



DOCUMENTATION

User Manual

How to build, run and analyse electromagnetic simulations in WaveFEM, from importing a model to exporting results.

CONTENTS

About WaveFEM	1	Simulation Settings	22
Installation	2	Running Simulations	25
The Interface	3	Results	26
How WaveFEM Works	8	Design Variants	31
Workflow	9	Horn Antenna	32
Importing Models	10	Microstrip Filter	34
Materials	14	Scripted Waveguide Bend	36
Ports	16	References	39

INTRODUCTION

About WaveFEM

WaveFEM is a three-dimensional electromagnetic simulator based on the finite element method (FEM). It is built for microwave and RF design: filters, waveguide components and antennas. Everything runs from one desktop application, from importing geometry to plotting fields, S-parameters and radiation patterns.

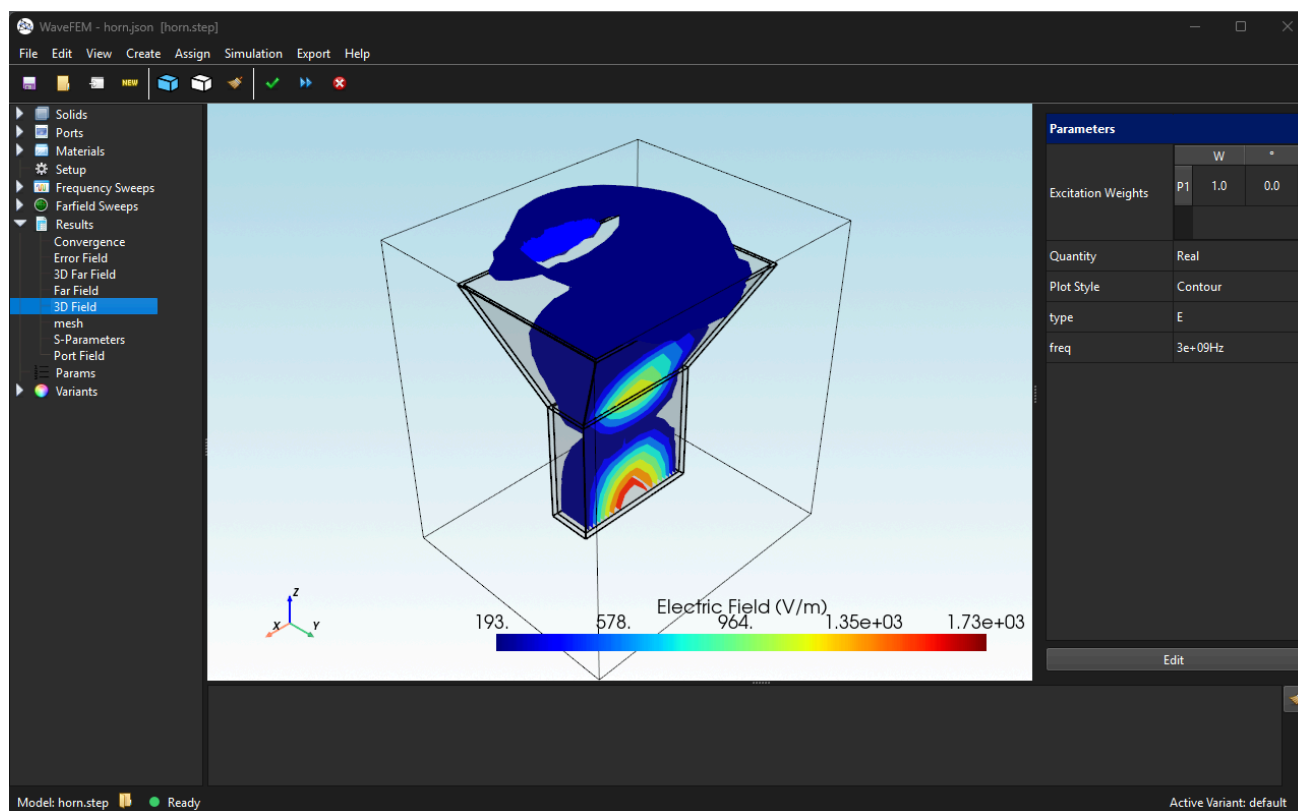


Figure 1 — Electric field inside a horn antenna

This manual covers

- Installing and starting WaveFEM.
- Importing models from STEP, DXF, GDS and Python (CadQuery) scripts.
- Setting up materials, ports, sweeps and solver settings.
- Running simulations and reading the console.
- Viewing, comparing and exporting results.
- Worked examples you can follow end to end.

It does not teach electromagnetic theory or derive the solver's equations; see the references for that.

GETTING STARTED

Installation

WaveFEM needs a 64-bit x86 PC running Windows 11. Unzip the download anywhere; no installer is needed. The folder contains:

assets/	Icons used by the program.
core/	The simulation engine.
gui/	The user interface.
examples/	Example projects and their model files.
run.bat	Starts WaveFEM.

Run **run.bat**. WaveFEM opens with an empty project:

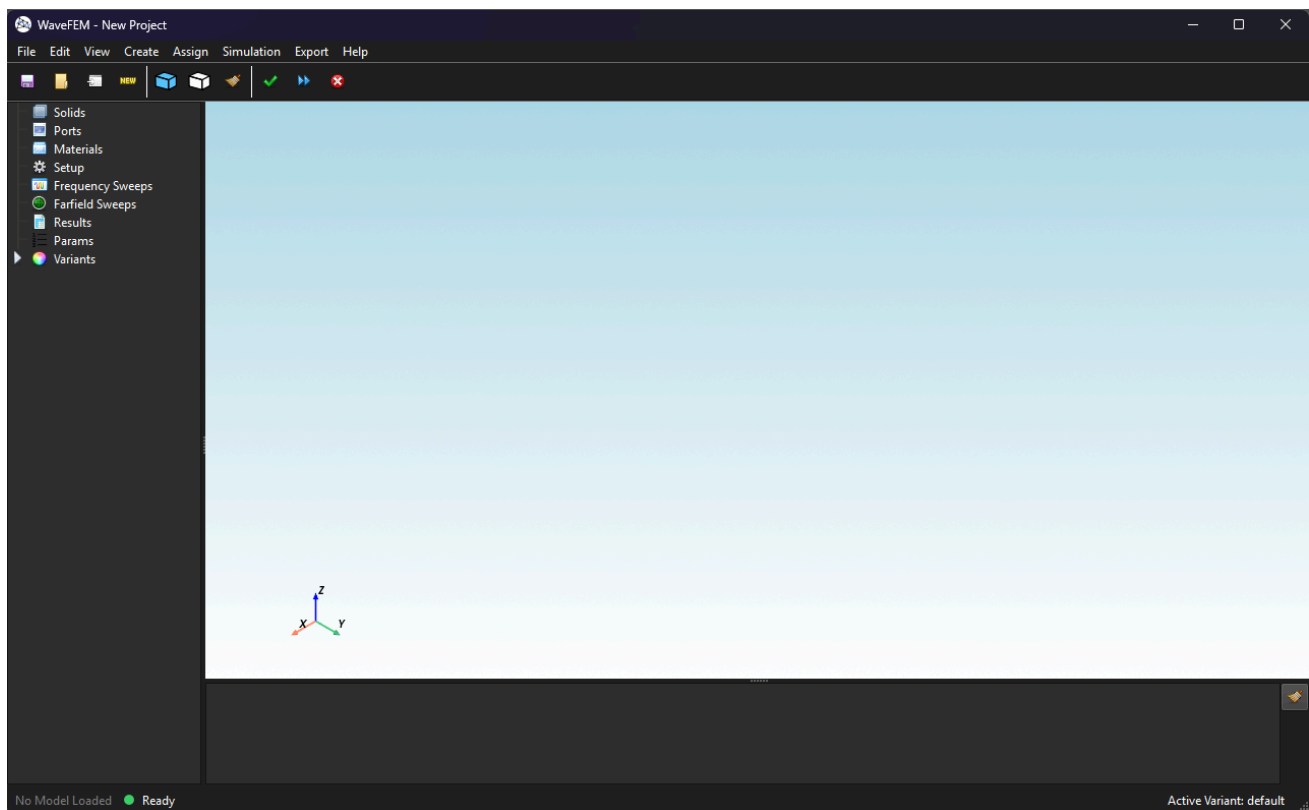


Figure 2 — WaveFEM after start-up

To explore, open an example with **File > Open** and pick a .json file in **examples/**. The [worked examples](#) walk through complete simulations.

GETTING STARTED

The Interface

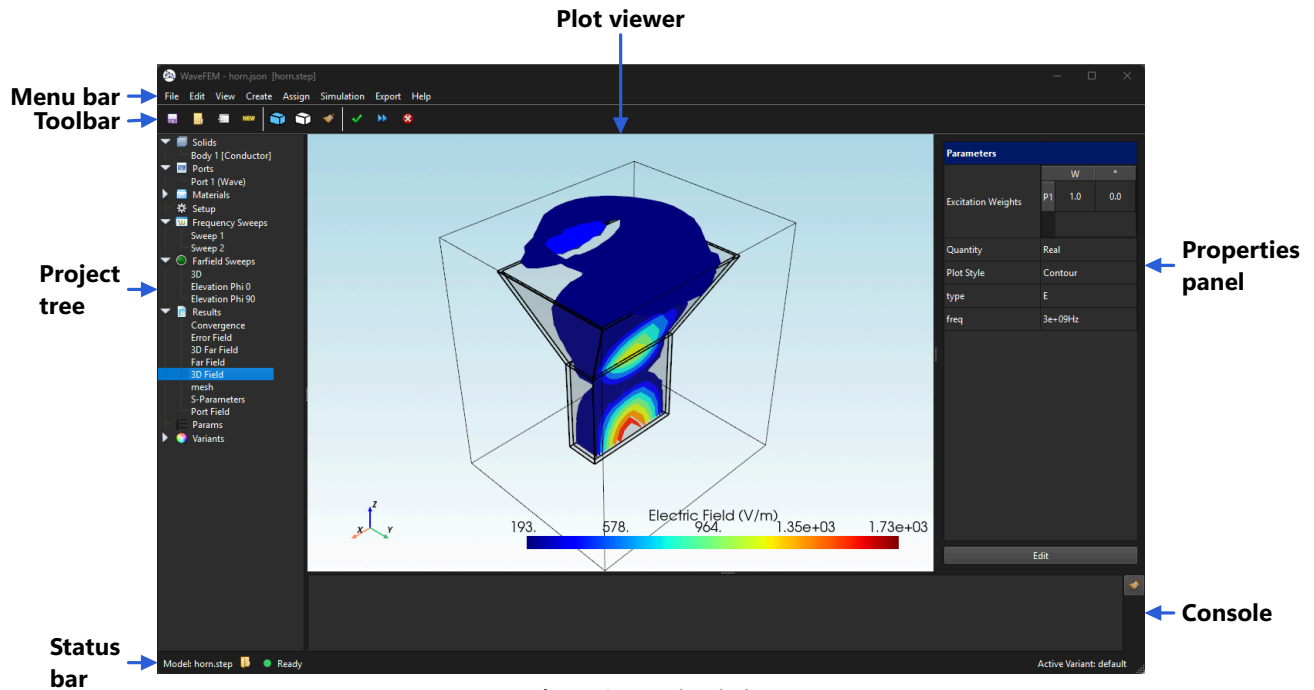
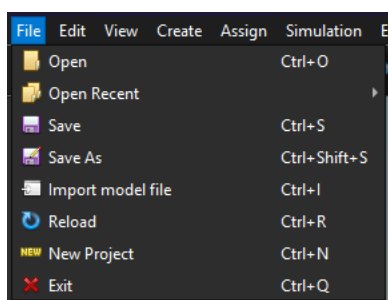


Figure 3 — Main window

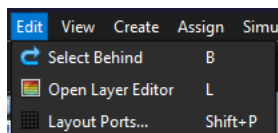
Menu bar	Every command, grouped by task. Most have a keyboard shortcut.
Toolbar	One-click access to the most used commands.
Project tree	Everything in the project: solids, ports, materials, settings, sweeps, results and variants. Click an item to inspect or edit it.
Plot viewer	The 3D model, or the plot of the selected result.
Properties panel	Settings of the selected tree item, with buttons to edit it.
Console	Messages from imports, validation and running simulations.
Status bar	Loaded model file, busy indicator and the active design variant.

Menus



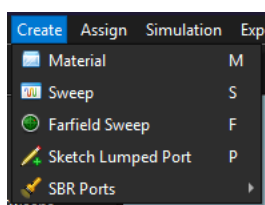
File

Open, save and create projects; import or reload the model file.



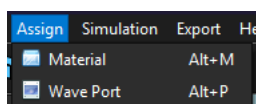
Edit

Select Behind (B) cycles to the face behind the selected one; **Open Layer Editor** edits the stack-up of a planar design and **Layout Ports...** draws ports on its layout.



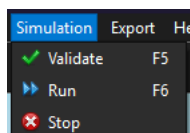
Create

New materials, frequency sweeps, far-field sweeps and lumped ports.



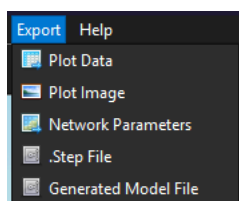
Assign

Turn the selected face into a wave port, or give the selected solids a material.



Simulation

Validate the setup, run the simulation, or stop it.



Export

Save the current plot's data or image, Touchstone S-parameters, or the model as a STEP file.

The **View** menu switches the model between shaded and wireframe display.

Toolbar



Figure 4 — Toolbar buttons

- | | |
|---------------------|--------------------|
| 1 Save | 2 Open |
| 3 Import model file | 4 New Project |
| 5 Shaded View | 6 Wireframe View |
| 7 Clear Plot | 8 Validate |
| 9 Run Simulation | 10 Stop Simulation |

The plot viewer has its own toolbar when a 2D plot is shown (zoom, pan and axis settings).

Project tree

Items with an arrow expand to show their contents. Click an item to show its settings in the properties panel or, for results, to plot it. Click empty space in the tree or the viewer to clear the selection.

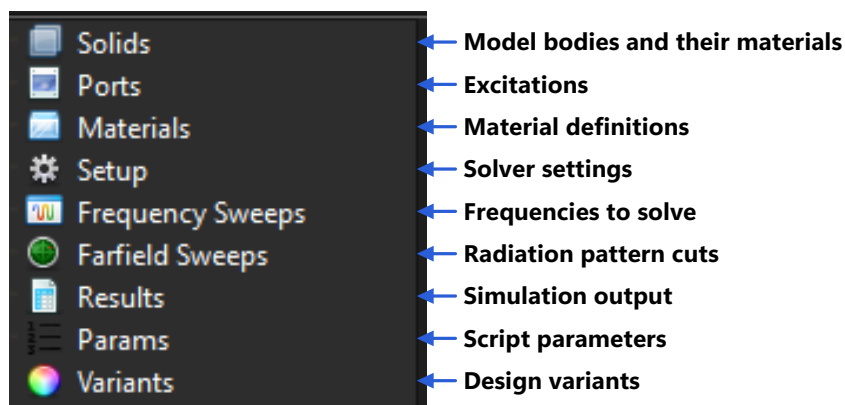


Figure 5 — Project tree

Plot viewer

The viewer shows the 3D model until a result is selected. Drag to rotate, scroll to zoom. Right-click the model to select a face; click a solid in the tree to select a body.

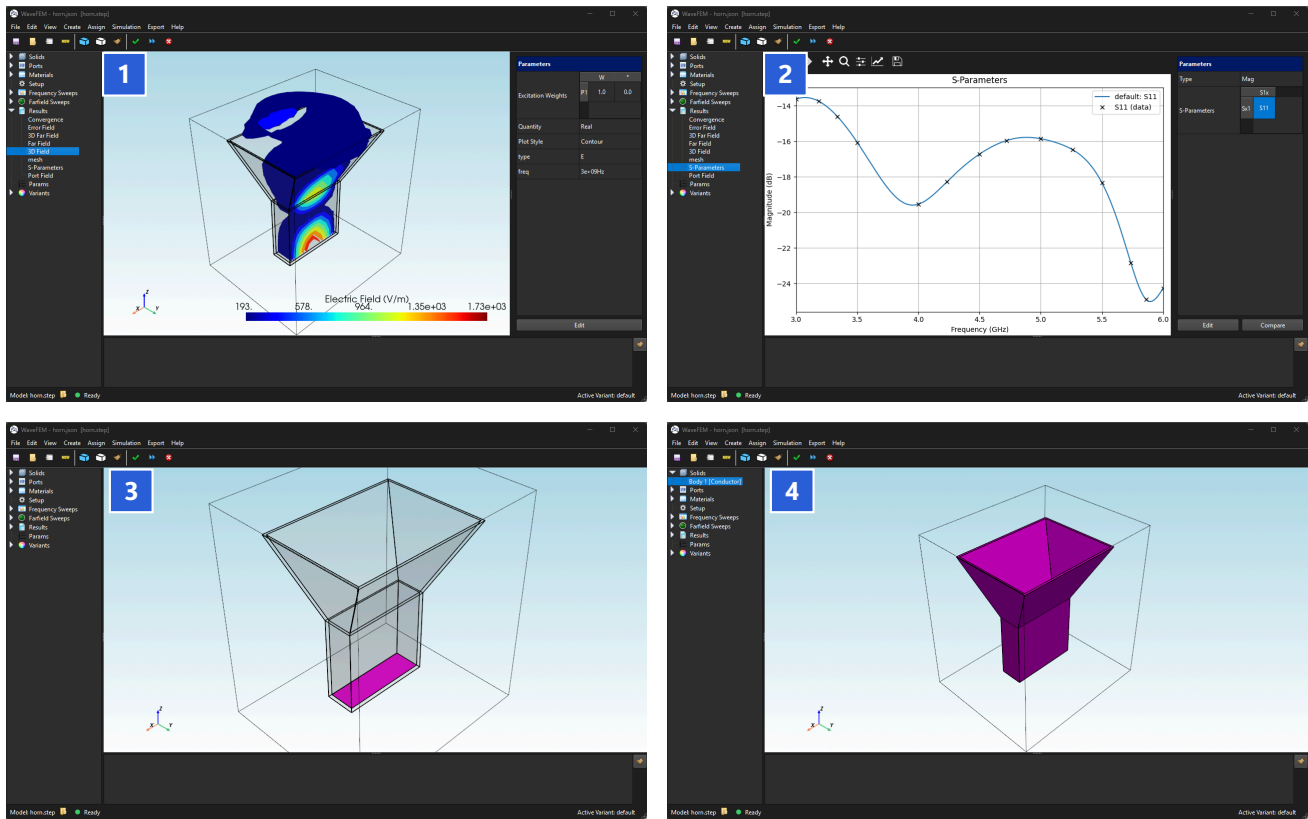


Figure 6 — Viewer modes: (1) 3D field, (2) 2D plot, (3) selected face, (4) selected solid

Console

The console reports what the program is doing: files loaded, validation results, and the progress of a running simulation. Errors are shown in red.

```
Validation Summary for Variant 'default':
Setup is Valid!
```

Figure 7 — Console after Simulation > Validate

Parameter tables

Settings are edited in tables like the one below. Related settings are grouped into tabs. Hover over a name for a description; greyed-out rows are disabled by another setting and their tooltip says which.

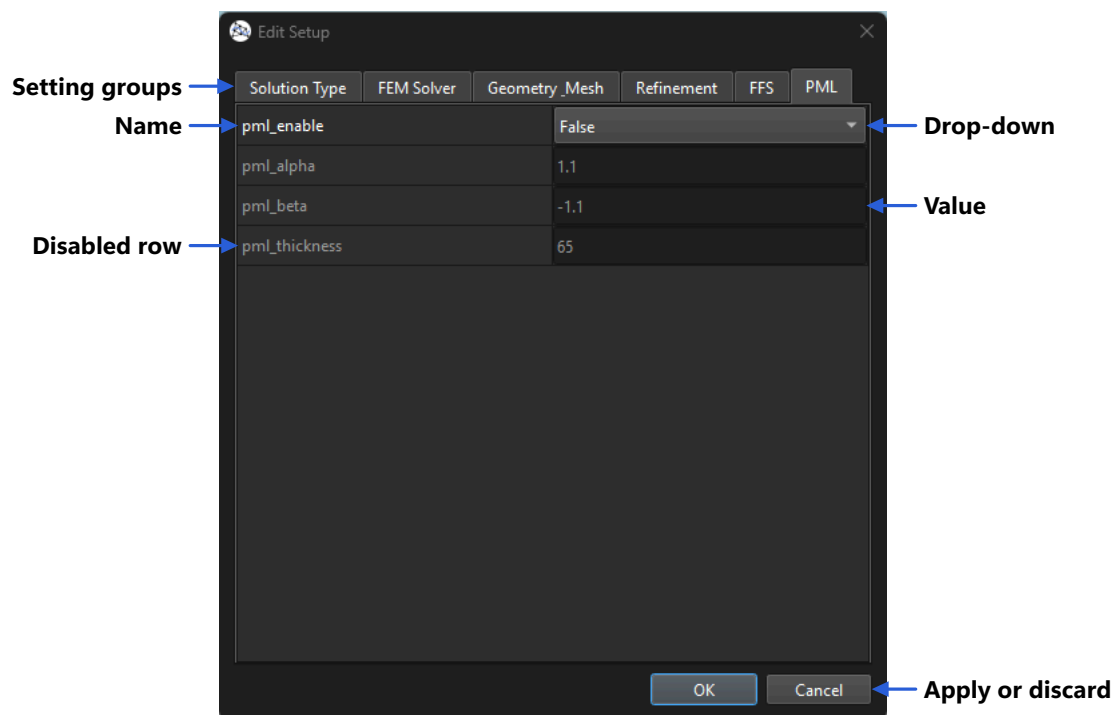


Figure 8 — A parameter table (Edit Setup, PML tab)

BACKGROUND

How WaveFEM Works

Physics

WaveFEM solves the source-free vector Helmholtz equation for the electric field:

$$\nabla \times \left(\frac{1}{\mu_r} \nabla \times \vec{E} \right) - k_0^2 \epsilon_r \vec{E} = 0$$

where μ_r and ϵ_r vary through the model. The magnetic field is computed afterwards:

$$\vec{H} = \frac{\nabla \times \vec{E}}{-j\omega\mu}$$

The finite element formulation follows [1].

Solution procedure

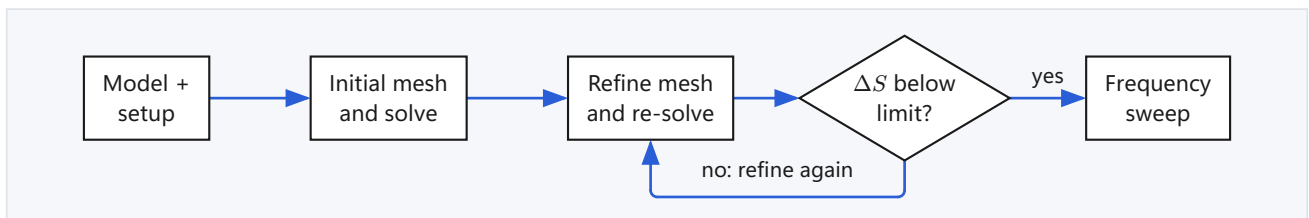


Figure 9 — Solution procedure

WaveFEM refines the mesh at one frequency until the S-parameters stop changing (ΔS below the limit). The final mesh is then used for every frequency in the sweeps. A sweep can solve fixed points or be **adaptive**: WaveFEM fits a rational function [2] to the solved points and adds points until the fit stops changing, then evaluates the fit across the band.

Ports

Ports are where energy enters and leaves the model.

- **Lumped ports** apply a voltage across a small rectangle between two conductors, such as the end of a microstrip trace and its ground plane.
- **Wave ports** inject the modes of a waveguide or transmission line cross-section. The face must be backed by a conductor.

Results

- **Electric and magnetic fields** in the model: where energy is stored, coupled or confined.
- **S-parameters**: reflection and transmission between ports, e.g. return loss (S_{11}) and insertion loss (S_{21}).
- **Far-field patterns**: directivity or gain against angle for radiating structures.
- **Convergence and error data**: how the adaptive mesh refinement progressed.

USING WAVEFEM

Workflow

A simulation follows the same steps every time. Each has its own section.

1. **Model** — import a STEP, DXF/GDS or Python file. Planar files need a layer stack-up.
2. **Materials** — create materials and assign one to every solid.
3. **Ports** — add wave ports on faces, sketch lumped ports, or draw layout ports on a planar design.
4. **Settings** — set the refinement frequency and mesh options, add frequency sweeps and, for antennas, far-field sweeps.
5. **Validate and run** — check the setup, save, run, and watch the console.
6. **Results** — plot fields, S-parameters and patterns; export what you need.
7. **Iterate** — create variants, change the design and compare results.

USING WAVEFEM

Importing Models

Choose File > Import model file and select the file. Pick the file type at the bottom right of the dialog.

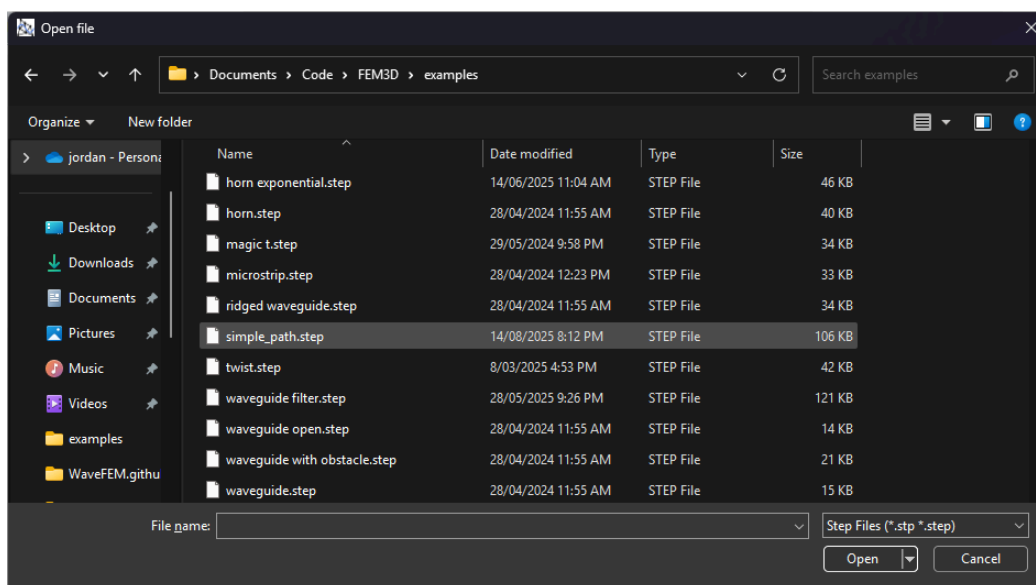


Figure 10 — Selecting a model file

Note Saving the project copies the model file next to the project file, so the project stays self-contained. The original file is not changed.

STEP files

A STEP file loads straight into the viewer. Each body becomes a solid in the tree.

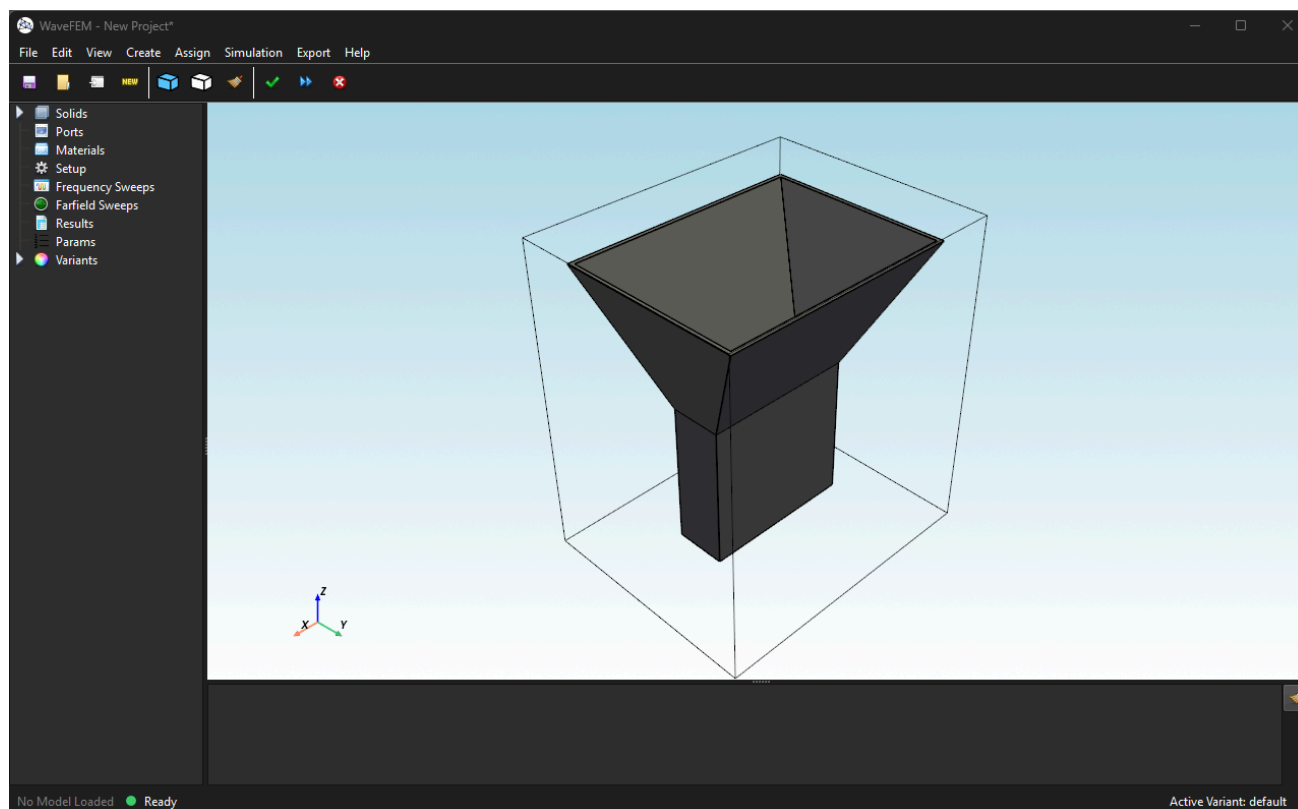


Figure 11 — Imported STEP model

Planar files (DXF and GDS)

DXF (PCB) and GDS (IC) files are 2D, so WaveFEM asks how to build them into 3D with the **Layer Stack Configuration** window.

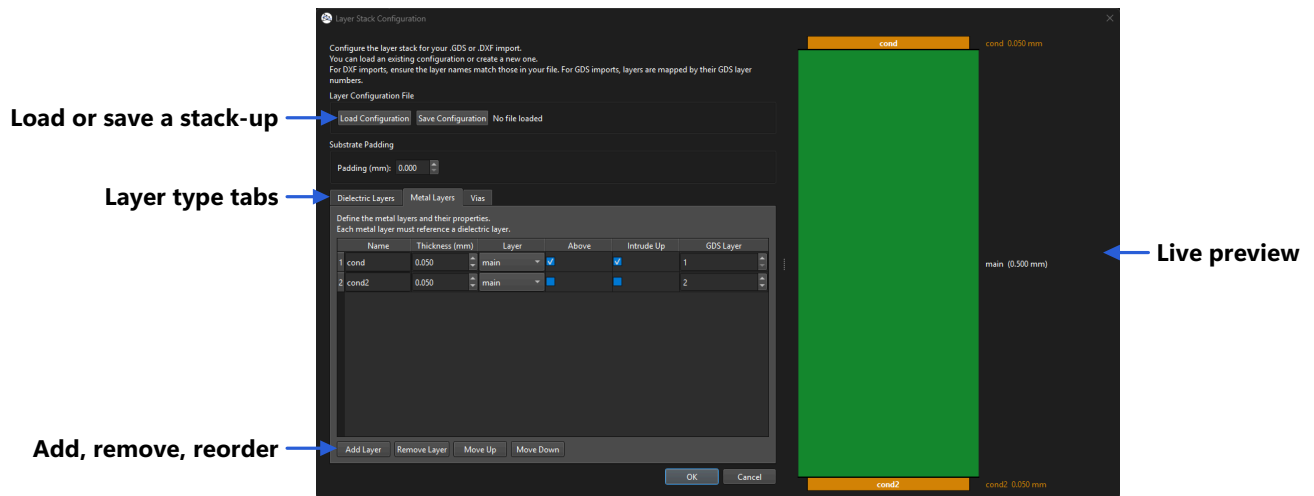


Figure 12 — Layer Stack Configuration

Dielectric layers	Substrates, listed bottom to top, each with a name and thickness.
Metal layers	Conductors. Each has a thickness, the dielectric it sits on, and the layer in the file it takes its shapes from (layer name for DXF, layer number for GDS). Above puts it on top of the dielectric instead of below; Intrude Up lets it extend into the dielectric above.
Vias	Vertical connections through a dielectric, taken from a file layer.
Padding	Extra substrate around the drawn shapes.

The stack-up is saved in the project. **Save Configuration** writes it to a `.json` file for use in other projects.

Note The DXF importer reads closed polylines (LWPOLYLINE, POLYLINE) and circles.

Python scripts

A Python file that builds a CadQuery model loads like a STEP file. Variables defined at the top of the script appear under **Params** changing one rebuilds the model immediately. Parameters are stored per variant.

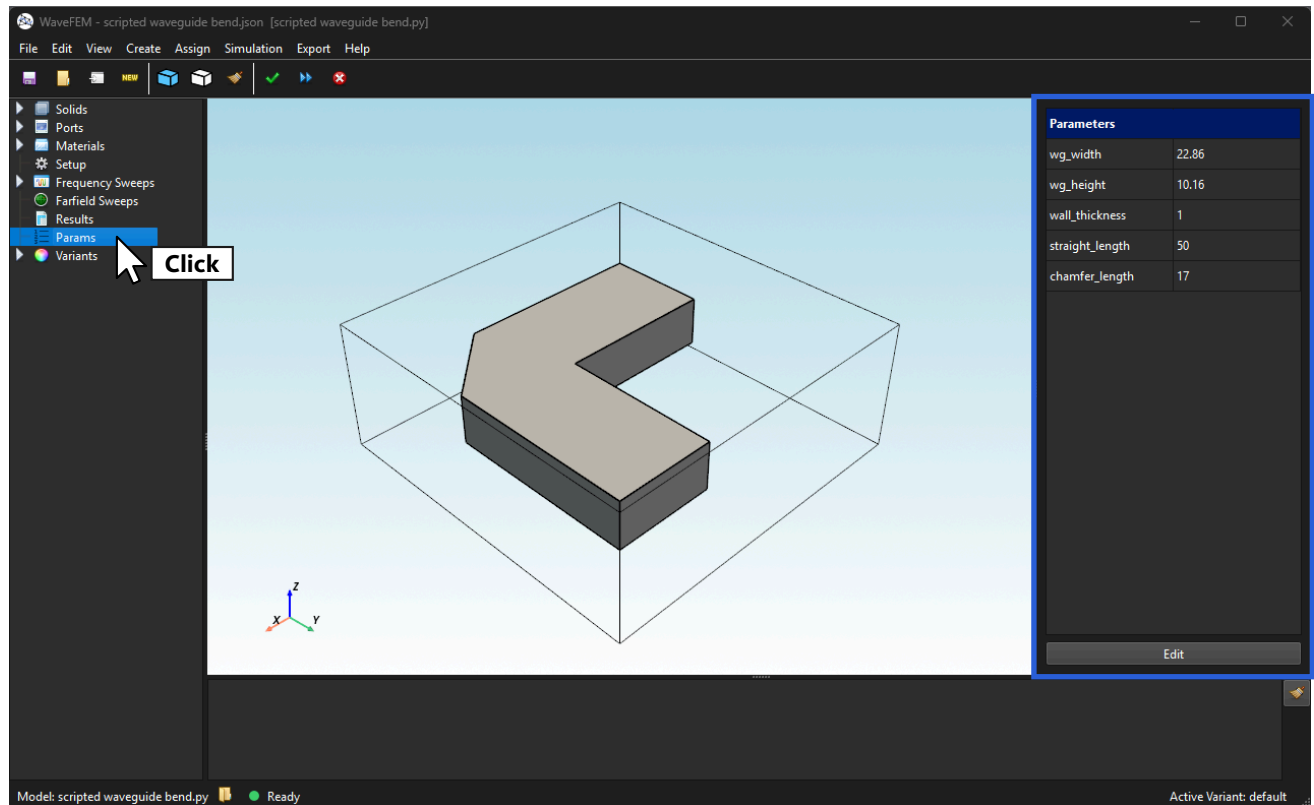


Figure 13 — Parameters of a scripted model

If the script fails, the console shows the Python error, for example:

```
Error Loading Model!  
unexpected indent (<cqscript>, line 12)
```

Note Debug scripts in a Python editor first; errors are easier to trace there.

Saving and opening projects

Projects are .json files: **File > Save** and **File > Open**. Keep the model file in the same folder as the project; if it is missing when a project opens, WaveFEM asks you to locate it.

USING WAVEFEM Materials

Every solid needs a material before a simulation can run.

Creating a material

Choose **Create > Material**, then copy one from the library or start from scratch.

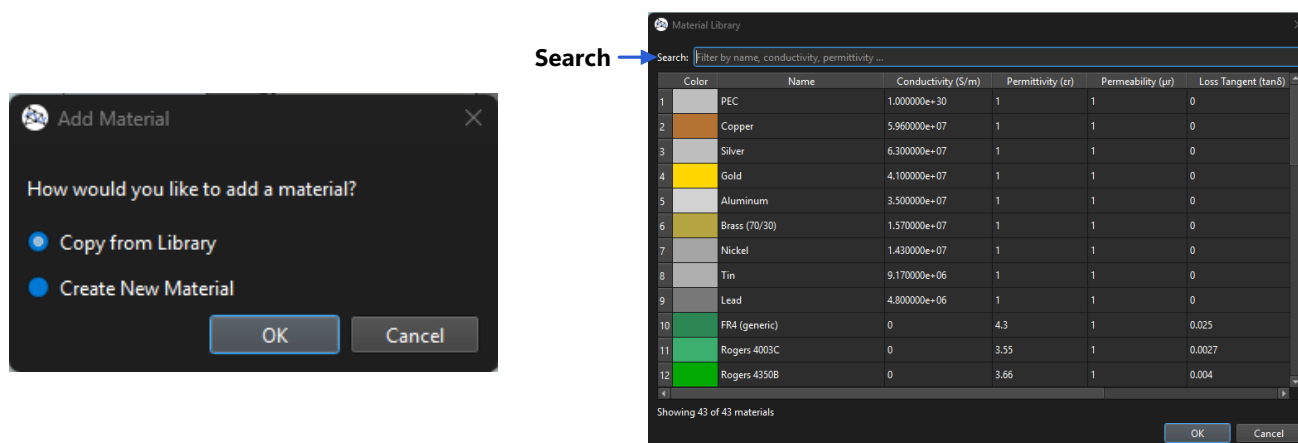


Figure 14 — Adding a material: choice dialog and library

The material opens in a parameter table:

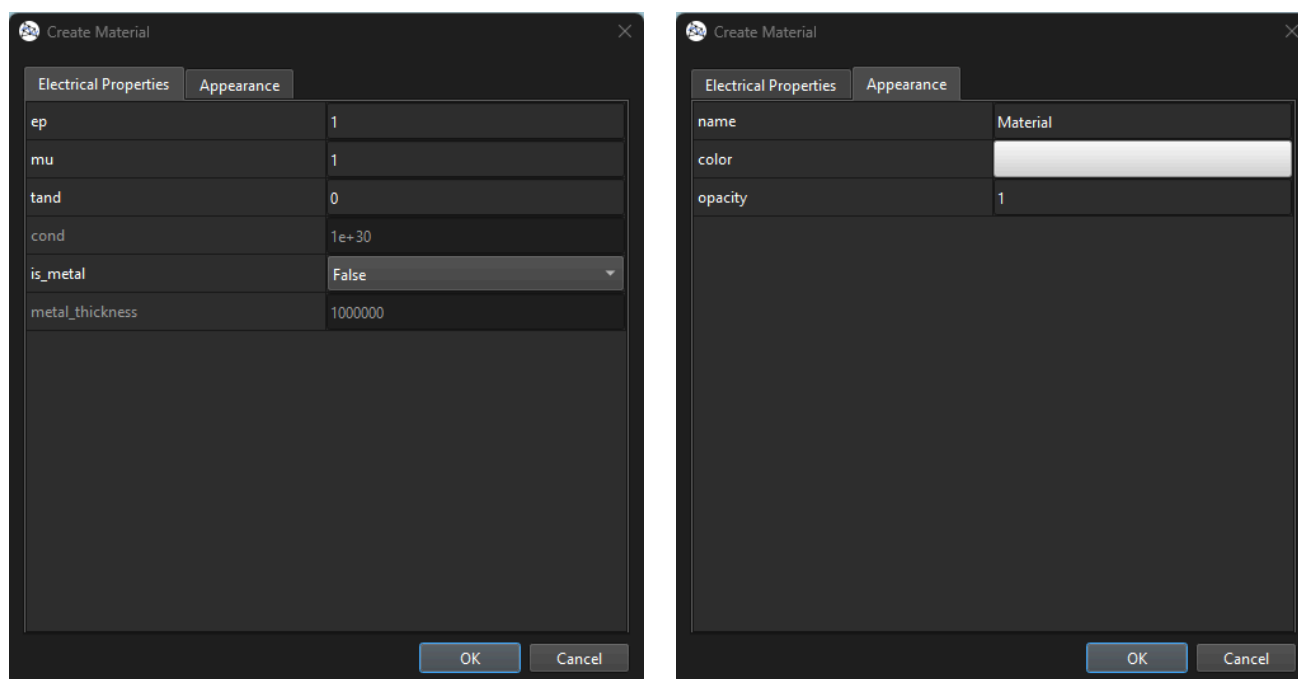


Figure 15 — Material properties

ep	Relative permittivity.
mu	Relative permeability.
tand	Dielectric loss tangent.
is_metal	Treat as a metal (see the note below).
cond	Conductivity in S/m (metals).
metal_thickness	Physical thickness of a metal, used when it is comparable to the skin depth.

name	Shown in the tree next to each solid that uses it.
color, opacity	Display colour and transparency (1 solid, 0 clear).

Note Metals are modelled with a surface impedance boundary and are not meshed inside. This is accurate when the skin depth is much smaller than the metal thickness. For thin or poorly conducting metals set `metal_thickness`.

Assigning materials

Click a solid in the tree (hold **Ctrl** or **Shift** to select several), then choose **Assign > Material** and pick the material. The tree then shows the material next to the solid, e.g. Body 1 [Conductor].

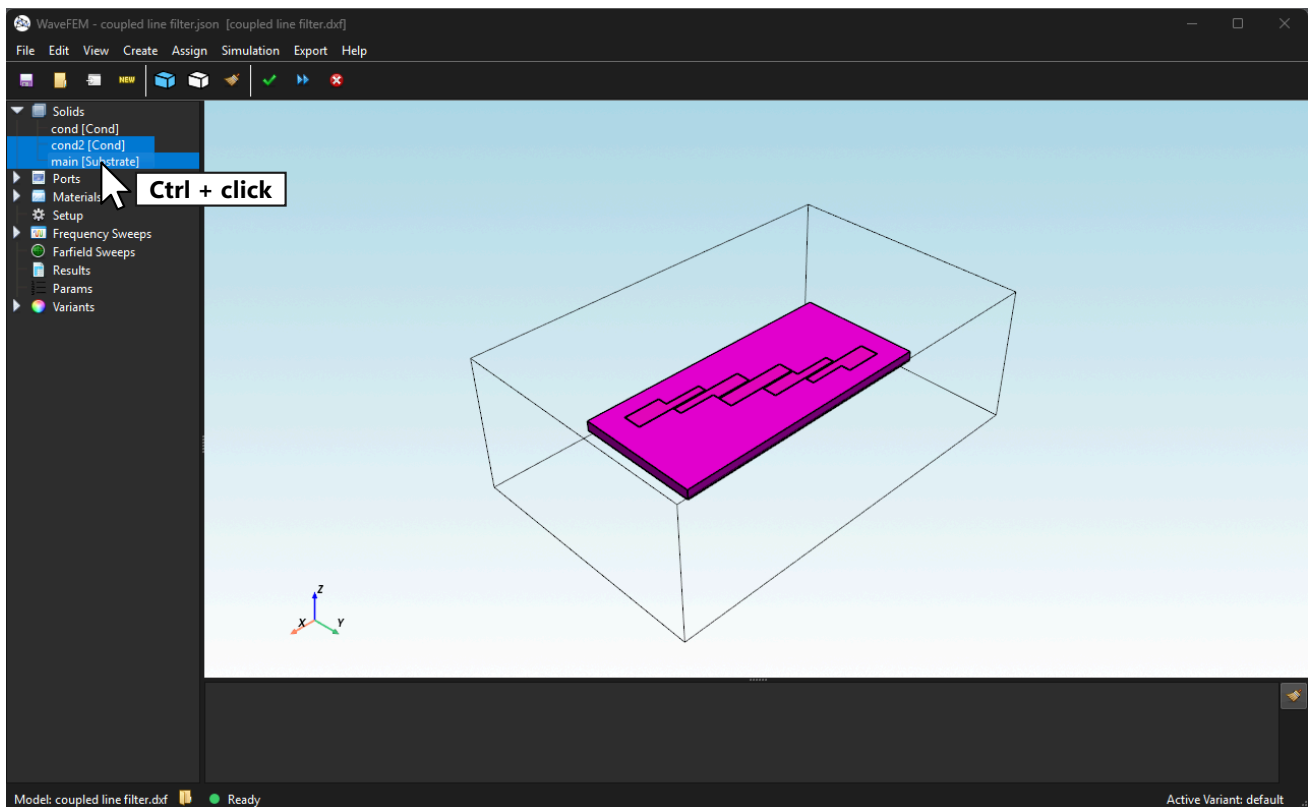


Figure 16 — Two solids selected

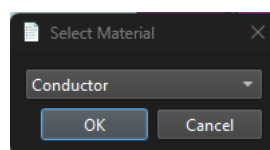


Figure 17 — Assign > Material

USING WAVEFEM

Ports

Wave ports

1. Right-click the port face in the viewer. If another face is in front, press **B** (**Edit > Select Behind**) until the right face is highlighted.
2. Choose **Assign > Wave Port**.
3. Check it: expand **Ports** and click the port to highlight it.

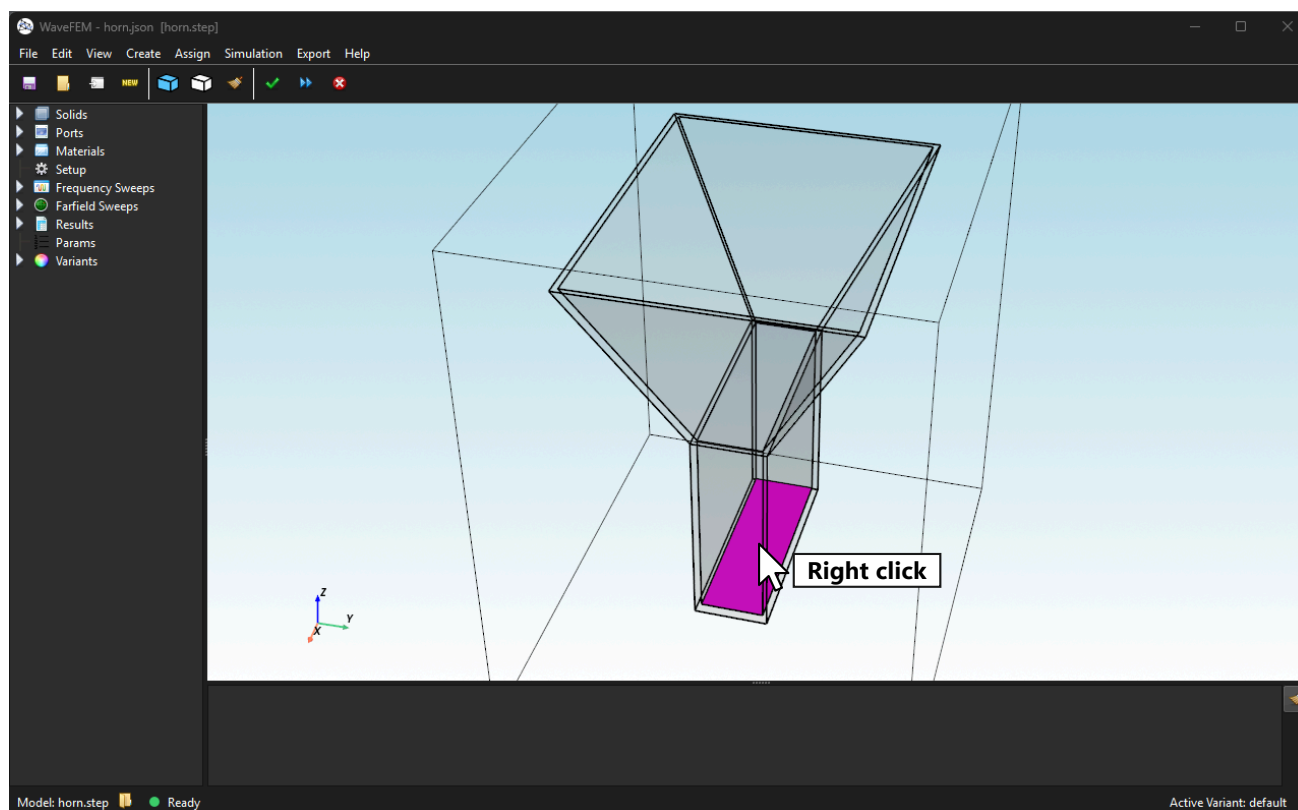


Figure 18 — Selecting a wave port face

Lumped ports

A lumped port is a rectangle drawn on a plane between two conductors.

1. Right-click a face that lies in the port's plane (here, the end of the feed line).
2. Choose **Create > Sketch Lumped Port** and keep **Parallel to face**.
3. In the **Sketch Drawer**, click **Rectangle**, then click the two corners. Points snap to the grid, corners and intersections. You can draw helper rectangles, but delete them so that exactly one remains.
4. Click **OK** and enter the port impedance.

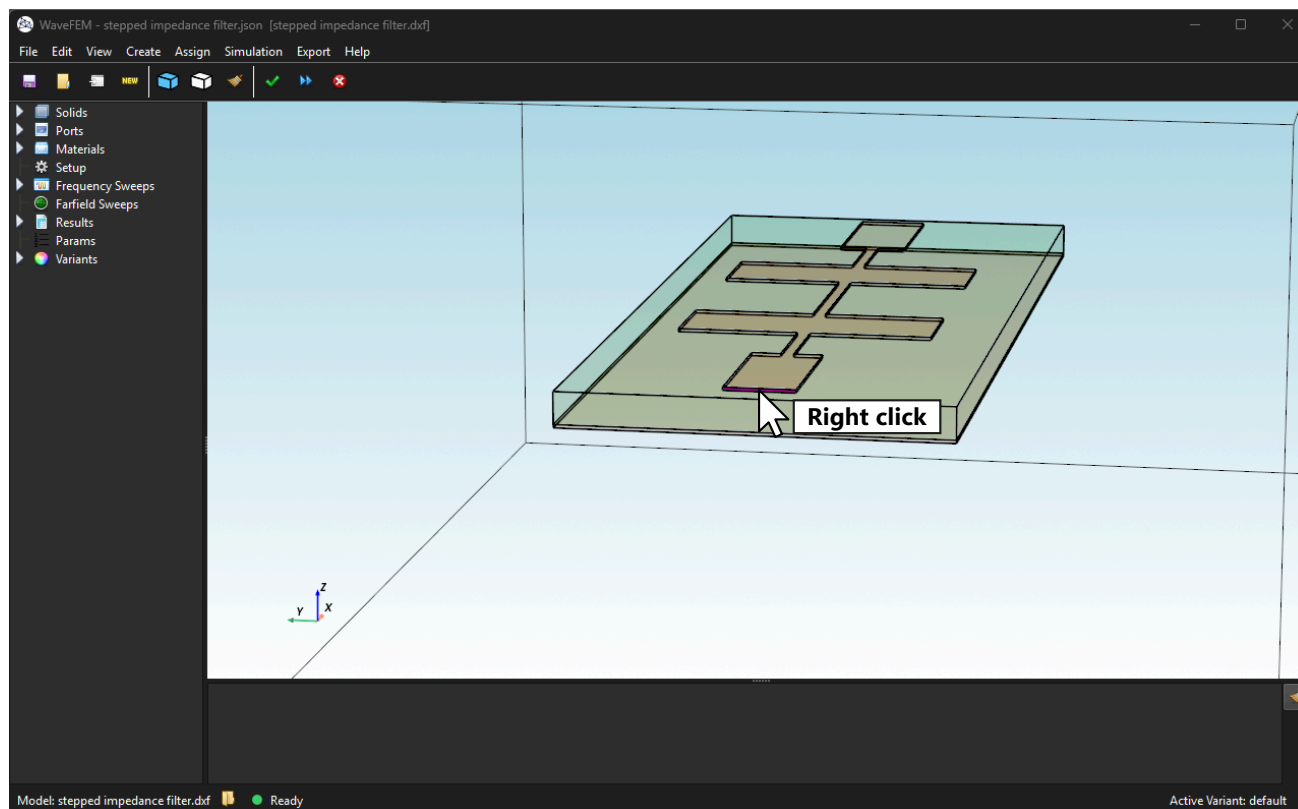


Figure 19 — Step 1: select a face in the port plane

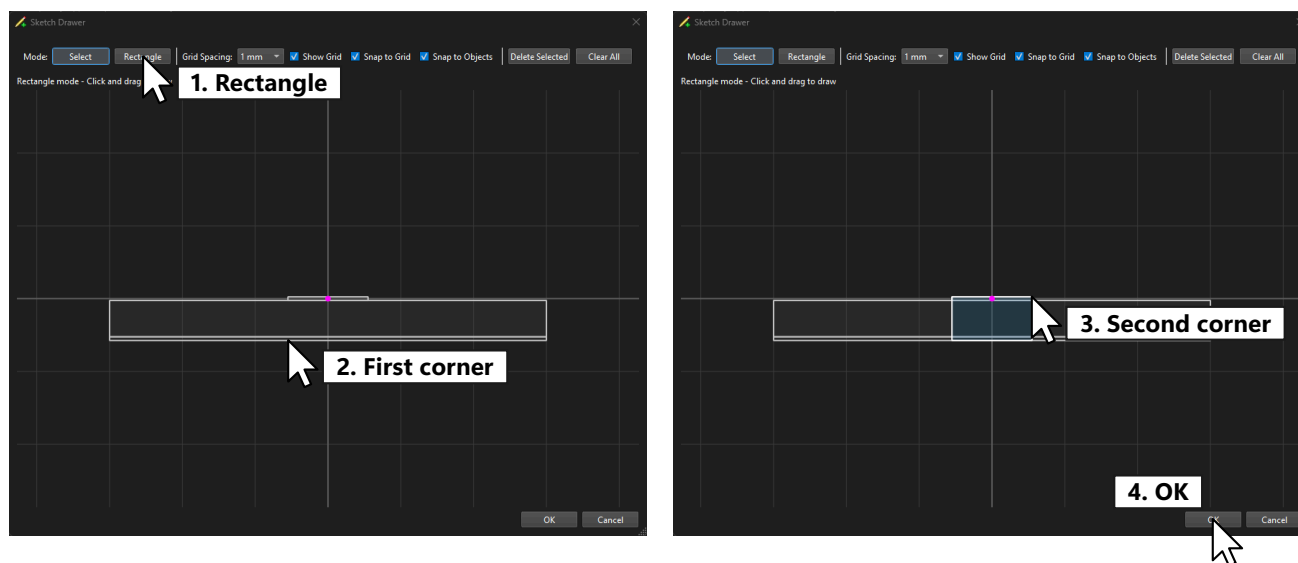


Figure 20 — Steps 3–4: drawing the port

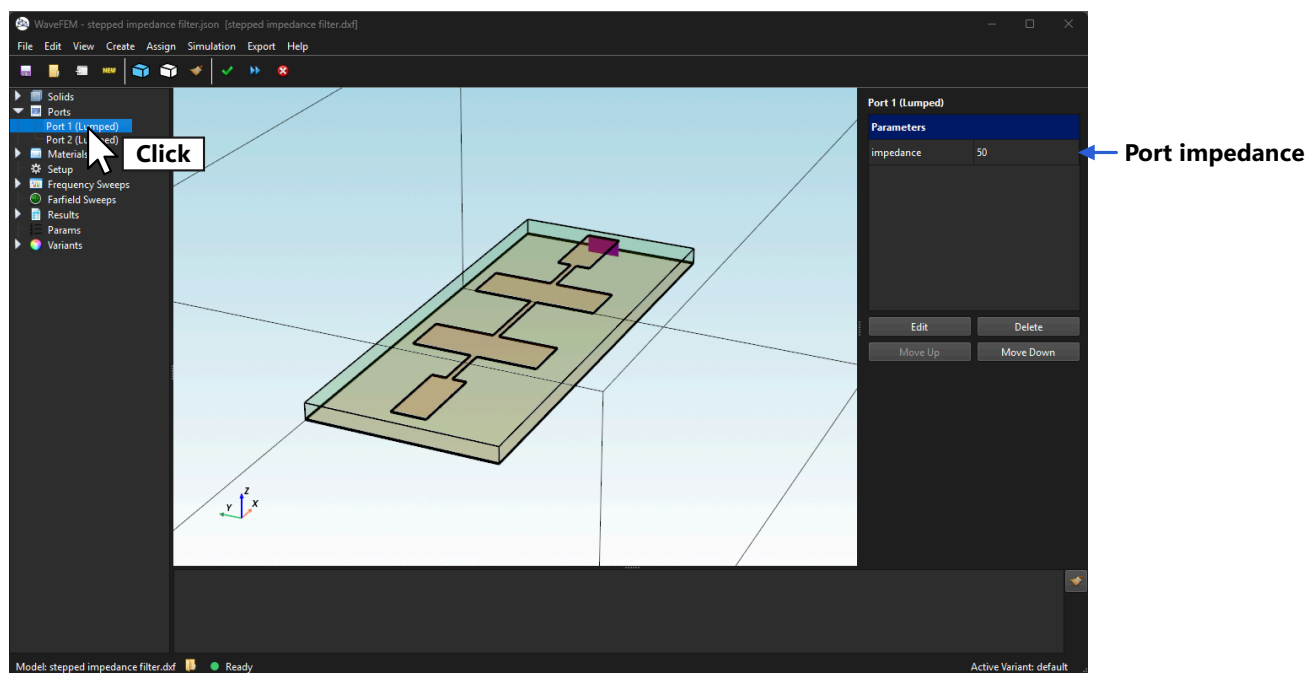


Figure 21 — The new port, selected in the tree

Note Lumped ports must be rectangles.

Layout ports

On a planar (DXF or GDS) design a lumped port can be drawn on the layout instead of sketched on a face. A layout port is a line in the layout plus a top and a bottom metal layer. WaveFEM builds the port rectangle from the stack-up: along the line, from the bottom face of the bottom layer to the top face of the top layer. Because it is not tied to a face, the port moves with the stack-up when layer thicknesses or the **Above** and **Intrude Up** options change, and it is kept when the model is rebuilt.

1. Choose **Edit > Layout Ports...** (**Shift+P**). The editor shows the metal layers from above.
2. With **Pick Edge** selected, click the edge of the trace where the port goes. The whole edge becomes the port.
The edge's layer is used as the top layer and a layer below it as the bottom layer.
3. Repeat for every port, then select each in the **Ports** list and check its layers and impedance.
4. Click **OK**. Every change made in the editor is applied together.

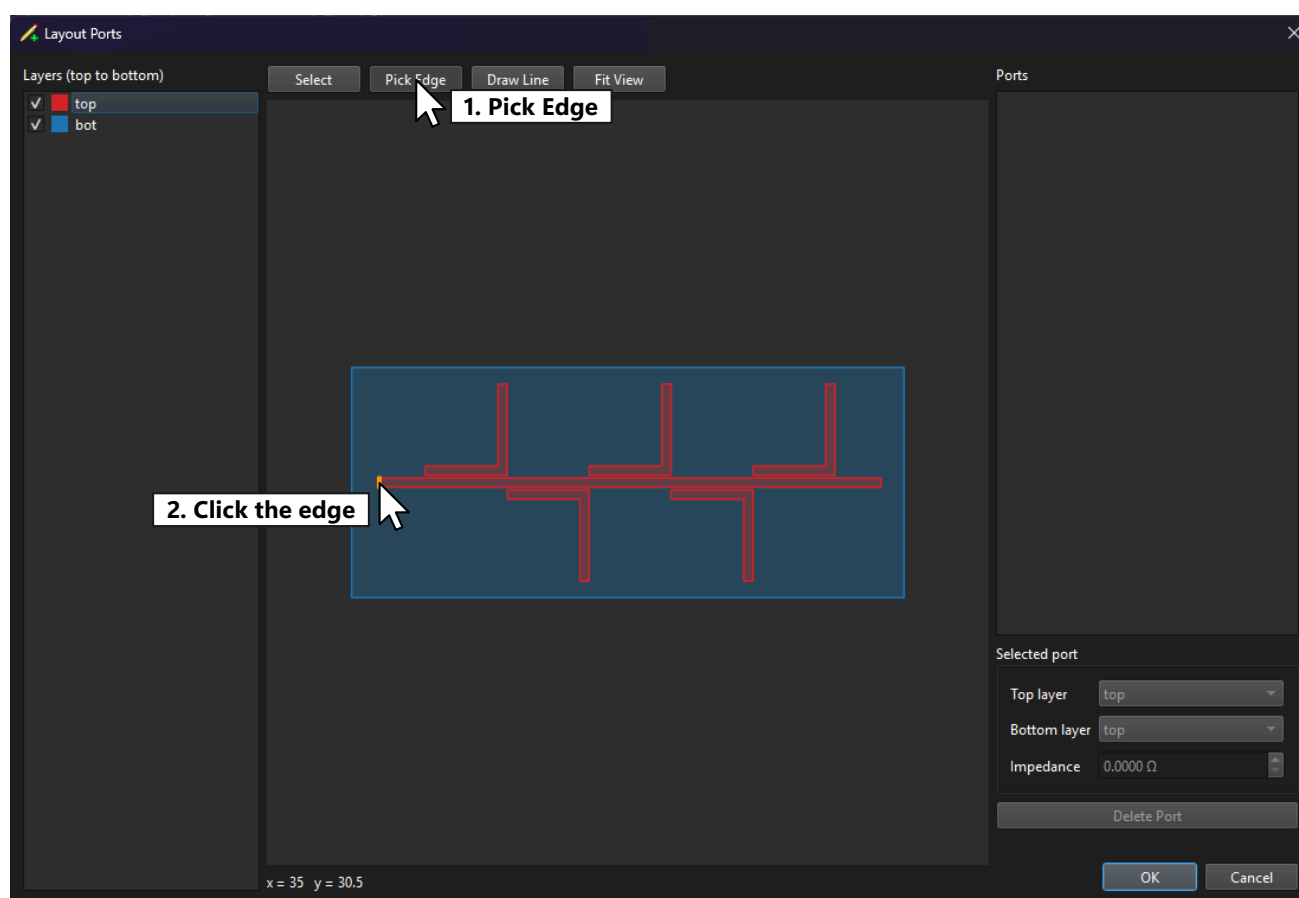


Figure 22 — Picking the edge at the end of the through line

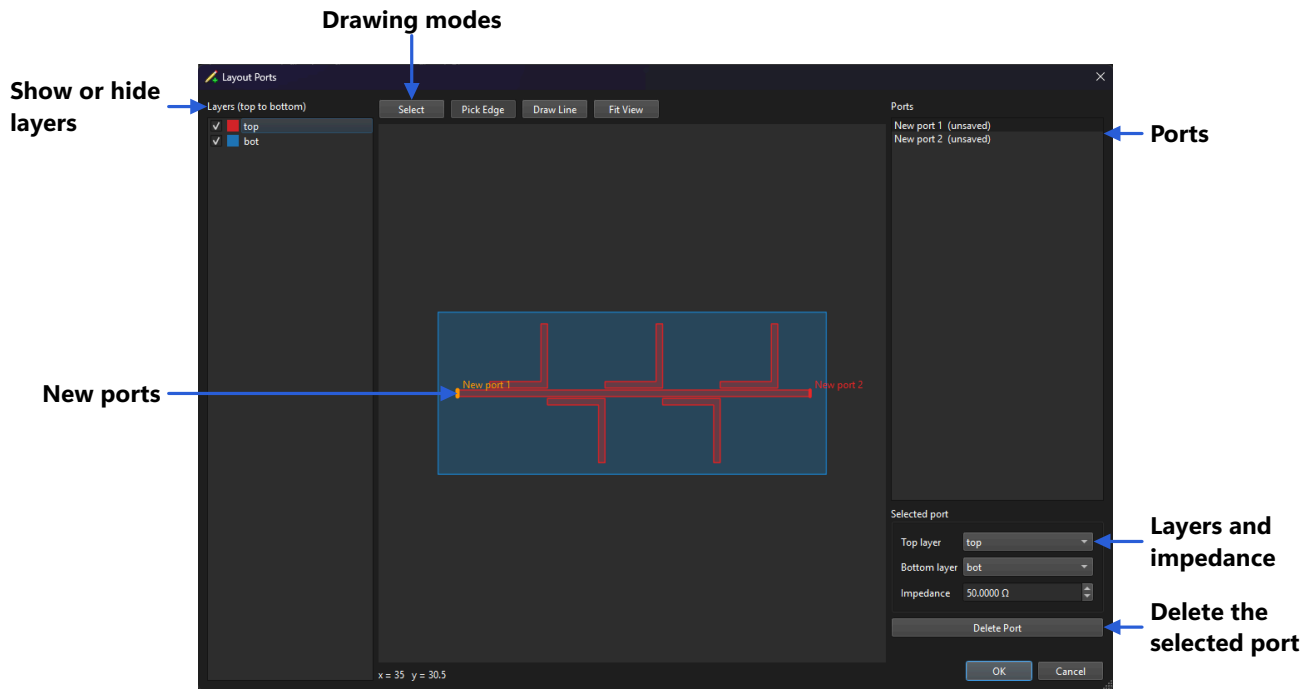


Figure 23 — Two layout ports, one at each end of the line

Select	Click a port on the layout to select it.
Pick Edge	Click a polygon edge to place a port along the whole edge.
Draw Line	Click the two ends of the port. Points snap to vertices, edge midpoints and edges. Esc cancels a line.
Fit View	Show the whole layout. The mouse wheel zooms; drag with the middle or right button to pan.
Layers	Untick a layer to hide it, so that clicks only pick edges of the layers you can see.
Top layer, Bottom layer	The metal layers the port connects. The bottom layer must lie below the top layer.

In the tree a layout port is named after its layers, e.g. Port 1 (Layout bot→top). **Edit in Layout...** in its properties panel reopens the editor with that port selected.

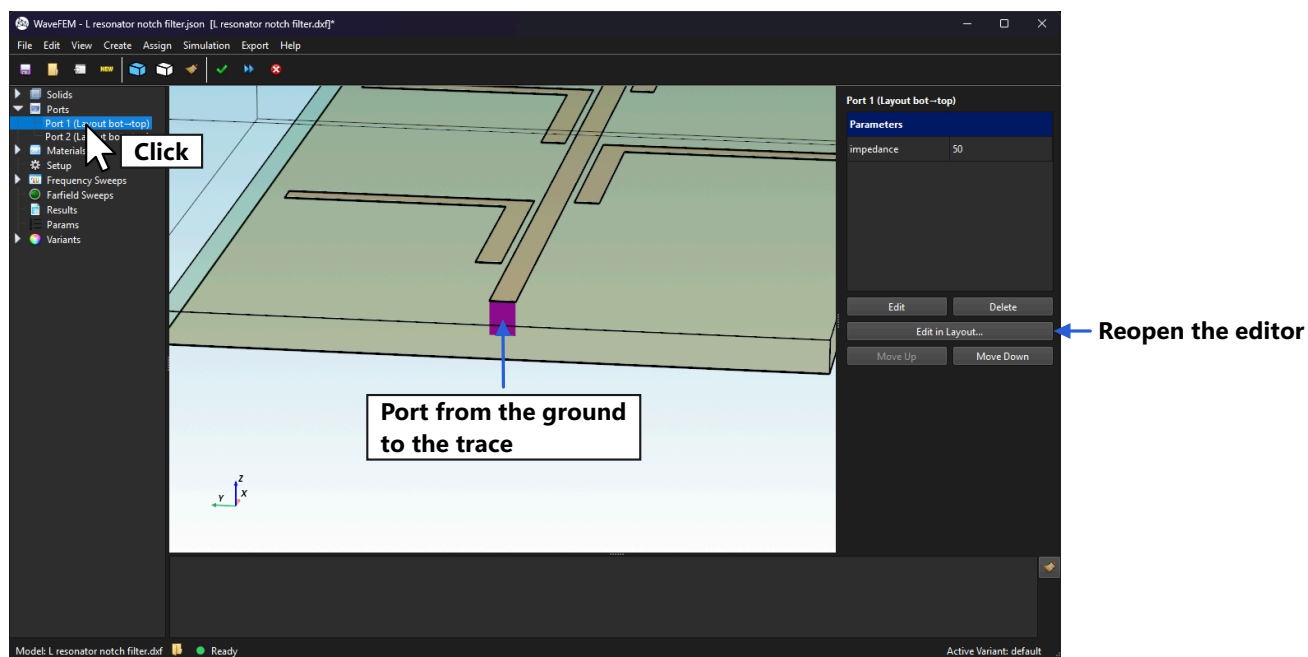


Figure 24 — A layout port between the ground plane and the trace

Note If the stack-up no longer has a port's layer, the port is marked - invalid and the setup does not validate until the port is fixed in the editor.

USING WAVEFEM

Simulation Settings

Solver settings

Click **Setup** in the tree, then **Edit**. The most important setting is `refine_frequency`: choose a frequency in or near the top of your band.

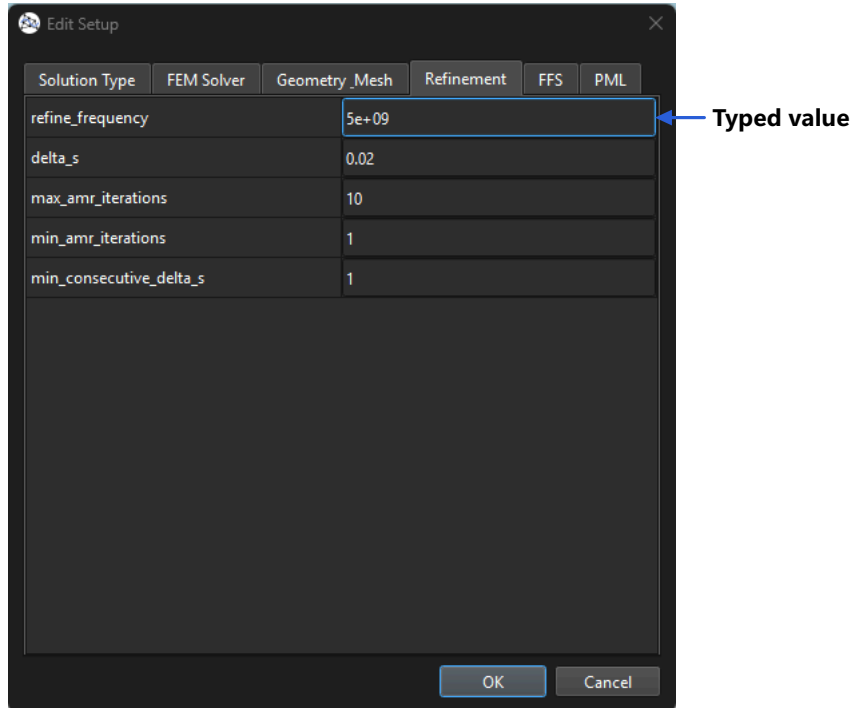


Figure 25 — Refinement settings

<code>base_unit</code>	Model length unit in metres (0.001 = mm).
<code>bounding_box_padding</code>	Air added around the model, in model units. Aim for about a quarter wavelength.
<code>target_elements_per_wavelength</code>	Initial mesh density.
<code>port_element_density</code>	Initial mesh density on ports.
<code>refine_frequency</code>	Frequency (Hz) at which the mesh is refined.
<code>delta_s</code>	Convergence limit for adaptive refinement.
<code>max_amr_iterations</code> / <code>min_amr_iterations</code>	Bounds on the number of refinement passes.
<code>min_consecutive_delta_s</code>	Passes in a row that must meet <code>delta_s</code> .
<code>max_ffs_iterations</code>	Maximum points in an adaptive frequency sweep.
<code>ffs_fit_tol</code>	Convergence tolerance of the adaptive sweep fit.
<code>pml_enable</code>	Absorb outgoing waves with a perfectly matched layer outside the box.
<code>pml_alpha</code> , <code>pml_beta</code>	Decay rate and propagation speed in the PML.
<code>pml_thickness</code>	PML thickness, in model units.

Frequency sweeps

Create > Sweep adds a sweep. A project can have several, e.g. a wide adaptive sweep for S-parameters plus a single frequency that saves fields.

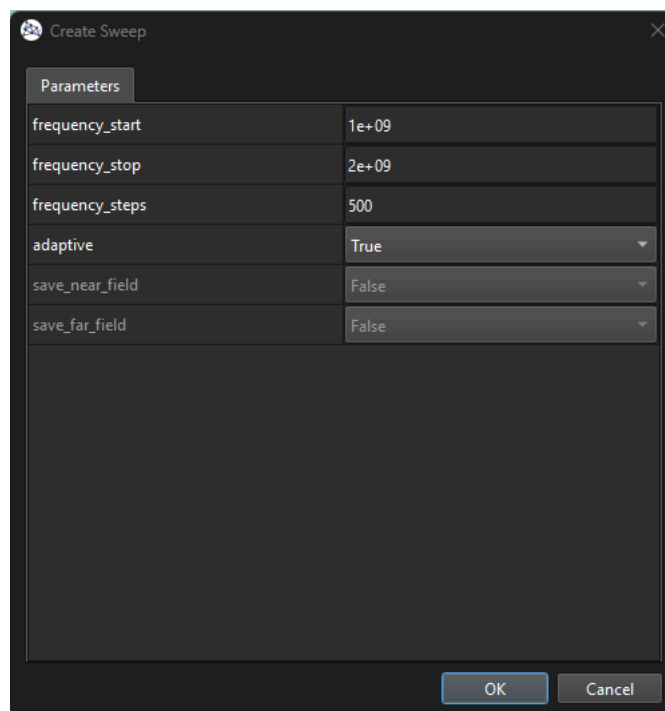


Figure 26 — Frequency sweep

frequency_start, frequency_stop	Band edges in Hz. Do not start at 0 Hz; use 1 Hz instead.
frequency_steps	Number of output points (for an adaptive sweep, the points the fit is evaluated at).
adaptive	Choose solve points automatically until the response is resolved.
save_near_field	Save fields inside the model.
save_far_field	Save radiation patterns (needs at least one far-field sweep).

Far-field sweeps

Create > Farfield Sweep defines a radiation pattern cut or a full 3D pattern.

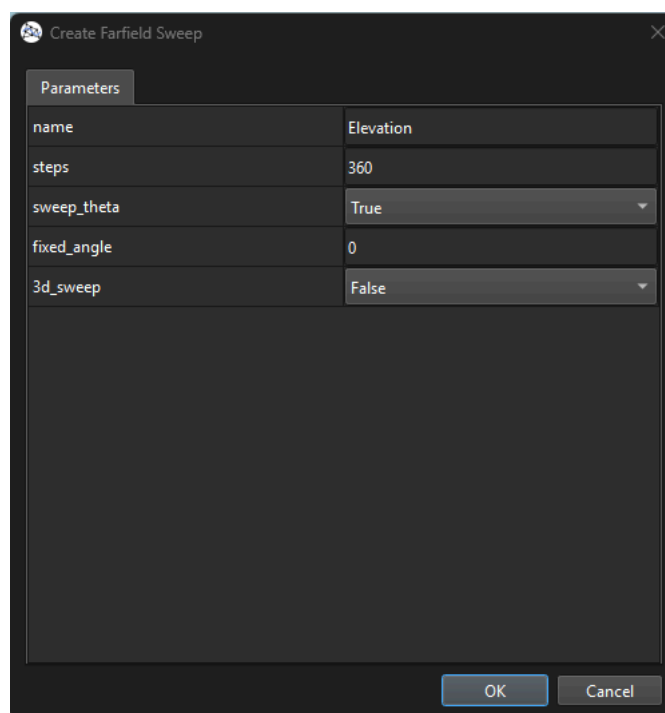


Figure 27 — Far-field sweep

name	Name shown in the tree and plots.
steps	Points per revolution. Keep this low (about 50) for 3D sweeps, which use many revolutions.
sweep_theta	Sweep theta at fixed phi; otherwise sweep phi at fixed theta.
fixed_angle	The angle that is held fixed (degrees; not used in 3D).
3d_sweep	Compute the full 3D pattern.

USING WAVEFEM

Running Simulations

Validate

Simulation > Validate checks the setup and reports in the console. A setup needs at least one port, a material on every solid, and at least one frequency sweep.

```
Validation Summary for Variant 'default':
No ports defined!
Material assignments not set.
Setup is Invalid!
```

Run and stop

Simulation > Run saves the project and starts the solver; only one simulation runs at a time. **Simulation > Stop** cancels it. Progress appears in the console:

```
===== Started Running at 19:31 =====
Adaptive Mesh Refinement: 6e+09Hz
  Iteration Num Elems Time (s) Delta S Peak Memory (mb)
          1      868      0      0      64
          2      931      0    0.254      85
          3     1152      0    0.111      94
          4     1384      0   0.0236     111
          5     1772      0  0.006995     134
Convergence reached (Delta S < 0.02).
Frequency Sweep:
  Frequency      Time
    3e+09         0
    6e+09         0
    4.5e+09        0
===== Finished Successfully at 19:31 =====
```

The first table is the mesh refinement: elements grow until ΔS falls below the limit. The second lists the frequencies solved by the sweep.

USING WAVEFEM

Results

When a run finishes, its results appear under **Results** in the tree. Click one to plot it. The properties panel shows the plot settings; **Edit** changes them.

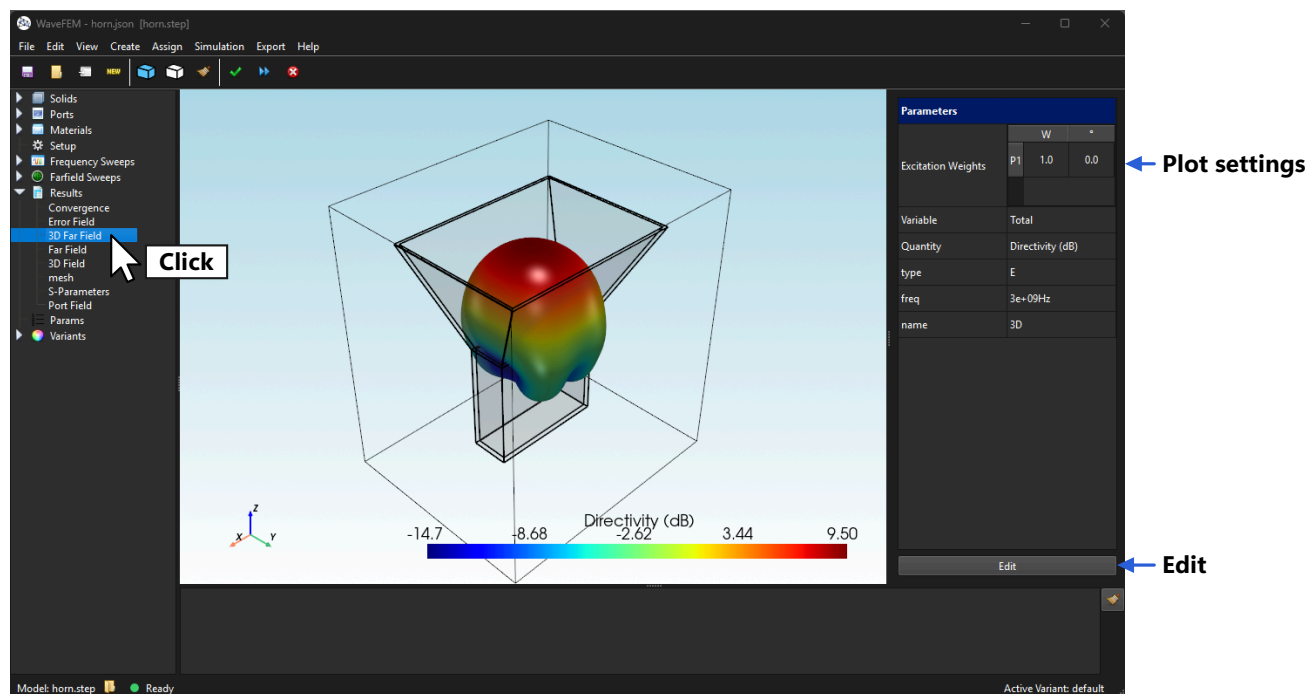


Figure 28 — Plotting a result

Changing what is plotted

1. Click **Edit** under the plot settings.
2. Change the settings, e.g. the frequency.
3. Click **OK**.

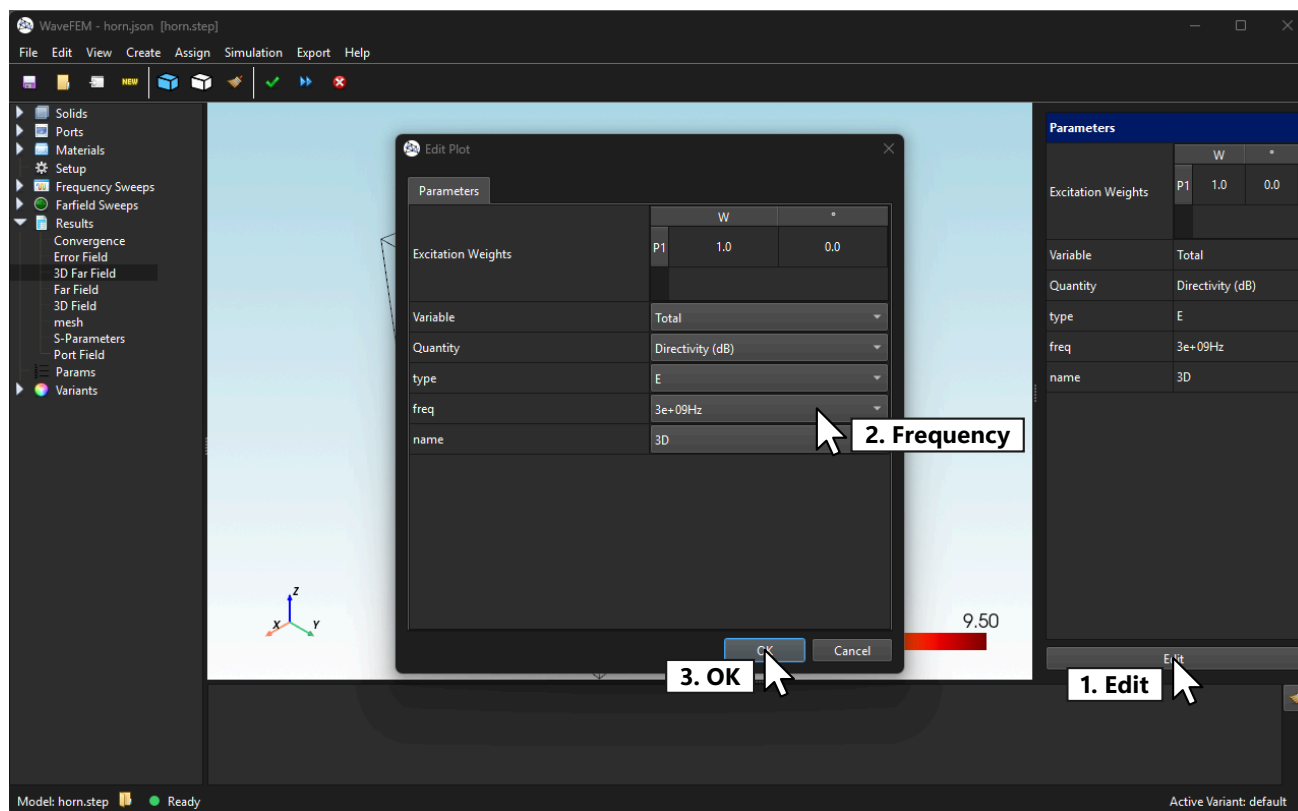


Figure 29 — Editing plot settings

Excitation weights

Field results can be shown for any combination of port excitations. The **Excitation Weights** table sets the power (W) and phase (degrees) of each port; edit it through **Edit** by double-clicking a value. Exciting only one port of two horn antennas, for example, shows the field in that horn alone; give both ports a non-zero power to see both at once.

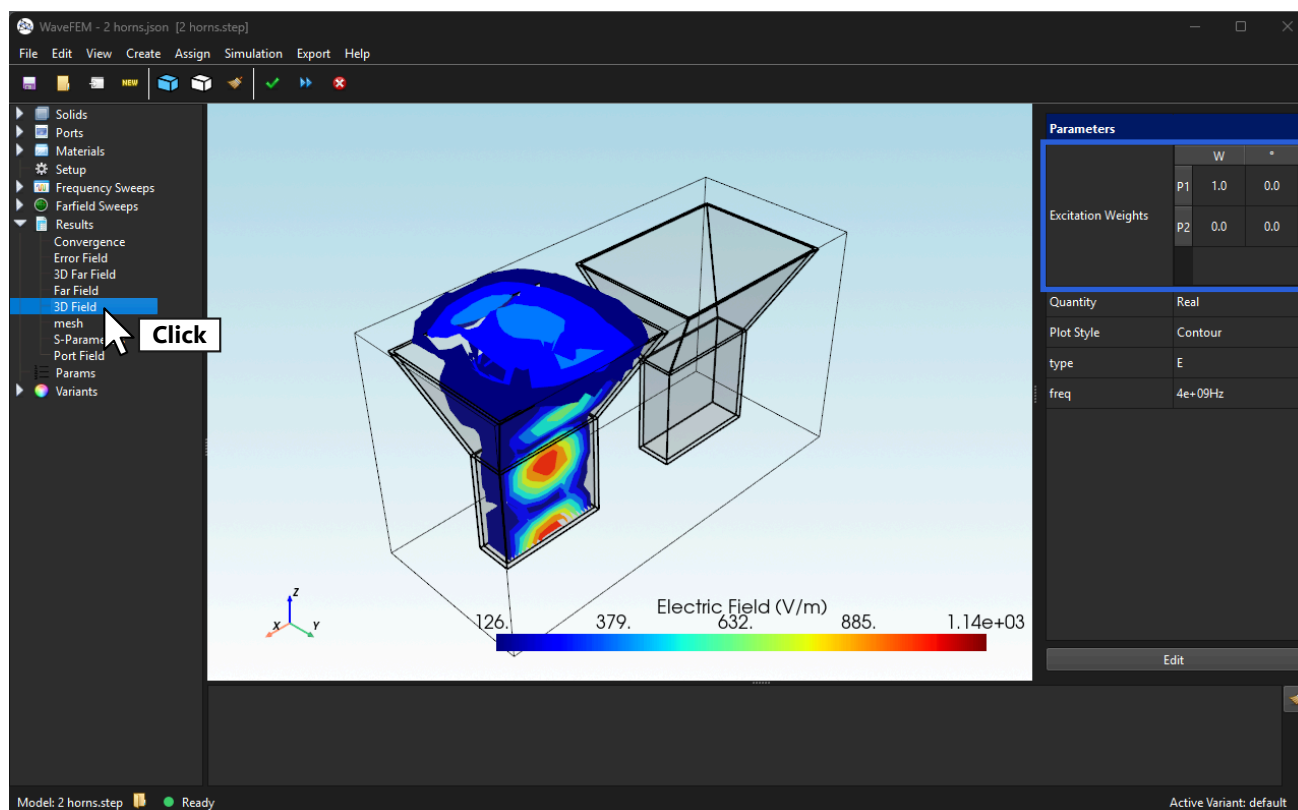


Figure 30 — Excitation weights for a two-port model

S-parameters

The S-parameter plot shows the parameters selected in its settings. In the edit dialog, click cells of the matrix to toggle them.

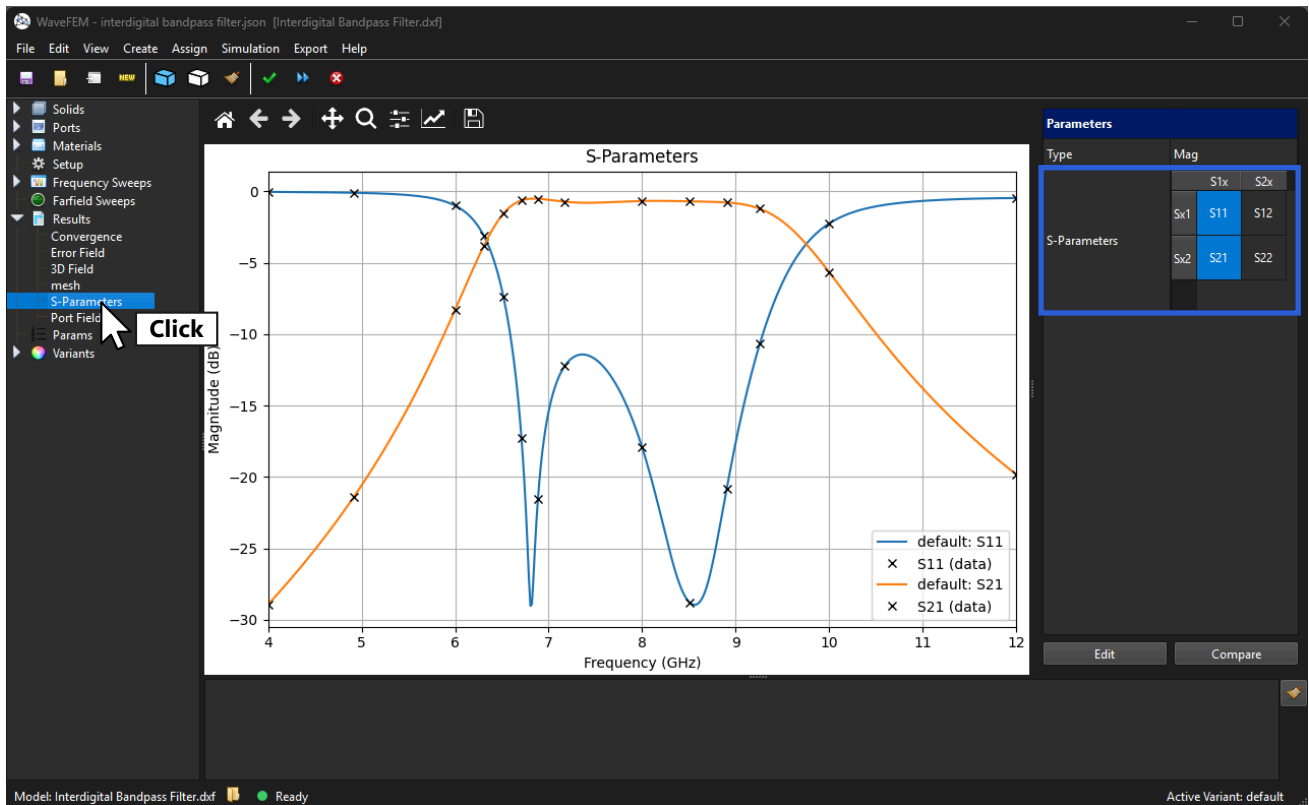


Figure 31 — S-parameters of a bandpass filter

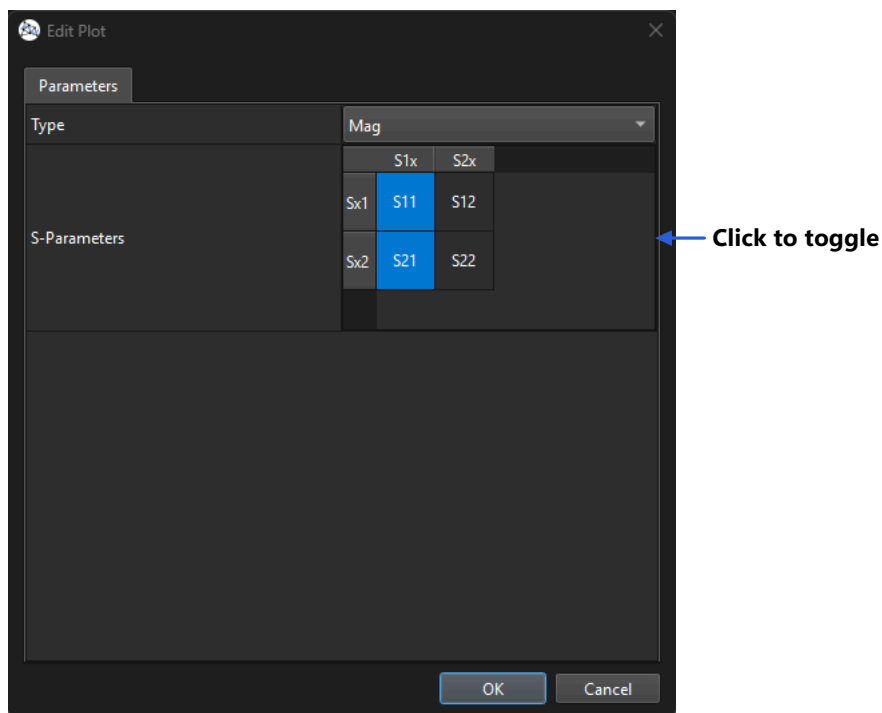


Figure 32 — Choosing S-parameters

■ Exporting

Display the result first, then use the **Export** menu.

Plot Data	The displayed data as .csv or .xlsx.
Plot Image	The displayed plot: .png, .jpg or .tiff for 3D plots; .png, .svg or .pdf for 2D plots.
Network Parameters	Touchstone .snp file of the S-parameters.
.Step File	The 3D model, e.g. of a scripted or layered design.

USING WAVEFEM

Design Variants

A variant is one version of the design: its own model file or script parameters, ports, materials and results. Every project starts with a default variant and always has at least one. Expand **Variants** and select one to copy, rename, delete or activate it.

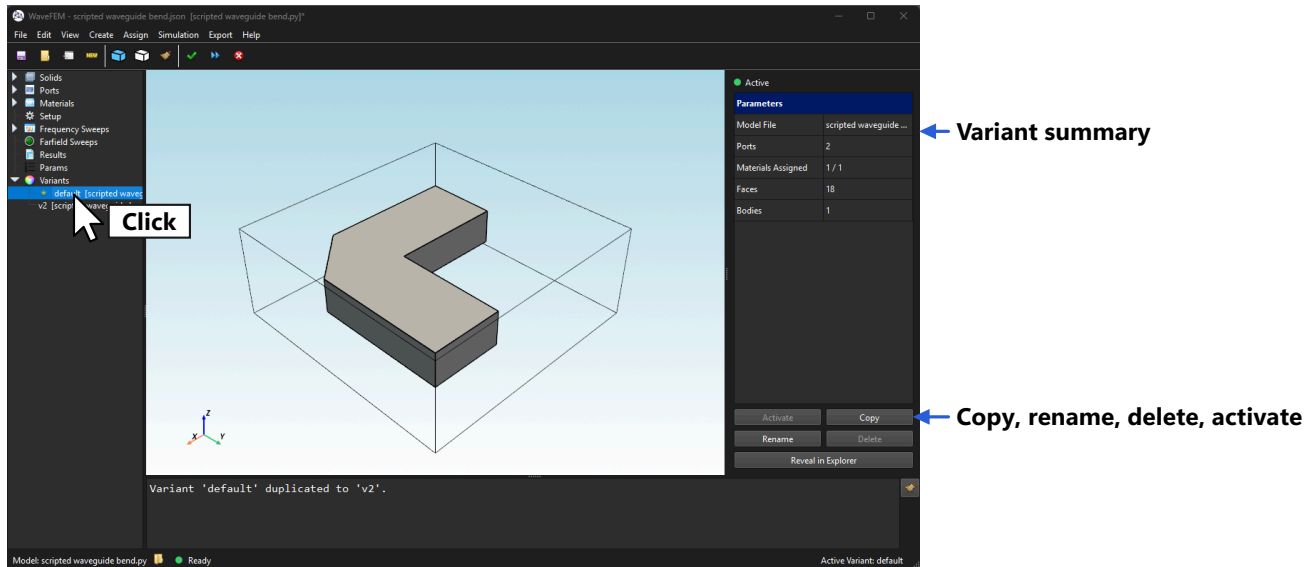


Figure 33 — Variants of a scripted waveguide bend

To compare variants, plot S-parameters or a far-field result, click **Compare** and pick the other variant. Both are drawn on the same axes.

EXAMPLES

Horn Antenna

This example simulates examples/regular/horn.step from 3 to 6 GHz.

1. **File** > **Import model file** and choose horn.step.
2. Right-click the inner face at the bottom of the waveguide, then **Assign** > **Wave Port**.
3. **Create** > **Material, Copy from Library**, choose PEC.
4. Click **Body 1** under **Solids**, then **Assign** > **Material** and choose PEC.
5. Click **Setup, Edit**, and set refine_frequency to 6e9.
6. **Create** > **Sweep**: 3e9 to 6e9 Hz, 500 steps, adaptive. This gives the S-parameters.
7. **Create** > **Sweep** again: 3e9 to 6e9 Hz, 4 steps, not adaptive, with save_near_field and save_far_field on. This saves fields at 3, 4, 5 and 6 GHz.
8. **Create** > **Farfield Sweep**: an elevation cut (sweep_theta true, fixed angle 90) and a 3D pattern (3d_sweep true, 50 steps).
9. **Simulation** > **Run**.

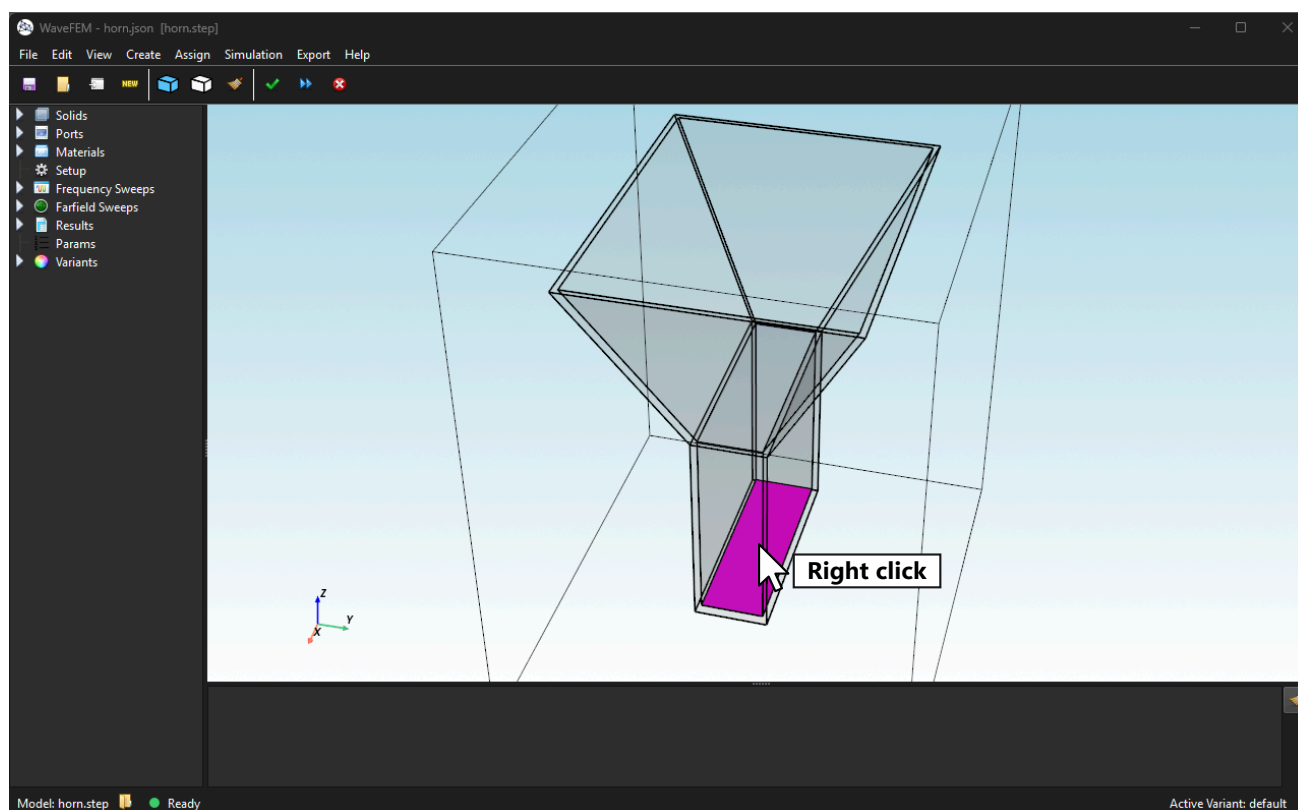


Figure 34 — Wave port at the bottom of the horn

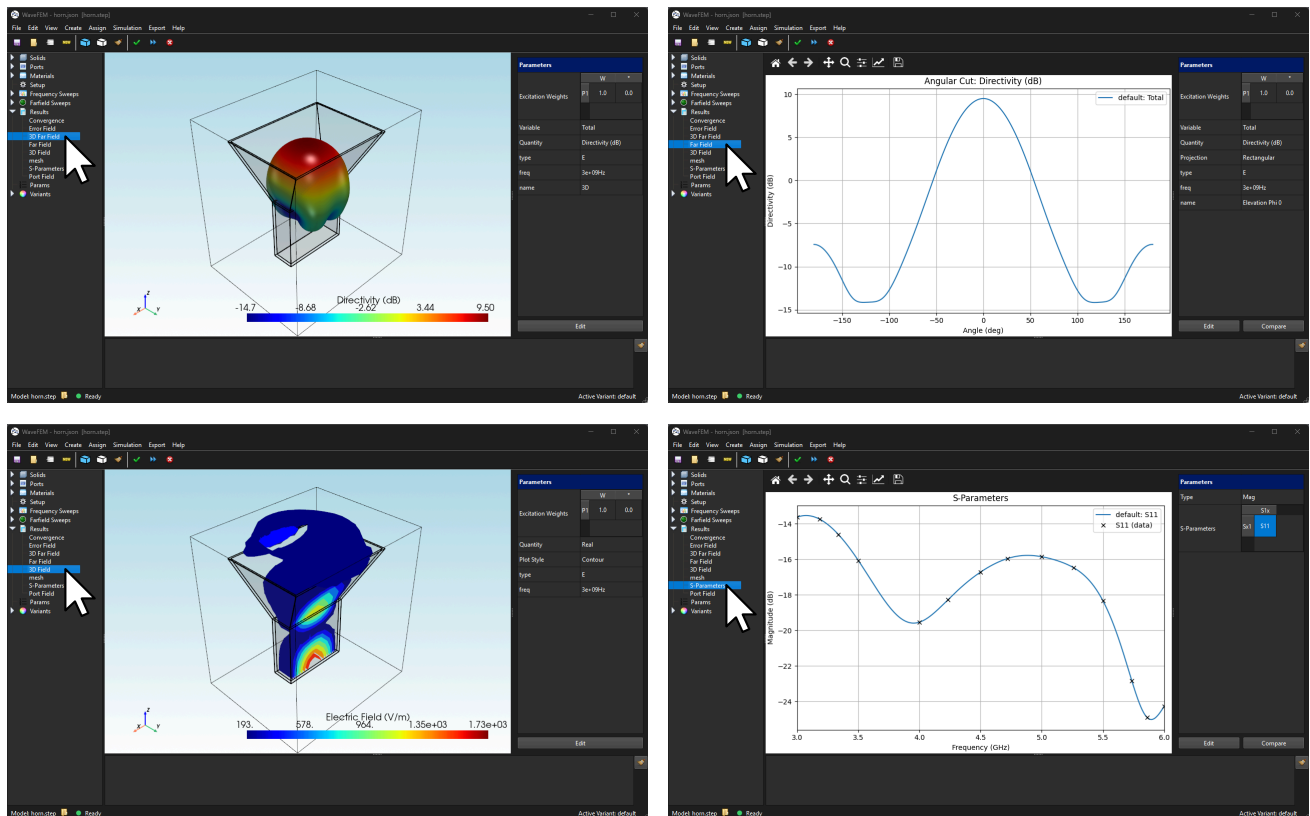


Figure 35 — Horn results: 3D pattern, elevation cut, electric field and S_{11}

EXAMPLES

Microstrip Filter

This example builds a stepped-impedance low-pass filter from `examples/regular/stepped impedance filter.dxf`. The file has two layers: `cond` (the traces, pink) and `cond2` (the ground plane, green).

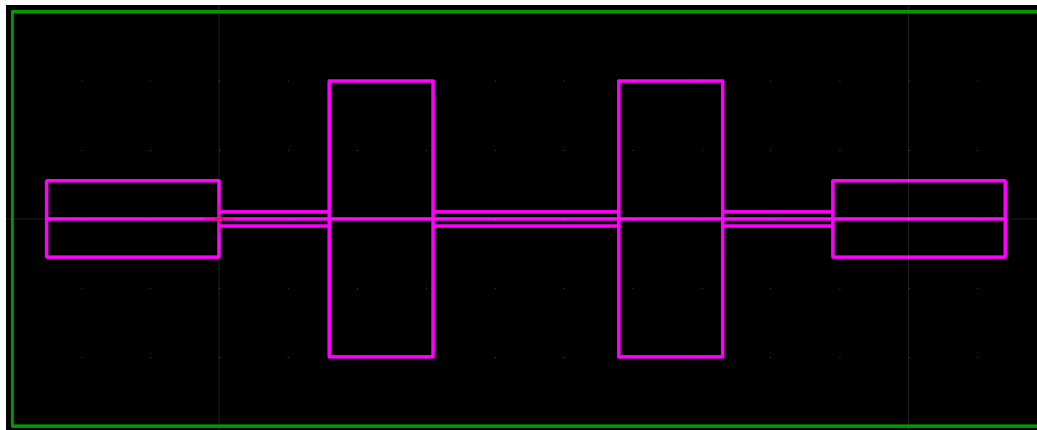


Figure 36 — Filter layout in the DXF file

1. Import the DXF. In the layer window add a dielectric `main`, 0.5 mm thick.
2. Add metal layers `cond` (0.05 mm, on `main`, **Above** and **Intrude Up** ticked) and `cond2` (0.05 mm, on `main`, below). Their names must match the DXF layers. Click **OK**.
3. Create materials `Cond` (PEC) and `R04003` (ep 3.55, $\tan\delta$ 0). Assign `Cond` to `cond` and `cond2`, and `R04003` to `main`.
4. Add a lumped port at each end of the line (see [Ports](#)).
5. Add an adaptive sweep from 1 Hz to 20 GHz, 500 steps.
6. In **Setup** set `refine_frequency` to `10e9` and `bounding_box_padding` to 4.
7. Run.

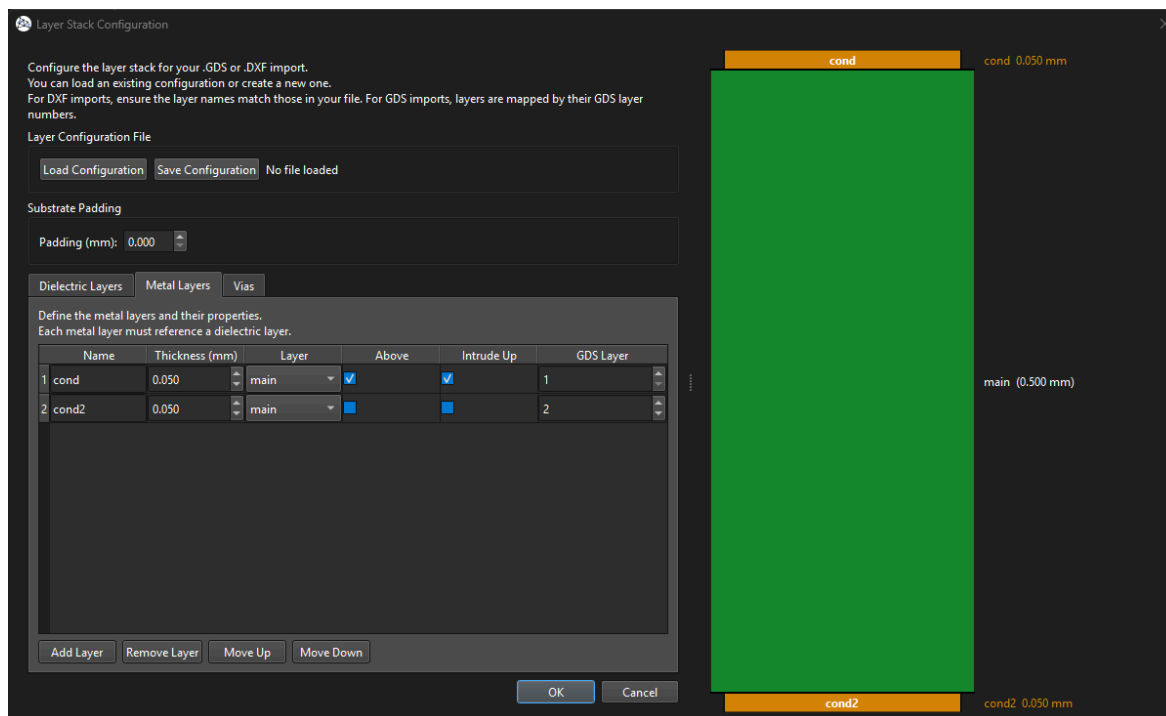


Figure 37 — Stack-up for the filter

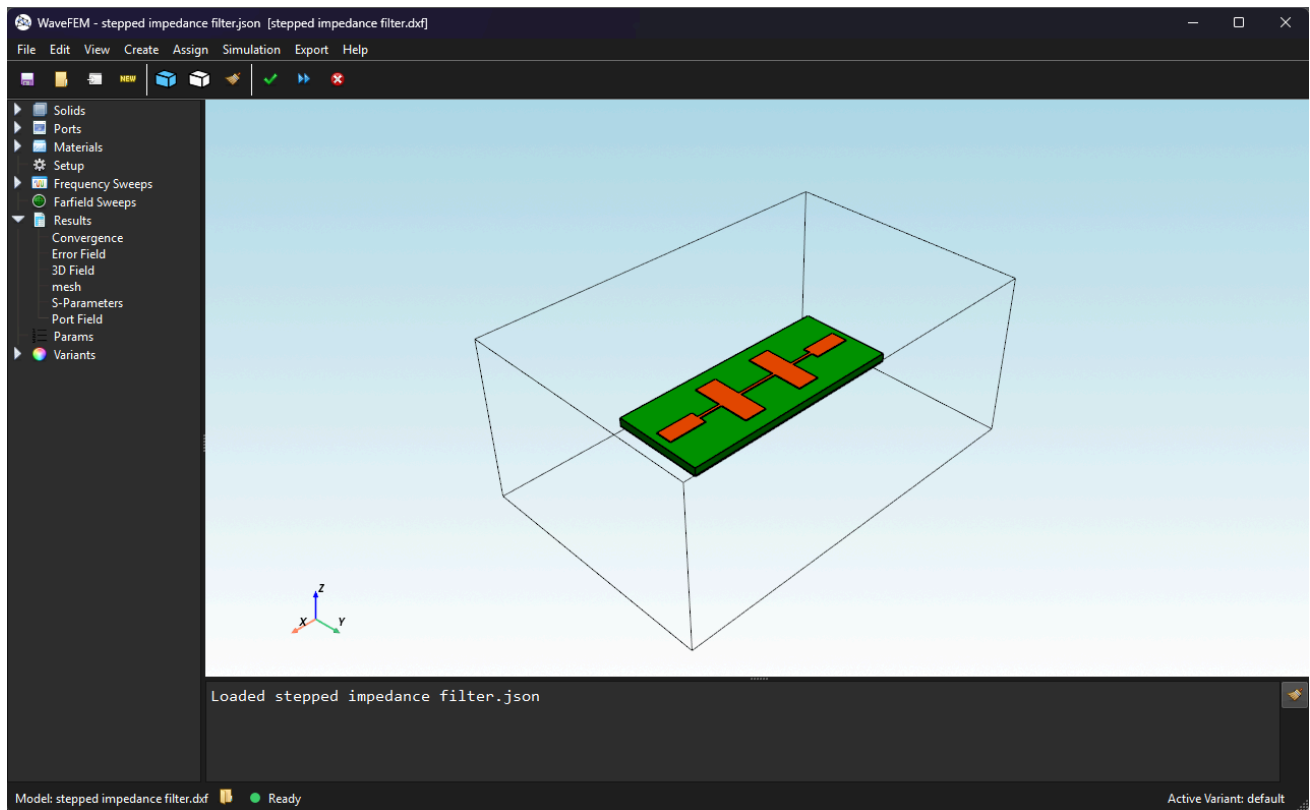


Figure 38 — The filter with materials assigned

EXAMPLES

Scripted Waveguide Bend

This example builds a WR-90 waveguide bend from a CadQuery script, examples/scripted/scripted_waveguide_bend.py, and shows how script parameters make it easy to try design changes.

The script

```
1 import cadquery as cq
2
3 # Waveguide dimensions
4 wg_width = 22.86 # Standard WR-90 width (mm)
5 wg_height = 10.16 # Standard WR-90 height (mm)
6 wall_thickness = 1.0
7 straight_length = 50.0 # Length of straight sections
8 chamfer_length = 17.0 # Length of chamfer at the bend
9
10 # Create the outer path for the waveguide (extends beyond inner for caps)
11 outer_path = (
12     cq.Workplane("XY")
13     .moveTo(-straight_length - wall_thickness, 0)
14     .lineTo(0, 0)
15     .lineTo(0, straight_length + wall_thickness)
16 )
17
18 # Create the inner path for the hollow interior
19 inner_path = (
20     cq.Workplane("XY")
21     .moveTo(-straight_length, 0)
22     .lineTo(0, 0)
23     .lineTo(0, straight_length)
24 )
25
26 # Create outer cross-section profile
27 outer_profile = cq.Workplane("YZ").rect(
28     wg_width + 2 * wall_thickness, wg_height + 2 * wall_thickness
29 )
30
31 # Create inner cross-section profile
32 inner_profile = cq.Workplane("YZ").rect(wg_width, wg_height)
33
34 # Create the outer and inner shapes by sweeping profiles along their respective paths
35 outer_shape = (
36     outer_profile.sweep(outer_path).edges(">(1, -1, 0)").chamfer(chamfer_length)
37 )
38 inner_shape = (
39     inner_profile.sweep(inner_path).edges(">(1, -1, 0)").chamfer(chamfer_length)
40 )
41
42 # Create hollow waveguide with caps by subtracting inner from outer
43 result = outer_shape.cut(inner_shape)
44
45 out_file = "waveguide_bend.step"
46 result.export(out_file)
```

python

How it works

- Parameters.** The variables at the top (`wg_width`, `wg_height`, `wall_thickness`, `straight_length`, `chamfer_length`) are plain numbers. WaveFEM lists every top-level assignment of a literal value under **Params**, so these can be changed without editing the file.
- Paths.** Two L-shaped paths trace the centre line of the guide: one for the outside of the walls and one for the hollow inside. The outer path is longer by `wall_thickness` at each end, which leaves a wall across each opening.
- Profiles.** Two rectangles on the YZ plane give the cross-sections: the inside is the standard 22.86×10.16 mm WR-90 aperture, the outside adds the wall thickness all round.
- Sweep and chamfer.** Each profile is swept along its path into a solid L-shaped bar, and the outer corner of the bend is cut off by `chamfer_length` to give a mitred bend.
- Cut.** Subtracting the inner solid from the outer one leaves the metal body: a hollow guide closed at both ends. The two end caps are the faces used as wave ports.
- Export.** The script must set `out_file` and write the model to it. WaveFEM redirects this to a generated file next to the script (here `waveguide_bend[auto_generated].step`) and loads that.

Note Build models from solids, as here: sweep, extrude or revolve closed profiles and combine them with cut and union. Do not use `shell()` to hollow a part. Shelling rennumbers faces whenever a parameter changes, so ports and other assignments end up on the wrong faces, and it sometimes leaves a solid that is not watertight, which cannot be meshed.

Debug a new script in a Python editor first. Errors show in the console when the model loads (see [Python scripts](#)).

Simulating it

- File > Import model file** and choose the `.py` file. Its parameters appear under **Params**.
- Right-click the end cap of one arm and **Assign > Wave Port**. Repeat on the other arm.
- Create a PEC material and assign it to the body.
- In **Setup** set `refine_frequency` to 8e9 and `bounding_box_padding` to 10.
- Add a sweep from 7 to 8 GHz and run.

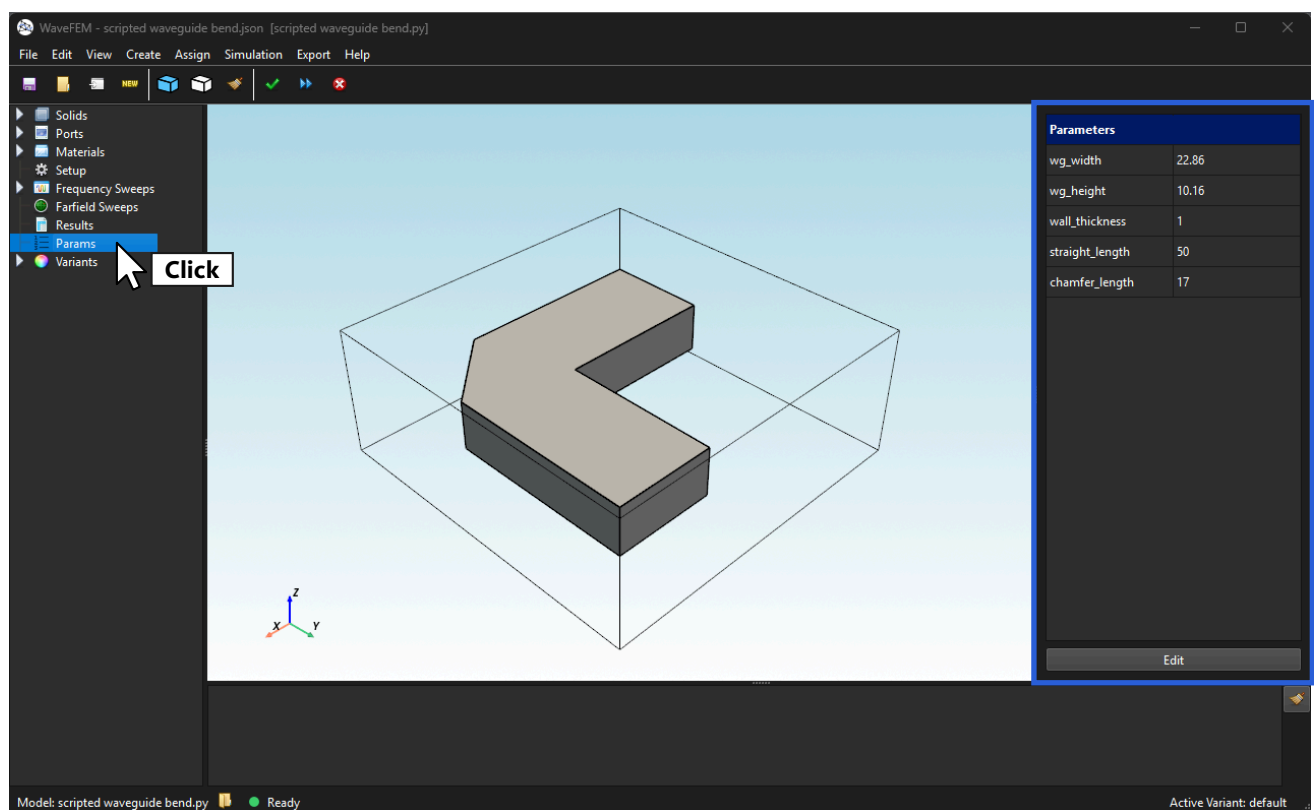


Figure 39 — The script's parameters in WaveFEM

■ Trying a design change

1. Select the default variant under **Variants** and click **Copy**.
2. Activate the copy, click **Params** and change `chamfer_length`. The model rebuilds straight away; the ports stay on the same end faces because the face order does not change.
3. Run the new variant, then open **S-Parameters** and click **Compare** to plot both variants together. A good chamfer gives the lowest S_{11} across the band.

APPENDIX

References

- [1] D. B. Davidson, "An evaluation of mixed-order versus full-order vector finite elements," *IEEE Transactions on Antennas and Propagation*, vol. 51, no. 9, 2003.
- [2] B. Gustavsen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Transactions on Power Delivery*, vol. 14, no. 3, pp. 1052–1061, 1999.