A 3D CAD model input pipeline for REFMUL3 full-wave FDTD 3D simulator

J. M. Santos[a,1] **Emanuel Ricardo**[a] **Filipe J. da Silva**[a] **Tiago Ribeiro**[b] **Stephane Heuraux**[c] **A. Silva**[a]

[1]*Instituto de Plasmas e Fusão Nuclear, Instituto Superior Técnico,*
  Universidade de Lisboa, 1049-001 Lisboa, Portugal

[2]*Max-Planck-Institut für Plasmaphysik, 85748 Garching, Germany*

[3]*Institut Jean Lamour, UMR 7198 CNRS-Université de Lorraine BP 50840, F-54011 Nancy, France*

  E-mail: jsantos@ipfn.tecnico.ulisboa.pt

ABSTRACT: The use of advanced simulation has become increasingly more important in the planning, design, and assessment phases of future fusion plasma diagnostics, and in the interpretation of experimental data from existing ones. The design cycle of complex reflectometry systems, such as the ones being planned for next generation machines (IDTT and DEMO), relies heavily on the results produced by synthetic diagnostics, used for system performance evaluation and prediction, both crucial in the design process decision making. These synthetic diagnostics need realistic representations of all system components to incorporate the main effects that shape their behavior. Some of the most important elements that are required to be well modelled and integrated in simulations are the wave launcher structures, such as the waveguides, tapers, and antennas, as well as the vessel wall structures and access to the plasma. The latter are of paramount importance and are often neglected in this type of studies. Faithfully modelling them is not an easy task, especially in 3D simulations. The procedure herein proposed consists in using CAD models of a given machine, together with parameterizable models of the launcher, to produce a description suited for Finite Difference Time Domain (FDTD) 3D simulation, combining the capabilities of real-world CAD design with the power of simulation. However, CAD model geometric descriptions are incompatible with the ones used by standard FDTD codes. CAD software usually outputs models in a tessellated mesh while FDTD simulators use Volumetric Pixel (VOXEL) descriptions. To solve this interface problem, we implemented a pipeline to automatically convert complex CAD models of tokamak vessel components and wave launcher structures to the VOXEL input required by REFMUL3, a full wave 3D Maxwell FDTD parallel code. To illustrate the full procedure, a complex reflectometry synthetic diagnostic for IDTT was setup, converted and simulated. This setup includes 3 antennas recessed into the vessel wall, for thermal protection, one for transmission and reception, and two just for reception.

KEYWORDS: Nuclear instruments and methods for hot plasma diagnostics, Computing, Simulation methods and programs

---

[1]Corresponding author.

Contents

1 Introduction

In reactor grade tokamaks, such as ITER or DEMO, diagnostic design must fulfill a very restrictive set of requirements due to burning plasma radiation loads and particle fluxes on the plasma facing components. The impact of these restrictions on the development of diagnostics and on measurement performance must be assessed and incorporated in the system design loop. Solutions found need to be iterated by constant verification of design performance versus feasibility, namely by incorporating neutronics and thermal/stress analyses. Microwave reflectometry was proposed [1] to supplement traditional magnetic diagnostics used for control on ITER[1] and was demonstrated on ASDEX Upgrade [2] as a potential replacement capable diagnostic for feedback position control. This control scheme is now under active design as a primary source for control position and shape measurements in DEMO [3, 4] and to be further tested in future devices such as IDTT [5]. Using this diagnostic technique for control purposes involves distributing poloidally a large set of reflectometers [2], probing the plasma in various radial Lines of Sight (LOS). In this way, it is possible to reach a separatrix shape approximation [6, 7] by sampling its position at various poloidal locations.

Over the years, Instituto de Plasmas e Fusão Nuclear's Engineering Group (IPFN-EG) has relied on the in-house developed REFMUL* family of Finite-Difference Time-Domain (FDTD) full wave Maxwell codes in inhomogeneous anisotropic media [8–11] to improve the interpretation of the complex measurements generated by its numerous deployed reflectometry diagnostics [12]. The most recent REFMUL3 3D code [13] allows a realistic simulation that fully incorporates the essential 3D effects affecting the waves propagation in the magnetised plasma, and their interaction with the more or less complex geometries of the launcher and receiver antennas, vessel wall antenna access cutouts, and other surrounding plasma facing structures. However, the lack of tools to easily build these real-world models in the voxelized format required by the FDTD simulators traditionally

---

[1]The microwave reflectometry plasma position control diagnostic was descoped from the ITER project during the preparatory design phase.

[2]On DEMO a set of 16 reflectometers is foreseen for this control application [3].
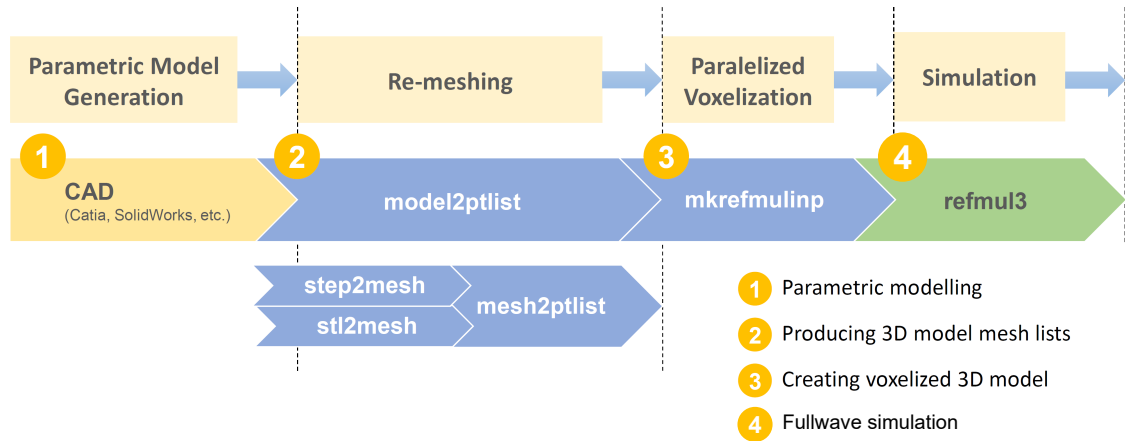
Figure 1. Microwave propagation simulation flow: 1) generation of parameterized real-world models, 2) re-meshing of standardized CAD model output and node and tetrahedron generation, 3) voxelized model generation, 4) fullwave simulation.

led to the oversimplification of these physical components. To integrate the simulation flow in the main tokamak diagnostic engineering design and validation flows, one needs to convert 3D models, in standard CAD formats, to the simulations voxelized input format.

Herein we present a multi-staged procedure to generate complex 3D model input for the REFMUL3 FDTD simulator from models generated by common CAD software tools in standard output file formats. The generation of the voxelized models involved the implementation of a software pipeline to post-process these files, generated from 3D drawings with parameterized antenna geometries and waveguide dimensions (band dependent), LOS angles (dependent on the facing separatrix configuration), simulation box sizes, and feature rich vessel and antenna access to plasma geometries. The developed pipeline capabilities are illustrated, throughout this paper, using data from the assessment of a Plasma-Position Reflectometry (PPR) antenna cluster for the future IDTT tokamak.

2   3D model generation flow

The design and validation flow of a reflectometer involves: defining the system design requirements according to the physics characteristics of the intended plasma regime(s), choosing the propagation mode and the standard microwave bands needed to cover the targeted density range, defining the type of antenna and waveguide to use according to in-vessel access and routing restrictions, and designing the optimal launcher/receiver antenna system and matching back-end electronics. It the last step, measurement simulations play a very important role. By incorporating the iterated antenna design, all the surrounding structures and geometry restrictions imposed by the design constraints, as well as realistic plasma and turbulence conditions, it is possible to assess the diagnostic's performance in the targeted plasma regimes. Simulation data can also be used to test and improve the data processing techniques used to reconstruct the density profiles in the selected configurations and assessed LOS. The simulation flow is depicted schematically in Figure 1. We will address herein its first three steps (1 - 3) and, in particular, steps 2-3 (blue arrows) which are related to the implemented
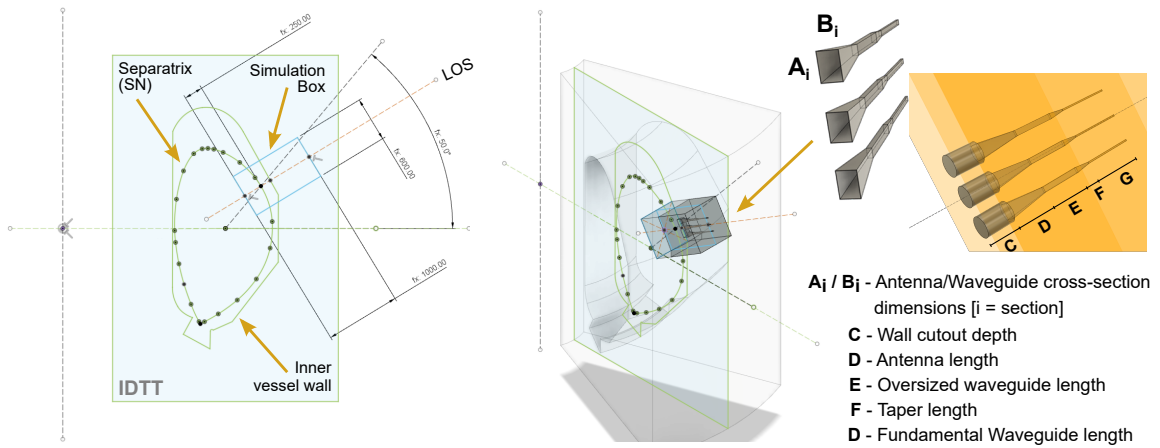
Figure 2. Parameterized drawings of a PPR cluster of antennas for IDTT. Poloidal (left), 3D rendering (center) and antenna cluster and protection hole geometry (right) drawings.

software pipeline that converts generic CAD models into input data to IPFN's REFMUL3 FDTD fullwave 3D simulator.

2.1 Parameterized CAD design

By using standard CAD tools, it is possible to easily incorporate in the simulation flow fine structural detail or to quickly produce finely tuned design iterations for performance assessment. To this end contribute the high parameterization capabilities and optimized drawing tools provided by this type of software. In our IDTT PPR assessment (see Fig. 2), it was possible to completely parameterize an antenna cluster in terms of its multiple component dimensions: antennas, waveguides, tapers, distance between antennas, shape and depth of the recessed holes in the vessel wall, etc. The LOS of the cluster was also parametrically defined, using a single angle in the poloidal plane and a perpendicularity constraint to the facing separatrix geometry, to facilitate the assessment and optimization of multiple PPR locations for the IDTT tokamak. Fine tuning any dimensional quantities, changing a system's LOS, the microwave band to simulate or the simulation box size becomes a matter of changing the values of a few parameters, and letting the CAD software update the corresponding model. Likewise, the underlying vessel and plasma facing structures can also be easily replaced or updated without a tedious and error prone hand-made re-drawing process.

2.2 3D Model re-meshing

The first step of the software pipeline, *model2ptlist* (figures 1 and 3), consists in remeshing the 3D CAD models into a tetrahedral volumetric mesh representation. Internally, CAD tools use constructive solid geometry techniques to create complex geometries from simpler objects (primitives, extrusions, revolutions, sweeps, filets, chamfers, etc). These objects can then be collectively translated into a boundary geometrical representation (BREP), a collection of connected surface elements representing the physical boundary of the solids, or into a tessellated representation, an approximation of this boundary by a mesh of triangles. The quality of this approximation is controlled by defining the maximum acceptable distance (or sag) between the triangles and the
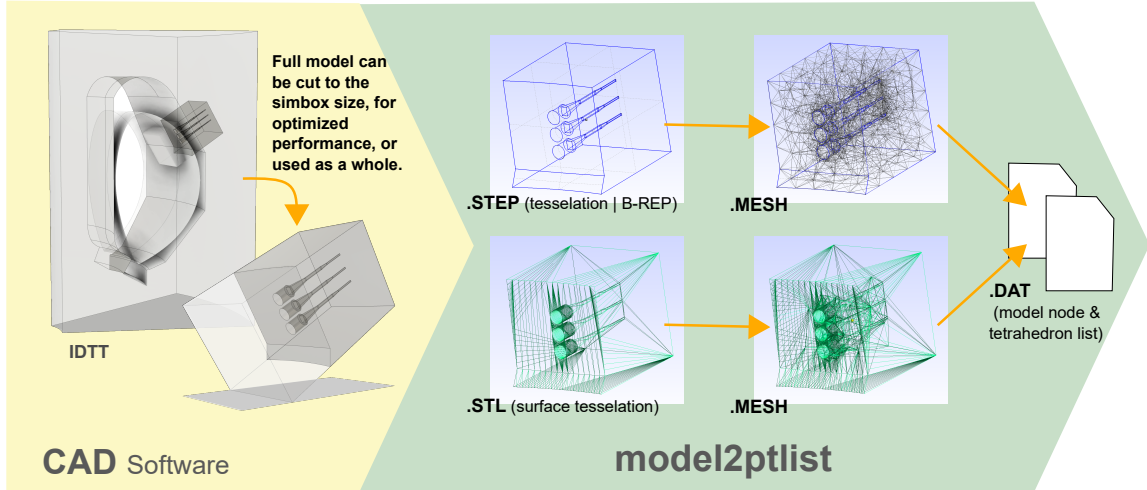
Figure 3.   Model remeshing *model2ptlist* pipeline stage: STEP model (top) and STL model (bottom) processing branches.

surfaces they approximate. *model2ptlist* can process standard non-proprietary CAD files using both representations: *step2mesh* handles BREP **St**andard for the **E**xchange of **P**roduct Data (STEP) files, and *stl2mesh* tessellated **S**tandard **T**essellation **L**anguage (STL) files. *model2ptlist* is completely written in Unix/Linux *bash* scripting language and makes use of standard Unix command line tools such as *awk*, *sed*, and *bc*, and also of *Gmsh* [14], an open-source 3D finite element mesh generator. Both *step2mesh* and *stl2mesh* use *Gmsh* to create the volumetric 3D tetrahedra mesh and generate an output ASCII *.mesh* file, containing the model's optimized conformal mesh. The fidelity of the approximation of *Gmsh*'s generated mesh to its internal BREP representation is controlled by setting the maximum size of the tetrahedron's edges and/or by computing the mesh element size from the curvatures, using a target number of elements per 2π radians. The latter parameter is particularly important when the models possess rounded surfaces or volumes. The last step of *model2ptlist*, *mesh2ptlist*, parses the *.mesh* files and creates two separate ASCII files with a list of all tetrahedrons in the mesh and the corresponding complete vertex list. *model2ptlist* can be used to generate tetrahedron lists of full tokamak vessel sections or just of the models generated from the intersection of the simulation box and the full model. Although the former leaves the definition of the size of the simulation box to the voxelization pipeline stage, increasing the process' flexibility, the penalty is an increased remeshing time, especially if finer resolution parameter values are used. This stage is, nevertheless, significantly faster than any FDTD simulation. For reference, the cut-to-the-simulation-box model in the left of figure 3, takes ≈ 5 s to remesh using a 5 cm maximum edge size and a 45 elements per 2π radians of curvature. As an output, *model2ptlist* also automatically produces the required *Slurm* [15] batch files to run the *mkrefmulinp* pipeline voxelization step in the batch queues of highly parallel HPC/supercomputers.

2.3   3D Model Voxelization

Classical Yee-cell-based FDTD simulators require data to be stored in a Cartesian uniform cell grid form. The spatial discretization of geometric entities into such cuboid grids, also known as
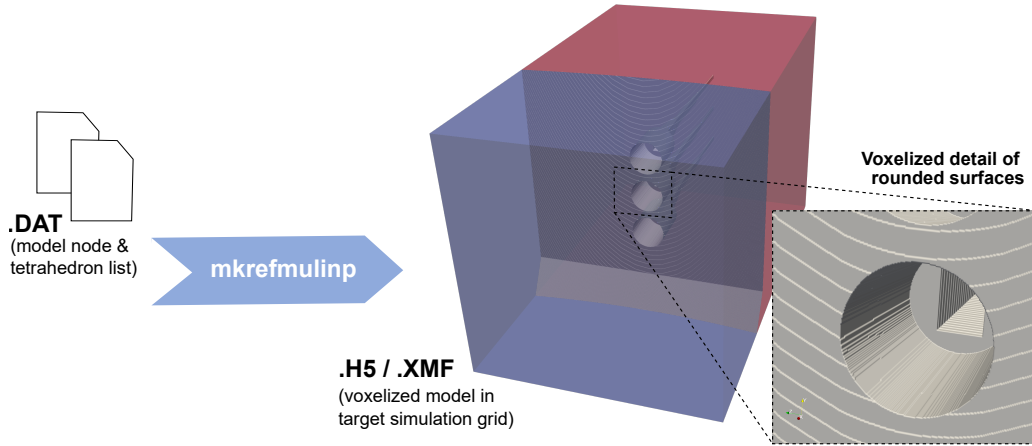
Figure 4. Voxelization of the re-meshed model of figure 3 generated by *mkrefmulinp*

voxelization, is a well-studied process (see [16] and references therein), usually performed in two steps: 1) determining the placement of structured mesh lines that best fit the model geometry, and 2) attributing the right properties to the cuboid cell, or voxel, also known as *material mapping*. Due to the specific characteristics of the reflectometry simulations, and the optimized volumetric meshing capabilities of *Gmsh*, it was possible to develop a simple voxelization stage that could be easily included in the REFMUL3 simulation flow. *mkrefmulinp* is a multithreaded C/OpenMP code, that uses the tetrahedra and vertex lists generated by *model2ptlist* and converts them into a voxelized representation stored in a HDF5 file compatible with REFMUL3 (figure 4). It also generates a *Slurm* batch file to run REFMUL3 in the batch queues of highly parallel HPC/supercomputers allowing the complete software chain, from the input of .STEP/.STL files up to the simulation output results, to be run in a fully automated/scripted way.

The grid spatial discretization step is driven by REFMUL3 phase accuracy constrains, that require that $Dxyz = dx = dy = dz = \lambda/20$, where λ is the wavelength in vacuum of the highest frequency wave in the simulation. Because REFMUL3 also requires that the electromagnetic fields be symmetric inside the fundamental waveguides cross-section, the frequency used to define $Dxyz$ is selected so that the discretization of the waveguide dimensions, where the signal is injected, always returns and even number of voxels in the corresponding plane. For simulating K, Ka, Q and V microwave bands, the optimal frequency/$Dxyz$ discretization pairs were, respectively: 30.00/0.50, 39.47/0.38, 49.18/0.31 and 74.63/0.20 [GHZ]/[mm]. Due to the high loads on plasma facing structures no components can have a thickness of the order of the highest $Dxyz$ value, i.e. 0.5 mm (K band). This fact and the conformal nature of the *Gmsh* generated volumetric meshes, guarantee that no single voxel layers will be encountered. For all these reasons, the implementation of more complex grid generation algorithms [16–18] could be avoided. Resolution of the voxelized models is further controlled during the previous re-meshing stage where the sag between the original arcs and corresponding approximated edges must be kept below $Dxyz$. In the example of figure 3, featuring a cylindrical hole with a 100 mm diameter, using a 45 edges/2π meshing parameter value guaranteed a sag of just 0.122 mm, below $Dxyz$ of all bands (K-V).

Since REFMUL3 admits excitation in the X plane, the simulation box is initially aligned so

that the LOS or the structure we want to excite (usually a waveguide) is aligned with this plane. The simulation box matrix, initialized with the value corresponding to vacuum, has its values changed only for voxels that lie inside the model's tetrahedra. Only tetrahedra entirely or partially inside the simulation box are dispatched to any available core. As no ray-casting approaches (such as [19]) were yet implemented, material mapping is performed by checking if every voxel, inside the parallelogram containing the received tetrahedron, belongs to it or not. A more balanced distribution of work between participating cores is obtained by setting a maximum value for the tetrahedra edge length during the re-meshing phase.

Table 1 shows the average time for the voxelization of a simulation box size of $1.0\,m \times 0.6\,m \times 0.6\,m$, using 1, 24, and 48 cores in a single *Marconi* [20] node (2 Xeon 8160 at 2.10 GHz, 48 cores, 192 GB DDR4 RAM) and for two microwave bands, K and Q. Depending on the band, and hence on the corresponding $Dxyz$, the size of the output voxelization matrix rises from $1999 \times 1199 \times 1199$ float32 (10.9 GB) to $3277 \times 1967 \times 1967$ float32 (47.2 GB). Three models were used in this benchmark: CSH – common antenna cluster squared hole (figure 5.1), SRH – separate cylindrical holes (figure 5.3) and FULL - CSH hole geometry inscribed in a full tokamak section model (left model of figure 3). In this optimized implementation, voxelization time depends little (<2-3%, for 1 core) on the total number of model's tetrahedra. Increasing the number of cores used it was possible to attain a computation time of the same order, but below, the time taken in memory allocation and, above all, IO time to write the voxelized matrix in the HDF5 format. Because the voxelization pipeline can run on the same computation infrastructure and file system as the REFMUL3 simulator it was possible to maintain all big data and file IO sharing in the same system, optimizing the total runtime of the simulations needed in the usual model validation simulation flow.

OpenMP Voxelization code running time					
Band / Simbox size	Model / #Tetrahedra	IO / Memalloc	1 core	24 cores	48 cores
K	CSH / 15874	45.67 s	1100.50 s	58.07 s	36.07 s
$1999 \times 1199 \times 1199$	SRH / 52617	45.17 s	1103.67 s	54.01 s	38.33 s
10.963 GB	FULL / 595373	45.73 s	1123.94 s	72.34 s	51.05 s
Q	CSH / 15874	160.37 s	4648.26 s	216.04 s	143.08 s
$3277 \times 1967 \times 1967$	SRH / 52617	161.10 s	4691.48 s	220.62 s	146.37 s
47.233 GB	FULL / 595373	161.70 s	4793.22 s	222.83 s	154.58 s

Table 1. Averaged run-time (3 runs) of the parallelized voxelization code on a single Marconi node (2 Intel Xeon 8160 CPUs at 2.10GHz / 48 cores / 192 GB DDR4 RAM).

3 Simulation of adapted models

REFMUL3 has been designed from scratch as a parallel code with a 3D domain decomposition, having a hybrid implementation using Message Passing Interface (MPI) and/or Open Multi-Processing (OpenMP). It possesses powerful input/output capabilities using either Hierarchical Data Format 5 (HDF5) with descriptors in eXtensible Data Model and Format (XDMF), as generated by *mkref-mulinp*, or using Visualization Toolkit (VTK) parallel format. Simulated transmission antennas (Tx) can also function as a receiving (Rx) ones, in a monostatic setup, or as part of a more com-
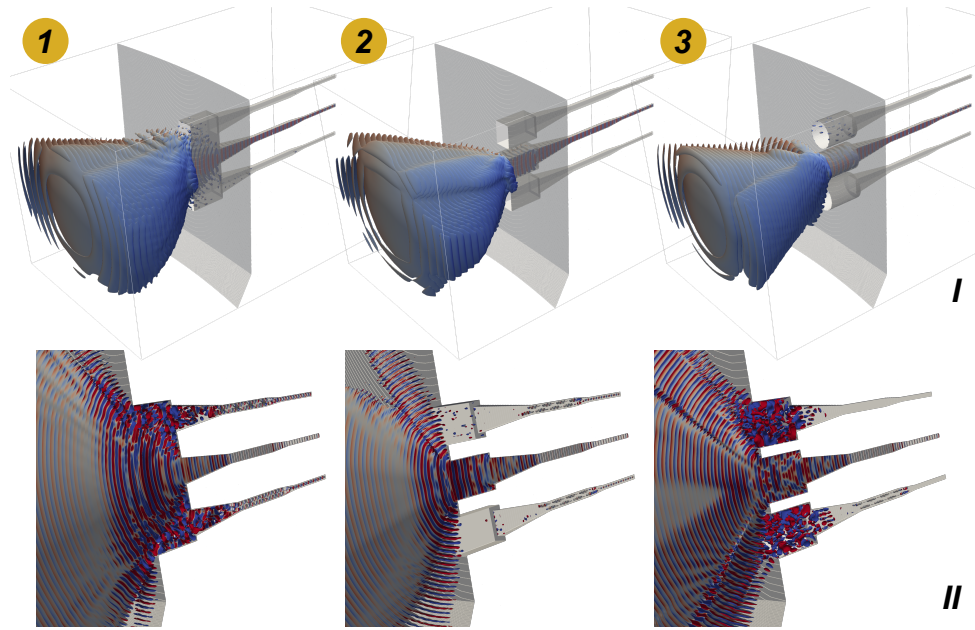
Figure 5. Beam propagation in vacuum, of a 3-antenna cluster, for different 10 cm recessed hole geometries: 1) single common rectangular hole, 2) individual rectangular holes, and 3) individual cylindrical holes.

plex cluster with several Rx antennas. Emitted signals, of fixed or continuously swept frequencies (Frequency-Modulated Continuous Wave - FMCW), are detected, after reflecting back to one or more Rx antennas, using a numerical homodyne or heterodyne detection. For each of the emission structures considered, and for each of the bands, three simulations are made: (i) one in vacuum, to assess the shape of the probing beam and evaluate the cross talk between antennas as observed in Figure 5; (ii) with a reflecting mirror, emulating the experimental technique used to determine the dispersion due to propagation in waveguides and antennas and (iii) with plasma as is the case appearing in Figure 6, where the reflectometer presented in Figure 5–1 is shown operating in the K band, probing a plasma surrounded with a Perfectly Matched Layer (PML). Although the detailed discussion of the presented simulation results is outside the scope of this publication, it is possible to observe how the different cluster hole geometries, Figure 5, and vessel wall geometry, Figure 6, affect the wave beam propagation patterns in vacuum and in plasma, respectively. Different beam shaping and crosstalk patterns due to hole geometry are exhibited in the former. Asymmetry and local compression of the interference patterns are seen in the latter, due to the wall geometry, to the curvature of the single-null magnetic flux distribution and corresponding density layer geometry, and to the wave-wall interactions. For reference, the full K band sweep (18-26.5 GHz) partially shown in Figure 6, performed in the $1.0\,m \times 0.6\,m \times 0.6\,m$ simulation box for 120000 iterations took ≈24 h to run on Marconi in 64 nodes of 48 cores (3072 cores in total).

4 Conclusions

We herein demonstrated a pipeline capable of producing 3D voxelized models for FDTD simulators, such as REFMUL3, from 3D models produced by standard CAD software. We proposed
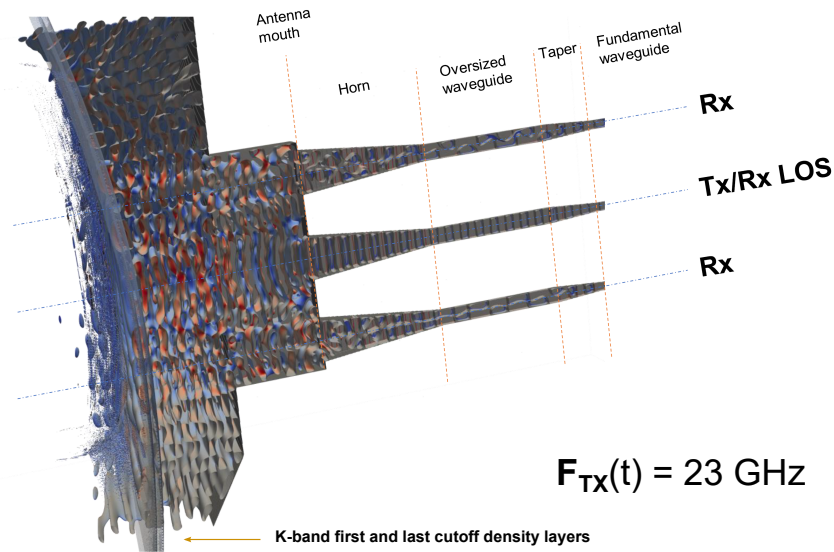
Figure 6. Propagation in plasma, K-band frequency sweep (snapshot taken when swept probing frequency reaches 23 GHz).

and validated a 3D model generation flow usable for producing highly realistic and detailed representations of tokamak vessel plasma facing structures, emitting and the receiving multi antenna clusters and complex plasma access geometries. The adaptation of standardized STEP ad STL CAD models guarantees that the models used for realistic full wave microwave simulations can be shared with neutronics, material stress and thermal analysis simulators. This add-on allowed REFMUL3 to become a full reflectometer simulator, capable of being used in sophisticated and integrated diagnostic design workflows. The analysis of the complex interaction between waves propagating in the plasma and the detailed antenna geometries and surrounding plasma facing structures, will be used in the future for the test and validation of new signal processing techniques to improve reflectometry measurements made in less favorable propagation conditions. One such example consists in probing the plasma along LOS oblique to the magnetic flux surfaces that requires the use of multi-antenna clusters. Finally, with the illustrated use case, we showed that the REFMUL3 FDTD software framework is now operational to accomplish the above-mentioned functionality.

Acknowledgments

References

[1] N. L. Bretz et al., *ITER position control reflectometry - conceptual design*, in *Diagnostics for experimental thermonuclear fusion reactors 2*, Plenum Press, (1998) 129–137.

[2] J. Santos, et al., *Reflectometry-based plasma position feedback control demonstration at ASDEX Upgrade*, *Nucl. Fusion* **52**(3) (2012) 032003.

[3] W. Biel et al., *Diagnostics for plasma control – From ITER to DEMO*, *Fusion Eng. Des.* **146** part A (2019) 465–472.

[4] Y. Nietiadi et al., *Nuclear and thermal analysis of a multi-reflectometer system for DEMO*, *Fusion Eng. Des.* **167** (2021) 112349.

[5] F. da Silva et al., *Assessment of measurement performance for a low field side IDTT plasma position reflectometry system*, *Fusion Eng. Des.* **168** (2021) 112405.

[6] G. Marchiori et al., *Study of a plasma boundary reconstruction method based on reflectometric measurements for control purposes*, *IEEE Trans. Plasma Sci.* **46**(5) (2018) 1285–1290.

[7] E. Ricardo et al., *Assessment of a multi-reflectometers positioning system for DEMO plasmas*, *J. Instrum.* **14**(08) (2019) C08010.

[8] F. da Silva et al., *Unidirectional transparent signal injection in finite-difference time-domain electromagnetic codes—application to reflectometry simulations*, *Journal of Computational Physics* **203** (2005) 467–492.

[9] F. da Silva et al., *Numerical advances in kernels for FDTD Maxwell codes using the Yee algorithm*, *11th Intl. Reflectometry Workshop for fusion plasma diagnostics — IRW11* (Palaiseau, April, 2013).

[10] F. da Silva et al., *Introducing REFMULF, a 2D full polarization code and REFMUL3, a 3D parallel full wave Maxwell code*, *13th Intl. Reflectometry Workshop for fusion plasma diagnostics—IRW13* (Daejeon, May 2017).

[11] Filipe da Silva et al., *Stable explicit coupling of the Yee scheme with a linear current model in fluctuating magnetized plasmas*, *Journal of Computational Physics* **295** (2015) 24–45.

[12] F. da Silva et al., *Modelling reflectometry diagnostics: finite-difference time-domain simulation of reflectometry in fusion plasmas*, *J. Instrum.* **14** (2019) C08003.

[13] R. Hatzky, *HLST Core Team Report*, *EUROFUSION WPISA-REP* (18) (2017) 20587.

[14] Geuzaine, C., and Remacle, J. F., *Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities*, *International Journal for Numerical Methods in Engineering* **79**(11) (2009) 1309–1331.

[15] https://slurm.schedmd.com/documentation.html

[16] Berens, M. K. et al., *Structured Mesh Generation: Open-source automatic nonuniform mesh generation for FDTD simulation*, *IEEE Antennas and Propagation Magazine* **58**(3) (2016) 45–55.

[17] T. Ishida, et al., *Efficient and Robust Cartesian Mesh Generation for Building-Cube Method*, *J. Comput. Sci. Technol.* **2**(4) (2008) 435–446.

[18] H.-S. Kim et al., *Generation of Non-uniform Meshes for Finite-Difference Time-Domain Simulations*, *J. Electr. Eng. Technol.* **6**(1) (2011) 128–132.

[19] Szucki, M., and Suchy, J., *A voxelization based mesh generation algorithm for numerical models used in foundry engineering*, *Metallurgy and Foundry Engineering* **38**(1) (2012) 43.

[20] https://www.hpc.cineca.it/hardware/marconi