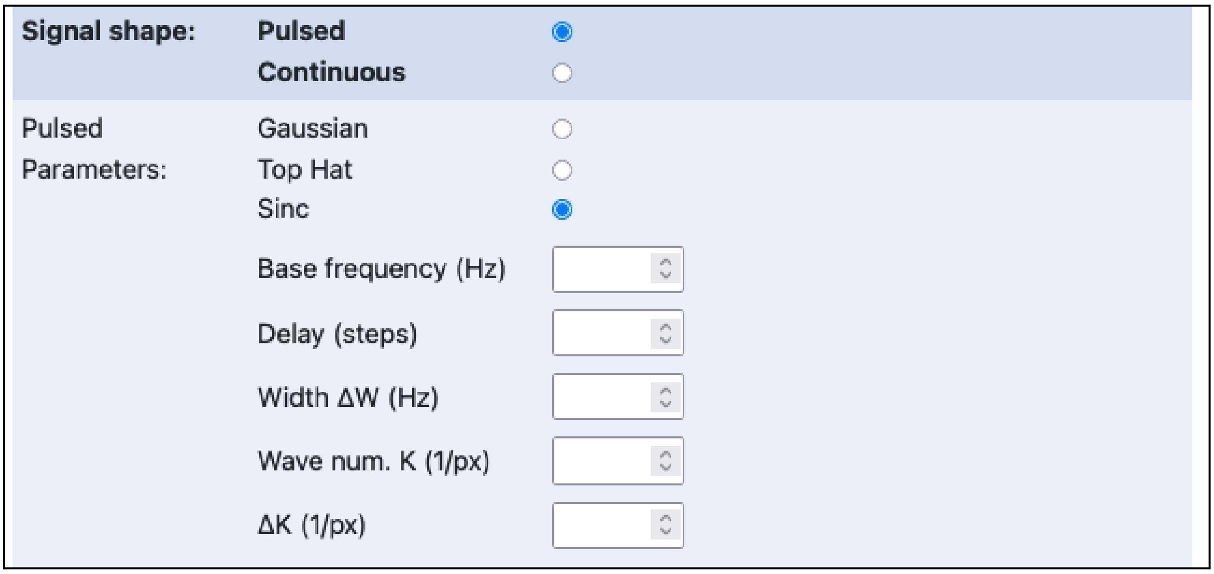


Figure 38: Pulsed Sinc parameters

Base frequency (Hz)

The centre frequency within the gaussian envelope.

Delay (steps)

Delay in time steps before the pulse starts.

Width ΔW (Hz)

The width of the gaussian envelope (in frequency).

Wave num. K (1/px)

The central wave number. Dimension is in $2\pi/\lambda$ where lambda is in pixels.

ΔK (1/px)

The width of the wave number region. The width is defined around the centre wave number.

Continuous Signal shape

Continuous excitation with a single frequency.

Frequency

Choose frequency for continuous excitation.

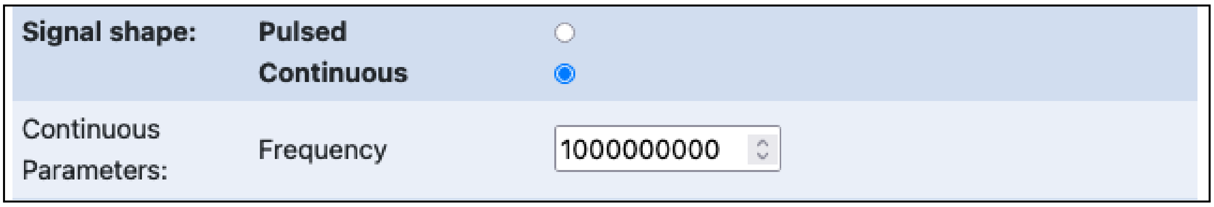


Figure 39: Continuous signal shape parameters

Initial Fields

Parameters can be loaded from previous output. Select ‘Load from file’ and choose the output directory and Frame number.

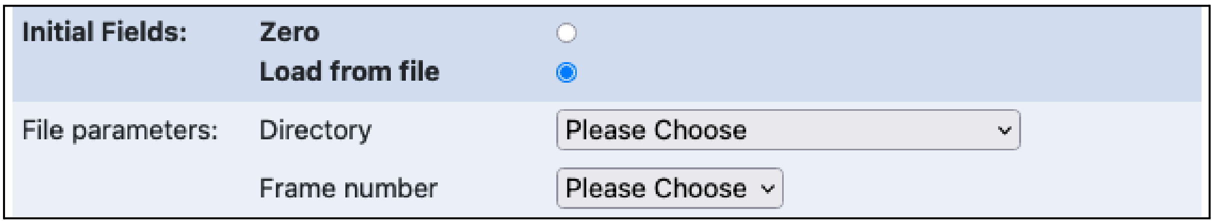


Figure 40: The ‘Input’ section showing the ‘Radial Signal’ and ‘Continuous Signal Shape’ sections

Constants

The options in the constants section are static:

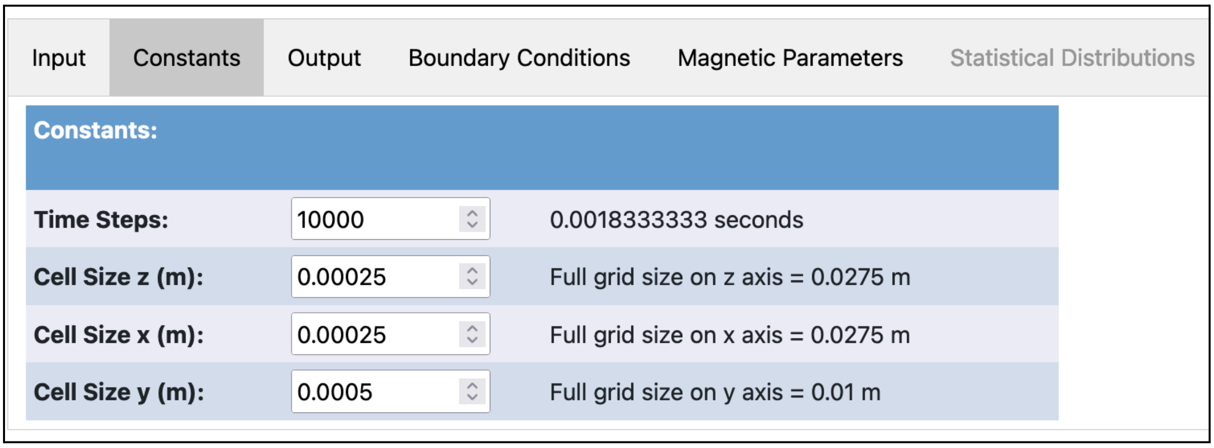


Figure 41: The ‘Constants’ pane

Time_steps

Total number of timesteps for the simulation. The conversion to seconds is displayed.

Cell sizes

Cell size of simulation in metres. Once entered, the actual grid size for each axis is calculated and displayed.

Output

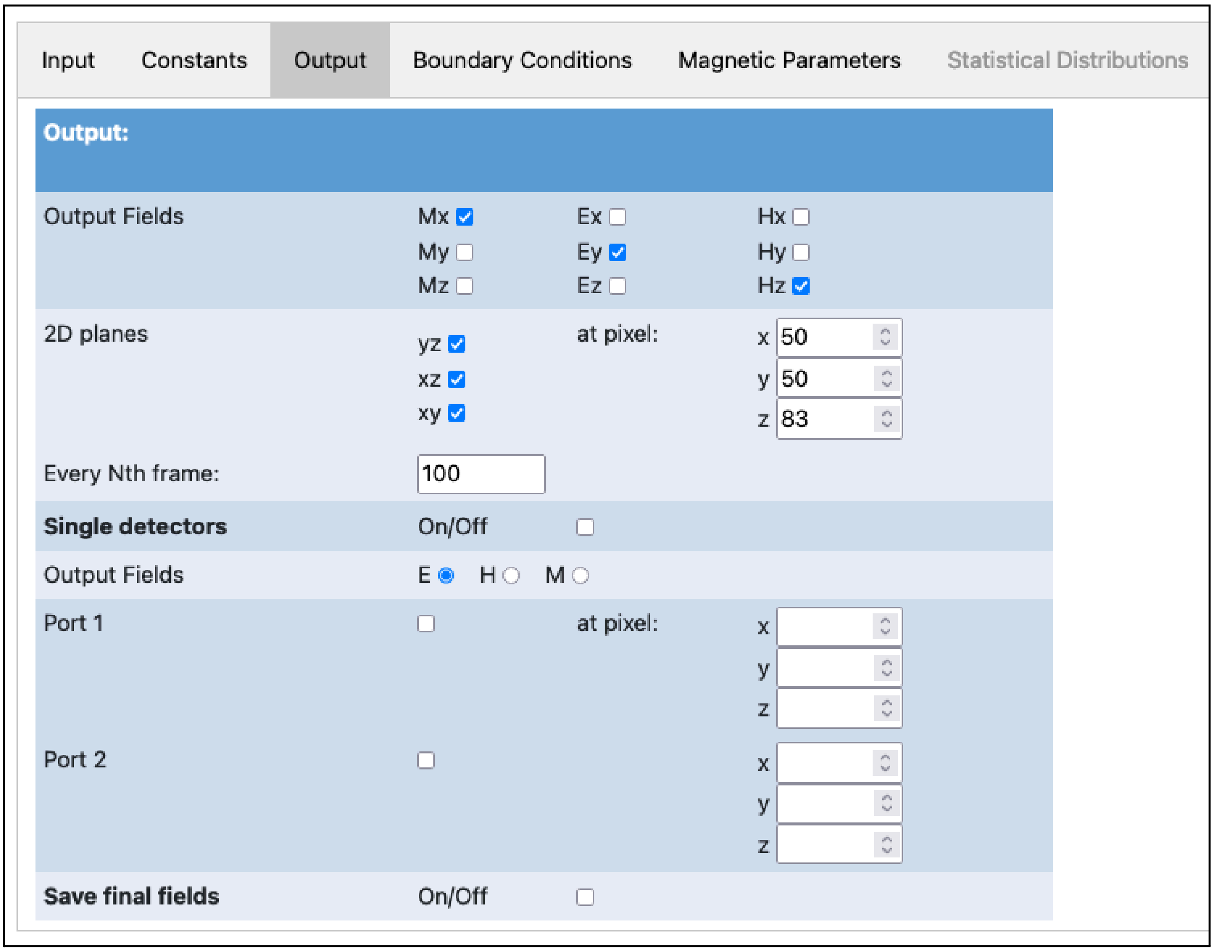


Figure 42: The 'Output' pane

The options in the Output section are also static:

Output Fields
Select which data fields to output using the interface.

2D planes xy, yz & xz
Output plane for vector data.

At pixel x, y & z
Pixel position for each output plane.

Every Nth frame
Output data at every Nth timesteps

Save final fields

Select whether to output the electromagnetic fields and magnetisation at the final state of the simulation.

Single Detectors

The single detector feature can be used to record the electric field at two selected positions in the 3D simulation space.

On/Off

Choosing ‘On’ will save files aTime1.dat and aTime2.dat recording the specified fields at the position of the two ports.

Output Fields - E, H & M

Two positions can be selected using each port to record transmission and reflection data.

Port 1 & 2

Two positions can be selected using each port to record transmission and reflection data.

at pixel

Choose the specific grid location using the x, y and z pixels.

Save final fields

Description required

Boundary Conditions

The solver contains several different options for boundary conditions, including perfectly matched layer (PML), perfect electric conductor (PEC) and periodic boundary conditions (PBC).

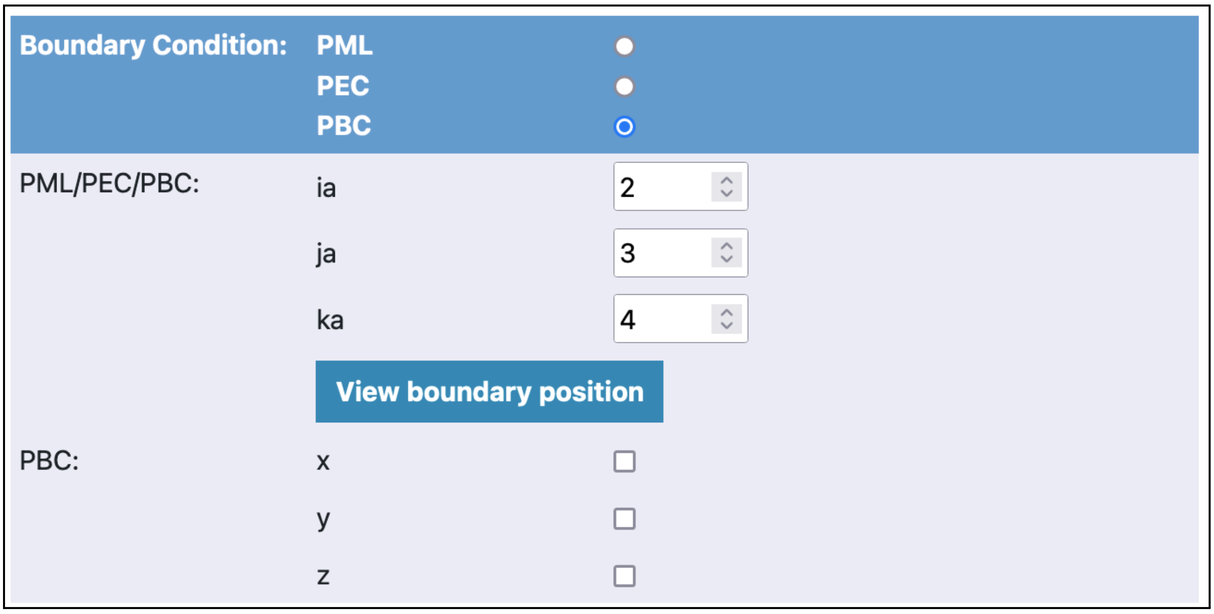


Figure 43: The 'Boundary Conditions' pane showing PBC options

Specifying boundary conditions

Each boundary condition can be selected by choosing one of the "PML", "PEC" or "PBC" options in the dropdown menu. The planes terminating the simulation space are assigned , and directions which can be enabled or disabled by ticking the neighbouring option box (see Fig. 31).

Controlling depth of PML layer

Electromagnetic wave absorption at the ends of the simulation space can be controlled by increasing or decreasing the number of cells in the PML layer. Increasing the number of PML cells will generally improve the degree of electromagnetic wave absorption at the cost of a longer computation time, due to the increased number of simulation cells. The number of cells in the PML layer can be chosen using the dropdown options "ia", "ja" and "ka" in the PML panel, which are integers corresponding to the x, y and z-axes, respectively. The PEC and PBC boundary conditions are also defined to begin at the points "ia", "ja" and "ka" and can be controlled using the same PML parameters.

View Boundary Position button

To view the positioning of the signal set by the x, y and z parameters, click this button:

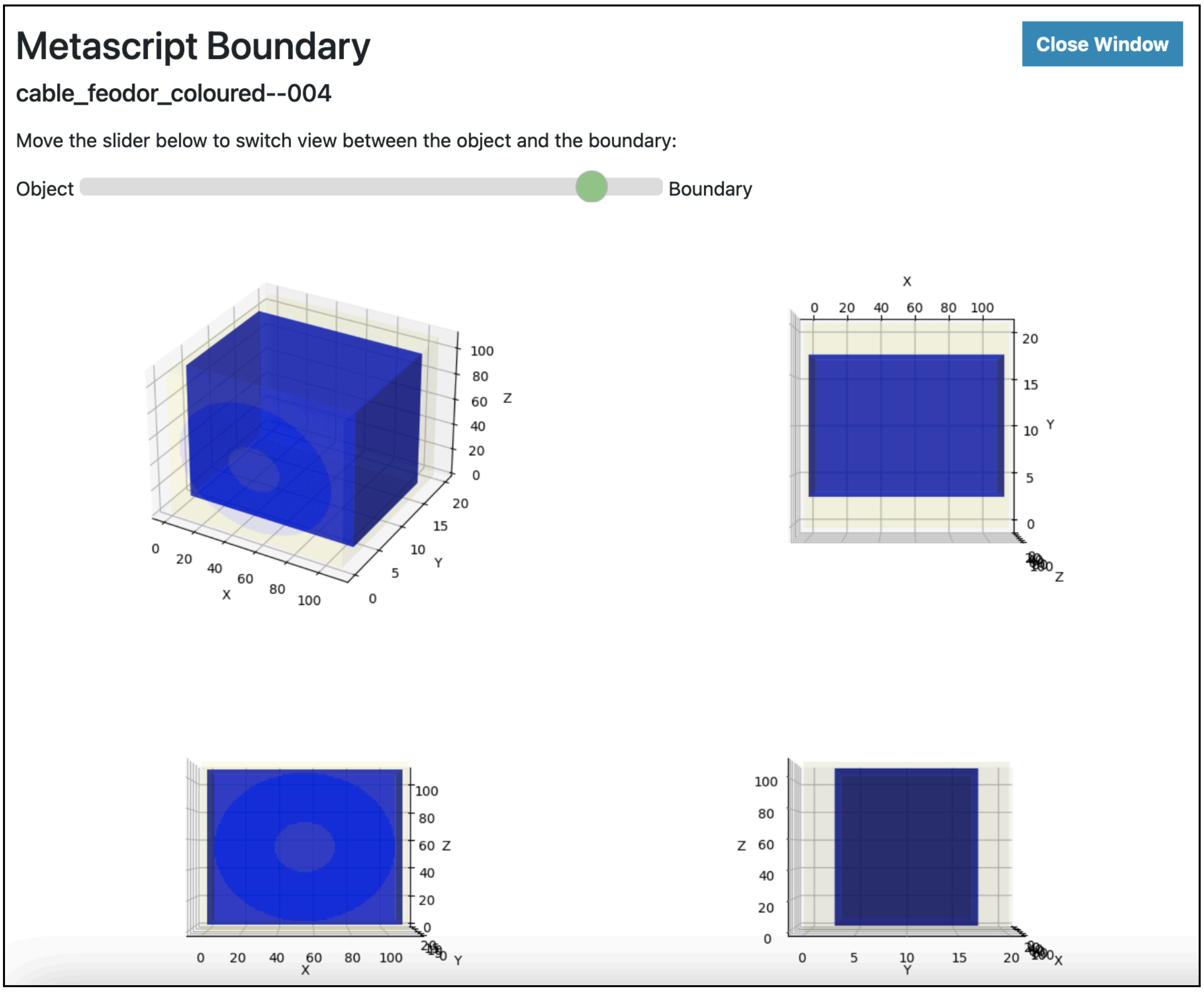
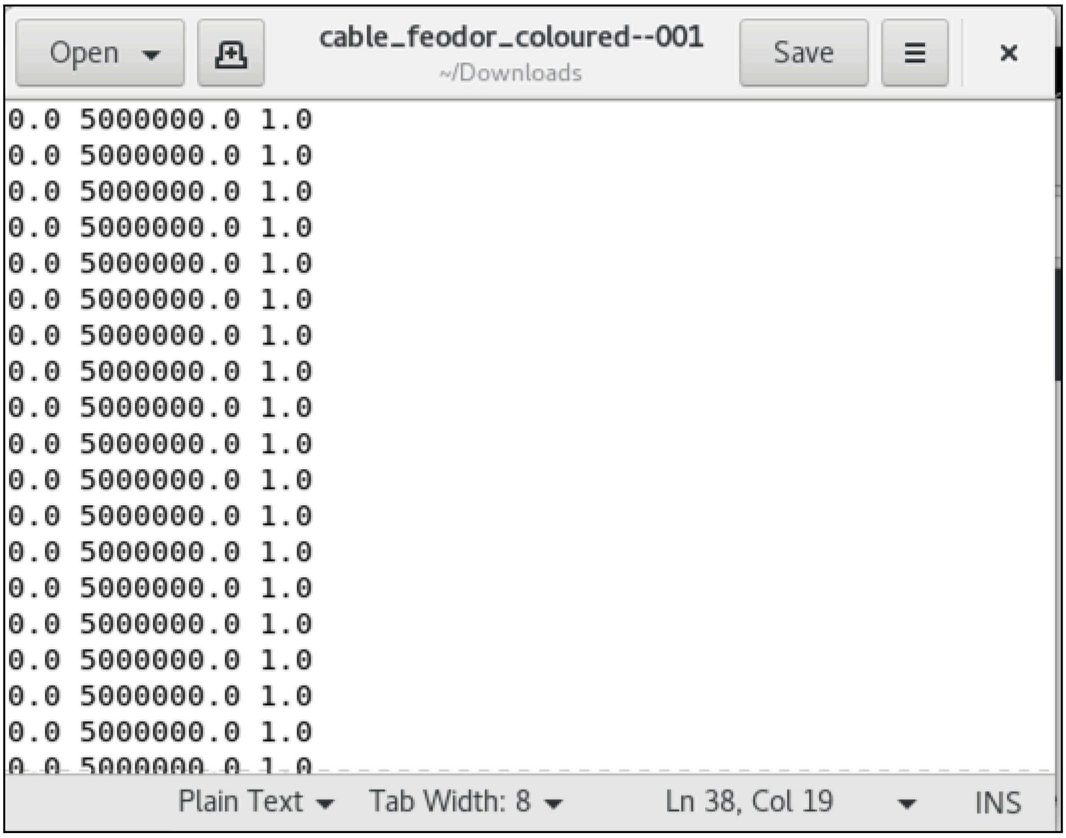


Figure 44: Displaying the Boundary position

Manual input of material parameters

As an alternative to GUI input, material properties can be imported manually using text file formats found in the “metascript” and “matrix file” folders.


Open | cable_feodor_coloured--001 ~/Downloads | Save

```
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
0.0 5000000.0 1.0
```

Plain Text | Tab Width: 8 | Ln 38, Col 19 | INS

Figure 45: 'material_input' file

In the above text file example, each row represents one cell in the grid which are populated sequentially according to a triple "for" loop. In this loop process, the x-axis represents the outer loop, and the z-axis represents the inner loop, with the y-axis in the middle. The value of conductivity, magnetisation and dielectric constant correspond to the first, second and third columns, respectively. The total number of rows equals the total number of cells in the simulation grid.

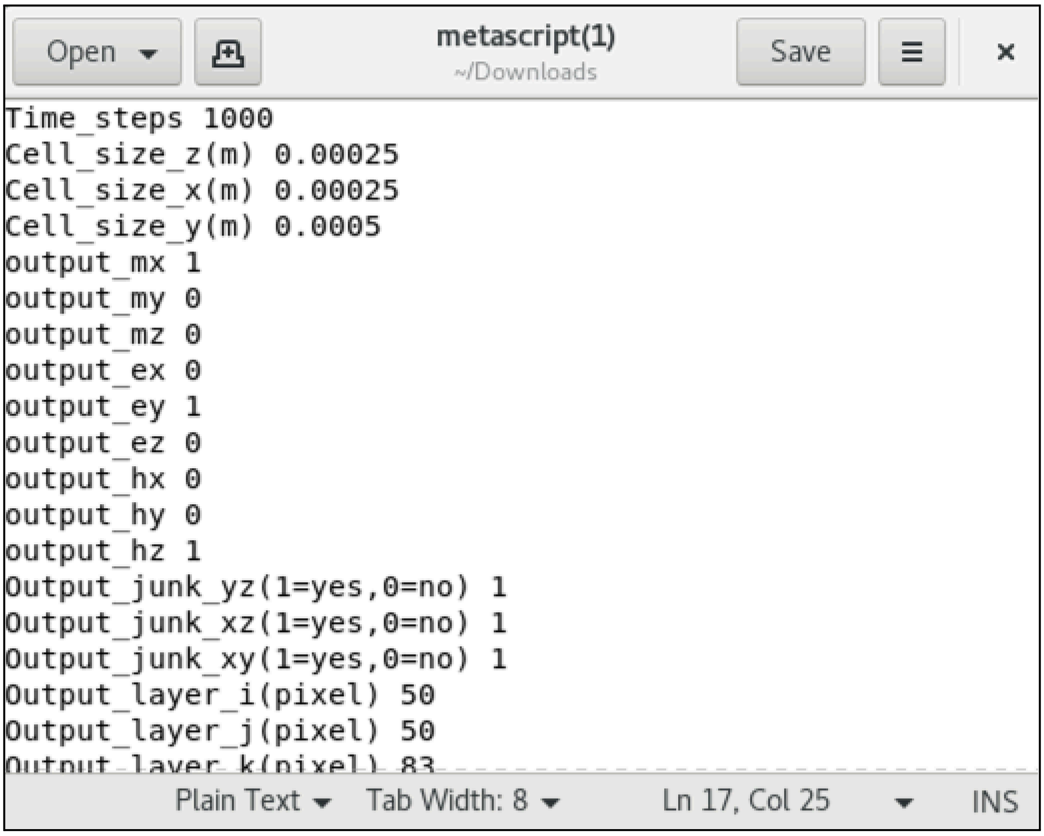


Figure 46: 'metascript' file

In the metascript folder, user-defined simulation parameters can be controlled by modifying the output text file following the creation of a metascript. Each row corresponds to a different physical parameter which can be controlled by varying its associated numerical value.

Magnetic Parameters

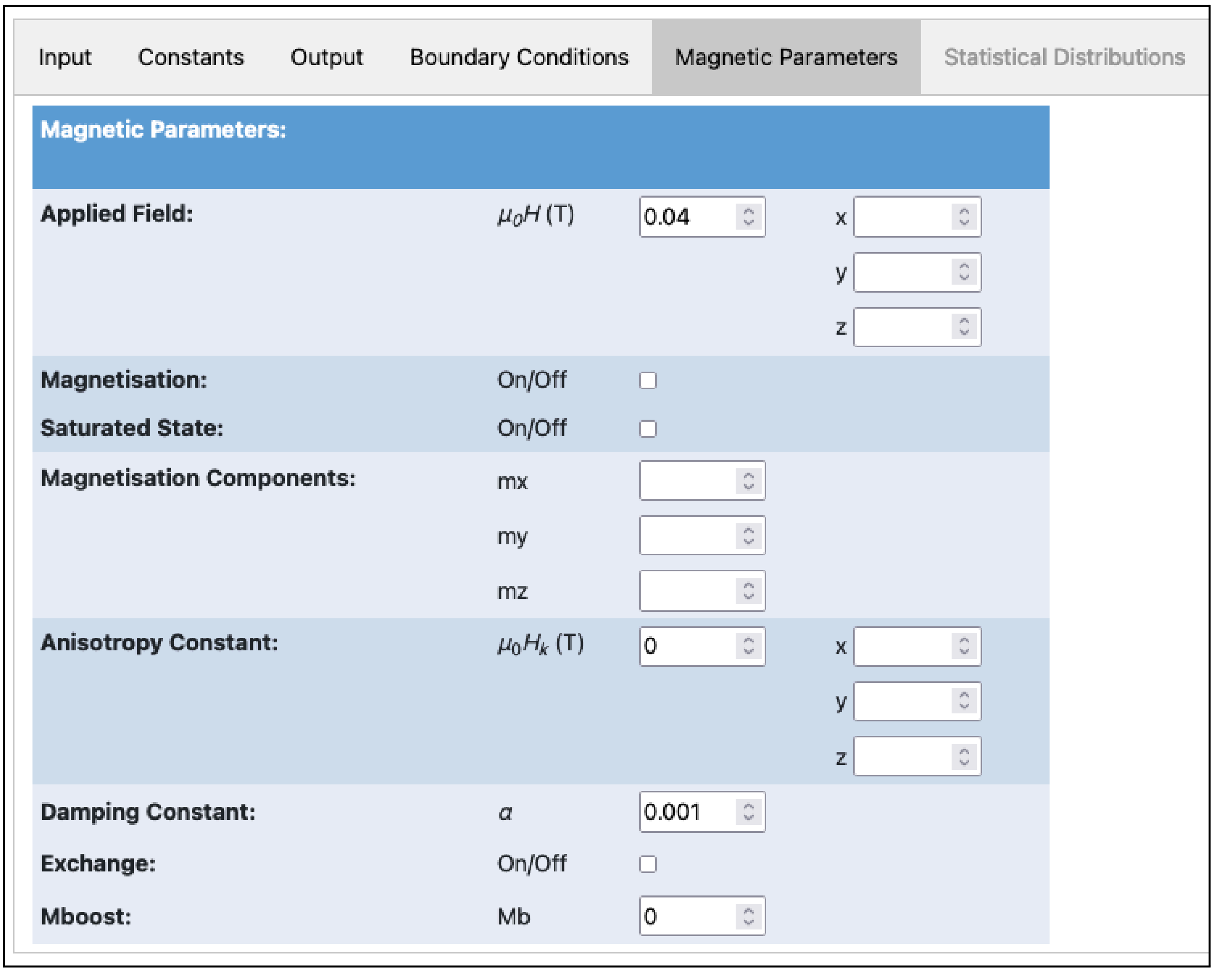


Figure 47: The 'Magnetic Parameters' pane

The options in the Magnetic Parameters section are static:

Applied Field μ_0H (T)

Absolute value of the applied magnetic field in Tesla.

Applied Field x, y & z

Directional cosines describing the orientation of the applied field.

Magnetisation (A/m)

Controls whether to enable or disable the magnetisation.

Saturated State

Control parameter to begin the simulation from an ideally saturated state or relaxed magnetic state.

Saturation Magnetisation $\mu_0 M_S$ (T)

Absolute value of the saturation magnetisation in Tesla.

Magnetisation x, y & z

Directional cosines describing the orientation of the magnetisation vector.

Anisotropy constant $\mu_0 M_S$ (J/m3)

Absolute value of the uniaxial anisotropy with second (K_1) order terms. In the easy axis case ($K_1>0$) the energy density E is computed by

$$E = K_1|\mathbf{u}\times\mathbf{m}|^2,$$

where **m** is the reduced (unit) magnetization and **u** is the easy axis. Here the energy is zero if the magnetization is aligned with the easy axis. In the hard axis case ($K_1<0$) the energy is offset so that the zero point occurs when the magnetization lies in the easy plane (i.e., orthogonal to the hard axis). In this case the energy density is given by

$$E = -K_1(\mathbf{u}\cdot\mathbf{m})^2.$$

Applied Field x, y & z

Absolute value of the applied magnetic field in Tesla.

Damping Constant a

Value for the damping parameter of the Landau-Lifshitz-Gilbert equation.

Exchange

Control parameter to enable or disable the exchange energy.

Exchange constant A_{ij} (J/m)

Absolute value of the exchange constant for a standard 6-neighbor exchange energy with an exchange energy density contribution from cell ***i*** given by

$$\boldsymbol{E}_\mathbf{i} = \sum_{j\in N_i} \boldsymbol{A}_\mathbf{ij} \frac{\mathrm{m}_i \cdot (\mathrm{m}_i - \mathrm{m}_j)}{\Delta_{ij}^2}$$

where N_i is the set consisting of the 6 cells nearest to cell ***i***, A_{ij} is the exchange coefficient between cells ***i*** and ***j*** in J/m, and Δ_{ij} is the discretization step size between cell ***i*** and cell ***j*** in meters.

Mboost

Multiples the time step used by the Landau-Lifshitz-Gilbert equation by the constant value Mboost.

Statistical Distributions

(Not currently available in the current version of the Solver)

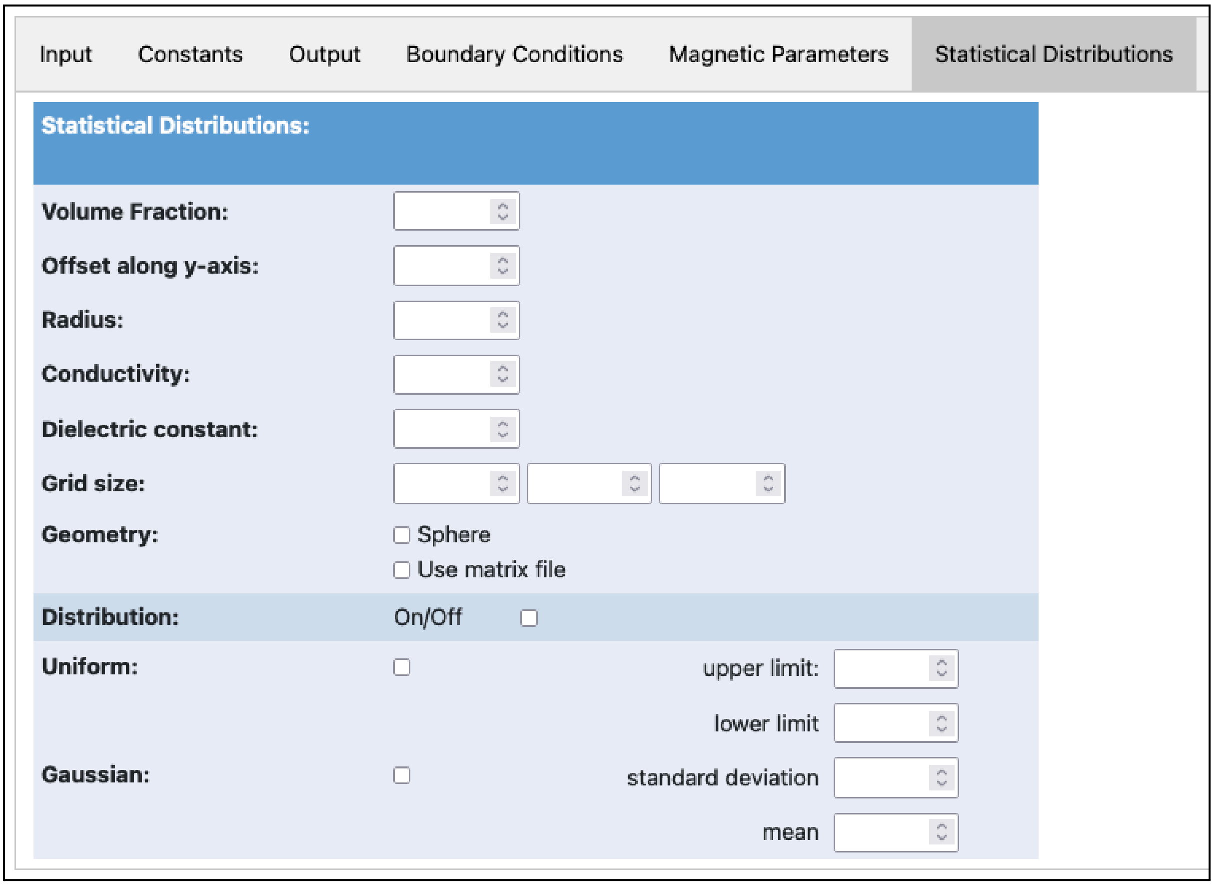


Figure 48: The 'Statistical Distributions' pane

Volume Fraction
Percentage of volume occupied by spherical inclusions.

Offset along y-axis
Description required

Radius
Description required

Dielectric constant
Description required

Grid size
Description required

Geometry – 'Sphere' or 'Use matrix file'
Populate the grid analytically with spherical particles or use standard matrix files.

Distribution – 'On' or 'Off'
Turn on or off the statistical randomization of chosen input particle.

Uniform
Statistically distribute material/geometric properties using a uniform probability density.

Uniform upper limit, lower limit & standard deviation

Upper and lower bounds on the uniform variation of material/geometric properties. For example, a range of 0.8 (lower) to 1.2 (upper) uniformly distributes properties between 80% and 120% of initial input.

Gaussian

Statistically distribute material/geometric properties using a Gaussian probability density.

Gaussian standard deviation & mean

User inputs for standard deviation and mean.

Exponentially Modified Gaussian

Statistically distribute material/geometric properties using an exponentially modified Gaussian probability density.

Exponentially Modified Gaussian standard deviation, mean & exponential mean

User inputs for standard deviation, Gaussian mean and exponential mean.

Saving the Metascript File

When finished, click the 'Update Parameters'' button at the bottom of the page to store the changes to the meta script parameters. This can be done from any pane.

There is also a link to ignore any updates and reload the original data.

Figure 49: The 'Matrix Files' page showing the 'Solve' link

Running the Solver and analysing the results

To run the Solver, simply click the ‘Solve’ button next to the desired matrix file on the Matrix Files’ page:

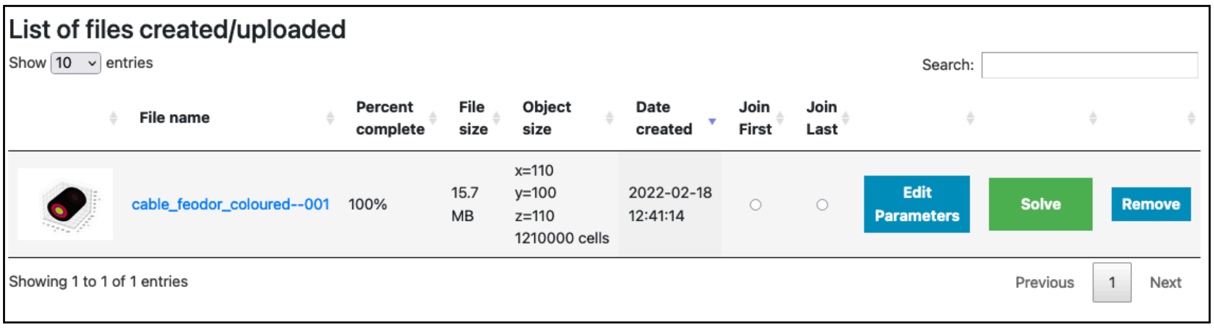


Figure 50: The ‘Matrix Files’ page showing the ‘Solve’ link

The browser will redirect to the ‘Output’ tab, in a new directory solely for the output of this run. The progress will update automatically every 10 seconds (this can be halted by clicking the link at the top of the page - see the Figure below).

Job Queued

The output page will show a status of ‘Job Queued’ until the solver starts.

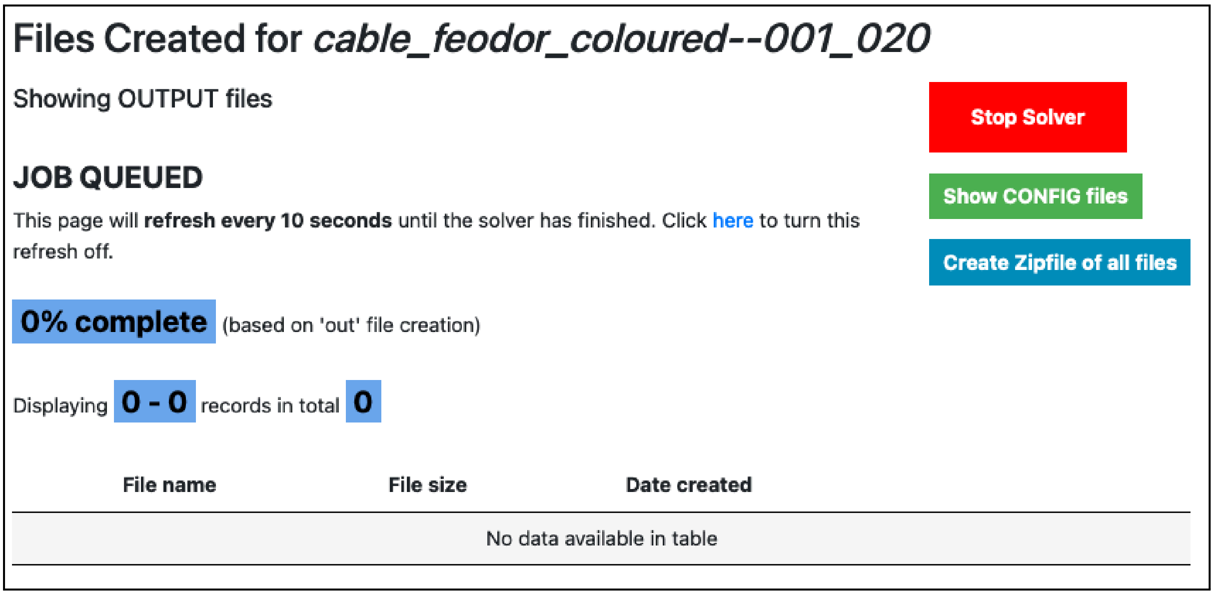


Figure 51: The Solver begins to run at 0% progress

Solver Running

The solver begins by calculating the number of files it needs to output and uses this to calculate the percentage complete metric.

As the solver runs, output is produced in the form of ‘out_’ files. Tese can be clicked on and viewed before the solver is finished.

The percentage completed metric will increase along with the number of files produced (see the Figure below).

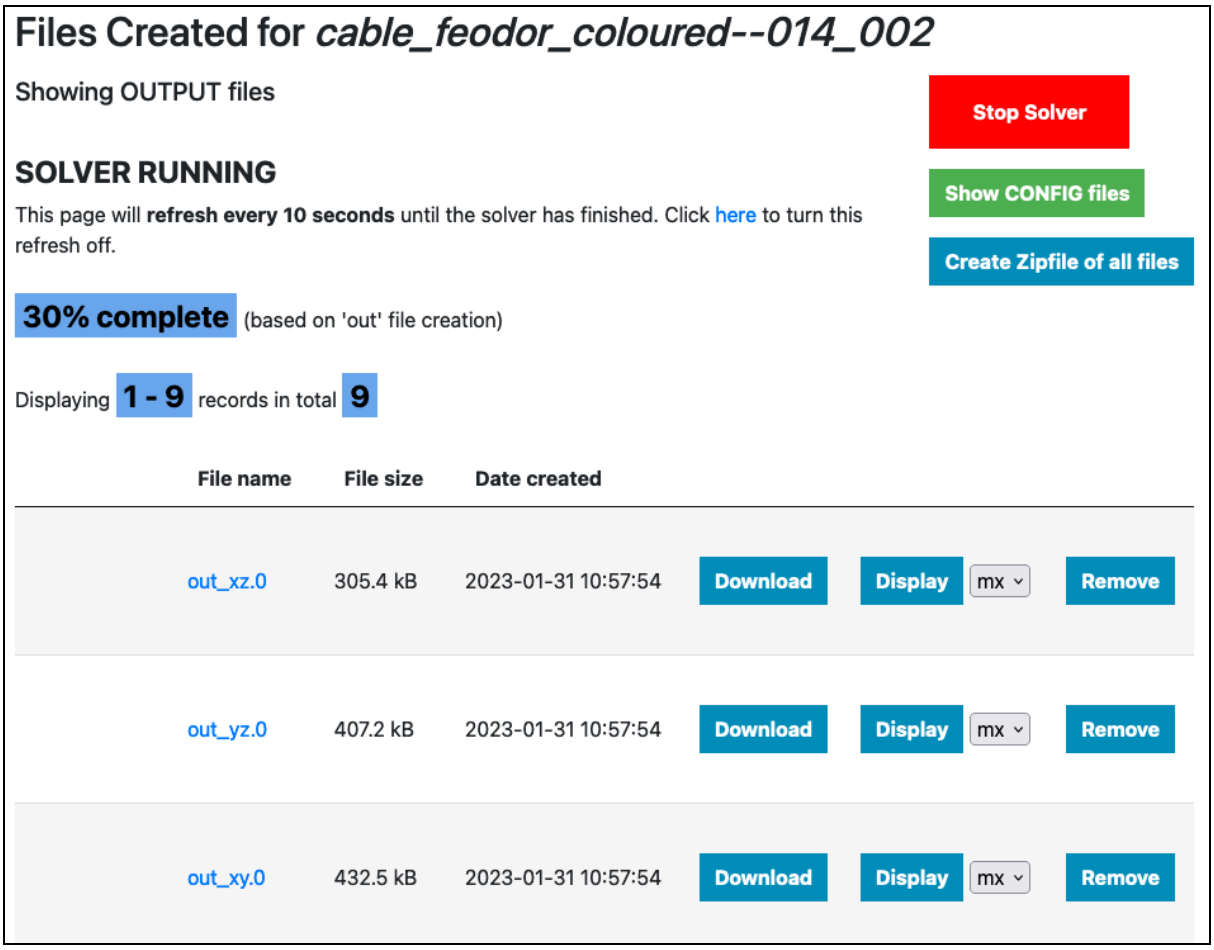


Figure 52: The page reloads automatically to show the new progress. This time at 30%

Manually Stopping the Solver

At any time it is possible to stop the solver by clicking the red ‘Stop Solver’ button.

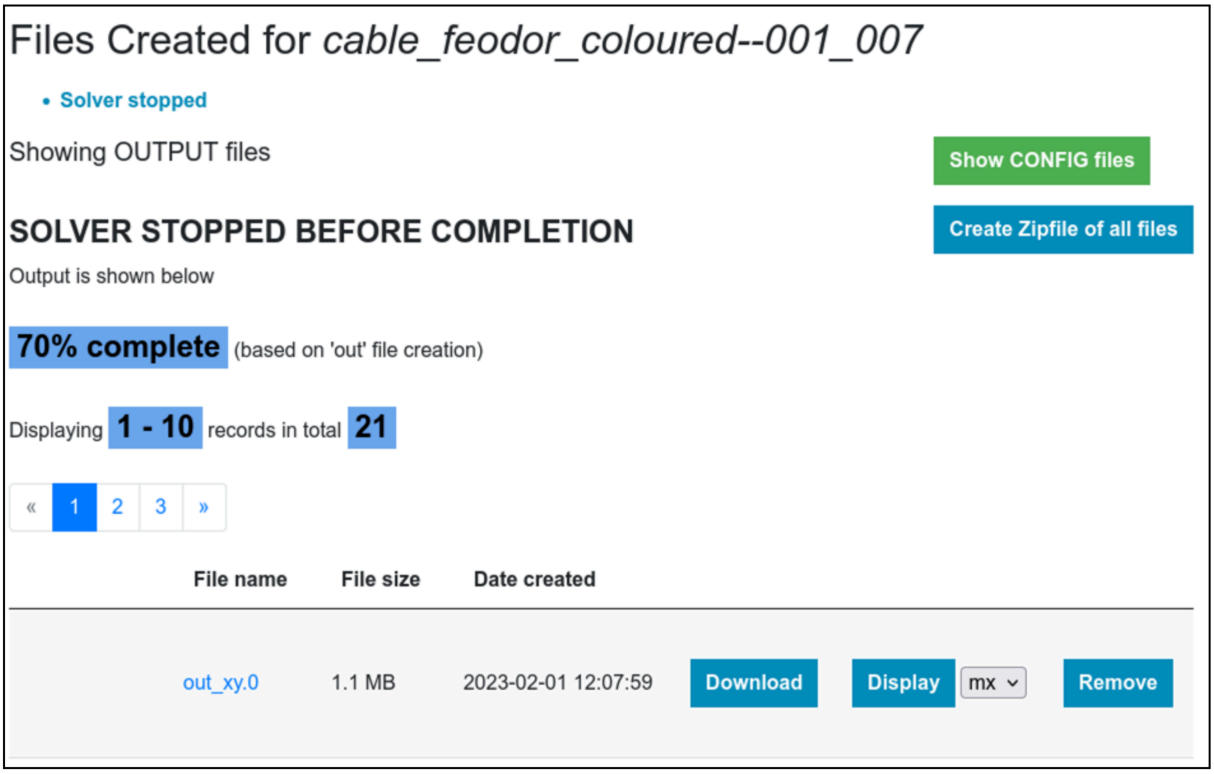


Figure 53: Stopping the solver before completion

Not a Number

After each refresh, the GUI checks the latest output file for the presence of ‘nan’ - Not a Number.

If ‘nan’ is found in the output file, this signifies an error and the Solver is stopped.

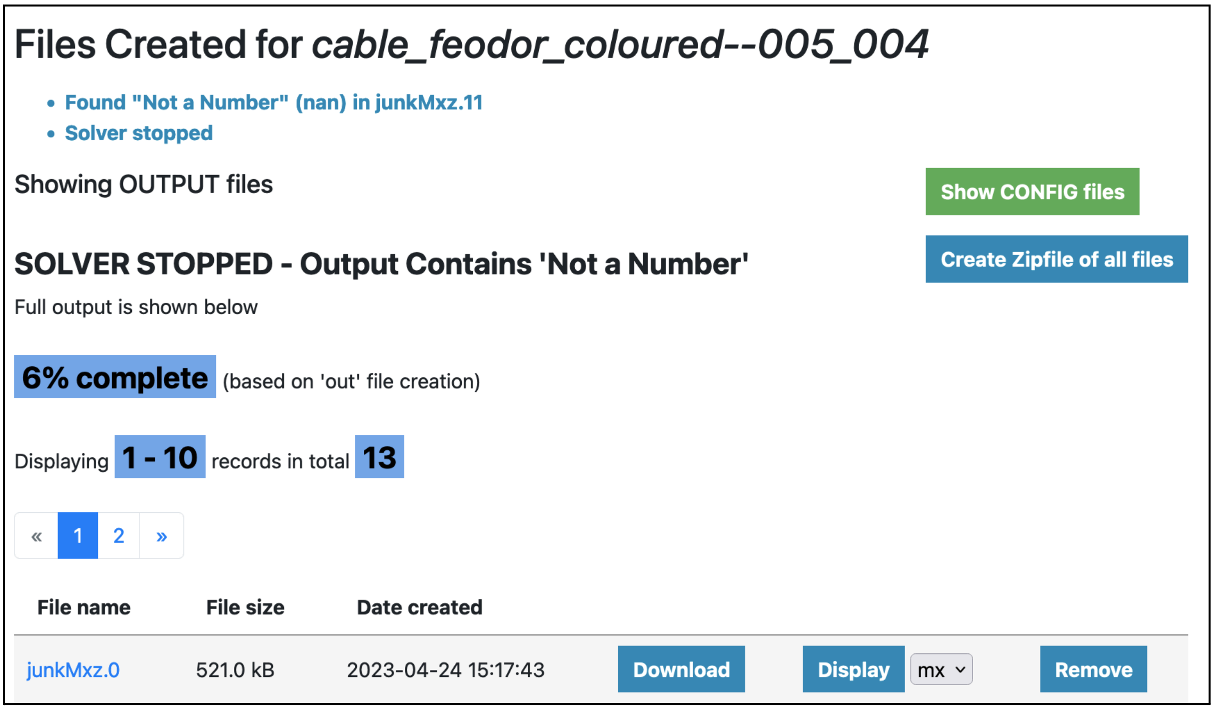


Figure 54: Checking for 'nan' - Not a number

Segmentation Fault

If an error occurs in the running of the solver causing it to stop (usually in the form of a segmentation fault), this is noted on the output screen.

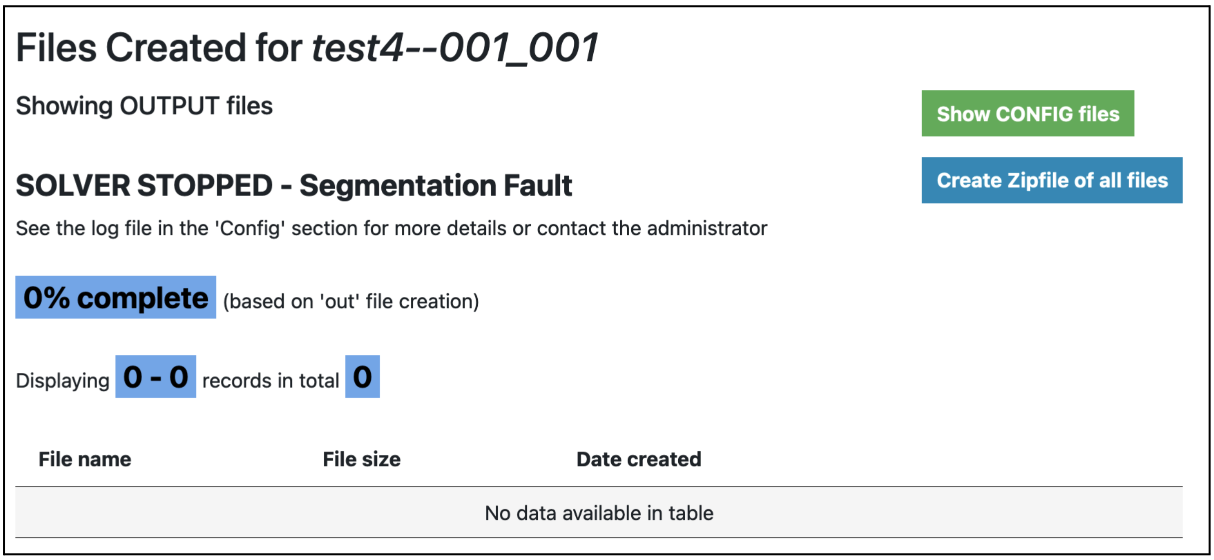


Figure 55: Segmentation Fault

Solver Finished

Once the Solver reaches 100%, all output files have been created.

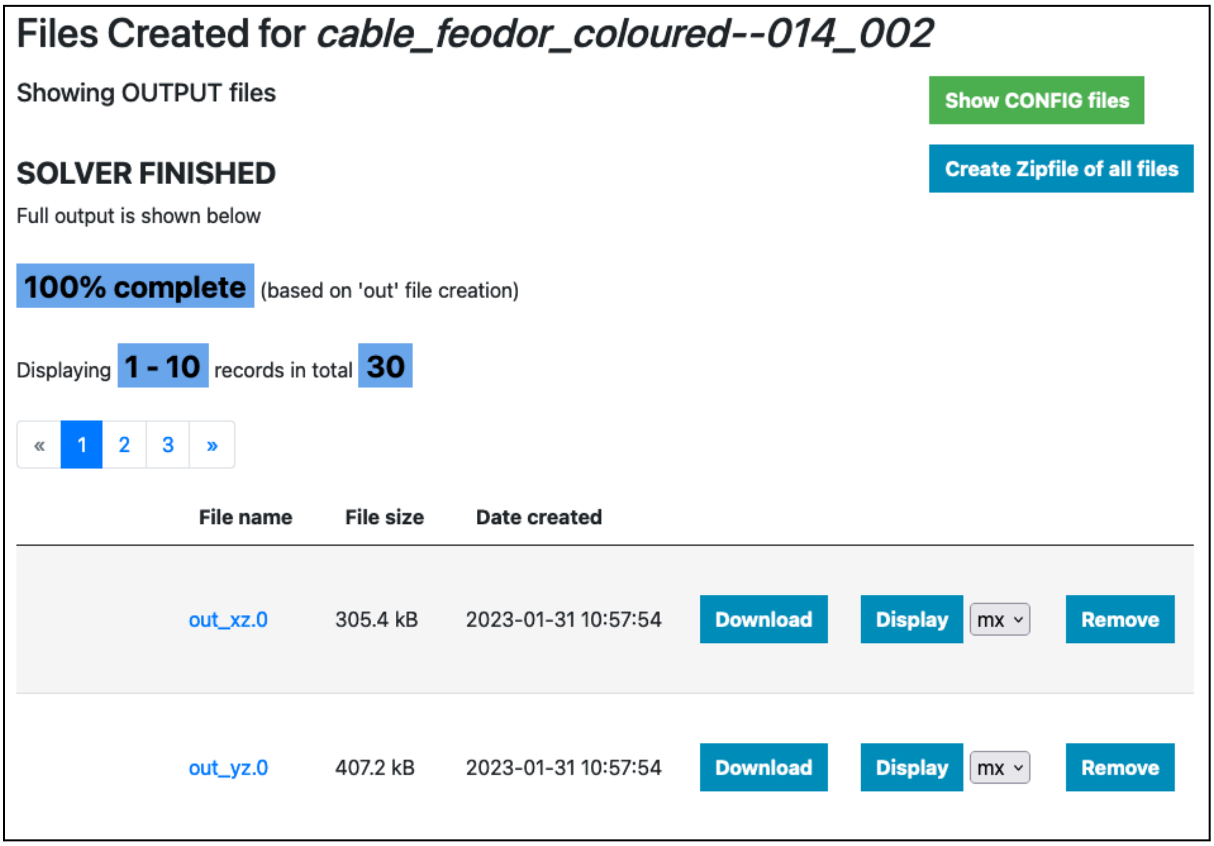


Figure 56: The Solver has finished and all output files are displayed for analysis

Creating a Zipfile

All output can be easily zipped into an archive file for download. Simply click the ‘Create Zipfile of all files’ button:

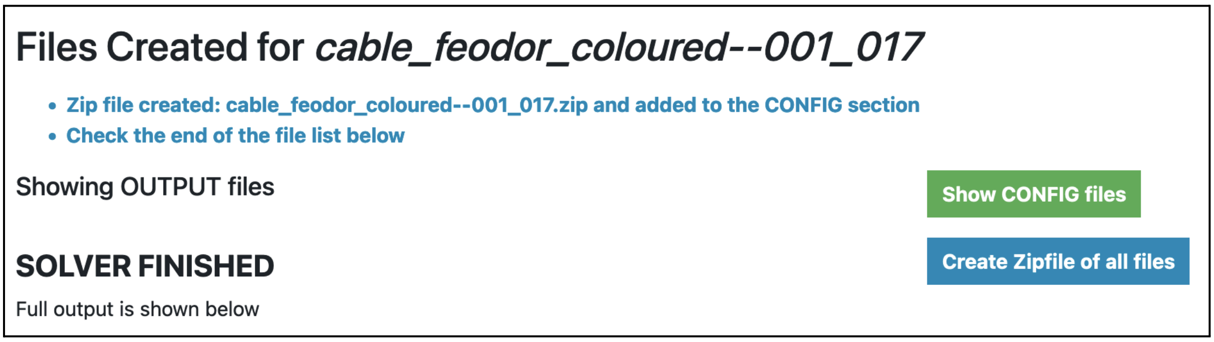


Figure 57: Clicking the ‘Create Ziipfile’ button

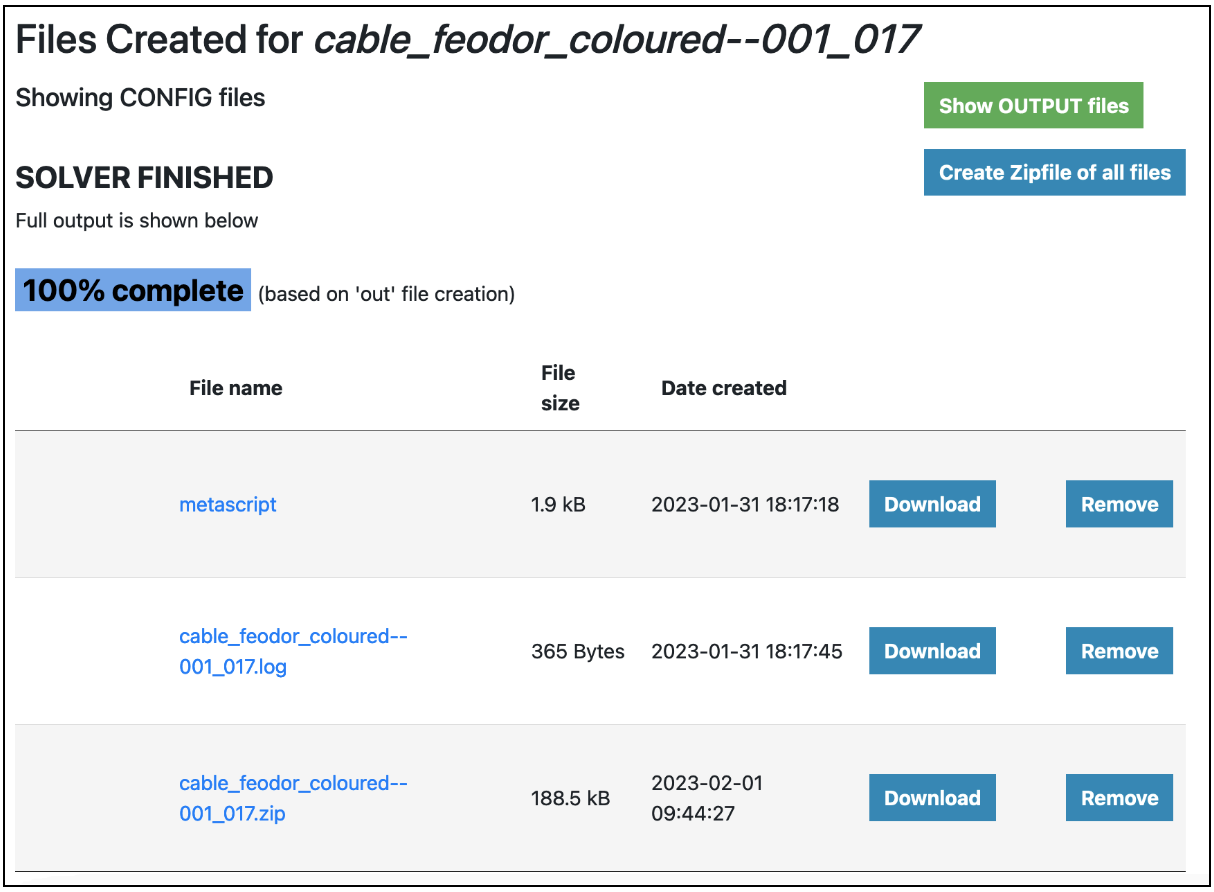


Figure 58: The created zipfile

Config Files

The main screen shows only the ‘out_’ files but several other files are also created by the solver, providing valuable configuration data. These are displayed on the Config screen, accessible by clicking on the ‘Show Config Files’ button:

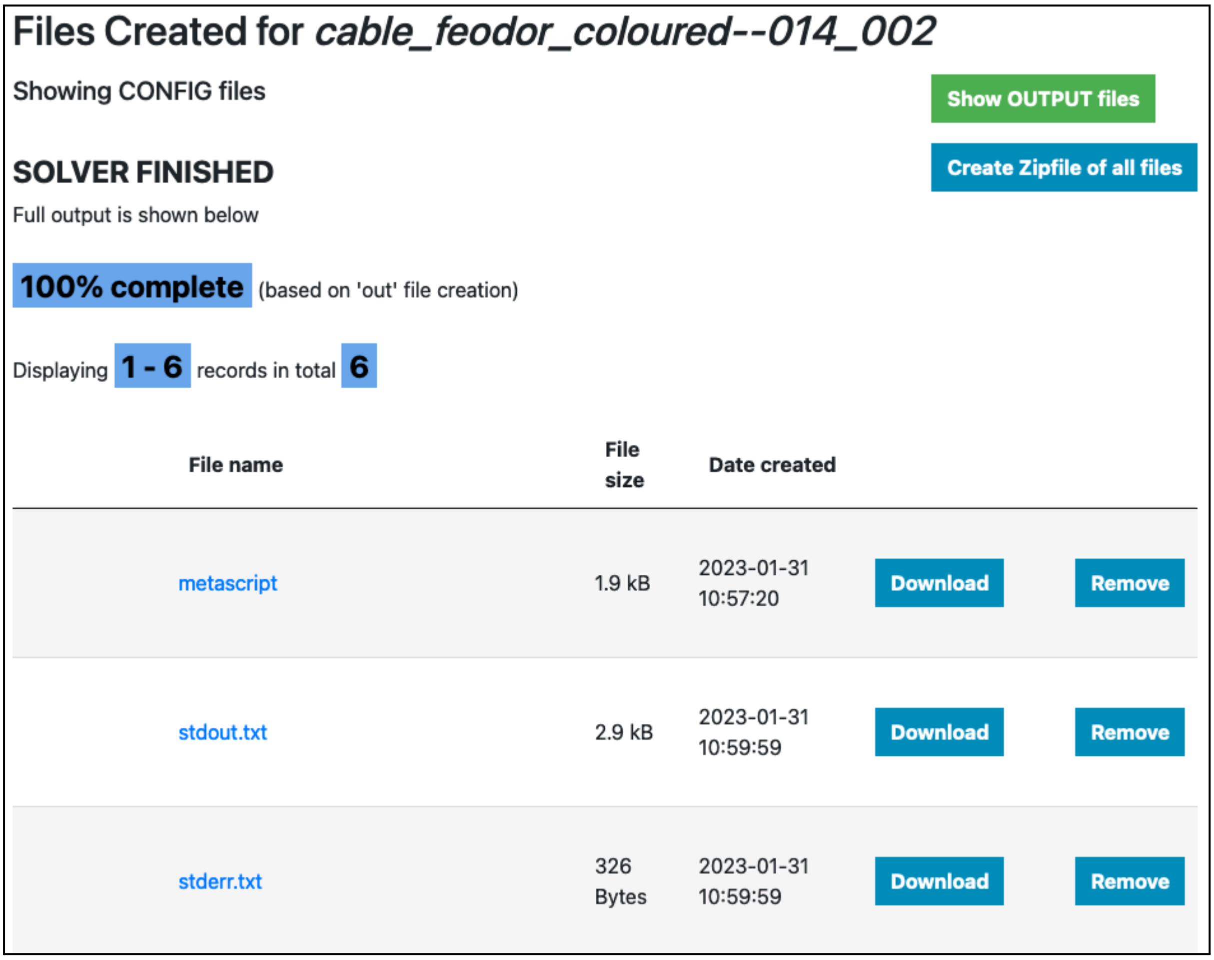


Figure 59: The Config files

Explanation of the config files:

Metascript
This is a copy of the meta script file passed with the matrix file to the solver

out.dat

Contains a list of the parameters, as read by the solver. (with no errors this should be the same information as in the Metascript file)

pid2
Description required (4051?)

aTimeM1(2).dat
Data outputs from ports 1 and 2. (Saved in single column format, each point a value of the chosen field per each step)

****input_file**.log**
This file contains a log of all output from the command line of running the Solver. It is named the same as the matrix file.

Analysing the results

All output files can be downloaded for analysis.

Out Files

The main output files are the 'out_' files.

The number and type of files outputted are controlled by the 'Constants' and 'Output' panes of the metascript parameters page.

Number of files

The number of 'out' files is calculated by the following equation:

Time Steps / Every Nth frame
E.g. 1000/100 = 10 output files

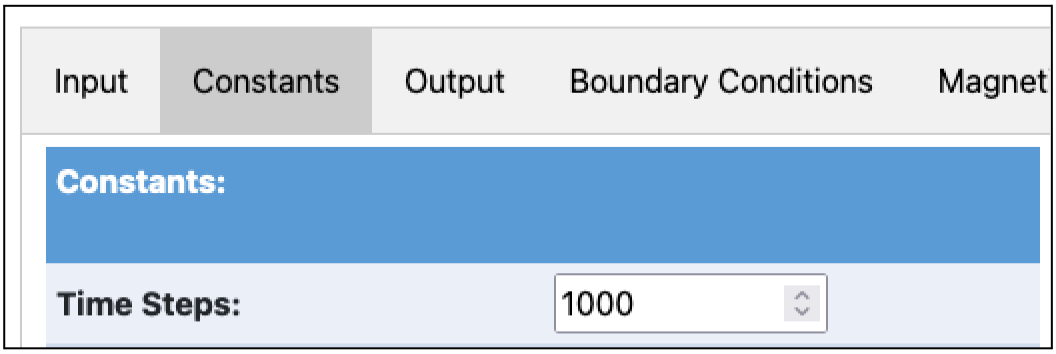


Figure 60: Showing the 'Time Steps' value on the meta script 'Constants' pane

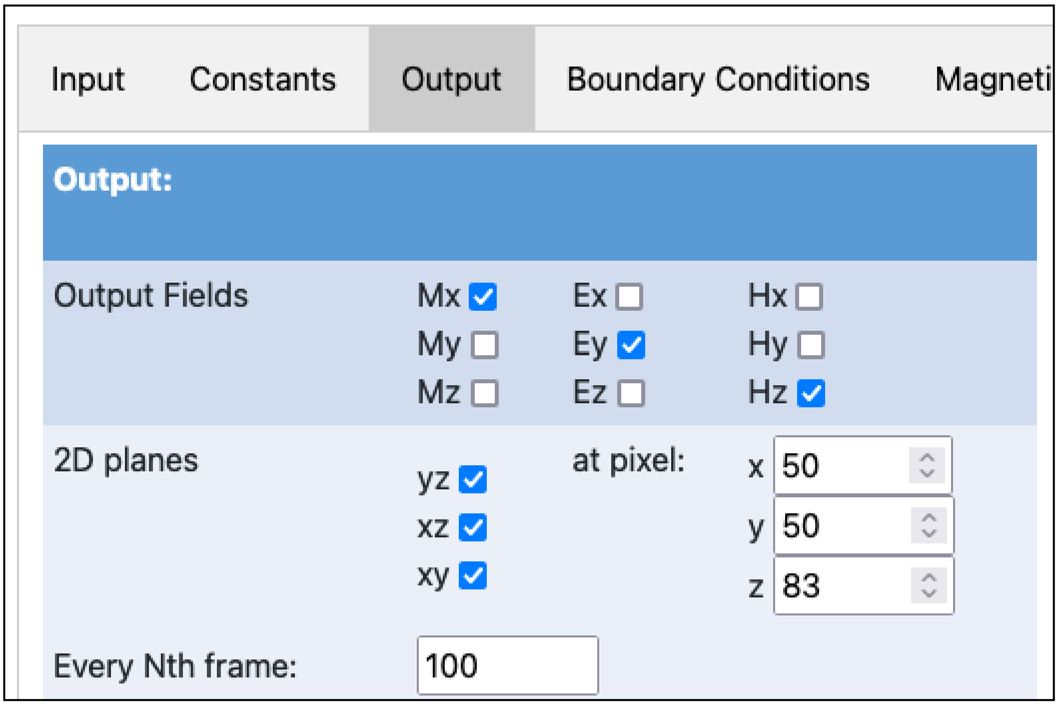


Figure 61: Showing the 'Every Nth frame' value on the meta script 'Output' pane

Type of output

Each output file can be split into a maximum of three files, based on the desired 2D planes chosen in the meta script 'Output' pane (see the Figure above).

- yz
- xz
- xy

Displaying output as Heatmaps

Each outputted file can be displayed as a heatmap.

The 'Output Field' must be specified when displaying the heatmap. The 'Output Fields' are selected on the meta script 'Output' pane (see the Figure above) before running the solver.

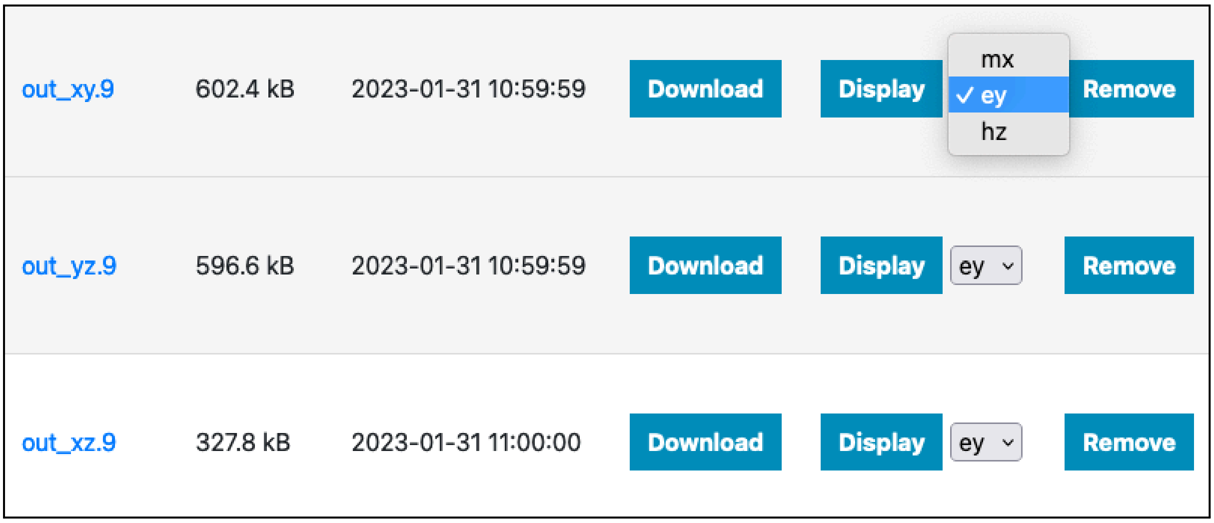


Figure 62: The ‘Display’ button and ‘Output Field’ select box

Output planes

out_xy

These contain data along the xy plane.

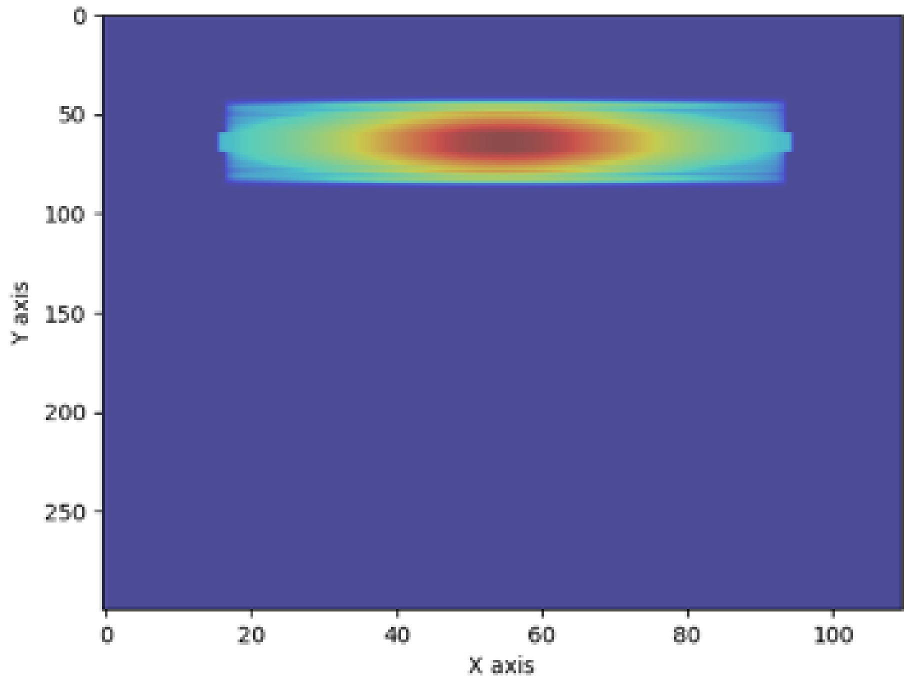


Figure 63: Heatmap of an ‘out_xy’ file create on the xy plane for a coaxial cable

out_yz

These contain data along the yz plane.

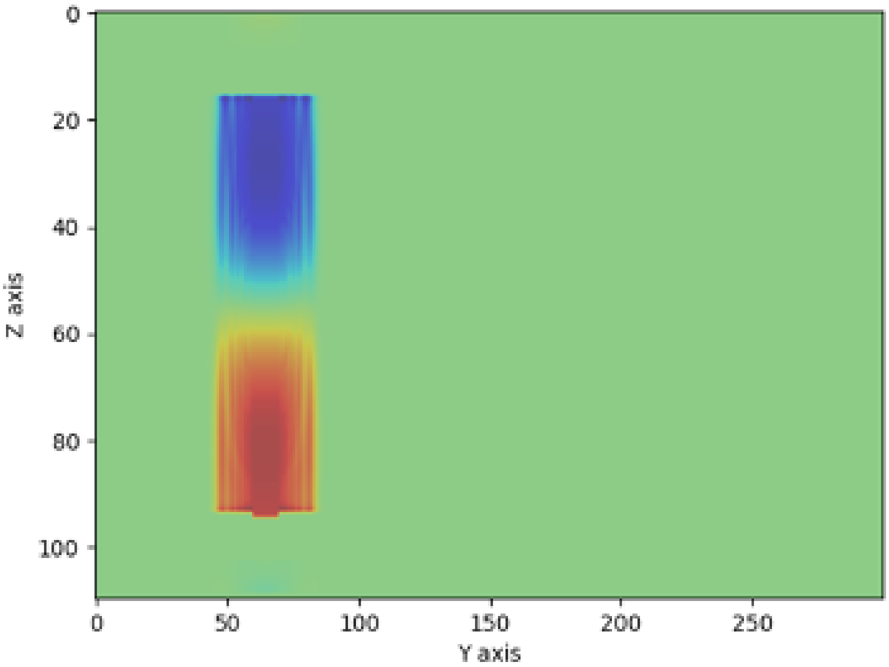


Figure 64 Heatmap of an 'out_yz' file create on the yz plane for a coaxial cable

out_xz

These contain data along the xz plane.

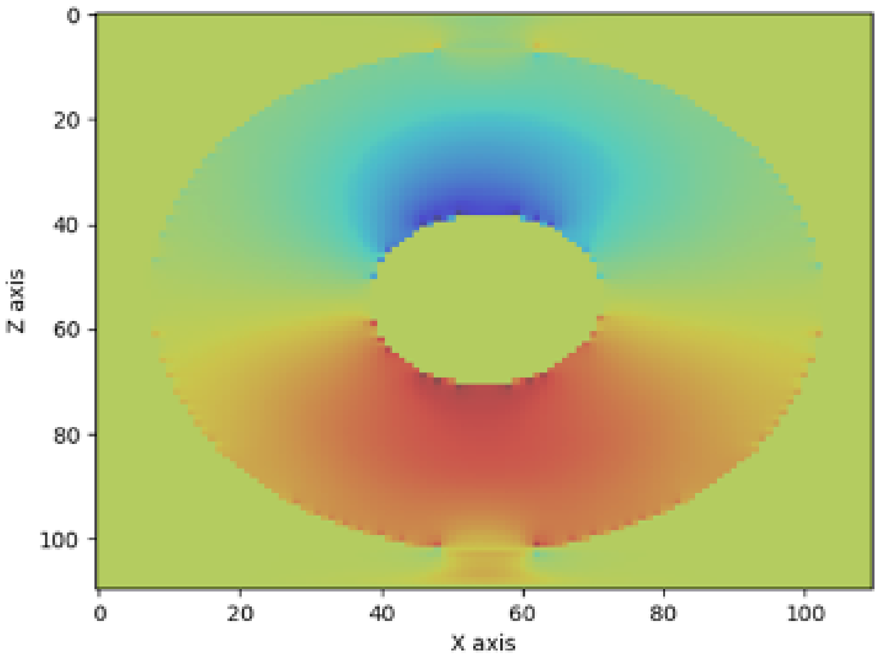
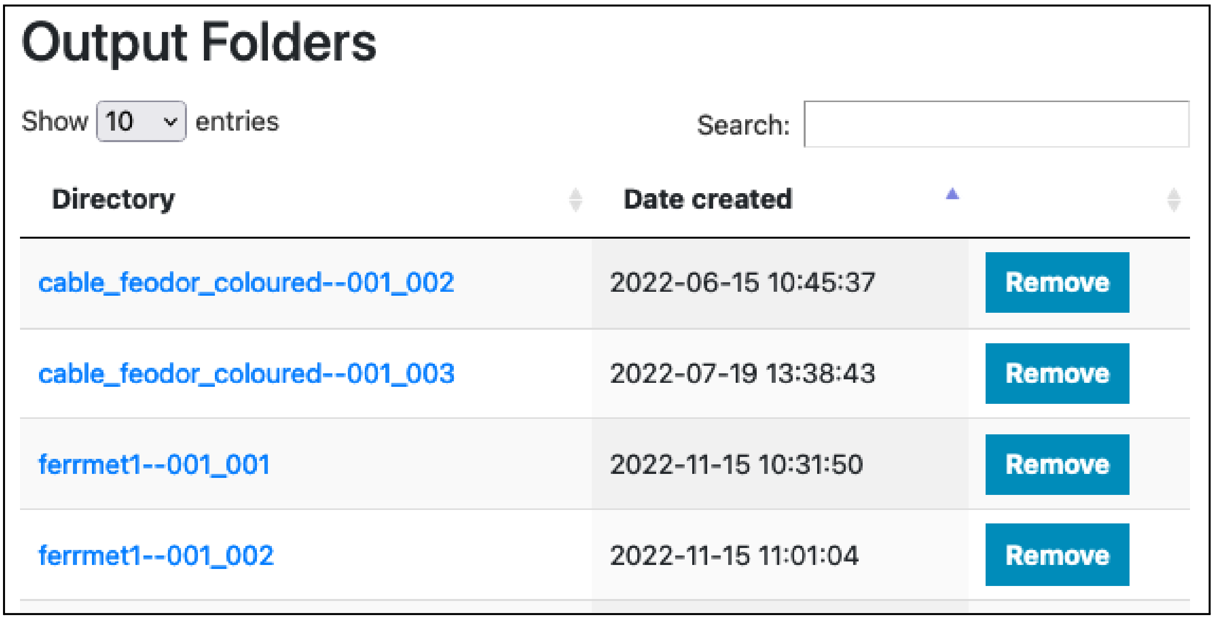


Figure 65: Heatmap of an 'out_xz' file create on the xz plane for a coaxial cable

The Output Directory List

The Output section can be revisited by clicking on the ‘Output’ tab in the top menu:



Output Folders

Show 10 entries

Search:

Directory	Date created	
cable_feodor_coloured--001_002	2022-06-15 10:45:37	Remove
cable_feodor_coloured--001_003	2022-07-19 13:38:43	Remove
ferrmet1--001_001	2022-11-15 10:31:50	Remove
ferrmet1--001_002	2022-11-15 11:01:04	Remove

Figure 66: The list of output directories

Running the Solver from the Command Line

The solver can be run directly from the command line. The following files are required from the GUI which are located in the following directories:

Matrix File:
/input_files/<<user id>>/

Metascript parameters File:
/metascript/<<user id>>/

NB. The user id can be obtained from the User table in the database. The ID of the 'admin' user is 1.

To run the solver via the command line, use the following command from the project root directory (inside maxllg-gui):

c_code/solver_icc /path/to/metascript/file /path/to/input/file /path/to/output/directory/ x_dimension z_dimension y_dimension

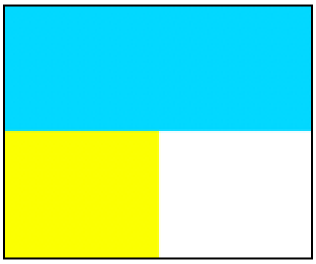
For Windows and Macs, use the solver_gcc compilation. Paths can be full or relative to the current location.

E.g.
c_code/solver_icc metascript/cable_feodor_coloured--001 input_files/cable_feodor_coloured--001 output/cable_feodor_coloured--001_002/ 110 300 110

Appendix

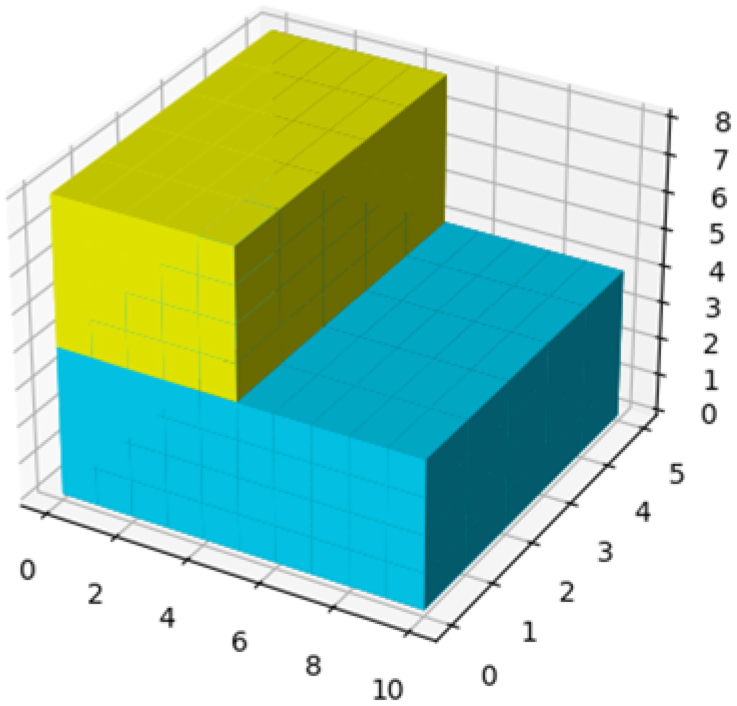
Creating the same matrix file in 2D and 3D

2D (using GIMP)



The X axis is horizontal
The Z axis is vertical
The Y axis (not seen in a 2D image) is the depth

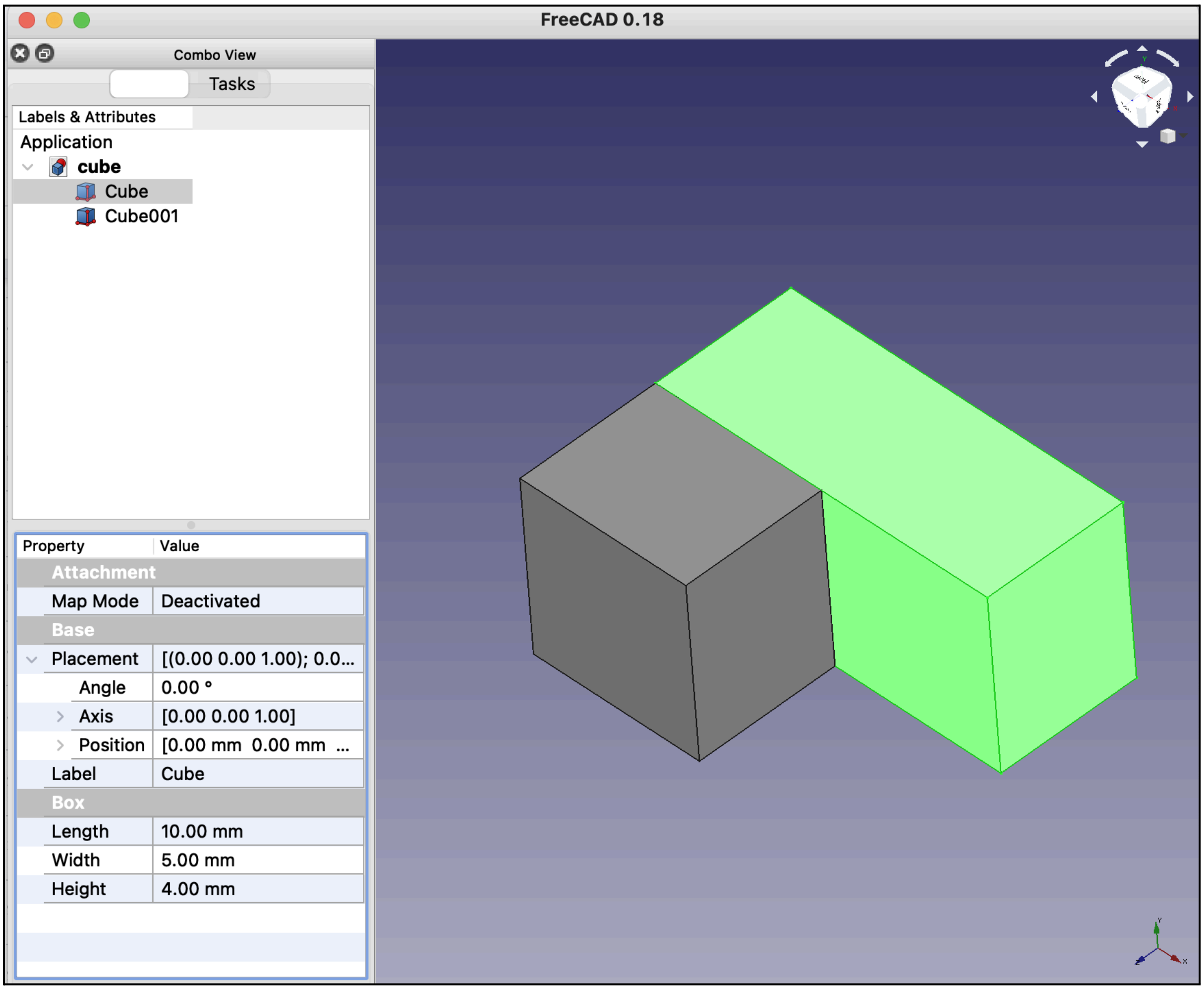
For an image of x=10, z=8 and y=5 this creates the following 3D matrix file rendered image:



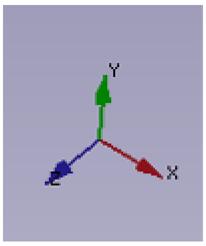
NB. The 3D image above appears to be ‘flipped’ along the x axis. This is because image editing software such as GIMP has it’s 0,0 pixel at the top left of the image. MatPlotLib, the python library that renders the matrix file in 3D, sets the 0,0,0 point in the bottom left. This means that, although the images are the same, they appear to be a mirror image of each other.

3D (using FreeCAD)

- The X axis is the 'length
- The Z axis is the 'height
- The Y axis is the 'width



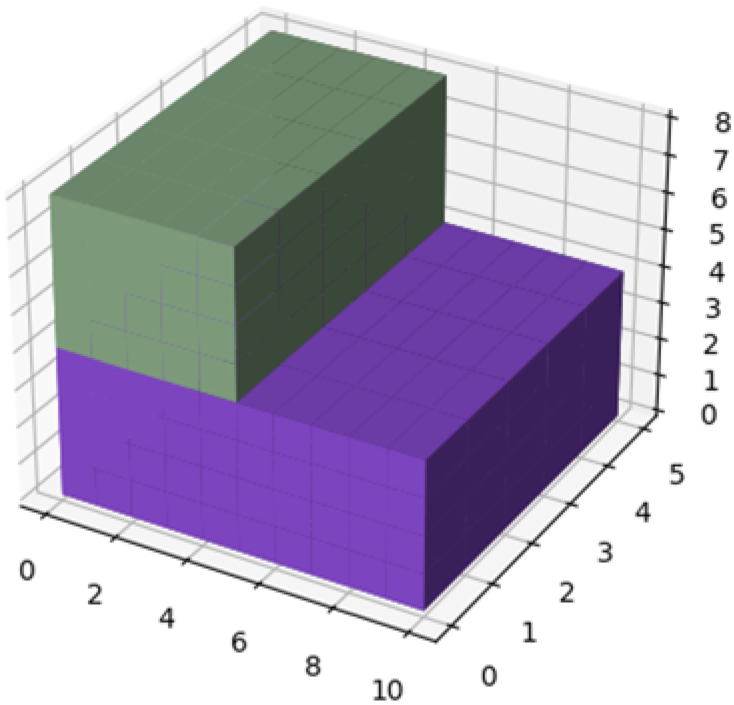
These axis can also be seen in the bottom right corner of the FreeCAD window:



(The obscured blue axis label is showing 'z')

The 'cube' FreeCAD and step files are included in the download for testing purposes.

This creates the following 3D matrix file rendered image:



0
2
4
6
8
10
0
1
2
3
4
5
0
1
2
3
4
5
6
7
8