ClickDT: Building Scalable and High-Resolution Wireless Digital Twins with a Few Clicks

Yiming Li, Scarlett Francini, Zhihui Gao, Tingjun Chen
Department of Electrical and Computer Engineering, Duke University
Email: {yl826, scarlett.francini, zhihui.gao, tingjun.chen}@duke.edu

***Abstract*—Ray tracing (RT) tools serve as a crucial component for building wireless digital twins (DTs) by simulating the interactions between radio frequency (RF) signal propagation and the physical environment. However, existing open-source RT frameworks often require complex setup and configuration procedures, hindering their rapid adoption by and accessibility to a broader audience. We demonstrate ClickDT, a streamlined framework for the rapid construction of high-resolution wireless DTs with just a few mouse clicks. ClickDT is open-sourced and designed to facilitate the generation of realistic 3D scenes and the execution of RT simulations through a web-based user interface.**

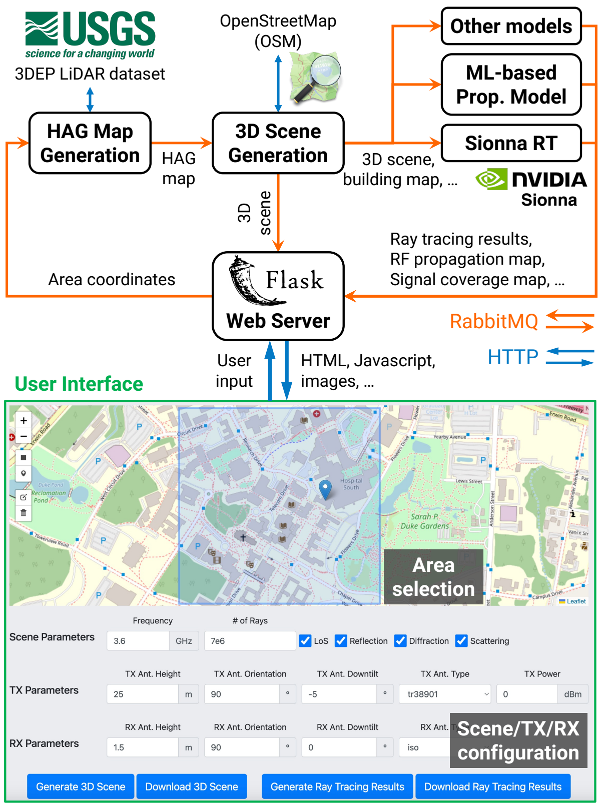
I. Introduction

Ray tracing (RT) tools [1], [2], [3] serve as a critical component for constructing wireless digital twins (DTs) by simulating the interactions between radio frequency (RF) signal propagation and the physical environment [4], [5], [6]. Such wireless DTs can be used for RF signal mapping and coverage prediction, which are crucial for cellular network planning and deployment [7], [8]. While RT can provide more accurate RF signal mapping leveraging real-world information, it comes at the cost of increased complexity and computational resources, and the need for real-world geographic information. Many advanced RT tools are also proprietary and thus not easily accessible by the broader research community.

Several open-source RT frameworks have emerged recently, such as NVIDIA's Sionna [3] and Opal [9]. However, they focus primarily on the RT engine itself and often lack adequate documentation or support for creating accurate spatial representations (e.g., 3D scenes) of the physical environments. Moreover, these tools typically require complex environment setup or configuration procedures, limiting their accessibility to only experienced researchers. To address these challenges, we present ClickDT, a streamlined framework for the rapid construction of high-resolution, realistic wireless DTs with just a few mouse clicks. ClickDT is open-sourced [10] and can be used by the broader community to facilitate the generation of realistic 3D scenes and execution of RT simulations through a user-friendly web-based user interface (UI).

II. Design and Implementation of ClickDT

Fig. 1 shows the overview of our developed ClickDT framework, which aims to facilitate the construction of precise wireless DTs. ClickDT consists of five key modules, whose inter-module communication is handled by HTTP and RabbitMQ, an open-source message broker software.



	Frequency	# of Rays			
Scene Parameters	3.6 GHz	7e6	✓ LoS	✓ Reflection	✓ Diffraction ✓ Scattering
	TX Ant. Height	TX Ant. Orientation	TX Ant. Downtilt	TX Ant. Type	TX Power
TX Parameters	25 m	90 °	-5 °	tr38901	0 dBm
	RX Ant. Height	RX Ant. Orientation	RX Ant. Downtilt	RX Ant.	
RX Parameters	1.5 m	90 °	0 °	iso	

Fig. 1. Overview of the ClickDT framework.

Web-based user interface (UI). The web UI consists of two sections as shown in Fig. 1. The top section is a worldwide map managed by Leaflet [11], with which the users can interact to select a rectangular geographical area of interest and place the transmitter (TX) at a specific location. The bottom section features multiple user input fields designed for entering the scene and RT parameters.

Flask web server. We use Flask as the web service to serve static files, including HTML, JavaScript, scene files, and RT results. It also functions as an API server and a user-facing gateway in ClickDT: it receives user input from the browser (e.g., coordinates for the selected area and RT parameters), and forwards the request to the pipeline via RabbitMQ.

HAG map generation. The United States Geological Survey (USGS) 3D Elevation Program (3DEP) provides public access to nationwide 3D LiDAR point cloud data in the Entwine Point Tiles (EPT) format. For the geographical area selected by the

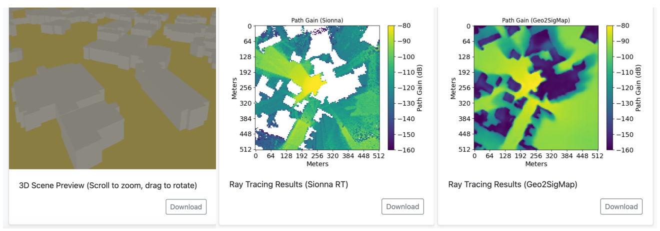

3D Scene Preview (Scroll to zoom, drag to rotate)
Download
Path Gain (Sionna)
Meters
Path Gain (dB)
Ray Tracing Results (Sionna RT)
Download
Path Gain (Geo2SigMap)
Meters
Path Gain (dB)
Ray Tracing Results (Geo2SigMap)
Download

Fig. 2. (*Left*) Constructed 3D scene for the selected area. (*Middle*) RT results generated by Sionna. (*Right*) Signal coverage map generated by ML-based method such as Geo2SigMap [10].

user, we first query the corresponding LiDAR points that fall within this bounding box and use the Point Data Abstraction Library (PDAL) to generate the HAG map with the point classifications information. The HAG map is represented by a 1-channel image, where each pixel value represents the height above the surrounding ground.

3D scene generation. For the selected area, a 3D mesh and a building map need to be generated for RT and other RF signal mapping methods. First, we employ OSM to extract the geographic information of the selected area, including the class of the objects (e.g., building, forest) with their corresponding locations and shapes. The extracted information is stored using key-value pairs in the OSM XML format (`.osm`), where the key describes the context of the location (e.g., buildings or highways) and the value contains the detailed information about the key (e.g., building shape/height). For a building described by a 2D polygon in the OSM XML file, we obtain its triangulation using Triangle and query the HAG map associated with this area to obtain the building height. Then, we use the linear extrude method to generate the 3D shape of the building given the 2D polygon and height value. Finally, we save the 3D mesh in the Polygon File Format (`.ply`), which can be used as input to Sionna RT and other RF signal mapping methods (e.g., Geo2SigMap [10]).

RT and RF propagation modeling. This module includes various RT and signal mapping tools, such as Sionna RT [3], ML-based propagation models (e.g., Geo2SigMap [10]), and empirical/statistical models such as the 3GPP urban macro (UMa) [12] model. It uses information passed from the previous modules to predict the RF path gain and/or signal strength distribution throughout the selected area in the form of coverage maps. Each signal mapping tool has a customized wrapper function to interface with other modules.

III. Demonstrations

During the demonstration, attendees will be able to navigate through the web-based UI to generate, view, and download 3D scenes along with the corresponding RT results and coverage maps for any chosen geographical area. Note that no prior knowledge or coding skills, such as Open3D, LiDAR, or Sionna RT, are required to use ClickDT. We will host ClickDT on our own server and request a monitor for this demo.

3D scene generation with accurate building height. Users draw an arbitrary rectangle on a nationwide map to select the area of interest. Clicking "Generate 3D Scene" creates and visualizes the corresponding 3D scene, which can be downloaded and used in Sionna RT. The buildings inside the generated scene are associated with the true building heights obtained from the USGS LiDAR dataset.

Sionna RT & ML-based propagation model. After selecting the area of interest, users can mark the TX location on the map and specify the RT parameters such as frequency and TX/RX antenna orientation. Then, users can click on the "Generate RT Results" button to generate two path gain maps based on Sionna RT and the ML-based propagation model leveraging Geo2SigMap [10]. After successful generation, users can view the path gain maps and download them in the `.npy` format.

Detailed RT analysis. Using the configured RT parameters, users can also specify an RX location, which will trigger ClickDT to run Sionna RT in a point-to-point setting and return the properties of each ray between TX and RX. The ray properties will be saved and can be downloaded as a CSV file, which lists each ray with its channel coefficients, delay, angle-of-departure (AoD), angle-of-arrival (AoA), and the type of ray (e.g., LoS, reflected, diffracted, scattered).

Acknowledgments

This work was supported in part by NSF grants CNS-2112562, CNS-2128638, CNS-2211944, AST-2232458, and ECCS-2434131, and the Center for Ubiquitous Connectivity (CUbiC), sponsored by Semiconductor Research Corporation (SRC) and Defense Advanced Research Projects Agency (DARPA) under the JUMP 2.0 program.

References

[1] Altair, "WinProp," https://web.altair.com/winprop-telecom, 2023.
[2] Siradel, "Siradel Volcano 5G Model," https://www.siradel.com/solutions/software/volcano-5g/, 2021.
[3] 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.
[4] X. Zhang, X. Shu, B. Zhang, J. Ren, L. Zhou, and X. Chen, "Cellular network radio propagation modeling with deep convolutional neural networks," in *Proc. ACM KDD'20*, 2020.
[5] S. Bakirtzis, K. Qiu, J. Zhang, and I. Wassell, "DeepRay: Deep learning meets ray-tracing," in *Proc. IEEE EuCAP'22*, 2022.
[6] J. Thrane, D. Zibar, and H. L. Christiansen, "Model-aided deep learning method for path loss prediction in mobile communication systems at 2.6 GHz," *IEEE Access*, vol. 8, pp. 7925–7936, 2020.
[7] 3GPP, "5G; NR; Base station (BS) radio transmission and reception," Technical Specification (TS) 38.211, 07 2020, version 16.4.0. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/138100_138199/138104/16.04.00_60/ts_138104v160400p.pdf
[8] A. Sufyan, K. B. Khan, O. A. Khashan, T. Mir, and U. Mir, "From 5G to beyond 5G: A comprehensive survey of wireless network evolution, challenges, and promising technologies," *Electronics*, vol. 12, no. 10, p. 2200, 2023.
[9] E. Egea-Lopez, J. M. Molina-Garcia-Pardo, M. Lienard, and P. Degauque, "Opal: An open source ray-tracing propagation simulator for electromagnetic characterization," *Plos one*, vol. 16, no. 11, p. e0260060, 2021.
[10] Y. Li, Z. Li, Z. Gao, and T. Chen, "Geo2SigMap: High-fidelity RF signal mapping using geographic databases," in *Proc. IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN'24)*, 2024.
[11] J. Cheng, B. Schloerke, B. Karambelkar, and Y. Xie, "Leaflet: Create Interactive Web Maps with the JavaScript 'Leaflet' Library," https://rstudio.github.io/leaflet/, 2024.
[12] 3GPP, "3GPP TR38.901," https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3173, 2023.