On the Digitization of the EM Environment: A Comparison of Ray Launching Solutions

Michele Zhu, Lorenzo Cazzella, Francesco Linsalata, Maurizio Magarini, Matteo Matteucci, Umberto Spagnolini
Department of Electronics Information and Bioengineering, Politecnico di Milano, Milan, Italy
Email: name.surname@polimi.it

***Abstract*—Digital Twins (DTs) have emerged as a promising method to accurately represent wireless propagation environments. The resulting virtual representation facilitates comprehensive insights into the behavior of the wireless channel, empowering multi-layer decision-making processes at the physical communication level. This paper investigates the digitization of wireless communication propagation, with a particular emphasis on the indispensable aspect of ray-based propagation simulation for ensuring real-time information within the DT up-to-date simulation. Through the introduction of a flexible evaluation framework, this work provides a comparative analysis of heterogeneous simulation software within an urban environment. Empirical analyses show the behaviour of ray-based solutions at increasing ray interactions depth with and without the activation of diffuse scattering simulation.**

***Index Terms*—Digital Twin, Ray Launching, Ray Tracing, Radio Propagation, Radio Map**

I. Introduction

The aim of a *Digital Twin* (DT) is to provide a high-fidelity digital representation of a physical entity or process. This involves a constant update of the virtual representation from the status of the real-world entity, and the definition of a suitable model for the evolution of the virtual representation. Taking decisions based on the information provided by the digital twin closes the loop by acting on the real-world entity.

In the wireless propagation modeling context, DTs can be a profitable way to represent the evolution of the electromagnetic (EM) environment [1]. Among the approaches available to model the wireless channel, ray-based simulation methods have emerged as one of the most accurate ones in high frequency communications settings when coupled with high-fidelity 3D maps of the environment [2].

High frequency wireless propagation is notably featured by multi-path propagation. Ray-based propagation simulation stands as a versatile modeling tool, offering estimates of path loss, angle of arrival/departure, propagation delay, and Doppler shift for each multi-path component. Ray-based simulation relies on the high frequency approximation of Maxwell's equations, resulting in the concept of ray. Moreover, it allows for effective integration with 3D maps of the environment to flexibly and faithfully model propagation environments presenting diverse geometric features [3].

By employing ray-based solutions, a DT can better simulate the real-world conditions of the EM environment in urban settings, taking into account the complexity of signal propagation in such scenarios by considering different possible interactions with the environment—e.g., reflection, diffraction and diffuse scattering. This enables us to achieve more accurate estimates and designs for wireless communication systems, specifically in dynamic vehicular environments where traditional models may fail to accurately represent the channel features [4].

The use of a suitable ray-based approach and the availability of an accurate 3D map of the propagation environment highly impact on the quality of an EM DT [1]. In [5] simplification of ray model based simulations are proposed and evaluated in end-to-end network simulations. In [6], the authors suggest an alternative dynamic ray method to alleviate the computational burden of ray tracing simulations, while a streamlined construction for radio propagation modeling through ray-based methods is designed in [7].

Ray-based wireless propagation simulation is an expensive operation [5]–[8], and can easily become a computational bottleneck towards achieving real-time performance in the update of a DT. Furthermore, the dynamism of the environment may render the DT information outdated even before the end of its computation.

To the best of our knowledge the current literature does not provide a comparison of available ray-based simulation solutions on common system architectures. In this paper, we consider a selection of commercial and open simulation software aiming at evaluating their computational performance over an urban propagation setting considering different link simulation conditions.

Contributions: In this paper, we propose the following main contributions:

- We propose an empirical method for evaluating and comparing the computational performance of heterogeneous ray-based simulation software over an urban wireless propagation scenario.
- We provide simulation results on the proposed urban scenario for a set of increasing ray interaction depths with the environment; we consider both propagation simulation with and without diffuse scattering, and two different link simulation conditions, i.e., single link vs. simultaneous multiple links, evaluating the latter on the GPU-accelerated available methods.

Organization: The remainder of this article is structured as follows: in Sec. II, we provide an overview of the basic ray-based propagation simulation algorithms. Sec. III describes

979-8-3503-0948-5/24/$31.00 ©2024 IEEE

the 3D maps of the urban environments considered for simulation, introduces the type of performed simulations, and the simulation software considered for comparison. In Sec. IV, we provide the simulation results and, finally, Sec. V draws the conclusions.

II. Ray-based propagation algorithms

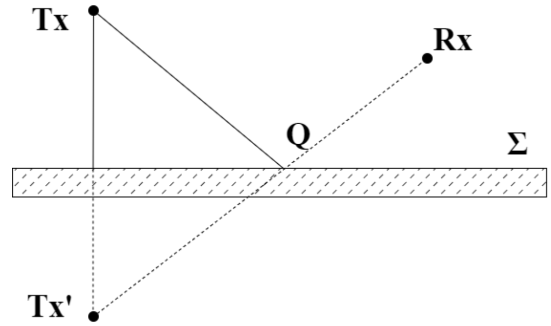
In this section, we introduce the main algorithms used for ray-based propagation simulation and we discuss the challenges of a direct comparison between heterogeneous propagation simulation software. Ray-optical methods can be divided into the following approaches:

- **Ray launching** is an approximate algorithm based on spawning rays from an EM emission source towards a set of angularly discretized directions. For each launched ray, after a number interaction steps (which is usually a simulation parameter) with the propagation environment, the reception of the ray at a target Rx is checked—e.g., by means of ray-tube modelling [9] or testing if the ray hits a sphere of given (possibly parametric) radius centered at the Rx point.
- **Ray tracing** (RT) aims at modeling point-to-point propagation by determining the geometrically-exact propagation paths between an EM emission source and a target Rx location. The computational complexity of the ray tracing task can easily become intractable with the increase of the number of surfaces to be taken into account or with the maximum number of interaction points per ray to be considered.

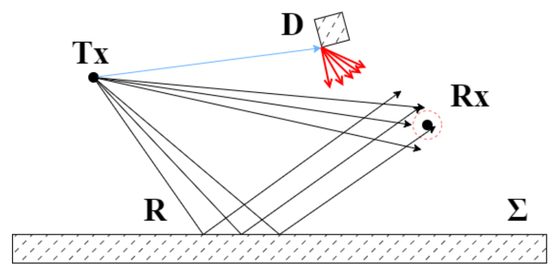
These approaches are implemented through a set of fundamental algorithms in high-frequency computational EM, among which are the main ones discussed below.

The **Image Method** is a point-to-point propagation simulation method. It is a realization of the above-discussed ray tracing approach. We provide in Fig. 1a a graphical representation of the image method for the single reflection case. The trajectory of a ray reflected from a plane surface can be determined by computing Tx′ as the image of Tx with respect of the reflection plane Σ, connecting Tx′ and Rx are connected though a segment intersecting the plane in point Q, and determining the reflected ray as composed by the segments connecting Tx, Q and Rx. In case of multiple reflections, the method can be extended recursively taking into consideration different reflection planes. The image method becomes computationally expensive when the environment presents a high number of reflecting surfaces or the number of considered sequential interactions for a ray increases [3].

The **Shooting and Bouncing Rays** (SBR) algorithm is a realization of the ray launching approach. We provide in Fig. 1b a graphical representation of SBR. A set of rays is launched from the Tx with a given angular separation. In conventional implementations, the launched rays are propagated in the environment up to a maximum number of interactions with surfaces (owing, e.g., to reflection, diffraction and scattering) and are considered to reach the Rx if they meet a reception condition—e.g., intersecting a sphere centered at the Rx. SBR implementations usually consider a variety of environmental interactions, e.g., reflection, diffraction, and diffuse scattering.



(a) Image method



(b) SBR

Fig. 1: (a) Image Method; Tx′ is the image of Tx with respect of the reflection plane Σ. (b) Shooting and Bouncing Rays method; R indicates relection points, D a diffraction point at an edge, and a path is considered at the receiver if it satisfies the reception condition—exemplified by hitting a reception sphere (red circle) at the Rx.

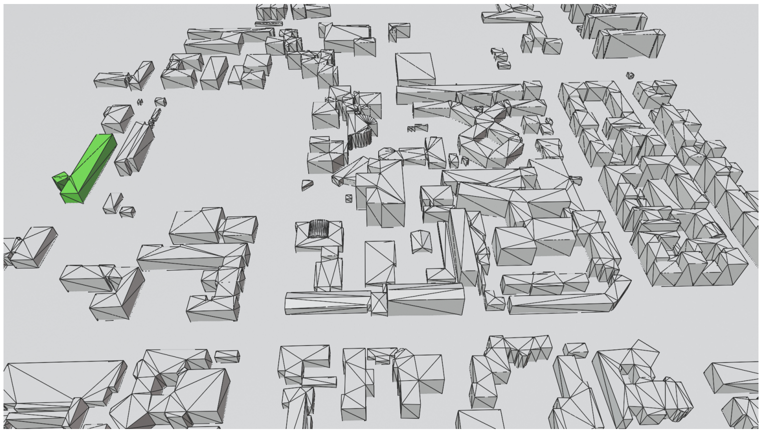
The latter has been shown to be particularly relevant to accurately model urban propagation environments [10].

Path correction methods (PCM) [3] were introduced to achieve exact geometric accuracy while exploiting the computational efficiency of SBR. Indeed, the *exact* reception of a ray at the Rx is highly unlikely owing to the discrete angular spacing of the launched rays of SBR. PCMs slightly change the positions of the intermediate path interaction points on the surfaces so that the launched ray can *exactly* reach the Rx.

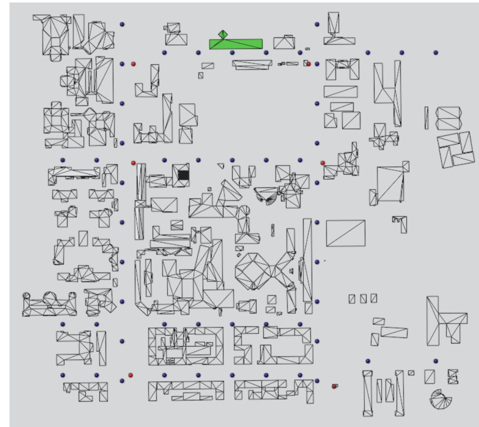
Differentiable ray tracing: A new approach to ray-based propagation simulation based on differentiable rendering (DR) [11] has been recently proposed in [12]. The developed ray tracing procedure relies on a differentiable rendering system to achieve differentiability of the ray tracing procedure with respect to environmental (e.g., radio-materials parameters) and system (e.g., antenna arrays positions) parameters. The challenges of rays obstruction in gradient-based optimization are tackled in [13], which proposes a new fully differentiable framework with everywhere-continuous loss functions by means of local smoothing.

III. Test scenario and ray-based solutions

In this section, we present the simulation set-up and a concise summary of different currently available ray-based propagation simulators, focusing on the solutions that we assess among them.



(a) 3D meshes detail



(b) Scenario top-view

Fig. 2: Selected test scenario. In green the main building of the High Frequency Campus in Milan, Italy. (a) shows a detail of the buildings' 3D meshes, while (b) presents a top view of the scenario along with the chosen Tx (red spheres) and Rx (blue spheres) positions.

TABLE I: Simulation parameters

Parameter	Value
Antenna Type	Isotropic
Carrier frequency	28 GHz
Buildings and ground material	Concrete
Material relative permittivity	5.31
Material conductivity	0.4838 S/m
Tx height	7 m
N_{Tx}	6
Rx height	1.5 m
N_{Rx}	51

A. Evaluation scenario

We consider an urban evaluation enviroment composed by buildings' meshes retrieved from the OpenStreetMap (OSM) online service for the High Frequency Campus urban geographical area in Milan, Italy. A section of the scenario is shown in Fig. 2a, presenting a view of the buildings' representations. For ray-based propagation simulation, we select the set of Tx and Rx depicted in Fig. 2b, where $N_{Tx} = 6$ transmitters (red circles) have been deployed at nodal positions in urban environment at a height of 7 m, while $N_{Rx} = 51$ receivers (blue circles) are distributed along the roads at a height of 1.5 m. Both Tx and Rx are equipped with isotropic antennas. We perform two simulation types: (i) *single link* (SL) simulations, where each Tx/Rx pair is separately considered to accurately measure the simulation performance for the single link case, and (ii) *multiple link* (ML) simulations, where simulations are performed for each Tx towards all the available Rx, allowing the simulation software to possibly exploit acceleration and parallelization techniques among different receivers.

Simulations are performed exhaustively for all the Tx/Rx communication pairs. When required, the simulation boundary has been set with a 50 m margin with respect to the bounding box comprising all buildings. The simulation boundary is assumed to be absorbing, so that only rays interacting with the buildings and ground within the scenario are considered during simulation. The ITU recommendation [14] has been considered as reference to determine the EM properties of the radio material, which are reported in Table I along with the general simulation parameters.

B. Compared ray-based simulation software

Several commercial and open solutions embed ray-based engines for EM propagation modelling. Among the most widespread commercial software are Remcom Wireless InSite, the MathWorks RF Antenna Toolbox, Siradel Volcano, Altair Feko, iBwawe Design, and EDX SignalPro. NVIDIA Sionna RT and the Ns-3 mmWave Module provide instead open source ray tracing solutions. Based on the availability of proprietary software and considering their use in academic and industrial contexts, we selected for evaluation the set of simulation solutions discussed in the following.

Remcom Wireless InSite (v3.3.3) [15] is a commercial solution that provides efficient and accurate modeling of the communication channel features in complex propagation EM environments. It offers two ray-based simulation methods: (i) full 3D, supporting simulations in the 0.1-20 GHz frequency range by means of the SBR and Eigen Ray algorithms, and (ii) X3D, which supports simulations in the 0.1-100 GHz frequency range, integrates SBR with a path correction method, and supports diffuse scattering simulation.

NVIDIA Sionna RT (v0.16.2) [12] is a recently proposed differentiable ray-based propagation engine that is part of the NVIDIA Sionna simulation library [16]. NVIDIA Sionna RT allows for accurate ray tracing simulations taking into account reflection, diffraction and diffuse scattering interactions. It relies on differentiable rendering system, leading to the desirable features discussed in Sec. II. NVIDIA Sionna RT provides two ray-based methods: (i) an exhaustive method, which tests all possible combinations of 3D primitives and paths, and (ii) a

Fibonacci method, which uses the SBR approach to efficiently compute the propagation paths.

The **MathWorks Ray tracing model (vR2023a U1)** [17] is part of the MathWorks Antenna Toolbox and offers two ray-based simulation methods: (i) SBR with exact path correction, supporting up to 10 reflections and 2 edge diffractions and providing an approximate number of propagation rays, and (ii) a ray tracing model based on the image method, supporting max. 2 path reflections and providing an exact number of rays featured by line-of-sight or reflection with exact geometry. Both methods enable simulation in the 0.1-100 GHz frequency range and support 3D indoor and outdoor environments.

IV. Simulation results

In this section, we evaluate the computational time efficiency of the selected ray-based simulation software on the considered scenario. Simulations are performed on a Windows 10 workstation equipped with Intel(R) Core(TM) i7-9700K CPU@3.60 GHZ, 8 cores, 16 GB RAM, and NVIDIA GeForce GTX 1070 Ti GPU with 8GB of dedicated memory.

The computational performance of ray-based simulation methods can depend on algorithmic approach, ray pruning, hardware acceleration, and 3D meshes complexity. To evaluate the efficiency, we measure the wall-clock time, which is the actual time elapsed during propagation simulation on a chosen system architecture. As discussed in Sec. III-A for SL simulations—considering a set of N_{Tx} transmitters and N_{Rx} receivers,—we measure the wall-clock time T_{ij} during the ray computation operation for each Tx/Rx pair (i, j) with $i = 1, \ldots, N_{Tx}$ and $j = 1, \ldots, N_{Rx}$. We evaluate the sample mean μ and variance σ^2 of the computational performance per Tx/Rx pair. For ML simulation, we measure the wall-clock time T_i during the rays simulation operation for each Tx, with $i = 1, \ldots, N_{Tx}$, towards all the available N_{Rx} Rxs, and we compute the sample mean $\mu\prime$ and variance $\sigma\prime^2$ of the performance over the simulation of multiple links.

We consider the X3D model for Remcom Wireless InSite (WI X3D), the SBR with exact path correction model for the MathWorks ray tracing model, and the *Fibonacci* model for NVIDIA Sionna RT. MW SBR has been performed within its Graphical User Interface (GUI), adding a small overhead in terms of computational time.

It is worth stating that the different implementations of ray-based propagation simulation methods present a heterogeneous set of input parameters. The maximum number of diffractions is set to 1 for WI X3D and MW SBR, while in Sionna diffraction is enabled by setting the corresponding boolean parameter. Moreover, the ray launching algorithms are significantly impacted by the initial number of rays sampled at the Tx. Both WI X3D and MW SBR offer the flexibility to specify the angular separation in degrees among the launched rays. We set this parameter to 0.5 deg. The Sionna *Fibonacci* method gives the possibility to specify the count of initially sampled rays. We use approximate equivalent number of rays of 1.6e5. This creates an equivalent starting condition for the heterogenous solutions, corresponding to 0.5 deg angular separation among rays.

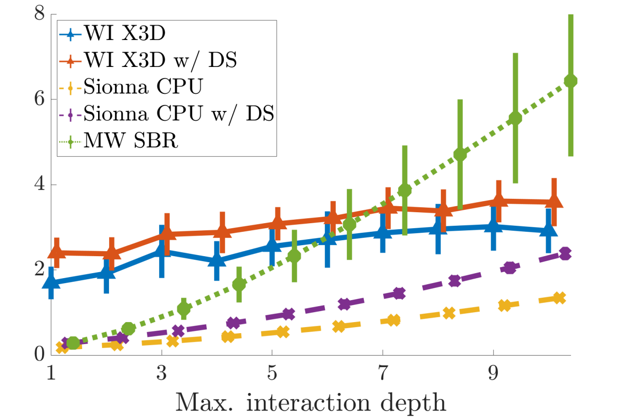


Fig. 3: Single link (SL) simulations with 1σ errobar w.r.t. the max. interaction depth with and w/o diffuse scattering (DS).

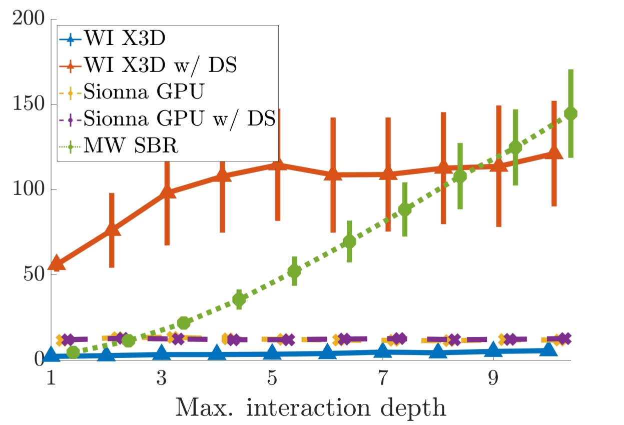


Fig. 4: Multiple link (ML) simulations with 1σ errobar w.r.t. the max. interaction depth with and w/o diffuse scattering (DS).

CPUs and GPUs are hardware resources providing different advantages. The former are optimized to handle serial operations and general purpose computing. The latter have proven to be particularly useful in tackling tasks requiring parallel operations and high throughput. In WI X3D and Sionna Fibonacci, we enable diffuse scattering (DS) simulation, showing the performance difference owing to the introduction of this interaction type during simulation. We remark that this preliminary investigation showcases the performance on a common computer architecture equipped with medium-level CPU processing and GPU acceleration capabilities.

Figure 3 provides the results for the SL simulations[1], measuring the wall-clock time of the ray launching operation per Tx/Rx pair. Considering maximum interaction depth at 1, 5, and 10:

- **WI X3D:** Without DS, this model experiences a time increase from 1.69 s to 2.9 s, while with DS it sees a

[1]The evaluated version of MW SBR does not model diffuse scattering.

time increase from 2.4 s to 3.59 s. In both conditions, the model shows an almost linear behavior.

- **Sionna Fibonacci:** Without DS, this model scales from 0.18 s to 1.34 s, and with DS ranges from 0.28 s to 2.38 s. In both cases, at maximum path depth of 5 the increase is of ≈ 2.5 times with respect to the initial case, and the behaviour is almost linear.
- **MW SBR:** This model varies from 0.28 s, to 2.32 s, up to 6.43 s—the DS case is not compared as currently not supported by MW SBR. The model shows an almost linear increase in computational time with a much higher variance with respect to the other examined solutions.

Fig. 4 reports the results of WI X3D, MW SBR, and NVIDIA Sionna, manually enabling GPU acceleration for the latter— GPU accelerations are enabled by default for WI X3D and MW SBR.We notice that the wall-clock time is evaluated for simulations performed separately at each Tx towards all the available Rx, with $N_{Tx} = 6$, and $N_{Rx} = 51$. The measurements are then averaged over the number of Tx. As the maximum interaction depth increases from 1 to 10:

- **WI X3D:** Without DS, this model experiences a time increase from 2.3 s to 5.5 s, while with DS the model shows a time increase from 56 s to 121 s. When comparing depth 1 and depth 5, the increase in computational time is ≈ 1.5 and ≈ 2 times, for the case without and with DS, respectively.
- **Sionna Fibonacci:** Without DS, this model scales from 11.58 s to 11.74 s, and with DS ranges from 11.92 s to 12.62 s. In both conditions, Sionna is slightly impacted by the variation of the maximum interaction depth and the introduction of DS interactions.
- **MW SBR:**This model varies from 4.5 s to 145 s, at depth of 5 the measured time is 50 s. The model shows a trend similar to that in the previous simulation.

V. Conclusion

This paper evaluated heterogeneous solutions for up-to-date wireless channel modelling within electromagnetic (EM) Digital Twins (DTs). We have examined state-of-the-art ray-based simulation algorithms and introduced a framework to compare heterogeneous propagation simulation software. The proposed framework is based on a standardized simulation scenario, and allows to efficiently asses the capabilities of various ray launching solutions in a DT pipeline. Numerical results in terms of wall-clock time were obtained on a selection of commercial and open simulation tools over different settings. Two simulation types have been conducted: *single link* (SL) simulations, where each Tx/Rx pair is separately considered and *multiple link* (ML) simulations, where simulations are performed for each Tx towards all the available Rx. In SL simulation, the wall-clock time is almost linear in the number of max. interactions per ray for all the tested tools with different slopes, and the addition of diffuse scattering does not significantly impact the computational time. The ML simulation shows that, when DS is not considered, WI X3D presents remarkable performance in performing ray tracing towards multiple Rx, while, when also DS interactions are enabled, Sionna shows higher computational efficiency. The presented results aim to provide a comparison of selected ray-based simulation software within an urban test scenario over multiple Tx/Rx communication pairs. As a future work, we plan to extend the proposed framework to take into account a wider set of environmental and simulation parameters.

Acknowledgment

The work of Francesco Linsalata, Maurizio Magarini, and Umberto Spagnolini was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART", Structural Project 6GWINET).

References

[1] A. Alkhateeb, S. Jiang, and G. Charan, "Real-time digital twins: Vision and research directions for 6g and beyond," *IEEE Communications Magazine*, vol. 61, no. 11, pp. 128–134, 2023.

[2] S. Salous, K. Haneda, and V. Degli-Esposti, "5g to 6g: A paradigm shift in radio channel modeling," *Radio Science*, vol. 57, no. 7, p. e2022RS007447, 2022, e2022RS007447 2022RS007447. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022RS007447

[3] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications," *IEEE Access*, vol. 3, pp. 1089–1100, 2015.

[4] C. Ding and I. W.-H. Ho, "Digital-twin-enabled city-model-aware deep learning for dynamic channel estimation in urban vehicular environments," *IEEE Transactions on Green Communications and Networking*, vol. 6, no. 3, pp. 1604–1612, 2022.

[5] M. Lecci, P. Testolina, M. Polese, M. Giordani, and M. Zorzi, "Accuracy versus complexity for mmwave ray-tracing: A full stack perspective," *IEEE Transactions on Wireless Communications*, vol. 20, no. 12, pp. 7826–7841, 2021.

[6] D. Bilibashi, E. M. Vitucci, and V. Degli-Esposti, "On dynamic ray tracing and anticipative channel prediction for dynamic environments," *IEEE Transactions on Antennas and Propagation*, vol. 71, no. 6, pp. 5335–5348, 2023.

[7] Z. Yun and M. F. Iskander, "Simplifying building structures for efficient radio propagation modeling," in *2023 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting (USNC-URSI)*, 2023, pp. 449–450.

[8] J. S. Lu, E. M. Vitucci, V. Degli-Esposti, F. Fuschini, M. Barbiroli, J. A. Blaha, and H. L. Bertoni, "A discrete environment-driven gpu-based ray launching algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 67, no. 2, pp. 1180–1192, 2019.

[9] M. F. Iskander and Z. Yun, "Propagation prediction models for wireless communication systems," *IEEE Transactions on microwave theory and techniques*, vol. 50, no. 3, pp. 662–673, 2002.

[10] V. Degli-Esposti, "A diffuse scattering model for urban propagation prediction," *IEEE Transactions on Antennas and Propagation*, vol. 49, no. 7, pp. 1111–1113, 2001.

[11] H. Kato, D. Beker, M. Morariu, T. Ando, T. Matsuoka, W. Kehl, and A. Gaidon, "Differentiable rendering: A survey," *arXiv preprint arXiv:2006.12057*, 2020.

[12] J. Hoydis, F. A. Aoudia, S. Cammerer, M. Nimier-David, N. Binder, G. Marcus, and A. Keller, "Sionna rt: Differentiable ray tracing for radio propagation modeling," *arXiv preprint arXiv:2303.11103*, 2023.

[13] J. Eertmans, L. Jacques, and C. Oestges, "Fully differentiable ray tracing via discontinuity smoothing for radio network optimization," *arXiv preprint arXiv:2401.11882*, 2024.

[14] I. T. U. R. Sector, "P.2040 : Effects of building materials and structures on radiowave propagation above about 100 MHz." [Online]. Available: https://www.itu.int/rec/R-REC-P.2040/en

[15] Remcom wireless insite. [Online]. Available: https://www.remcom.com/wireless-insite-em-propagation-software

[16] J. Hoydis, S. Cammerer, F. Ait Aoudia, A. Vem, N. Binder, G. Marcus, and A. Keller, "Sionna: An open-source library for next-generation physical layer research," *arXiv preprint*, Mar. 2022.

[17] Matlab ray tracing propagation model. [Online]. Available: https://uk.mathworks.com/help/antenna/ref/rfprop.raytracing.html