Modeling of SHF/EHF Radio-Wave Scattering for Curved Surfaces with Voxel Cone Tracing

Iuliia Tropkina, Alexander Pyattaev, Yekaterina Sadovaya, and Sergey Andreev

Abstract—**Efficient and accurate radio propagation modeling is essential for optimization of both radio sensing and communication systems. However, highly accurate full-wave methods remain inefficient at high frequencies, as unit of computation has to be made much smaller than the wavelength. On the other hand, ray-based approaches offer the desired speed, but the surface element (typically, a triangle) must be made much larger than the wavelength, thus making it difficult to represent complex curved surfaces of common objects, such as cars or unmanned aerial vehicles. As a result, for SHF/EHF bands, it is challenging to select a method that is both fast and capable of capturing curved surfaces correctly. To address this issue, we present a method that offers a reasonable trade-off between speed and accuracy for radio propagation modeling in the bands of interest. Specifically, we combine efficient voxel scene representation targeting a cone tracing algorithm with a statistical scattering model. To confirm the validity of our approach, we report the dependence of reflected power on the distance for basic primitives, such as cone and sphere, for which closed-form radar cross-section solutions are known. We demonstrate that the proposed approach is superior compared to the conventional ray-tracing solutions, while posing no restrictions for efficient voxel cone tracing implementations.**

Index Terms—**scattering, cone tracing, voxel-based representation, curved surface, radar, simulation, modeling.**

I. Introduction

Efficient wave propagation modeling is crucial to analyze and improve any radio system. It provides a realistic assessment of the overall system performance, and reveals possible challenges. Moreover, it reduces the need for costly experiments and channel measurements in the field. For instance, as the interest in unmanned aerial systems grows, simulation tools are required to predict issues with their communication links and RADAR sensors. In fact, there already exists a prototype tool for concurrent simulations of robotic and network aspects of drone swarms based on ROS/Gazebo [1]. The primary motivation of our work is to complement such tools with fast radio-wave propagation simulation in scenes with high complexity.

The commonly used electromagnetic channel modeling methods, their corresponding algorithms, and scene representations are summarized in Table I. These methods can be grouped into full-wave and asymptotic solutions. The full-wave methods include Method of Moments (MoM), Finite Element Method (FEM), and Finite-Difference Time-Domain (FDTD). For MOM and FEM, the channel characteristics are determined by employing numerical analysis to obtain solutions to Maxwell's equations, which, in turn, requires subdivision of environment surfaces into a number of small elements (cylindrical, rectangular, or triangular) [2]. The number of corresponding ordinary differential equations (ODEs) with complex or inhomogeneous scenes increases significantly, thus causing heavy computational and memory burden [3]. For the FDTD methods, the algorithms are based on a discrete-time solution to Maxwell's equations, and therefore call for a voxel (aka ”Yee lattice”) scene representation [4].

TABLE I: Algorithms and scene representation options in existing channel modeling methods

Method	**Key algorithm**	**Scene representation**
Curve fitting	Gauss–Newton	None
FEM/MOM	ODEs	Mesh, NURBS
FDTD	Huygens principle	Voxels
GO (GTD/UTD)	RL, RT, photon mapping	Mesh, parametric
Proposed	Cone tracing (UTD)	Voxels

A typical performance bottleneck of FDTD and other full-wave methods is in the voxel size, which must be made much smaller than the wavelength used in the modeling [3]. As a result, simulations of larger SHF/EHF band scenarios may become infeasible due to memory consumption and computational cost. Asymptotic methods rely on geometric optics (GO), and algorithms, such as ray tracing (RT) or ray launching (RL), have both been used successfully for high-frequency modeling of large scenes in multiple works [5]. A scene is typically represented by triangular meshes, which are easily handled by dedicated hardware. Despite the fact that the modern ray-based algorithms can process scenes of high complexity with acceptable speeds, they are not capable of fully capturing the details of curved objects. This is because any curve ends up broken down into multiple flat faces, with exact curvature information lost in conversion.

In addition, for ray-based methods, the surface element must be made much larger than the wavelength. Consequently, it becomes cumbersome to represent complex objects in detail. While this may not be a problem for models of buildings [6], it does become one where objects of interest have intricate ”organic” shapes, such as aerodynamic surfaces of drones or environmental features, such as trees, fences, and wires. To address this challenge, several authors applied Nonuniform Rational B-Splines (NURBS) for surface representation [7], [8]. Multidimensional Halton sequences were used for scene representation with conventional RT in dynamic scenes, by

I. Tropkina, Y. Sadovaya, and S. Andreev are with Tampere University, (e-mail: firstname.lastname@tuni.fi), Tampere, Finland.

A. Pyattaev is with YL-verkot Oy, (e-mail: ap@yl-verkot.com), Finland.

showing reasonable accuracy [9]. However, in such solutions, the computations become nearly as expensive as those for the full-wave models, thus limiting their benefits. Further, NURBS do not solve any of the issues with trees and thin wires.

Based on the above, the following observations can be made. Today, to accurately model the interaction of, for example, an unmanned aerial vehicle (UAV) with RADAR of another UAV, one may utilize any of the full-wave methods, or asymptotic methods with NURBS scene representation. However, both of these approaches have very low framerate. In some cases, this can be mitigated by multi-core computation, but if the frames are temporally correlated (such as in UAV simulations), radio channel modeling becomes a bottleneck. Hence, the target of this work is the development of a method using ray-based algorithm (for speed) that is capable of processing curved surfaces (to capture e.g., UAVs) with higher accuracy than what is available for triangular meshes, but without the computational costs associated with NURBS.

The rest of this text is organized in the following way. The conventional ray-based algorithm challenges are briefly discussed in Section II-A. Section II-B and II-C present the proposed method. Section III describes the theoretical foundations and test scenarios. The comparison with alternative methods is made in Section IV. Our simulation results are reported in Section V. Conclusions are made in Section VI.

II. Proposed Modeling Method

A. Challenges of Ray-Based Algorithms

Before being processed in a RT or RL, 3D surfaces must be represented by triangles. This type of primitive is preferred because the ray–triangle intersection algorithm is relatively straightforward, and has adequate GPU support. Moreover, it is possible to represent an object of arbitrary shape through triangulation. Hence, a ray-based model typically treats ray–triangle interaction as a ”kernel”, which is then applied across all possible interactions. Specifically, if a ray hits a triangle, it can be reflected or refracted following the laws of GO. In certain cases, ray–edge interaction is also traced, thus allowing for knife-edge diffraction to be modeled. This approach causes the following widely known issues:

1) If the number of rays radiated from a transmitter (Tx) is low, some faces may not be illuminated, thus producing gaps in reflections of sharply curved objects.
2) Every flat surface (even if represented by multiple adjacent triangles) should be ”counted” by the rays emitted from a given source only once. In particular, only one reflection should be made from a wall that may consist of several triangles. Hence, this presents issues when aiming to distinguish a genuinely flat object from a gently curved surface, which should produce unique reflected rays for each triangle in the mesh.
3) A curved object is visible at the Rx as a set of discrete rays rather than a continuous spread of rays. If the number of those rays is sufficiently small, this could cause discontinuities in reflected power when environment objects, such as drones, move.

B. Voxel-based Scene Representation

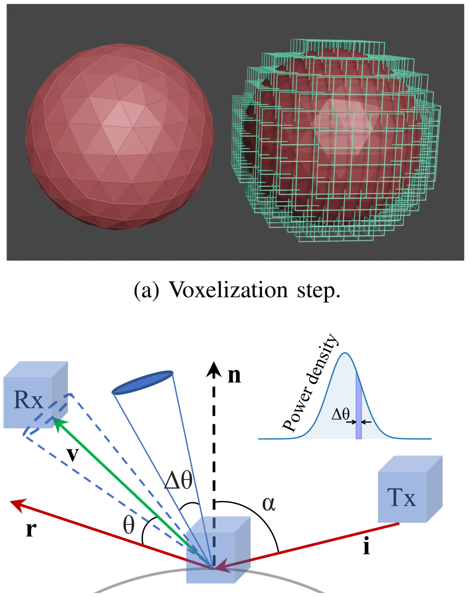
The voxel-based scene representation has been successfully applied to graphics engines for global illumination in [10], which is reasonably similar to radio propagation modeling. The authors of that paper use a voxel octree structure to capture the scene geometry, which is built once for the static part of the scene, and is then updated as needed to reflect modifications of the environment. The authors demonstrate high fidelity simulation of light with reflections, scattering, and transmission at above 30 frames per second on extremely complex scenes. Currently, there are no studies that adapt that structure and associated algorithms for radio propagation simulations. Inspired by this, we investigate the applicability of a voxel scene structure in combination with a cone tracing algorithm for radio propagation modeling. There are, naturally, important differences that are worth noting.

First, the voxel size has to be made as small as possible to capture the details of the environment (as voxel size directly impacts the spatial resolution), yet it has to remain much larger than the wavelength, such that GO could still be applied to it. Hence, for most practical frequencies, voxels end up rather large, and special care needs to be taken to capture their interactions with the radio waves. In our implementation, during scene voxelization (see Fig. 1a), every voxel stores normals of all surfaces it encompasses. This provides a sensible compromise between the amount of information stored and the possibility for the calculation of scattering at the later stages. Further, one has to store the speeds of the objects in every voxel, such that appropriate Doppler calculations could be applied later on.

C. Calculation of Voxel’s Scattering

The rationale behind the proposed method is in the possibility to distinguish flat and curved surfaces, and in the calculation of the scattering coefficient for the latter. Hence, the local surface curvature can be captured. A surface is considered curved within the voxel if the angle between any two stored normals is larger than a threshold of T_{flat}. In this case, we consider an object as a scatterer. There are multiple approaches in computer graphics for fast scattering calculations [11]–[13]. Typically, the authors assess the intensity of reflected light via one of the following scattering models. The Lambertian model implies that the surface is diffusely reflecting, where luminous intensity of the reflected light obeys Lambert’s cosine law. According to the Phong model, the reflected light is a combination of diffuse Lambertian reflection with specular reflection.

This approximation demonstrates adequate results for curved objects with natural-looking pictures, yet neither of the above is directly applicable to radio propagation modeling. We follow the steps that are similar to Phong model for scattered power calculations, by taking into account the specifics of a cone tracing algorithm as follows. If a slightly curved surface with the average normal of $\mathbf{n}$ is illuminated by an incident cone propagating along the direction $\mathbf{i}$ as shown in Fig. 1b, one can calculate the expected direction $\mathbf{r}$ along which most



(a) Voxelization step.

(b) Ray propagation near a curved surface.

Fig. 1: Calculation steps.

power would be reflected, i.e. the specular component in the Phong model. However, some energy is reflected along other directions as well, and its amount depends on the angle of θ, which is between **r** and the direction of each reflected ray **v**. The latter is the main difference between our method and the Phong model.

We propose to estimate the power loss within each reflected cone as the integral of a certain power density function over the apex angle $\Delta\theta$ of the reflected cone:

$$K_p = \int\limits_{\Delta\theta} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} dx. \tag{1}$$

Then, coefficient K_p is utilized for the calculation of power reflected from the voxel along a specific reflected cone. Alternatively, if the reflected cone is found to hit the Rx, we can compute $\Delta\theta$ based on the Rx effective aperture and the distance between the voxel and the Rx.

In addition, it is necessary to define the limit cases for the types of reflected surfaces. For this, the following approach is utilized. Wherein the maximum angle between the normals is less than T_{flat}, the surface is considered as flat, and the distribution should be set to a delta function. The reflection can be calculated according to existing techniques, for example, GO physics solutions. This method provides the desired accuracy for flat surfaces [6]. Then, to take into account diffraction, one can set the diffraction threshold T_{dif}. Hence, if the angle between normals exceeds T_{dif} degrees, we consider the object to be an idealized scatterer with the radar cross-section (RCS) equivalent to a sphere inscribed into the voxel. In this case, the distribution should be set to uniform.

In this letter, we only consider PEC materials, but our method can be extended to support dielectric materials and associated refractions. This would require an additional distribution per voxel for refraction, as well as a dedicated study for different polarizations, with the rest of the logic remaining unchanged. However, the implementation would become more complicated.

III. Test Scenarios and Interpretation

We consider several test scenarios to evaluate the accuracy of the proposed method and compare it with the conventional ray-based approaches. We use primitives for which the derivation of the RCS expressions is possible. From this limited set of curved objects, we chose a metal sphere to demonstrate the scattering on the uniformly curved face and a cone to consider the case of a sharp-end element. The analytical expressions were derived using PO algorithm for a sphere and a cone, correspondingly [14]:

$$\sigma = \begin{cases} \pi r^2, & r \gg 2\lambda; \\ 144\pi^5 r^6/\lambda^4, & r \ll 2\lambda, \end{cases} \tag{2}$$

where r is the sphere radius; λ is the wavelength; and

$$\sigma = \pi a^2 \text{tg}^2(\alpha), \tag{3}$$

where a is the cone radius; α is the cone apex angle.

If the Tx and the Rx points are co-located, the power reflected from an object can be estimated by using the following equation [15]

$$P_{Rx} = \frac{\sigma P_{Tx} G_{Tx} S_{Rx}}{(4\pi)^2 R^4}, \tag{4}$$

where P_{Tx} is the Tx power; G_{Tx} is the Tx antenna gain; R is the distance between the object and the Tx/Rx point; S_{Rx} is the Rx antenna area.

IV. Computational Complexity

In this section, we compare the complexity of the proposed voxel-based method with that of the existing solutions. Firstly, the computational complexity of highly accurate MOM approach is much higher than that of our proposed method. As an example, the well-known NEC++ MOM engine may represent a sphere of radius 5λ with about $N = 1000$ elements, where each element is internally an ODE. However, to solve the system of N equations, one needs at least $O(N^2)$ operations, and a substantial amount of memory to store the intermediate results. In our case, for a sphere of the same size and the same number of elements N, the solution speed is $O(N)$, with no extra memory requirements. Further, in our approach, the number of instructions per point is minimal, and the code has no branching. For the sphere example, the difference in runtime speeds is on the order of 1000.

Secondly, regarding the conventional RT method, the simulation is divided into two steps: searching for the intersections and physics calculation. If we consider Moller–Trumbore ray–triangle intersection algorithm, it requires at least four scalar product operations per triangle. For our method, we exploit the operation of a linear iteration of the voxel indices, which is much less computationally expensive per surface element. At the second stage, one should reach similar computational

TABLE II: Comparing the number of elements for the conventional RT and the proposed method

	Sphere		Cone	
R, m	Voxels (new)	Faces (RT)	Voxels (new)	Faces (RT)
3	674	320	138	136
4	1250	1280	174	178
5	1898	5120	186	190

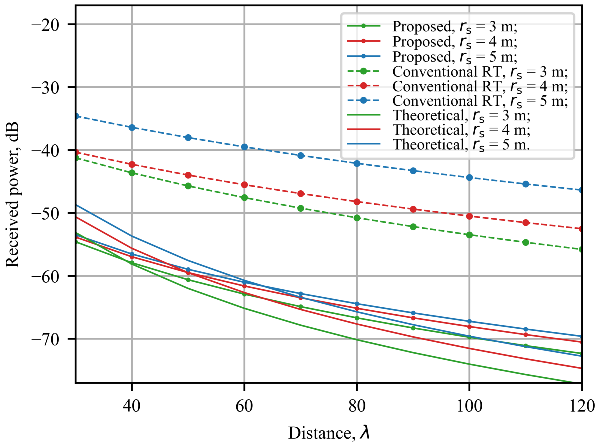


Fig. 2: Received power vs. distance for the sphere with the radius of 3, 4, 5 m.

complexity for RT and the proposed approach, but with substantially improved accuracy in the latter case. This is considered further in Section V.

Comparison with volume subdivision methods like FDTD is even more one-sided due to the cubic growth in the number of cells needed to represent the environment.

V. Modeling Results

Here, we compare the accuracy of the conventional RT with that of our method for the same operating costs. The computational complexity under test scenarios for the two approaches in terms of the number of processed elements per object is presented in Table II. Because the number of elements is limited by the subdivision value of an object, the values were chosen as close to each other as possible. We report the dependence of the reflected power at the Rx on the distance from the object to the Tx/Rx for different object parameters. The results are obtained using the conventional RT modeler, our proposed method, and theoretical equations [15]. For our scene representation, the voxel size is set equal to 0.5 m. It is proposed to consider the conventional RT with Lambertian scattering function enabled, which is employed in REMCOM software widely accepted for industry-grade simulations [16].

As can be seen in Fig. 2 and 3, the proposed method achieves better agreement with the theoretical results. Quantitatively, our solution demonstrates the mean dB error of 0.06 dB (vs. 17.16 dB for the conventional RT), and the mean squared dB error of 3.06 dB (vs. 17.93 dB for the conventional RT). Hence, we establish that our approach provides a significant reduction in the mean squared dB error (over 82%). In addition, since the authors of other works consider that the dB error level of 1 dB is acceptable for such scattering simulation tools, and that the level of 0.01 dB is very accurate, we can conclude that the achieved precision of the proposed method is sufficient for scattering modeling unlike the conventional RT [17].

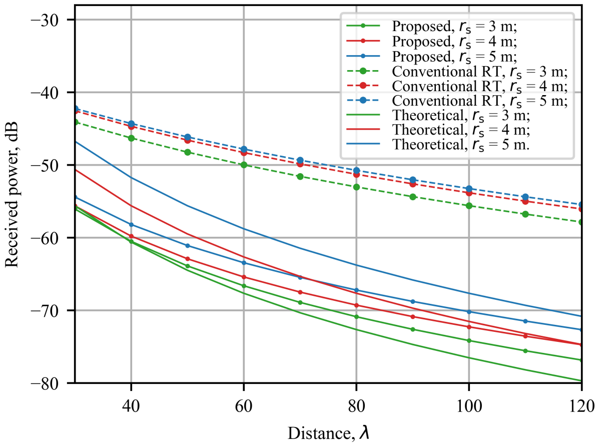


Fig. 3: Received power vs. distance for the cone with the base radius of 3, 4, 5 m.

It is worth adding that the conventional RT without Lambertian scattering enabled is known to be unable to find any paths for a cone shape. This causes difficulties where the scene contains multiple objects, as one may not be certain that all the objects are ”captured” in the results. In contrast, our approach guarantees that all the objects are represented in the results, irrespective of their shapes and sizes. It can be concluded that when the computational costs of the second stage are the same, our solution is much more accurate. On the other hand, when the stage of intersection searching is taken into account, our method is faster and more accurate compared to the conventional RT.

VI. Conclusions

A new modeling method for SHF/EHF radio-wave scattering, as shown in this letter, is indeed capable of addressing most of the requirements for fast and accurate radio-wave propagation simulation in scenes with high complexity. It allows to capture the details of sharply curved surfaces across a wide range of frequencies without having to sacrifice performance, and is thus suitable for studies of diffuse scattering, even in the EHF bands. The proposed scene representation is known to be suitable for a voxel cone tracing algorithm; the modified Phong reflection model was demonstrated to be comparable in terms of accuracy with ray-based modelers. Taken together, they make it possible to run radio propagation simulations at superior speeds using algorithms and data structures that are well-suited for hardware acceleration on commodity GPUs.

Acknowlegments

This work was supported by a personal grant from the Foundation of the Association of Electrical Engineers, by the Academy of Finland (Projects RADIANT and IDEA-MILL), and by the JAES Foundation (Project STREAM).

References

[1] M. Calvo-Fullana, D. Mox, A. Pyattaev, J. Fink, V. Kumar, and A. Ribeiro, “ROS-NetSim: A framework for the integration of robotic and network simulators,” *IEEE Robotics and Automation Letters*, pp. 1–1, 02 2021.

[2] K. Yee, “Numerical solution of initial boundary value problems involving Maxwell’s equations in isotropic media,” *IEEE Transactions on Antennas and Propagation*, vol. 14, no. 3, pp. 302–307, 1966.

[3] M. Särestöniemi, M. Hämäläinen, and J. Iinatti, “An overview of the electromagnetic simulation-based channel modeling techniques for wireless body area network applications,” *IEEE Access*, vol. 5, pp. 10622–10632, 2017.

[4] Z. Shi, P. Xia, Z. Gao, L. Huang, and C. Chen, “Modeling of wireless channel between UAV and vessel using the FDTD method,” in *10th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2014)*, pp. 100–104, 2014.

[5] Q. Hu, Y. Cai, A. Liu, G. Yu, and G. Y. Li, “Low-complexity joint resource allocation and trajectory design for UAV-aided relay networks with the segmented ray-tracing channel model,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 9, pp. 6179–6195, 2020.

[6] V. Semkin, D. Solomitckii, R. Naderpour, S. Andreev, Y. Koucheryavy, and A. V. Räisänen, “Characterization of radio links at 60 GHz using simple geometrical and highly accurate 3-D models,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 4647–4656, 2017.

[7] F. Weinmann, “SBR ray tracing on NURBS for electromagnetic scattering simulations,” in *2012 6th European Conference on Antennas and Propagation (EUCAP)*, pp. 2948–2951, IEEE, 2012.

[8] J. Barz, O. Abert, and S. Muller, “Interactive particle tracing in dynamic scenes consisting of NURBS surfaces,” in *2008 IEEE Symposium on Interactive Ray Tracing*, pp. 139–146, IEEE, 2008.

[9] F.-M. Wei, J.-X. Lin, L. Zhu, and W. Gao, “Terahertz scattering computation of complex structures modeled by NURBS surfaces,” in *2019 International Applied Computational Electromagnetics Society Symposium-China (ACES)*, vol. 1, pp. 1–2, IEEE, 2019.

[10] C. Crassin, F. Neyret, M. Sainz, S. Green, and E. Eisemann, “Interactive indirect illumination using voxel cone tracing,” in *Computer Graphics Forum*, vol. 30, pp. 1921–1930, Wiley Online Library, 2011.

[11] S.-P. Chuah, N.-M. Cheung, and C. Yuen, “Layered coding for mobile cloud gaming using scalable Blinn-Phong lighting,” *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3112–3125, 2016.

[12] X. Zhang and Y. Gao, “Generalised ambient reflection models for Lambertian and Phong surfaces,” in *2009 16th IEEE International Conference on Image Processing (ICIP)*, pp. 3993–3996, IEEE, 2009.

[13] J. Pascual-García, J.-M. Molina-García-Pardo, M.-T. Martínez-Inglés, J.-V. Rodríguez, and N. Saurín-Serrano, “On the importance of diffuse scattering model parameterization in indoor wireless channels at mm-wave frequencies,” *IEEE Access*, vol. 4, pp. 688–701, 2016.

[14] C. A. Balanis, *Advanced engineering electromagnetics*. John Wiley & Sons, 2012.

[15] S. Kingsley and S. Quegan, *Understanding radar systems*, vol. 2. SciTech Publishing, 1999.

[16] V. Degli-Esposti, F. Fuschini, E. M. Vitucci, and G. Falciasecca, “Measurement and modelling of scattering from buildings,” *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 1, pp. 143–153, 2007.

[17] B. Kolundzija, A. Krneta, D. Olcan, and M. Kostic, “Highly Accurate 3D EM Modeling Based on Ultra High Order Basis Functions,” in *2019 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting*, pp. 1449–1450, IEEE, 2019.