Laguerre-FDTD Code Base Description

1. This code base contains mainly two parts – MATLAB & C++
   1. MATLAB:
      1. DARPA\_Cylinder.m is the main code. It defines the cylinder geometry through the ConformalGrid\_3D\_ML.m MATLAB functions. This code itself can run the entire EM simulation, but it has poor computational performance. You should use it as a starting point to get familiar with the code and simulation flow.
      2. ConformalGrid\_3D\_ML.m is the MATLAB function that defines the entire DARPA cylinder model. It can also control the location and diameter of the bottom aperture according to the input parameters. Comments are provided inside the function to help you better understand it.
      3. LagPos.m: MATLAB function that does the data post-processing, including 2D plot of E-field and H-field. I don’t know what kind of data output you need to export, so currently this function will not export anything. But you can modify it very easily to have it export any data in any format that you need
      4. MatrixExport.m: Run this function if you want to convert the Cylinder Geometrical data generated from MATLAB into C++ code for improved computational speed. The break point location has been indicated in the DARPA\_Cylinder.m code for you. Essentially, this function will convert the sigma\_x/y/z, dxe, dye, dze, dxh, dyh and dzh variables from the MATLAB to their corresponding text file, and the C++ code will then read in those files to do the computation
   2. C++:
      1. The C++ code base is built upon Microsoft Visual Studio. Download the VS 2022 community version and that will allow you to open the LFDTD\_CUDA\_PARDISO.sln file.
      2. To run the C++ code, please refer to my github repository: <https://github.com/YWang1998/Laguerre-FDTD> and download the required Intel oneAPI MKL and C/C++ complier. You should also download the C++ Boost library (version 1.83.0): <https://archives.boost.io/release/1.83.0/source/>
      3. Unzip and put the boos library under the C:\Program Files folder in your local machine.
      4. Similar to the MATLAB code, current C++ code will not generate any output. Once you’re comfortable with the simulation flow and get familiar with the code, we can have the code to generate any output you need.