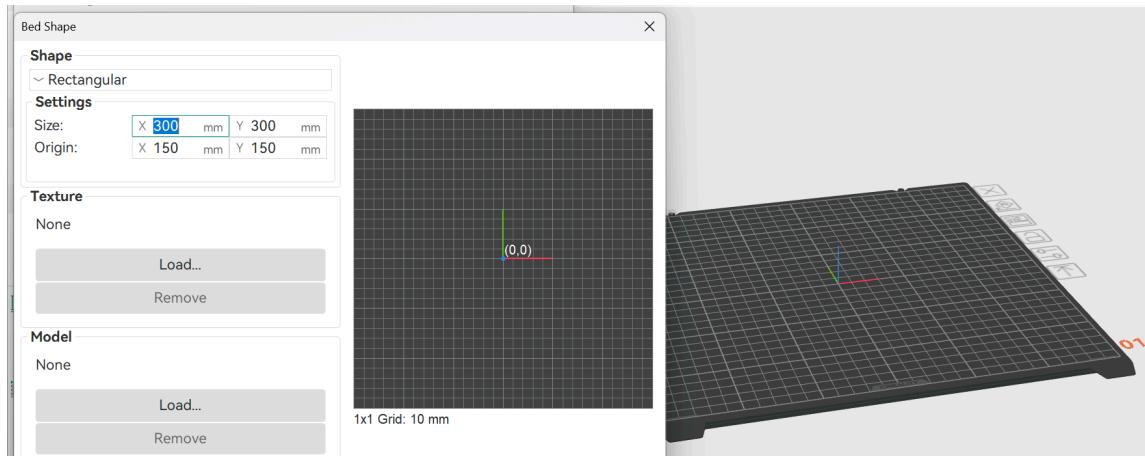


EasyConical is meant to be used alongside a regular 3D print slicer for the best results. This guide will explain how to use our software along with the steps you need to take in the slicer you are using. All photos in this guide were taken in [OrcaSlicer](#).

## Initial Slicer Setup

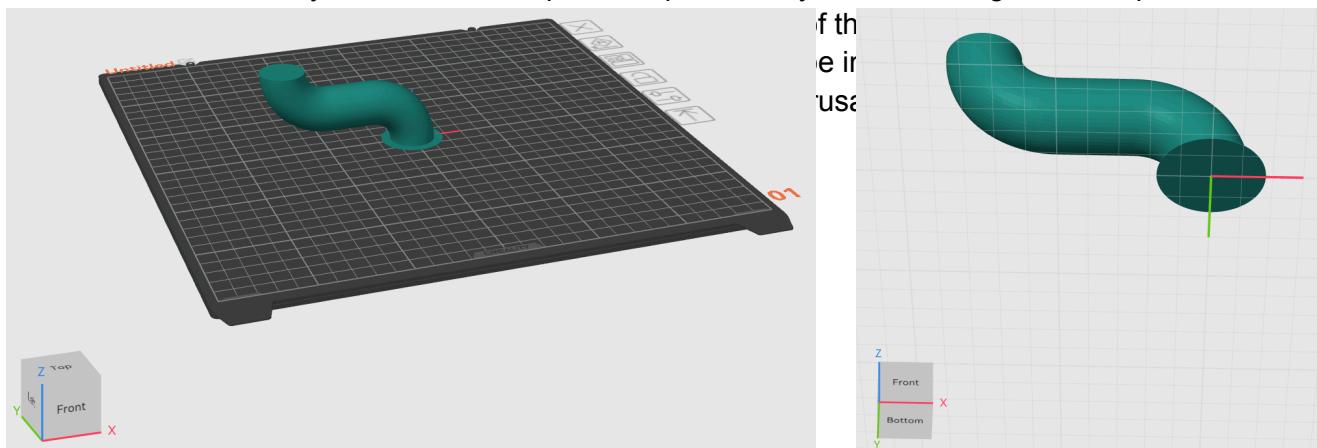
Before you begin, make sure the slicer you are using is able to use relative E distances. If it cannot, you must use another slicer, as EasyConical can only process relative E distances, not absolute.

EasyConical works by warping the print file around the vertical origin axis(usually Z). Unless your slicer can process parts outside of the buildplate, you will need to move the origin to the center of the buildplate. This can be done by setting the origin X and Y positions to exactly half of the buildplate's X and Y size. Most slicers give you the ability to customize the buildplate settings, including moving the origin.



The buildplate customization menu in OrcaSlicer, with the origin set to the center of the buildplate.

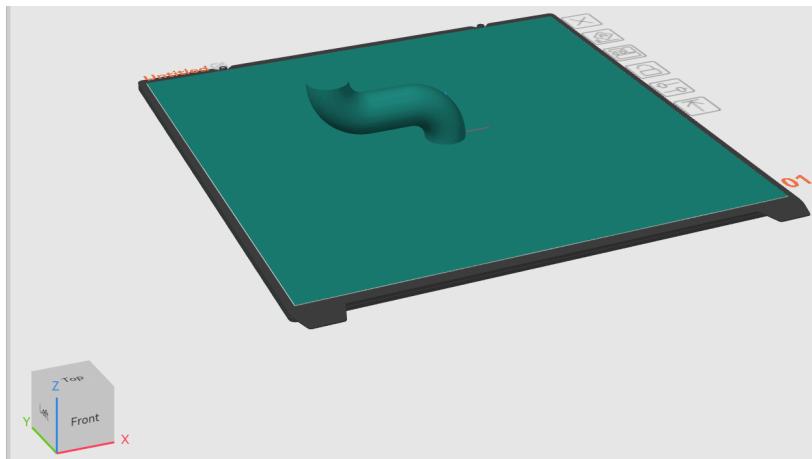
Once this has been done, the next step is to import the file you are working with, in a format compatible with the slicer(commonly an STL). Currently, EasyConical can only process one model at a time, so you will need to repeat this process if you are working with multiple files. The



Multiple angles of a part in OrcaSlicer centered on the vertical origin axis, in this case 'Z'

## The Workaround

If you are using a PrusaSlicer or another slicer that has been derived from it, there is an additional process that must be done, you can skip this step if you are not using a PrusaSlicer based slicing software. EasyConical works by warping models around the origin, but PrusaSlicer creates additional part origins not relative to the buildplate origin. To force the part origin and the buildplate origin to be the same, create a thin rectangular model the same exact size as the buildplate, and move it to the center of the buildplate. Now this whole setup can be exported as a single STL file.

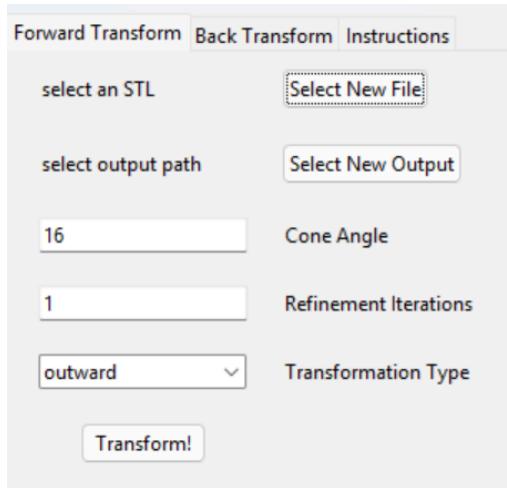


The OrcaSlicer buildplate with the part and a rectangle the size of the buildplate.

## Forward Transformation

Now that the file has been prepared, it's finally time to open up EasyConical. Run the .exe file, and in a few moments the menu will pop up. Make sure that the 'Forward Transform' tab at the top is selected. Use the select buttons to select the STL you have exported from your slicer and the folder in which to output the next file. Make sure you select the file that you have exported from the slicer, and not the original model file. The three settings, 'Cone Angle', 'Refinement Iterations', and 'Transformation Type', can be left at default, and are explained further in the 'Advanced Settings' section of this tutorial. Once you are satisfied with the inputs, click the

'Transform!' button and wait for a message that says 'Operation finished'.



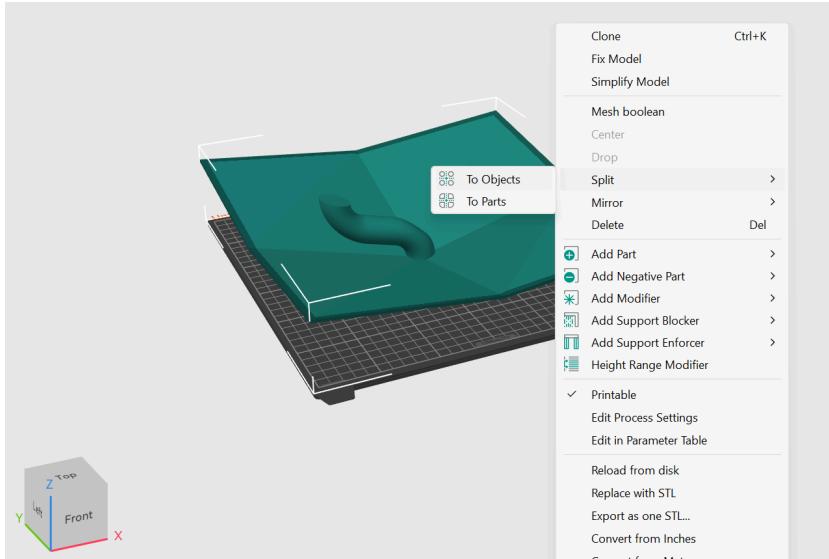
The 'Forward Transform' tab in EasyConical

## Slicing the Model

If you did not perform the workaround steps, take the model exported by EasyConical and put it into the slicer. You should see a distorted version of your model in the center.

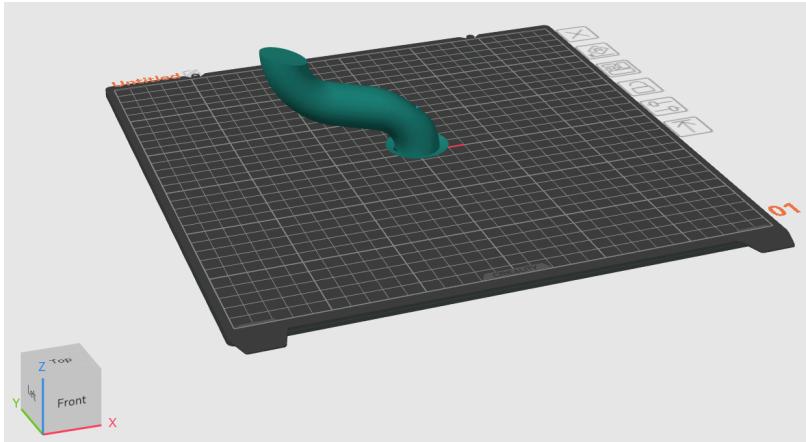
### If you used the workaround

Import the file that was exported from EasyConical into your slicer. You should see a distorted version of your model in the center, as well as a distorted version of the rectangular model used in the PrusaSlicer workaround. Right click on the model, and select the 'split' option present in most slicers. If given a choice between splitting to 'parts' or 'objects', select 'objects'. Once this has been done, deselect everything and select only the distorted rectangle. You may now safely delete this plane, as it is not needed anymore.



Splitting the distorted objects in OrcaSlicer.

Now that you have an isolated, distorted version of your model in your slicer, set up the normal settings for your printer. The settings for conical slicing may need to be slightly different than your regular settings, and the only setting that must be set a certain way is ‘use relative e distances’, which must be turned on for the model to process correctly. Most slicers should have this option, but certain slicers (like FlashPrint) do not, and are sadly incompatible with EasyConical. Once the desired settings have been dialed in, slice the part using the processes in the slicer you are using, and export the gcode file to a known location.

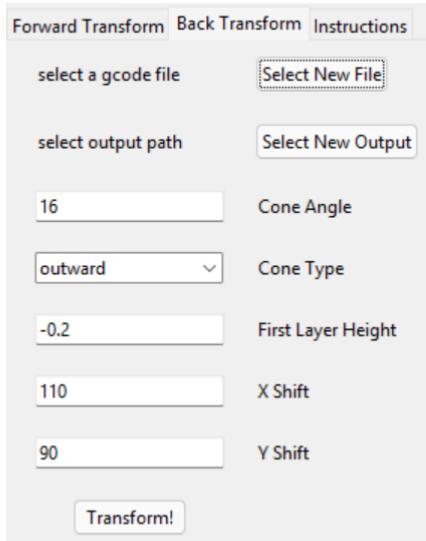


The distorted STL loaded in OrcaSlicer.

## Back Transformation

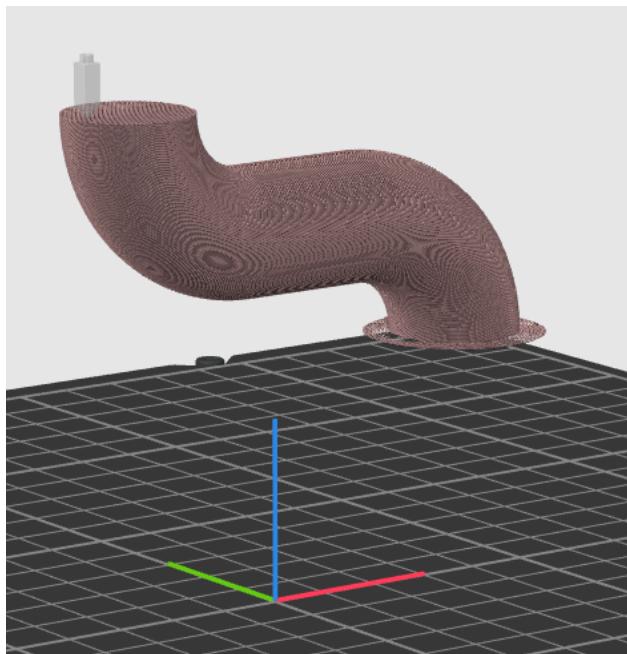
Open up the second tab in EasyConical, labelled ‘Back Transform’. Use the select buttons to select the gcode file exported by your slicer and the folder in which to output the final file. The parameter for ‘Cone Angle’ should be the same number used when forward transforming, and the parameter for ‘Cone Type’ should be set to the same value used in the ‘Transformation

Type' parameter during forward transformation. 'First Layer Height' is different for every printer, but it should be relatively close to zero, as this affects the height of where the model is printed. 'X Shift' and 'Y Shift' should be set to half the size of the buildplate, respectively to the X and Y sizes of the buildplate. Once you are ready, click the 'Transform!' button, and wait until an 'Operation finished' message shows up.



The 'Back Transform' menu in EasyConical

This process can take up to 2-3 minutes, and the GUI will appear to be frozen during this time. Don't worry, the program IS still running as intended despite it looking broken, and once it is done processing it will return to normal. Do not close the window while it is in this state, as this will cancel the processing. Once the file exports, check it by using any gcode viewer, and it should look exactly like your original model. Congrats, you have successfully used EasyConical!



The final product in OrcaSlicer's gcode viewer

## **Advanced Settings**

This section will discuss the optional settings found in EasyConical.

**'Cone Angle'**: This setting controls the angle(in degrees) to which the model is distorted. It should be adjusted in accordance to the size of your printhead, as this is the angle layers will be at once the process is finished.

**'Refinement Iterations'**: This setting controls the number of refinements the STL mesh will go through. Simpler models work fine with a single iteration, but slightly more complex models may need two or three passes.

**'Transformation Type'**: This setting determines whether the model is distorted inwards or outwards. The 'outward' option should be selected if there are outward overhangs, and the print will radiate outwards from the inside. The 'inward' option should be selected if you have an inner overhang and the print starts from the outside.