The Development of a VR Wireless Signal Propagation Simulator in Unreal Engine: A Device and Performance Testing

Louis Nisiotis, *University of Central Lancashire, Cyprus*

Anna Anikina, *University of Central Lancashire, Cyprus*

Marios Raspopoulos, *University of Central Lancashire, Cyprus*

Abstract—The wireless communication industry and research community focused on developing accurate and efficient models for signal propagation extensively over the past years. High-fidelity simulation software often comes with high costs and relies heavily on offline computations, limiting flexibility and scalability. This paper introduces a low-cost prototype for simulating signal propagation, leveraging advancements in Gaming technologies, Virtual Reality, and Digital Twins. The system is capable of conducting ray tracing simulations of wireless signal propagation through various configurable parameters and fidelity levels, generating signal strength heat-map and real-time ray visualization in immersive VR and 3D desktop modes. Thorough device and performance testing validated the prototype's development and assessed implementation feasibility. The findings establish its operational capabilities and performance metrics, providing insights and guiding further development and refinements, ensuring its reliability and effectiveness in simulating complex wireless propagation.

Accurate radio propagation modeling has been a significant research focus explored through empirical, semi-empirical, and deterministic path-loss calculation models. Empirical models are efficient and simple but often lack accuracy and broad applicability [1]. On the contrary, deterministic models use electromagnetic theory and methods like Ray Tracing, offering higher accuracy but requiring substantial computational resources [2]. Recent advancements in wireless communication and GPU technology shifted the focus to these deterministic models using Ray Tracing approaches [3]. Considering these rigid requirements of those models, there is a need for more accessible, and cost-effective solutions for simulating and visualizing signal propagation. Recent advancements in computer graphics, GPU and CPU hardware, and eXtended Reality (XR), offer new opportunities to address complex requirements and develop innovative and accessible tools for signal propagation simulation and visualization that can overcome existing limitations.

This paper presents a prototype in development that aims to create a signal propagation simulator using a sophisticated game development engine through the concept of Digital Twins, visualised through VR and 3D desktop modes. It extends our paper presented at the 48th IEEE International Conference on Computers, Software, and Applications (COMPSAC 2024) [4], where we introduced a preliminary version of our prototype. While the initial paper laid the groundwork of using VR, Digital Twins, and Gaming technology for simulating wireless signal propagations, this extended paper presents the results of a comprehensive performance and device testing, assessing the feasibility of deploying such an immersive low-cost simulator using Unreal Engine. The aim is to ensure robust performance under various simulation parameters, by identifying the optimal configurations for a usable experience. The core research question is: *What configurations and technical settings are necessary to optimize the simulator's performance for best operational and user experience?*. Our objective is to systematically evaluate the prototype's performance across a range of simulation parameters, to establish standardized operational settings. This work sets the premises for future developments to establish the simulator as an effective tool for academic research and practical applications in wireless network design.

Digital Object Identifier 10.1109/XXX.0000.0000000

BACKGROUND AND CONTEXT

Ray Tracing (RT) is a well-known technique in computer graphics to simulate light interactions in virtual environments to create 2D images from 3D scenes. In radio signal propagation, RT assumes propagation can be approximated as rays spherically launched from the transmitter. It applies principles of Geometrical Optics to identify ray paths and intersections with objects between transmitter and receiver locations, using high-frequency electromagnetic theory to calculate the amplitude, phase, delay, and polarization of each ray. This method accurately estimates signal spatiotemporal parameters, which is key considering the advancements in wireless communications, such as millimeter-wave spectrum usage, phased-array antennas, massive-MIMO systems, and Reconfigurable Intelligent Surfaces [5], [6], [7].

Historically, the RT modelling computational processing demands was a challenge, but this has been mitigated by parallel and distributed computing, and leveraging GPU technology. Recent works, such as the GPU-based kD-tree-accelerated beam-tracing method (GKBT) [8], and environment discretization techniques [9], have significantly increased computational efficiency. A key development is the 5G simulation platform by Ericsson and NVIDIA, utilizing NVIDIA's Omniverse cloud platform to enable real-time calculations of spatiotemporal signal characteristics in dynamic environments, demonstrating the concept of Digital Twins [10].

Digital Twins in Wireless Communications

Digital Twins is a key emerging technology driving digital transformation, gaining significant academic and industrial attention. Digital Twins are typically defined as virtual representations of objects or systems, updated with real-time data, and utilizing simulations and reasoning to aid decision-making [11]. This synchronization allows near-real-time insights into the physical system's operation, aligning the real and virtual worlds through data, analytics, and visualization tools [12].

The growing need for Digital Twins is driven by advancements in hardware, software and wireless communications [12], and the industrial demands for real-time monitoring, operational flexibility, improved management, and personalized services among other needs [13]. In the telecom industry, Digital Twins are particularly valuable for real-time analysis, enabling visualization of signal propagation within physical wireless networks. Such an approach enables to deploy virtual models mirroring real-world network entities and processes, including environmental and user interactions, regularly updated to maintain fidelity and accuracy. Recent studies focused on real-time simulations using Digital Twins, integrating 3D mapping, sensing, RT, and AI to create and update accurate models of environments and signal propagation [14]. Various use cases demonstrate the potential of networked Digital Twins, particularly for network performance evaluation and radio resource management in industrial settings [13]. However, while much of the literature addresses methodologies and data integration challenges, there is a need for practical implementation examples within the radio propagation domain to showcase tangible outcomes and benefits.

Gaming Technologies and eXtended Reality

The advancement of 3D gaming and GPU hardware technology presents significant opportunities for developing models of radio networks [10]. Gaming technology is capable of managing complex scene geometry, high-fidelity graphics, and dynamic AI behaviors, which can be repurposed to accurately simulate radio networks within a Digital Twin framework [15]. eXtended Reality (XR) encompassing Augmented, Virtual, and Mixed Reality technologies, has gained considerable research and industrial attention, including from the gaming industry. XR offers immersive experiences, enhancing user presence and interaction, and has the potential to revolutionize our perception of real and digital worlds. In Digital Twins, XR is frequently used for visualizing and interacting with real-time data, with positive results across a plethora of domains [12].

Exploring the integration of gaming technologies, Digital Twins, and XR to create a low-cost, flexible tool for real-time deterministic modelling offers unique opportunities. Our work aims at the development of a cost-effective simulator, providing the desired accuracy and resolution in real-time simulations of radio environments using a sophisticated and accessible game engine development tool, the Unreal Engine. Real-time in the context of our simulations refers to the system's ability to compute simulation data and render the ray paths' for visualisation purposes without perceptible delays to the user. Previous efforts in this domain include prototypes such as 'DeepWiSim,' a wireless simulator for deep learning automation [16], and 'See-the-Radio-Waves', a VR demo for visualizing and optimizing wireless systems [17]. Other projects involve visualizing Electromagnetic Simulations [18], and large-scale wireless emulators applying radio wave propagation models in 3D spaces [19]. Our work uniquely integrates VR with Digital Twin technologies, advancing visualization and interaction capabilities that are yet to be explored in existing research. It relies on accurate mathematical models and high-performance tethered VR to offer access to immersive experiences, aiming to incorporate user interactions directly within the simulations and to integrate real-time data from sensors, setting the groundwork for future real-time digital twin applications.

Signal Propagation Simulation Prototype

The key objectives of this prototype development are: i) Assess the feasibility and effectiveness of Unreal Engine in simulating wireless signal propagation and accurately representing key propagation mechanisms; ii) Determine the resulting simulation resolution and accuracy; iii) Integrate the Digital Twin concept for visualizing an immersive, interactive radio environment in VR; iv) Understand the technical requirements and user experience challenges associated with developing such prototypes.

The use of Unreal Engine 5 is justified by its flexible development tools, sophisticated geometry and rendering system. Two main operational components are deployed: i) a radio propagation modelling tool, operating in a 3D environment, with real-time ray generation; ii) an immersive visualisation of the resulting signal strength through a heatmap, which can be navigated and explored in Desktop and VR modes.

The prototype is based on a *Transmitter-Obstacle-Receiver* architecture meaning that its model is built around virtual rays travelling from the *Transmitter* to the *Receiver*, interacting with *Obstacles* along the way.

The simulation calculates the complex electric field at every interaction point and *Receiver* cell, considering polarization effects. The geometrical and morphological properties of obstacles are defined through the property interface within the game engine.

The *Transmitter* simulates a real-world source (e.g. Wi-Fi access point) and emits primary rays in a user-defined spherical area. Each ray is assigned an initial electric field vector $\vec{E^i}_{xyz} = \begin{bmatrix} E^i_x & E^i_y & E^i_z \end{bmatrix}$, which is composed of the electric field's x, y, z components and depends on the field E_0 emitted from the antenna, the antenna gain $P(\theta, \phi)$ and its polarization $\hat{p}$ ($\hat{\theta}$), which vary according to the azimuth (θ) and elevation (ϕ) angle of each emitted ray i^{th}.

$$\vec{E^i}^{\perp,\parallel}_{xyz}(r_i) = E_0 \frac{e^{-jkr_i}}{\sqrt{4\pi} * r_i} P(\theta, \phi)\hat{p} \quad (1)$$

When a ray hits an *Obstacle*, it generates reflected and refracted rays according to Snell's law. The reflection $R^{\perp,\parallel}$ transmission $T^{\perp,\parallel}$ coefficients, influenced by the obstacle's electrical properties are used to calculate the new electric fields.

The *Receiver*, implemented as a 3D grid cell, captures rays overlapping with it and calculates their received electric field at a total travelled distance R_i as:

$$\vec{E}^{\perp,\parallel}(R_i) = E_0 \frac{e^{-jkR_i}}{\sqrt{4\pi} * R_i} P(\theta, \phi)\hat{p} \prod_{m=1}^{n_R} R_m^{\perp,\parallel} \prod_{m=1}^{n_T} T_m^{\perp,\parallel} \quad (2)$$

where n_R and n_T are the total number of reflections and refractions encountered by the ray. The total Electric field at the receiver is the norm of the vectorial sum of all the captured rays (N) calculated using:

$$E_{total}^{\perp,\parallel} = \sqrt{\sum_{i=1}^{N} E_x^2 + \sum_{i=1}^{N} E_y^2 + \sum_{i=1}^{N} E_z^2} \quad (3)$$

The received signal power is finally calculated using:

$$P_{dBm} = 10 \log_{10} \left| \frac{E_{total}^2 \lambda^2 G_R}{8 Z_0 \pi} \right| \quad (4)$$

where Z_0 is the impedance of free space (377Ω) and G_R is the receiver gain.

Digital Twin Environment

The prototype we used in this experiment replicates a 100 m^2 computer lab at UCLan Cyprus. To accurately model this physical space, the Turtlebot2 robot equipped with sensors was used to create a 2D map, which was then used in Unreal Engine to reconstruct the physical space in 3D (Figure 1a). Further details were achieved using Unreal Engine's built-in level design tools. Static mesh actors were used to represent the physical structures of the room, converted into *Obstacles*, including data on material properties such as thickness, relative permittivity, and conductivity.

Heatmap A key prototype feature is the dynamic generation of an intuitive heatmap to visually represent signal strength across the digital twin's space, to aid the understanding of signal propagation under various transmitter and base signal configurations (Figure 1b). The digital space is divided into discrete *Receiver Cells* using an influence map method registering incoming rays. Signal power distribution and intensity received by each cell are translated into pixel colors, with color depth indicating the signal strength levels. The user defines the heatmap coordinates and resolution prior to simulation for detailed visual analysis. To convert signal power in dBm (equation 4) into pixel colors, the system: i) obtains the minimum and maximum expected signal power, ii) calculates the percentage of the current signal power relative to these expected values, iii) converts this percentage to a value between 0 and 255, and iv) applies this value to the red color channel for visualization.

System Testing

The significance of software testing is well-established and methodologies such as functional, performance, and usability testing, among others are typically used [20]. For the prototype to be complete, both accuracy and system/functional testing must be performed to ensure correctness of calculations, system functionalities, and the ability to provide effective user experience. At the current development stage, our focus is on technical testing rather than simulation accuracy, as we are still calibrating the signal propagation modelling. Currently, we focus on implementing the basic visualization functionalities within the VR environment to refine the core visualization components before integrating more complex 3D interactions. Once these are fully developed and optimized, we will conduct thorough testing to validate the simulator's accuracy using commercial simulators and in-situ measurements.

Testing Strategy

To evaluate the current version of the prototype prior to further usability and simulator accuracy evaluations, we devised a comprehensive testing strategy exploring important performance and device-related aspects including: i) *Performance Testing* to evaluate key metrics such as Frame Per Second (FPS), successful execution of simulations, system stability, and heatmap generation completion, with emphasis on performance criteria; ii) *Device Testing* to verify functionality across different platforms, including desktop, tethered VR, and pixel streaming for VR and web browsers, emphasising on modality confirmation, performance, and quality of the visual experience. This systematic performance and device compatibility assessment will help us identify optimal configurations, potential issues, and optimize the system to improve user experience. The requirements guiding our testing strategy are:

- The simulator must successfully complete the propagation processes.
- The simulator should generate a complete heatmap.
- Users need to visualize real-time ray generation and their interactions with the environment.
- Users should navigate the environment smoothly.

Our testing strategy focused on i) Desktop mode, ii) Tethered VR, and iii) Pixel Streaming in VR and web-browser. For Desktop simulations, the tests were conducted in the Unreal Editor using the 'Play In New Window' mode, enabling to simulate the experience directly in the engine while having access to the scene attributes and system controls. For each desktop test, we were terminating and relaunching Unreal Engine to avoid potential memory and caching issues that could result from multiple and repeated simulations.

In Tethered VR mode, a wired connection between the VR headset and the computer is required for data and power to be transmitted for the VR headset to display high-quality visuals rendered by the computer hardware. We want to test whether this mode would enable to harness the processing power of the PC to run the intensive simulations and render the environment efficiently. We have conducted the tests using the Meta Quest 2 headset in tethered mode through Quest Link using USB-C 3.2 cable connection. For each VR testing condition, the Unreal Engine was terminated and relaunched, the VR headset was disconnected, restarted, and reconnected to the PC.

Pixel Streaming, is a feature of Unreal Engine, where the graphical output of the engine is rendered on a server and streamed to a web browser, making it accessible to a wide range of devices including VR headsets. A PC is acting as a server for generating the 3D scene, continuously encoding it into a media stream, and then broadcasts it through web services to standard web browsers, hence removing the computational load on the user's device. The user can also send input and events to control the experience back to Unreal Engine in real-time.

For this study, we explored the efficiency of pixel streaming in desktop web-browsers and in VR. Pixel streaming testing was conducted on local network conditions, through Chrome web browser for PC, and through the Meta Quest's 2 internal VR browser. For each pixel streaming testing condition, both the engine and the server instance were relaunched.

Testing Parameters The tests have been carried out on a mid to high-end Desktop PC equipped with an Intel Core i7-10700K CPU, 2.9 GHz, NVidia 4060 RTX GPU, and 32GB of DDR4 RAM. We have used Unreal Engine 5.2.1 version. To evaluate the prototype's performance, we focused on key configurable parameters for optimizing the visualization of wireless signal propagation: the azimuth increment (θ), the elevation angle increment (ϕ) within the antenna's coordinate system, and the radius of each cubic cell in the heatmap. Additionally, we examined ray visibility during simulations, and its impact on performance.

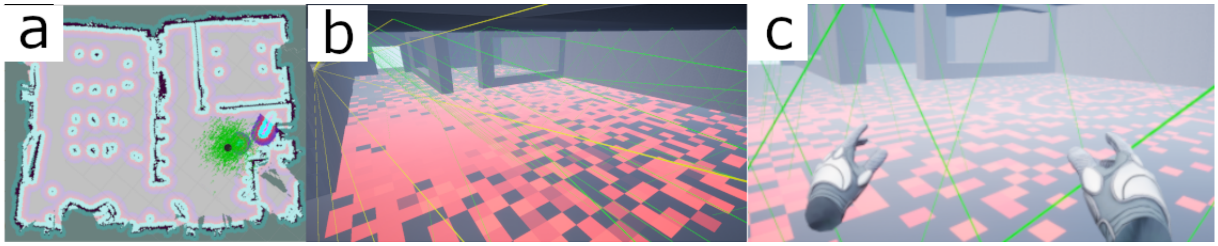

FIGURE 1. Preliminary Version of the Digital Twin development process, and simulation examples. (a) 2D layout generated through Turtlebot2; (b) Example ray tracing simulation and heatmap visualisation; (c) Example immersive experience in VR

TABLE 1. Performance and Device Test Results

Performance Testing							
Configuration No:	**1**	**2**	**3**	**4**	**5**	**6**	**7**
θ Increment (degrees)	5	5	5	5	5	1	1
ϕ (degrees)	5	5	5	5	5	1	1
Cell Radius (cm)	10	15	25	15	25	5	10
Ray Visibility	OFF	OFF	OFF	ON	ON	OFF	ON
Mdn FPS PC (during simulation)	>100	>100	>100	<60	<60	>100	>100
Mdn FPS VR (during simulation)	65-74	65-74	65-74	<60	<60	<60	<60
Simulation Completion	YES	YES	YES	YES	YES	NO	NO
Heatmap Generation	YES	YES	YES	YES	YES	NO	NO
Device Testing							
PC Simulation	PASS	PASS	PASS	PASS	PASS	FAIL	FAIL
Tethered VR	PASS	PASS	PASS	FAIL	FAIL	FAIL	FAIL
Pixel Streaming VR	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
Pixel Streaming Web-browser	PASS	PASS	PASS	PASS	PASS	PASS	PASS

Our controlled variables for the tests were the number of θ and ϕ positions, a maximum of 4 bounces per ray, a maximum ray length of 40 meters, and the visibility duration of rays set at 0.5 seconds. We have also set a maximum lost ray length at 20 meters to evaluate performance under conditions of partial signal loss. The results we considered in our analysis included Median (Mdn) and Frames Per Second (FPS) to assess simulation smoothness, System Stability to ensure operational reliability, Simulation Completion, and Heatmap Generation completion for spatial data visualization. The FPS rate threshold we aimed to achieve for PC simulation was set to 45 FPS, which is known to provide a usable experience in desktop gaming. The threshold for VR was set at 60 FPS to ensure a smooth and responsive experience for users, as maintaining consistent frame-rate equal and higher than 60 FPS is critical preventing user discomfort in VR settings.

Test Results Numerous simulations were performed using different parameters and across devices, with key results presented in Table 1. As anticipated, using the simulator in Desktop mode proved to be the most efficient and performant. A key finding is that the real-time ray visualisation substantially reduces frame rates below 60 FPS during the simulation calculations phase. In Desktop mode, despite the lower frame rates, the system remained usable and the experience was relatively smooth, maintaining between 45-55 FPS (Mdn). However, in Tethered VR, low frame-rate between 20-30 FPS (Mdn) significantly affected the user experience. Once the calculations were completed, frame-rates return to stable rate exceeding 60 FPS (Mdn) in both Desktop and Tethered VR, indicating that post-calculation stage provides good system performance.

In Tethered VR mode, configurations 1-3 demonstrated that the system can successfully complete all simulations while maintaining good performance, consistently staying above the 60 FPS threshold. Under these configurations and with real-time RT disabled, users are can visualize the generated heatmap and navigate in the environment smoothly, without noticeable delays and framedrops that are known to hinder the VR experience. Thus, these configurations provide an optimal setting for users to engage effectively with the simulator.

Pixel streaming through desktop web browser was found as a viable alternative for remote access to the simulator. Due to the nature of pixel streaming, some occasional delays were observed, but the overall functionality was preserved, allowing users to explore the digital space, visualize ray propagation, and generate heatmap effectively. However, the performance of pixel streaming for VR was problematic. While the VR device successfully rendered the stream in its built-in web browser with a consistent frame rate of 75-80 FPS, the user experience was significantly impaired by network latency issues that could potentially induce VR sickness, making it unsuitable for prolonged use in its current form. Future efforts should focus extensively on network configuration and optimization to mitigate these issues if pixel streaming is to be considered a feasible option for accessing the simulator remotely in VR.

DISCUSSION

The findings of our work demonstrate the potential and challenges of simulating wireless signal propagation in our prototype simulator. We have shown that a medium-high end PC can handle simulations of the current prototype effectively, and identified certain configurations where Tethered VR can deliver consistent and satisfactory performance. However, results on real-time ray visualization for VR highlights the current limitations of the prototype. Additionally, pixel streaming was identified as a functional alternative for remote access through web browsers. However, it faces significant challenges in VR due to latency issues that could affect user comfort, and should be explored in more detail.

While our VR system provides an immersive experience, the accuracy of signal propagation simulations is still not verified and may be limited by the computational model used. Furthermore, the simplifications necessary for real-time rendering can result in deviations from real-world scenarios. A key limitation of the prototype relates to the fact that the effectiveness of the VR experience is dependent on the underlying hardware. Moreover, the current prototype is tested using a controlled digital twin environment and may not scale seamlessly to more complex scenarios. Extensive field tests are required to validate its performance across different settings and conditions.

Current development focus on refining the complex propagation calculations, validating calculation results, fine-tuning the digital twin setting, and optimizing the algorithms and the environment to ensure a smooth user experience. Future work will focus on further system performance, user experience (i.e. VR sickness and usability evaluation), and simulation accuracy evaluations, to determine the extent to which the prototype effectively supports real-time simulations and visualizations in efficient and immersive ways. These findings will set the stage for future research directions and help to assess the practicality, benefits, and limitations of this approach, opening new research opportunities for simulation, analysis, and interaction.

CONCLUSIONS

Unlike traditional 2D and 3D simulators, VR offers an immersive experience that allows users to interact within 3D spaces, enhancing their understanding and decision-making in complex scenarios. VR's capability to simulate physical presence in virtual environments is beneficial for tasks requiring a spatial understanding of signal propagation and interference areas where traditional 2D and even 3D simulations fall short. We envision the development of a low cost tool that would enable network engineers to optimize antenna placement in diverse settings, visualising changes in signal strength and quality to adjust antenna positions and settings. Such tool can be used for research and development activities, and also in education, providing students with practical, hands-on experience for exploring of complex concepts like wave propagation and network planning among other topics.

REFERENCES

1. K. Haneda *et al.*, "5G 3GPP-like channel models for outdoor urban microcellular and macrocellular environments," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, pp. 1–7, 2016.
2. M. Born and E. Wolf, eds., *Principles of Optics - Electromagnetic Theory of Propagation, Interference and Diffraction of Light*. Pergamon, 6th edition ed., 1980. ISBN: 978-0-08-026482-0.
3. Y. Liu, D. Shi, A. Li, and P. Zeng, "Radio wave propagation prediction based on the NVIDIA Optix GPU ray tracing engine," in *2019 IEEE 6th Intern. Symposium on Electromagnetic Compatibility*, pp. 1–3, 2019.
4. L. Nisiotis, A. Anikina, and M. Raspopoulos, "Exploring gaming technologies, digital twins, and VR to visualise wireless propagation simulations," in *48th IEEE International Conference on Computers, Software, and Applications, COMPSAC*, 2024.

5. V. Degli-Esposti, F. Fuschini, E. M. Vitucci, M. Barbiroli, M. Zoli, L. Tian, X. Yin, D. A. Dupleich, R. Müller, C. Schneider, and R. S. Thomä, “Ray-tracing-based mm-wave beamforming assessment,” *IEEE Access*, vol. 2, pp. 1314–1325, 2014.
6. J. Huang, C.-X. Wang, Y. Sun, J. Huang, and F.-C. Zheng, “A novel ray tracing based 6G RIS wireless channel model and RIS deployment studies in indoor scenarios,” in *IEEE 33rd Intern. Symp. on Pers., Indoor and Mobile Radio Comms (PIMRC)*, pp. 884–889, 2022.
7. J. Huang, C.-X. Wang, Y. Sun, R. Feng, J. Huang, B. Guo, Z. Zhong, and T. J. Cui, “Reconfigurable intelligent surfaces: Channel characterization and modeling,” *Proc. of the IEEE*, vol. 110, no. 9, pp. 1290–1311, 2022.
8. J. Tan, Z. Su, and Y. Long, “A full 3D GPU-based beam-tracing method for complex indoor environments propagation modeling,” *IEEE Trans. Antennas and Propagation*, vol. 63, no. 6, pp. 2705–2718, 2015.
9. 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.
10. Ericsson, “Next-generation simulation technology to accelerate the 5G journey,” 2021. https://www.ericsson.com/en/blog/2021/4/5g-simulation-omniverse-platform, Accessed on 28/11/2024.
11. M. Armstrong, “Cheat sheet: What is Digital Twin?,” *Business Operations*, 2020. [Online]. Available: https://www.ibm.com/think/ topics/digital-twin, Accessed on 28/11/2024.
12. N. Napp and L. Nisiotis, *Digital Reality Technology, Challenges, and Human Factors*. IEEE Technology and Engineering Management Society Body of Knowledge (TEMSBOK), John Wiley & Sons, Ltd, 2023.
13. S. Olcott and C. Mullen, “Digital twin consortium defines digital twin,” 2020. [Online]. https://www.digitaltwinconsortium.org/2020/12/digital-twin-consortium-defines-digital-twin/, Accessed on 28/11/2024.
14. 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.
15. P. Öhlén, C. Johnston, H. Olofsson, S. Terrill, and F. Chernogorov, “Network digital twins – outlook and opportunities,” *Ericsson Technology Review*, vol. 2022, no. 12, pp. 2–11, 2022.
16. Z. Lai, D. Yuan, W. Bao, Y. Zhang, and B. B. Zhou, “DeepWiSim: A wireless signal simulator for automatic deep learning,” in *Proceedings of the SIGCOMM ’22 Poster and Demo Sessions*, (New York, NY, USA), p. 13–15, Association for Computing Machinery, 2022.
17. P. Tang, J. Zhang, Y. Zhang, Y. Guan, P. Qi, F. Wang, L. Yu, and P. Zhang, “Demo abstract: "See the Radio Waves" via VR and its application in wireless communications,” in *IEEE INFOCOM 2021 Workshops*, pp. 1–2, 2021.
18. J. Sköld, “Augmented visualisation of radio path propagation,” 2021. Masters Dissertation, Umeå University, Department of Physics.
19. B. Levchenko, A. Chukhray, and D. Chumachenko, “Development of game modules with support for synchronous multiplayer based on Unreal Engine 4 using artificial intelligence approach,” in *Integrated Computer Technologies in Mechanical Engineering*, pp. 503–513, 2020.
20. D. E. Rzig, N. Iqbal, I. Attisano, X. Qin, and F. Hassan, “Virtual reality (VR) automated testing in the wild: A case study on Unity-based VR applications,” in *Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis*, ISSTA 2023, (New York, NY, USA), p. 1269–1281, Association for Computing Machinery, 2023.

Louis Nisiotis is an Assistant Professor at the University of Central Lancashire, Cyprus, in Pyla, 7080, Cyprus. His research interests include eXtended Reality, Digital Twins, Serious and Entertainment Videogames. Dr Nisiotis received his PhD in Computing from Sheffield Hallam University, UK. He is a member of IEEE, and IEEE Computer Society. Contact him at LNisiotis@uclan.ac.uk.

Anna Anikina is a Research Assistant at the University of Central Lancashire, Cyprus, in Pyla, 7080, Cyprus. Her research interests include Gamification in Learning and Training, Communication Signal Analysis and Perception, and Interconnected Metadata Architecture. Mrs Anikina received her BSc (Hons) in Computing from the University of Central Lancashire, Cyprus. Contact her at AAnikina@uclan.ac.uk.

Marios Raspopoulos is an Associate Professor at the University of Central Lancashire, Cyprus, in Pyla, 7080, Cyprus. His research interests include Radio Propagation and Channel Modelling, localization and Tracking. Dr Raspopoulos received his PhD in Telecommunications Engineering from the University of Surrey, UK. He is a member of the IEEE. Contact him at https://mraspopoulos.github.io/.