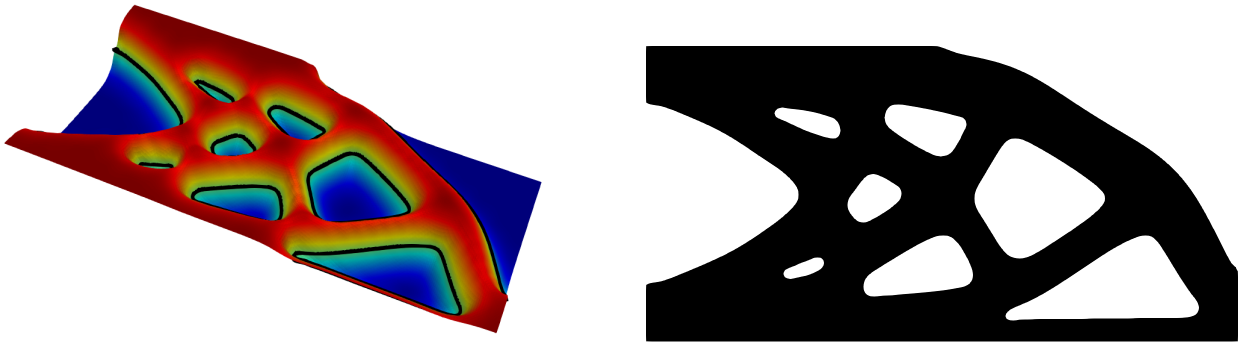
Final Project Proposal

Level-Set Function Visualizer for MORIS: a Topology Optimization Software

Brendan Chong

1 Background

MORIS is an immersed Finite-Element software capable of performing topology optimization for a variety of physics problems. MORIS models geometry with either implicit Level-Set functions or surface mesh (.obj) objects. Level-Set functions are scalar valued functions in $\mathcal{R}^d$ where the zero-isocontour line defines the geometry interface. An example can be seen in Fig. 1



(a) Example Level-Set function with black zero isocontour line. (b) Resulting geometry from Level-Set function

Figure 1: Geometry modeling using Level-Set functions.

Multiple of each geometry type may be combined to create a multimaterial problem. An example can be seen in Fig. 2. In this case, the number of possible phases is determined by $n_m = 2^{n_\phi}$, where n_m is the number of phases and n_ϕ is the number of geometries (Level-Set fields or surface mesh objects). However, oftentimes, the full 2^{n_ϕ} phases are not used, and multiple phases are combined into a single material phase for the problem. To specify this, the user provides a phase table that maps a name to every possible combination of phases. An example phase table can be seen in Table 1.

$\phi = 0$	1	2	Name
in	in	in	p0
in	in	out	p1
in	out	in	p2
in	out	out	p3
out	in	in	p4
out	in	out	p5
out	out	in	p6
out	out	out	p7

Table 1: Example phase table for a $n_\phi = 3$ problem. Note that in this case, each phase was assigned a different name. This does not necessarily have to be the case.

To create the phase table, the user must either carefully visualize the geometries by hand or run the entire physics problem with all phases active to visualize the combination. This causes a lot of confusion and wastes users' time creating the table without convenient visualization software. Thus, as a final project, I would like to develop a visualizer that plots multiple Level-Set functions or imported surface meshes and allows the user to specify any phase combination to view.

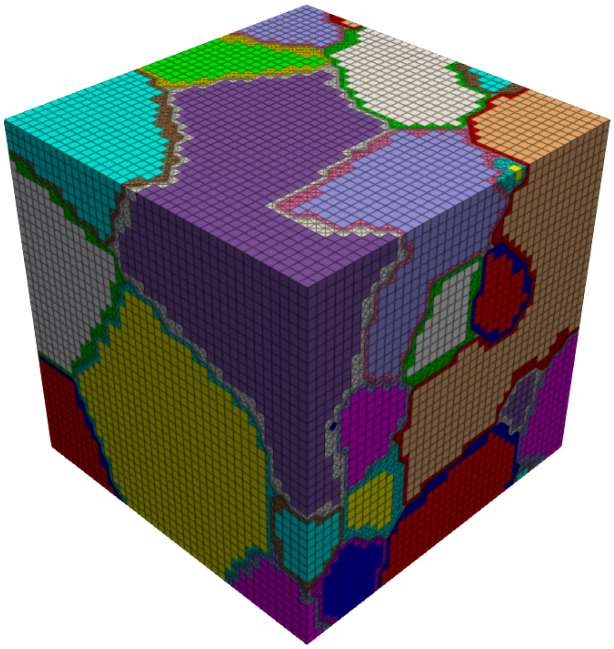


Figure 2: Example render of phases for multimaterial problem.