MATERIALS THAT MATTER

FVMat

DESIGN, SIMULATION & OPTIMIZATION
OF NOVEL META-MATERIALS

Heat Transfer Meta-Material Design Beta Module – User Manual

General description

What does it do?

The module serves as a powerful ML-boosted tool for simulating thermal behavior of 3D arrays of Unit Cells (UCs), each designed with a hexagonal shape housing a central void. These UCs encapsulate a precise volume of liquid, with the current configuration featuring Plastic ABS as the solid material and Water as the enclosed liquid. Integrated seamlessly into Ansys Mechanical Steady State Thermal, the module operates via a custom User Programmable Feature (UPF), employing a Machine Learning (ML) algorithm to determine the effective conductivity of each individual element, thereby enabling comprehensive analysis and prediction of thermal behavior.

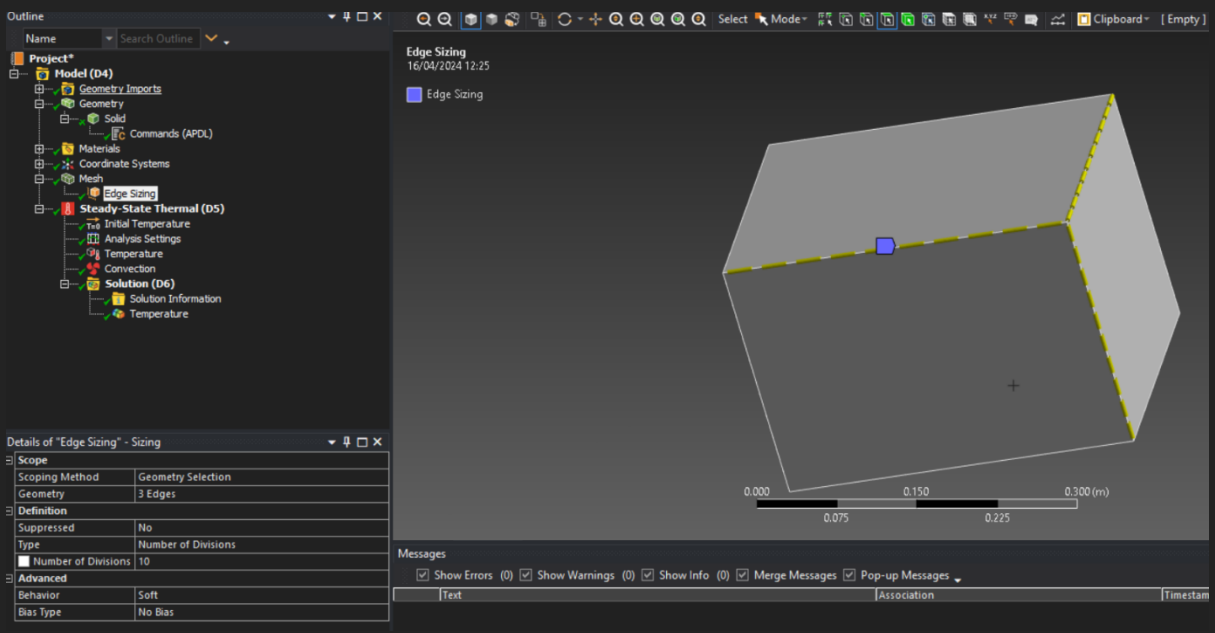
What is it good for?

When comparing simulations utilizing conventional Finite Elements (FE) techniques with those employing this innovative module, a dramatic reduction in computational time becomes evident. For instance, in a scenario involving a 10x10x10 UCs structure, computational time is slashed by a staggering factor of 60. Remarkably, this acceleration in processing speed is achieved without compromising the fidelity of results, ensuring that simulations remain highly accurate and reflective of the physical behavior of the UCs.

How to use it?

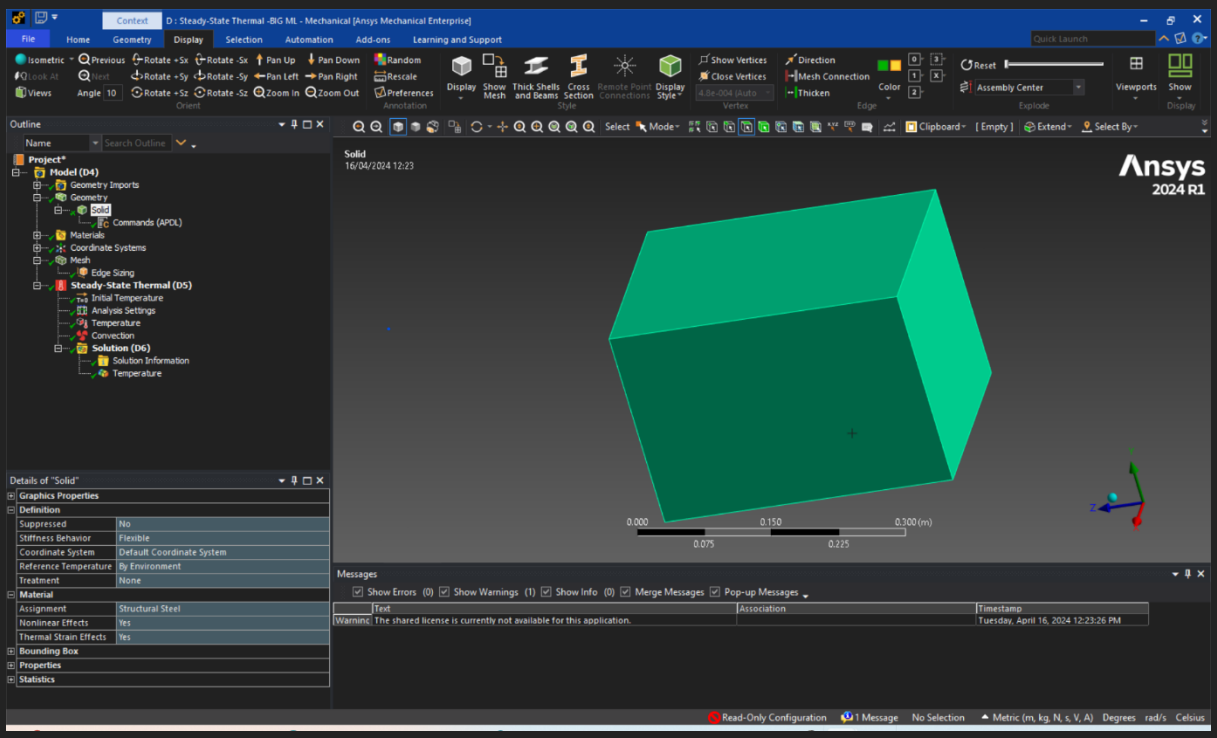
The UCs are characterized by their geometric characteristics, namely thickness of the solid part (d), height and depth of the void (H and L respectively), and amount of water filling it, in terms of relative height ($0 < y < 1$).

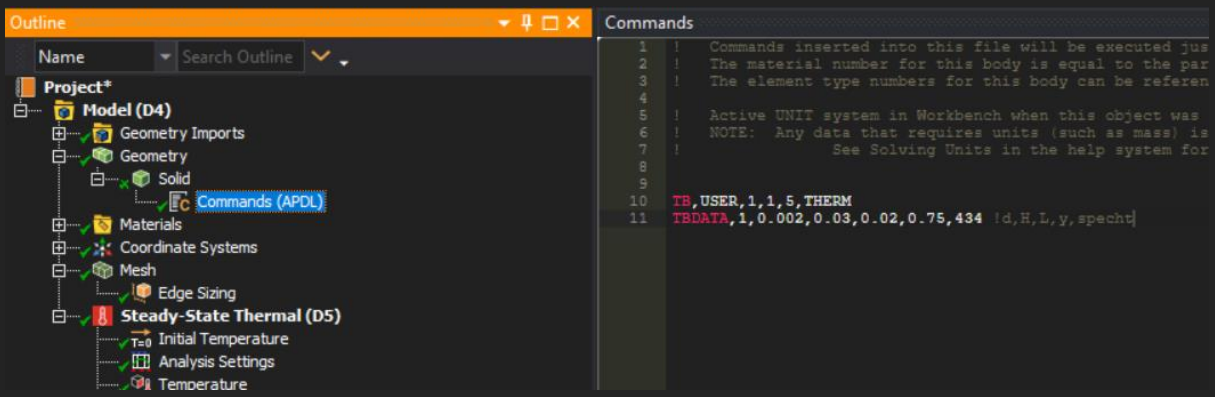
It is important to decide the geometry of the UCs and the number of them; given d, H, L and y, the total height of each UC will be $H+2d$, and its depth $L+2d$. Importantly, this will be the size of each FE element. If N_x, N_y and N_z are the number of UCs along the three directions, then the volume of the simulated object will be N_x $(L+2d)$ x N_y $(L+2d)$ x N_z $(H+2d)$. An “Edge Sizing” method in the meshing needs to be adopted, choosing “Number of Divisions” as Type and for each edge select the corresponding number of UCs (N_x, N_y and N_z), see figure:



The UPF subroutine is activated by the APDL command “TB,USER”, where the user needs to specify the material ID, the number of temperatures for which data will be provided, and the number of data points to be specified for a given temperature, followed by the keyword THERM. These data points will be provided by the command TBDATA, where the user specifies the temperature number and the values of the data. For this UPF, the data that need to be given are 5, *in the following order*: *d*, *H*, *L*, *y*, and specific heat.

The subroutine will be automatically called for the material ID specified.





How does it work?

The UPF calls a ML algorithm that for every element calculates the temperature-dependent effective heat transfer coefficient (conductivity, k) depending on the geometric characteristics of the element/UC and its current temperature.

Why does it work?

We built a dataset of results obtained through PyMAPDL simulations of a UC with periodic boundary conditions and used it to calculate the temperature-dependent effective conductivity of such UC. A Machine Learning model was then trained on this dataset to predict the value of k given the values of d, H, L, y, and T (temperature). Moreover, our model predicts the conductivity along different axes (k_x, k_y and k_z), which are in general different from each other since the UC is not fully symmetric.

We tuned the model to get a good accuracy of the predictions (relative mean error on the order of 10^{-3}).

We tested the procedure in several scenarios, for example a 10x10x10 array of UCs, for which a full simulation took 21 minutes to complete, while a simulation employing our module took only 23 seconds. The agreement between the temperature profiles in the two cases is highly satisfactory.