2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC), 1-5 October 2023, Barcelona, Spain
DOI:10.1109/DASC58513.2023.10311124

Co-simulation Digital Twin Framework for Testing Future Advanced Air Mobility Concepts: A Study with BlueSky and AirSim

Junjie Zhao, Christopher Conrad, Rodolphe Fremond, Anurag Mukherjee, Quentin Delezenne, Yu Su, Yan Xu, and Antonios Tsourdos
School of Aerospace, Transport and Manufacturing
Cranfield University
Bedford, United Kingdom
Email: {junjie.zhao, christopher.conrad, rodolphe.fremond, anurag.mukherjee.436, quentin.delezenne.464, Yu.Su, yanxu, a.tsourdos}
@cranfield.ac.uk

Abstract—**The UK Future Flight Vision and Roadmap outlines the anticipated development of aviation in the UK by 2030. As part of the Future Flight demonstration segment, project HADO (High-intensity Autonomous Drone Operations) will develop, test, and deploy fully automated unmanned aircraft system (UAS) operations at London Heathrow Airport. Cranfield University is leading the synthetic test environment development within the HADO project, and a digital twin (DT) prototype was developed to enable mixed-reality tests for autonomous UAS operations. This paper enhances the existing DT by introducing new co-simulation capacities. Specifically, a co-simulation DT framework for autonomous UAS operations is proposed and tested through a demonstrative use case based on BlueSky and AirSim. This prototype integrates the traffic simulation capabilities of BlueSky with the 3D simulation capabilities of Airsim, to efficiently enhance the simulation capacities of the DT. Notably, the co-simulation framework can leverage the 3D visualization modules, UAS dynamics, and sensor models within external simulation tools to support a more realistic and high-fidelity simulation environment. Overall, the proposed co-simulation method can interface several simulation tools within a DT, thereby incorporating different communication protocols and realistic visualization capabilities. This creates unprecedented opportunities to combine different software applications and leverage the benefits of each tool.**

Keywords—HADO, Unmanned Aircraft System, Digital Twin, Co-simulation, BlueSky

I. INTRODUCTION

A. Context

As future flight technologies and capabilities advance, new classes of vehicles are being developed with novel technologies and capabilities [1]. These include unmanned aircraft systems (UASs), advanced air mobility (AAM) and regional air mobility vehicles. To address the challenges of these emerging technologies, a collective and collaborative effort is required to develop the systems, products and services needed to safely integrate new aerial vehicles and operations. Consequently, considerable research is underway to develop innovative solutions for future flight traffic management.

The progress of ongoing AAM research projects and initiatives in the EU, Switzerland, the US, and the UK has been reviewed by the same authors of this paper in [2]. In general, the US Federal Aviation Administration (FAA) NextGen Office released an initial Concept of Operations (ConOps) V1.0 in 2018 and V2.0 in 2020 to describe the essential components of a UAS traffic management (UTM) ecosystem. In Europe, U-space was established to manage UAS traffic in 2017. This is defined as a set of services and procedures relying on a high level of digitalization and automation to support the safe, efficient, and secure integration of dense UAS operations [3]. In 2020, the Swiss U-Space ConOps was proposed in alignment with the EU U-space concept, describing the high-level requirements for developing and deploying crewed and uncrewed operations within the Swiss U-Space system [4]. In the UK, Connected Places Catapult (CPC), in collaboration with the Department for Transport (DfT), Civil Aviation Authority (CAA) and several industrial stakeholders, developed a UTM framework termed Open-Access UTM (OUTM) between 2018 and 2021 [5]. This OUTM concept demonstrated how UTM could effectively complement air traffic management (ATM) services [6].

These ConOps highlight the principles, challenges, assumptions, services, operating methods, requirements, regulations and standards of AAM and UTM. Notably, all four proposals advocate a similar evolutionary and phased approach to introducing and implementing AAM. Nonetheless, airspace management proposals vary in the literature, offering different definitions, operational structures and stakeholder roles within the AAM ecosystem.

The UK Future Flight Challenge aims to establish a roadmap for the development of a new aviation system in the UK by 2030. The four-year program is creating the aviation system of the future and will demonstrate the safe integration and operation of UASs, AAM operations and regional aircraft by 2024, backed by significant advancements in electrification and autonomy [1]. As one of the seventeen Future Flight Challenge Phase 3 projects, project HADO (High-intensity Autonomous Drone Operations) will develop, evaluate, standardize, and operationally deploy fully automated UASs at London Heathrow Airport to conduct 24/7 commercial UAS operations. This will involve a 4-month evaluation of beyond visual line of sight (BVLOS) UAS operations in both physical and synthetic test environments.

The HADO project involves depolying a UTM system for BVLOS UAS operations. Specifically, Cranfield University is leading the development of a synthetic test environment within project HADO. In this context, the initial capacities of a digital twin (DT) mixed-reality system were demonstrated in [2], and shown to facilitate virtual and mixed-reality testing of the UTM systems. These initial capabilities were enhanced in [7] through a robust software architecture, and tested through several multi-agent and UTM use cases. This paper

---

This research is funded by the UKRI Future Flight Challenge Phase 3 project HADO (High-intensity Autonomous Drone Operations), grant number 10024815.

further builds on this work to introduce co-simulation capabilities within the DT system.

B. Related Work

The authors comprehensively review related work on DT systems in [2], and extend this work to a fully functional prototype in [7]. This paper builds on prior work to demonstrate the benefits of co-simulation within a DT development platform.

Numerous simulation platforms have been developed for UAS operations, as summarized in Table 1. Notably, Microsoft released AirSim for high-fidelity large-scale testing of autonomous solutions [8]. Moreover, Quantum 3D [9] and MIT [10] have developed UAS pilot simulators to train UAS stakeholders across a range of mission profiles. Other well-established simulators include CoppeliaSim (formerly V-REP) [11], Gazebo [12], Hexagon Flight Simulator [13], MATLAB / Simulink [14], Simulator and Testbed for MAV Swarm Experiments (Simbeeotic) [15], Flight Gear [16], and DroneSim Pro [17]. These simulation tools, however, all have a range of benefits and limitations, and no simulation tool in isolation can meet all the requirements of a DT mixed-reality system. Consequently, co-simulation is required to leverage and integrate the capabilities of different software platforms.

In general, co-simulation is a simulation technology that can integrate and coordinate different simulation models or tools to create a unified and synchronized simulation environment. Co-simulation has been studied in numerous DT applications, such as transportation [18], aerospace [19], marine systems [20], smart cities [21], energy systems [22] and elements within the automotive industry [23]. A literature review was conducted on the application of co-simulation in DTs, and a summary of these applications is presented in Table 2. To date, however, most of these DT implementations do not exceed a maturity level of three, and few have started integrating the system with real-time data streams. Co-simulation therefore plays a crucial role in increasing this maturity level, by better replicating real-world scenarios and interactions. By combining models from diverse domains, such as physics, control systems, and data analytics, co-simulation enables a more accurate and comprehensive representation of the physical system being modelled. This integrated approach facilitates the evaluation of system behaviour, performance optimisation, and the identification of potential issues or vulnerabilities within the DT.

C. Contributions

It is needed to develop an accurate, efficient and scalable co-simulation-based DT framework for the development, verification and validation of UAS and AAM operations. This must enable a standardizable architecture and communication interface to ensure interoperability, compatibility, and synchronization with multiple software packages.

This study thereby proposes and demonstrates a co-simulation DT framework for autonomous UAS and AAM operations. It highlights how different co-simulators contribute to the DT framework, and showcases a prototype based on BlueSky and AirSim for enhanced air traffic simulation capabilities. The main contributions of this work are as follows.

- A generic co-simulation DT framework for testing autonomous UAS operations is developed based on the DT prototype developed in prior work. This enables a more efficient way to enhance the simulation capacities of the DT by leveraging, integrating, and visualizing them through external simulation tools, rather than building them from scratch.
- A co-simulation DT prototype based on AirSim and BlueSky is developed. In particular, the traffic simulation capacity of BlueSky is leveraged and integrated within the DT to enable air traffic simulation within the mobility layer of the DT framework. Additionally, AirSim and the 3D digital environment can enhance the simulation capacities with unique and high-fidelity data model.
- A stepwise method for building a high-fidelity digital environment is proposed, based on AutoCAD, Blender, Substance 3D Painter and Unreal. This improves the fidelity, accuracy and efficiency of the DT model.

D. Paper Structure

The remaining part of the paper is organized as follows: Section II presents the proposed methodology to develop a co-simulation DT framework; Section III presents up-to-date DT framework capacities in terms of the digital environment; and Section IV presents and discusses the experiments and results on the enhanced DT capacities based on co-simulation with AirSim and BlueSky. Finally, in Section V, the main conclusions of this work are summarized, and the future direction of this research is proposed.

II. GENERIC CO-SIMAUTION DT FRAMEWORK

To support the exploration and development of co-simulation capabilities within a DT, a generic co-simulation framework is proposed in this paper. Notably, co-simulation is a key enabler of the DT as well as a key capacity of the DT framework for AAM development and testing. The proposed generic architecture is illustrated in Fig. 1 and is composed of four main elements: co-simulation tools, the server module, the 3D visualization module and the physical world.

The physical world can be decomposed into UTM / UAS operations and the environment in which they operate. The physical world module in Fig. 1 shows the generic architecture for UTM and UAS operations and highlights various actors and components, their contextual relationships, and the associated high-level functions and information flow. The 3D visualization module is the module to create, integrate and visualize the elements of the DT. The DT encompasses both the digital environment and the digital elements needed to support UTM and UAS operations. Five layers are defined for the DT, with the first three layers (terrain, buildings and weather/atmosphere) building up the digital environment and the last two layers (infrastructure and mobility) relating to the digital elements required for UTM and UAS operations.

The co-simulation tools refer to the simulation tools that can be leveraged, integrated, and visualized in the DT framework. In the context of UTM and UAS operations, the simulation tools can include simulation software with low-fidelity or fast-time simulation capacities. These simulation capabilities from the co-simulation tools shall be integrated and visualized in a particular layer of the DT. Two or more simulation tools shall be interfaced and used simultaneously together to contribute to the overall DT framework.

TABLE 1. REVIEW OF SIMULATION TOOLS FOR UAS OPERATION

Simulator	Ref.	Description	Benefits	Limitations
AirSim	[8]	An end-to-end platform for safely creating, training, and validating autonomous agents through simulation	• Open-source and high-potential in DT	• Technical knowledge needed
Quantum3D Fixed-wing UAV Simulator	[9]	An operator/pilot-centric simulator with a generic aerodynamic model, integrated with sensors and camera simulations, which supports stand-alone or networked simulations for a wide variety of missions	• Good portability and transportability	• Limited in test capacities
MIT Media Lab UAV Pilot Simulator	[10]	A web-based simulation system aimed at providing real-time photorealism without requiring a high-end computing system	• Effective design choices to limit the need for high-end hardware	• Limited in DT capacities
CoppeliaSim, formerly V-REP	[11]	A robotics simulation software with a distributed control architecture	• User-friendly interface and IDE • Multiple entities can be independently controlled • Compatible with many programming languages	• High resource usage • Slow simulation speed • Limited scalability
Gazebo	[12]	An open-source robot simulation software	• Wide variety of features • Compatible with many programming languages and robotics platforms • Extensive community support is available	• Complex and hard to learn • Very resource demanding
Hexagon Flight Simulator	[13]	A professional flight simulator	• Supports dynamic tuning of mission parameters. • Includes a joystick and RC for manoeuvring micro aerial vehicles and an interface RC simulator for better realism	• Not aimed at multi-agent simulations • Limited flexibility and customisability
MATLAB / Simulink	[14]	A widely used programming language and simulation platform developed by MathWorks, capable of simulating robotics and UAV operations	• Wide range of modelling and simulation tools • Offers enhanced UAS simulation capabilities through specialised tools and toolboxes • Easy-to-use and well-documented • Supports SIL and HIL • One of the most important and widely used simulation and development environments for UAS systems and operations	• Unable to support high-fidelity DTs • Simulations developed using MATLAB lack realism and are more focused on backend computations • Limited scalability • Resource-demanding • Slow simulation speed
Simulator and Testbed for MAV Swarm Experiments (Simbeeotic)	[15]	An open-source simulator that can simulate UAV swarm and UAVNet communication infrastructures in a 3D virtual world	• Efficiently models sensing, actuation, and communication within a UAS swarm • Enables collaborative and distributed simulations, whereby multiple users can interact with the same simulation in real time from different locations	• Limited realism • Limited ability to simulate complex frameworks, ground infrastructures and other conventional aircraft
Flight Gear	[16]	An advanced flight simulator, primarily developed for fixed-wing aircraft, that can also be used to simulate UASs	• Compatible with a wide variety of operating systems and platforms • Supports the integration of real-time data for improved decision-making capabilities • Contains several different and customisable aircraft models • Supports multiplayer simulations and flight training scenarios • Open-source, with considerable community support	• Not catered for multi-UAS operations • Not conducive to the DT application under consideration
DroneSim Pro	[17]	An affordable drone simulator that utilises precise drone flight physics within practical scenarios centred around the Phantom Vision+ model	• Suitable for both beginners and advanced pilots • Compatible with a wide variety of operating systems and platforms • Provides detailed analysis reports of pilot performance after a simulation	• Does not readily support new vehicle models. • Not designed to test and deploy large-scale UAS / AAM operational frameworks

The server module is the core of the co-simulation framework. It is responsible for communication between the physical world and the 3D visualization module, and the synchronization of all components within the DT framework. The data transmitted to the 3D visualization module can be information from the physical world and information from the co-simulation tools.

TABLE 2. OVERVIEW OF EXISTING CO-SIMULATION APPLICATIONS IN DT FRAMEWORKS

Domain	Ref.	Application	Simulation Software	Communication Protocol	Sensors	Eval.	Matur. Level
Airspace	[18]	Manned aircraft flights	Matlab / Simulink, Xplane, FlightGear	UDP	Speed, control, position, etc.	-	0
Airspace	[19]	Unmanned aircraft flights	Matlab / Simulink, Xplane, Autopilot	UDP	Speed, control, position, etc.	-	0
Marine	[20]	Ship	Libcosim, Matlab, Python, Java, C/C++	TCP / IP	pre-recorded position, dimension, etc.	-	2
Smart Cities	[21]	Urban space	Unity3D, SUMO	IoT	IoT sensors	-	2
Energy Systems	[22]	Motorized spindle	Ansys, Matlab, and LabVIEW	TCP / IP	Temp., speed	Accuracy testing	3
Automotive	[23]	Automated car systems	Unreal, Matlab Simulink, Python, SUMO	5G	LIDAR, RADAR, GPS, CAN	Accuracy testing, ISO standards	3

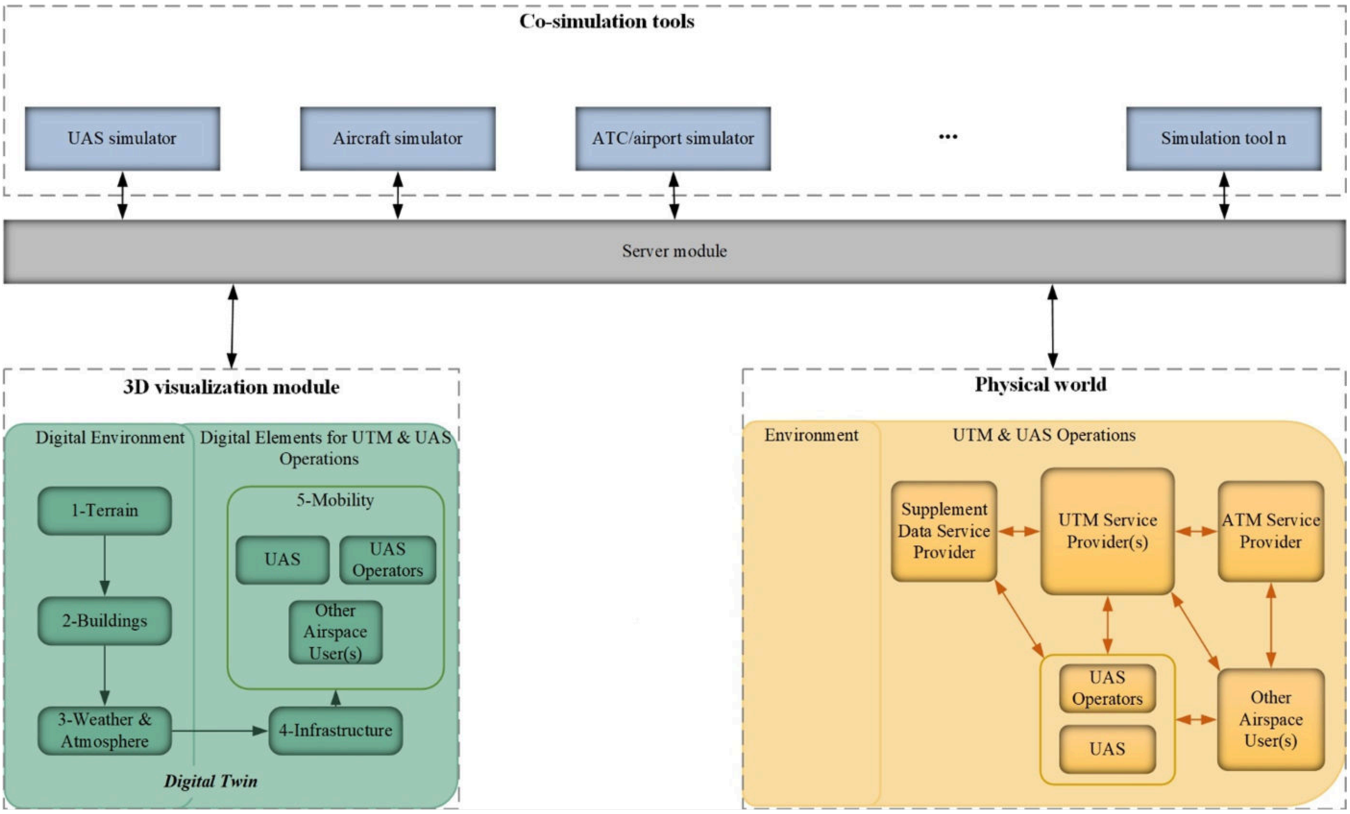

Fig. 1. Generic co-simulation DT framework.

III. ENHANCED DIGITAL WORLD

A DT prototype was developed in [2] and extended in [7] for testing autonomous UAS operations. The Unreal 4.27 graphical engine was used for 3D creation and visualization within this prototype, to develop, integrate and visualize the components of the digital environment. AirSim was further used to simulate the UAS and UAS operations within the digital environment. This is an open-source and cross-platform UAS simulator developed by Microsoft Research. This section describes how the digital world of the DT was further enhanced in this work.

The digital environment of the DT prototype was enhanced using the methodology proposed in [2] and [7]. Notably, the Cesium plugin provided World Terrain, Bing Maps Aerial Imagery, and OpenStreetMap (OSM) Buildings as static layers within the digital world. Nonetheless, the layers provided by Cesium are not updated regularly and cannot be easily modified. Consequently, static layers were also built in Blender to replace preliminary layers provided by Cesium, thereby enhancing the accuracy of the digital world. Specifically, the OSM plugin for Blender was used, and the OSM objects were modified to meet the fidelity requirements of DT under development.

The region of interest was first selected using the OSM plugin, as depicted in Fig. 2. The experimental area considered was Cranfield University Campus (including Cranfield Airport). The terrain and satellite image overlay were subsequently created in Blender, followed by the default building infrastructure. As opposed to directly importing the OSM layer Cesium, this methodology enables 3D buildings and entities to be created separately using other software and manually introduced within the OSM layer set.

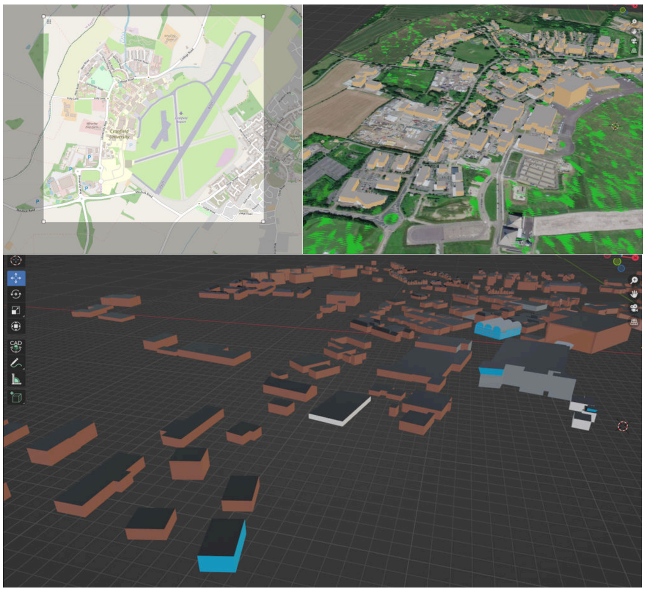

Fig. 2. Blender OSM plugin for choosing regions of interest (top-left), providing data for Blender (top-right), and manually modifying the OSM objects (bottom).

A workflow is developed for high-fidelity building modelling. Firstly, the AutoCAD layout of the area of interest was used to determine the dimensions and features of the buildings. The information was subsequently imported inside Blender, as illustrated in Fig. 3. This offers details on the building shape projection, road lines and markings, and coverage area of roof extensions. Secondly, the object meshes were created in Blender. For objects and features on which data could not be found, the models have been made with creative liberty to achieve a high-fidelity aesthetic replica of the real-world entities. Thirdly, the objects were exported to Substance 3D Painter for texturing. The textures of all objects were generated using the export presets for Unreal Engine already embedded in the software. The textures can also be imported back to Blender in the rendering engine, allowing a preview of the result before implementation in the simulation environment. Fig. 4 illustrates the process of developing high-fidelity models of the ARIC and DARTeC buildings at the Cranfield University with the proposed workflow.

After preparing the buildings and other information in Blender, they were exported as 3D object files (.fbx) and imported into the Unreal engine as a static mesh. Specifically, a .fbx file format is preferred as it ensures that the exported models, materials, animations, and other data are preserved and can be accurately imported into the Unreal engine.

Inside the Unreal engine, further modifications can be carried out to improve the fidelity and realism of the digital environment. Static meshes, such as windows on buildings, fences, aircraft, trees, and lampposts can be appropriately placed. Such assets can be obtained from the 3D object *Quixel Megascans* library. Materials for the buildings, roads and other surfaces were further created using Blueprint programming within the Unreal engine, which is a visual scripting system that allows developers to create gameplay materials, mechanics, interactions, and functionality without writing traditional code. Fig. 5 shows the enhanced digital environment at the Cranfield University campus.

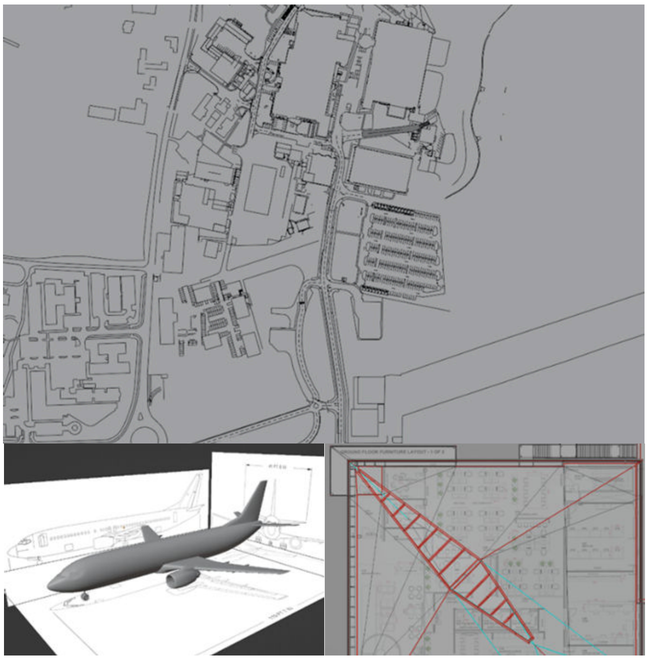

Fig. 3. View of the AutoCAD Cranfield model (top), Boeing 737 model (bottom-left) and DARTeC building model (bottom-right) designs.

In terms of the digital elements for UTM and UAS operation, For this study, the default flight controller, sensor models, dynamic vehicle model, physics engine, rendering engine, and public API layer provided by AirSim were used. Additionally, the AirSim Python APIs were utilized to interact with the simulated UASs. The custom vehicle mesh is available. It can be created with Blender, textured with Adobe Substance 3D Painter and integrated into AirSim in sequence. Custom sensor models are also being developed to supplement the default sensor models provided by AirSim. For example, the Spirent GSS7000 platform is being deployed to get more realistic IMU and GNSS data. MATLAB / Simulink is being used to develop other custom components, such as sensor models and flight controllers.

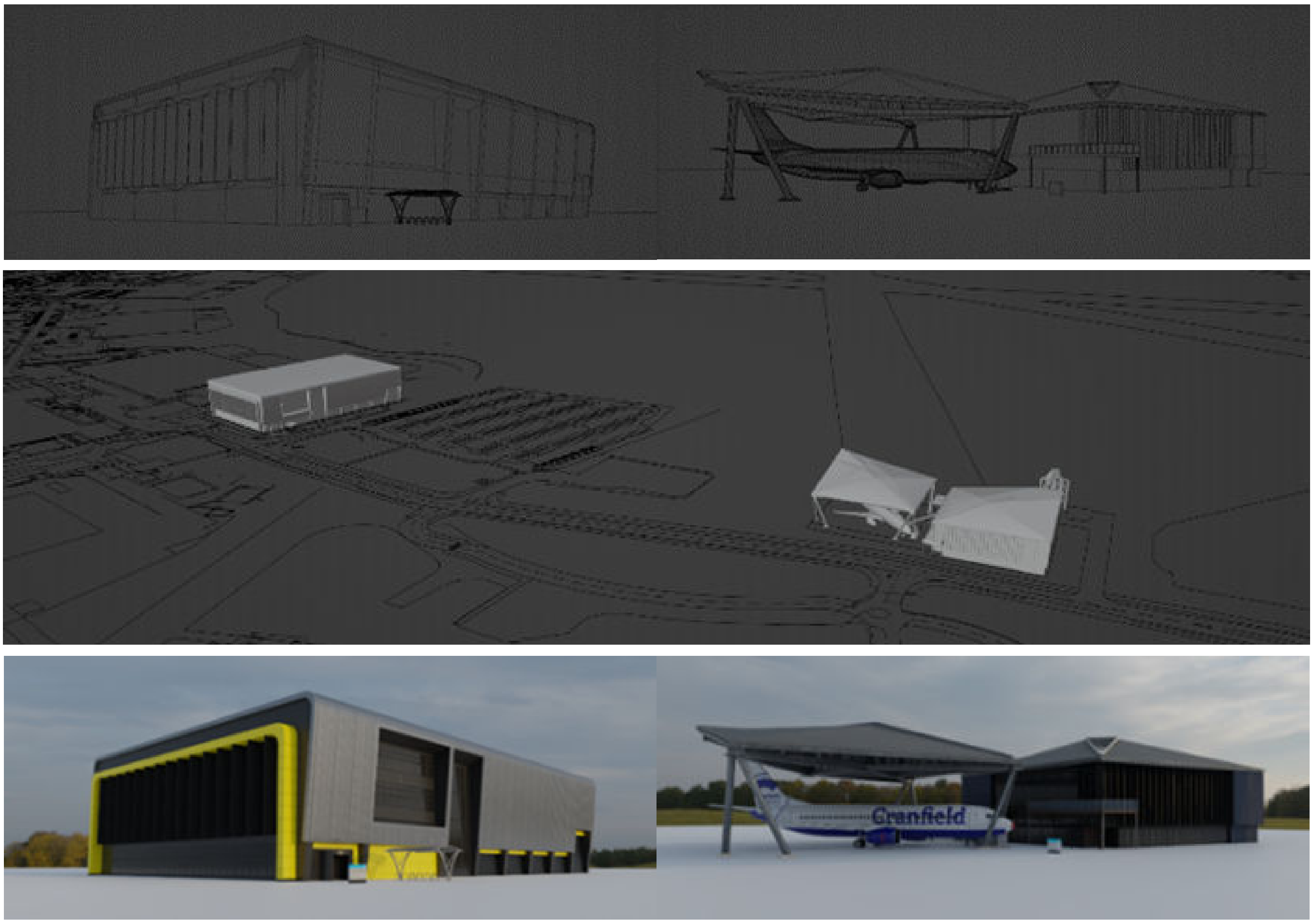

Fig. 4. Process of developing high-fidelity models of the ARIC and DARTeC buildings.

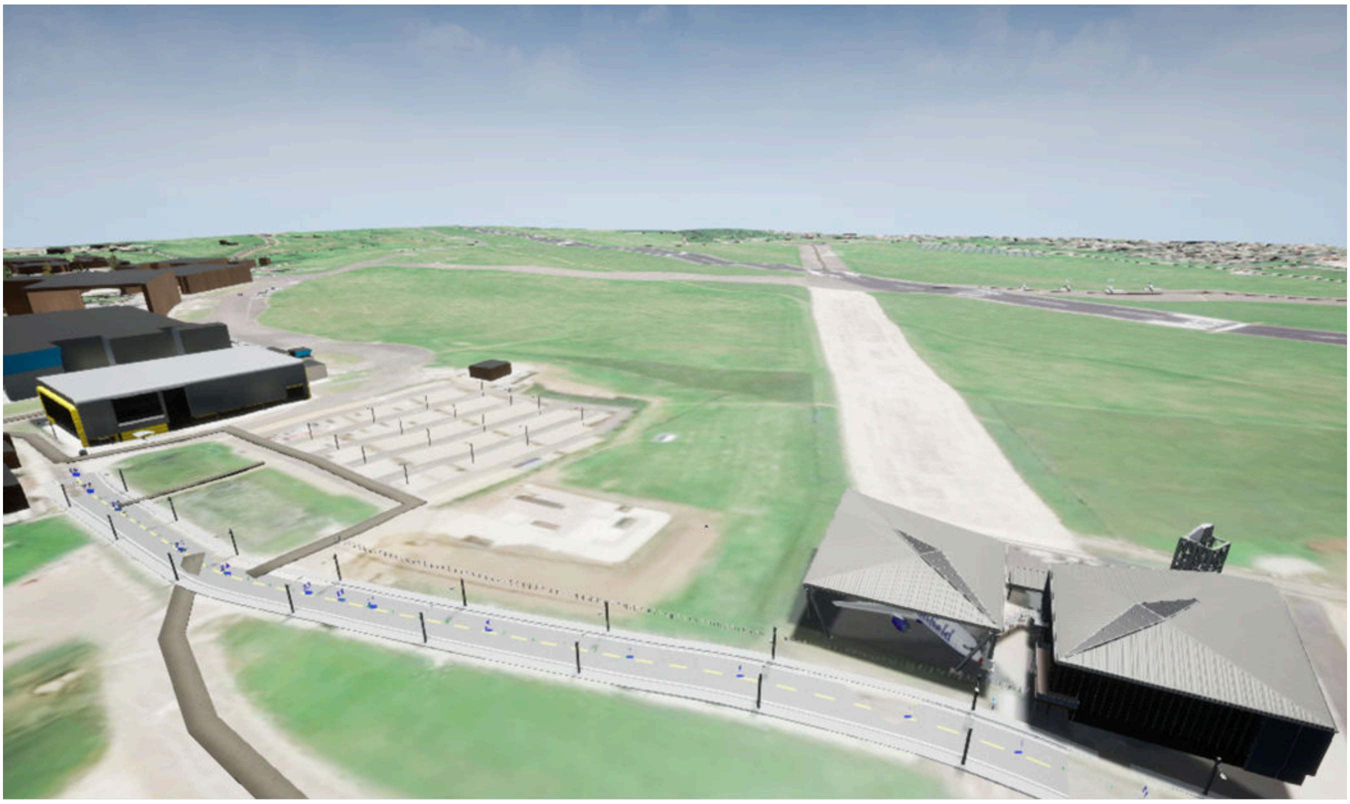

Fig. 5. High-fidelity 3D models of the ARIC and DARTeC buildings.

IV. ENHANCED DT CAPACITIES BASED ON CO-SIMULATION WITH BLUESKY AND AIRSIM

This section describes preliminary experiments, results and discussions that were conducted to integrate BlueSky as a co-simulation tool for the DT framework.

The co-simulation DT framework is developed based on the DT prototype. Specifically, the Unreal engine is the core of the 3D visualization module to develop, integrate and visualize elements of the co-simulation DT framework. The AirSim further acts as one of the co-simulation tools to enable 3D UAS simulations in the mobility layer of the 3D visualization module.

A. BlueSky – Mobility Layer Traffic Simulation

BlueSky is a fully open-source and open-data tool for air traffic simulation [24], and has been enhanced to accommodate unmanned traffic simulation by the authors of this work [25]. Additionally, the authors have enabled the possibility of establishing and displaying live/simulated manned/unmanned traffic in BlueSky. Consequently, BlueSky is an important co-simulation tool, to provide traffic information for the mobility layer of the DT framework.

Fig. 6 shows a stepwise process of the proposed co-simulation method for traffic simulation of the mobility layer of the co-simulation DT framework. This is enabled through the following sequential process:

1. The traffic data is acquired from the BlueSky co-simulator.
2. The traffic data is transmitted through ZMQ (ZeroMQ) TCP communication provided by the server module.
3. The traffic data is processed and formatted in the server module.
4. The other layers of the digital world are calibrated, synchronized and displayed to align with the visualization in BlueSky.
5. Traffic information is visualised, including vehicle creation, traffic flow display, and state updates.

Each step within the process is hereby discussed for the demonstrative use case under consideration:

a) Data Acquisition: The traffic scenario was generated in BlueSky for the case study under consideration. Specifically, the unmanned traffic within the experimental area around the DARTeC building at Cranfield University was simulated, for a set of UAS operations lasting 2 minutes and 22 seconds. Five unmanned aerial vehicles (UAVs) were included in the scenario, each executing a unique mission along a different route. The ID, GPS location, velocity and status of each UAV were obtained.

b) Communication: ZMQ is an open-source and high-performance messaging library that provides asynchronous and lightweight communication between applications or processes. It is being used in BlueSky for multiprocess document communication. In this study, ZMQ TCP communication is applied for the communication between BluSky and the other components within the co-simulation DT framework. ZMQ TCP communication uses the TCP transport in ZMQ for communication between ZMQ sockets. The communication was built up following the steps: ZMQ context creation, socket creation with TCP transport option, socket connection/binding to an IP address and port, and messages transmitting with the socket.

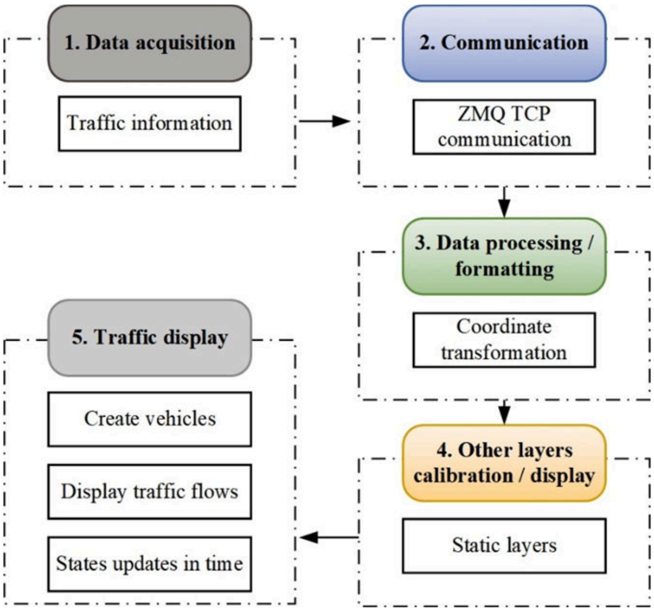

Fig. 6. Stepwise method for traffic simulation of the mobility layer of the co-simulation DT framework.

c) Data Processing / Formatting: The way points of the aircrafts are indicated with coordinates at each time step. Consequently, a coordinate system transformation was also developed to align the coordinate systems of the various simulation tools. BlueSky uses general polar coordinates (latitude, longitude, altitude) to express the location on the map. AirSim and Unreal, however, use cartesian coordinates to define the location. Specifically, AirSim uses the North-East-Down (NED) coordinate system, whereby +X is North, +Y is East, and +Z is Down, and all units are in metres. In Unreal Engine, however, +Z is up instead of down, and the length unit is in centimeters instead of meters. The coordinates, therefore, need to be transformed into the proper formats before being used.

d) Other Layers Calibration / Display and Traffic Display: The other layers of the 3D digital environment within the DT framework were calibrated with BlueSky and displayed in the Unreal environment. The other layers, including buildings and weather / atmosphere were developed and evaluated for the DT framework in [7]. The traffic information was further displayed in the 3D digital environment as components of the mobility layer of the digital world. Moreover, the vehicle states were continually updated as the simulation progressed.

Fig. 7 shows a snapshot of the 2D map in BlueSky (left) and the 3D digital world in the Unreal engine (right). The static layers of the digital world were thereby aligned with the 2D map in BlueSky. The traffic information acquired in BlueSky was successfully transmitted to the server module for processing and formatting. Additionally, the traffic information was successfully displayed in the BlueSky 2D map and 3D digital environment as components of the mobility layer of the digital world, as shown in Fig. 8. The vehicle states were also successfully updated in time.

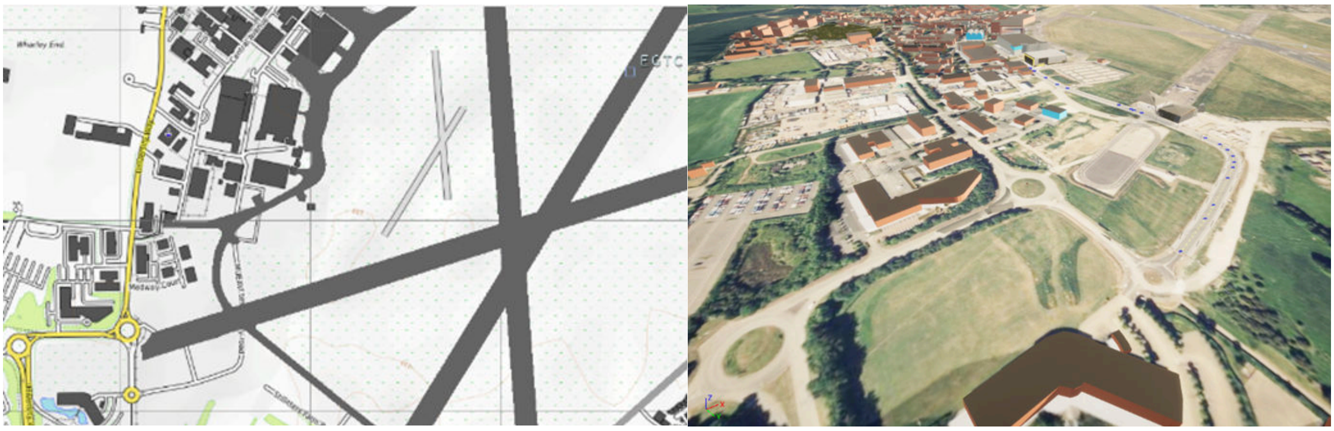

Fig. 7. 2D map in BlueSky (left) and the 3D digital world in Unreal (right).

B. *Co-simulation of BlueSky and AirSim*

As co-simulators of the co-simulation DT framework, BlueSky and AirSim can communicate and interoperate in real time. Specifically, BlueSky can provide traffic information to AirSim and the 3D visualization module for traffic simulation. Conversely, AirSim can provide readings for advanced sensor modules to enhance the BlueSky simulation.

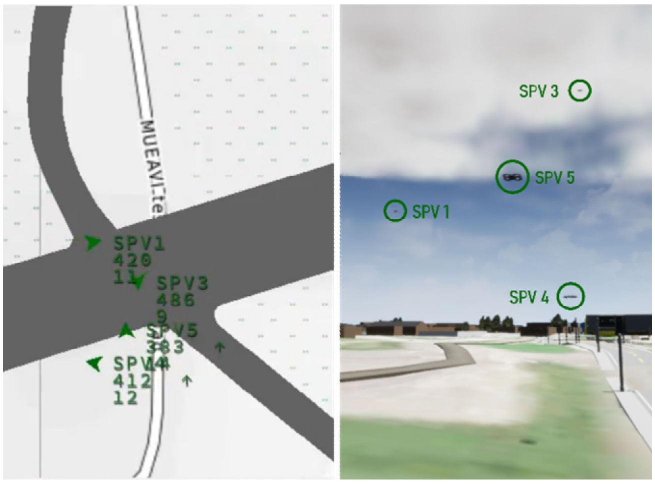

Fig. 8. Traffic co-simulation with BlueSky and AirSim.

Fig. 9 illustrates the co-simulation and bi-directional communication between BlueSky and AirSim. At each time step, track and velocity data are transmitted from BlueSky to AirSim, and the data are processed and displayed in the Unreal engine as traffic information of the mobility layer within the digital environment. Meanwhile, the sensor data from each UAV are recorded. These data are subsequently accessed by BlueSky and used to enhance the original simulation with additional data from AirSim. In the considered use-case displayed in Fig. 9, altitude data of the terrain is recorded using altitude sensors mounted on the UASs within AirSim. This data is subsequently used to include the above ground level (AGL) readings within the BlueSky simulation.

C. *Discussions*

The DT prototype discussed in [2] and [7] was further developed, and enhanced co-simulation capacities were introduced. Moreover, detailed AutoCAD information was included, and a stepwise method for building a high-fidelity digital environment was proposed, based on AutoCAD, Blender, Substance 3D Painter and Unreal. This improves the fidelity, accuracy and efficiency of the DT model.

A systematic stepwise method for building a generic co-simulation DT framework for autonomous UAS operations was proposed, and a co-simulation DT prototype was created with AirSim, Unreal and BlueSky. The benefits of the co-simulation method are described as follows,

- The traffic simulation capacity of BlueSky is leveraged and integrated to enable the traffic simulation of the mobility layer of the DT framework. This proves the proposed method is a more efficient method to enhance the simulation capacities of the DT by leveraging, integrating, and visualizing them from other simulation tools instead of building from scratch.
- The 3D visualization module can enhance the simulation capacities with its unique and high-fidelity simulation capacities. In this case, BlueSky benefits from the 3D digital environment with more realistic sensor data.
- Potentially, as a combination, the co-simulation tools can be used in the integrated and standardized framework in parallel or sequence. In the case of BlueSky, the visualization view in BlueSky is better in showing the traffic states as a 2D view, and the 3D visualization in the Unreal engine is better in UAV operation details. It is useful to combine the views and switch from them as needed.

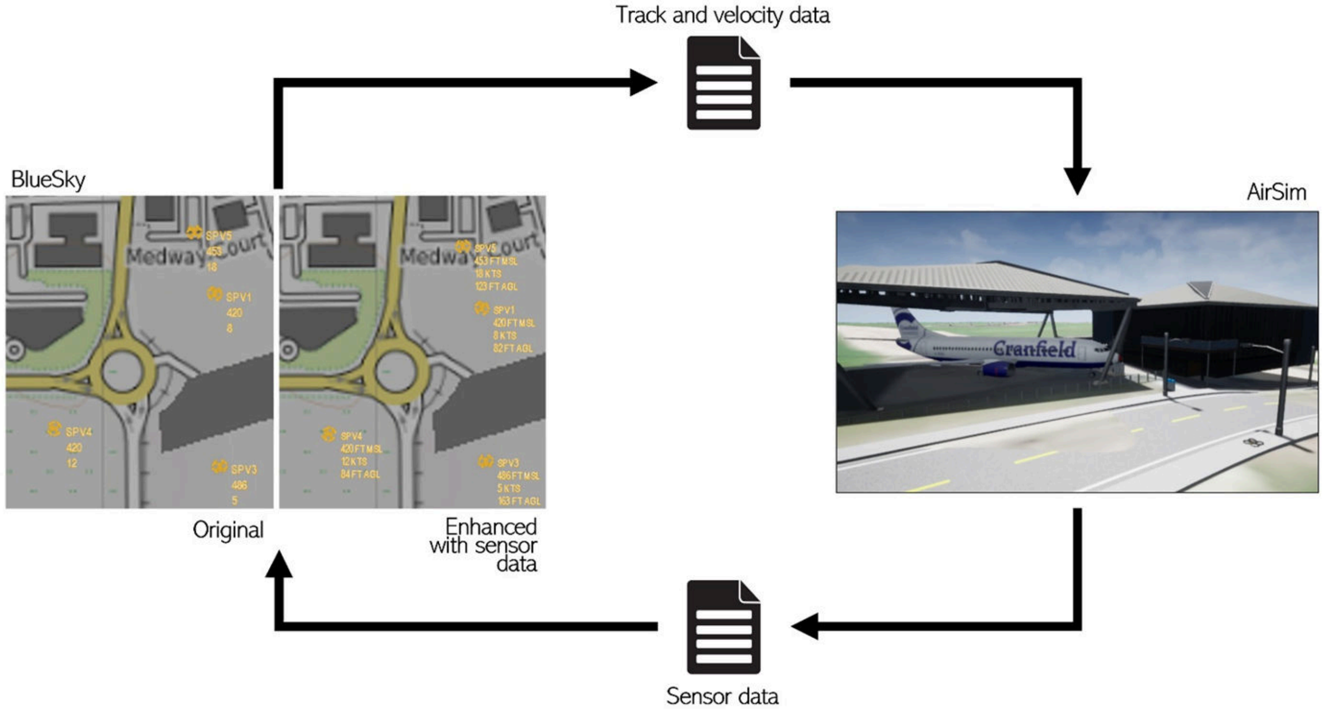

Fig. 9. Effects of the developed co-simulation use case between BlueSky and AirSim.

V. CONCLUSIONS

This study proposes a co-simulation method to develop an accurate, efficient and scalable DT framework throughout the entire development process of autonomous UAS operations. The proposed method is evaluated through an experiment and shown to exhibit promising results. Overall, the proposed co-simulation method can support the existing DT framework by interfacing several simulation tools, incorporating different communication protocols and adding realistic visualization, to create unprecedented opportunities for software tool combinations. In the context of digitalization and autonomy for AAM, the proposed DT system is envisioned to support the development, integration and testing of ATM / UTM ecosystems by enabling the seamless integration of simulation and physical test environments. This will be pivotal to enabling the widespread certification and adoption of these emerging operations.

Several areas shall be further explored in future research work. In terms of co-simulation, standardized and universal co-simulation architecture and communication interface to ensure interoperability, compatibility, and synchronization are required. Additionally, other simulation tools, such as MATLAB / Simulink, will be integrated into the co-simulation DT framework to enable more simulation capacities. Finally, rigorous evaluation and validation of the co-simulation DT framework will be conducted.

ACKNOWLEDGMENT

This research is funded by the UKRI Future Flight Challenge Phase 3 project HADO (High-intensity Autonomous Drone Operations), grant number 10024815. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) license to any Author Accepted Manuscript version arising.

REFERENCES

[1] UK Research and Innovation, Future Flight Vision and Roadmap, no. August. 2021. [Online]. Available: https://www.ukri.org/wp-content/uploads/2021/08/UKRI-130821-FutureFlightVisionRoadmap.pdf

[2] J. Zhao, C. Conrad, Q. Delezenne, Y. Xu, and A. Tsourdos, "A Digital Twin Mixed-reality System for Testing Future Advanced Air Mobility Concepts : A Prototype," in 2023 Integrated Communication, Navigation and Surveillance Conference (ICNS), 2023, pp. 1–10. doi: 10.1109/ICNS58246.2023.10124310.

[3] SESAR Joint Undertaking, Supporting Safe and Secure Drone Operations in Europe. 2020. [Online]. Available: https://www.sesarju.eu/sites/default/files/documents/u-space/U-space Drone Operations Europe.pdf

[4] Federal Office of Civil Aviation (FOCA) and the Swiss U-Space Implementation (SUSI) Public-Private Partnership, Swiss U-Space ConOps. 2020.

[5] CPC, "Connected Places Catapult Enabling UTM in the UK," no. May, 2020.

[6] UK Department for Transport, "Implementing an Open-Access UTM Framework for the UK," no. April, 2021, [Online]. Available: https://cp.catapult.org.uk/wp-content/uploads/2021/06/Implementing-an-Open-Access-UTM-Framework.pdf

[7] C. Conrad et al., "Developing a Digital Twin for Testing Multi-Agent Systems in Advanced Air Mobility : A Case Study of Cranfield University and Airport," unpublished.

[8] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles," Springer Proc. Adv. Robot., vol. 5, pp. 621–635, 2018, doi: 10.1007/978-3-319-67361-5_40.

[9] Quantum3D, UAV Simulator: fixed wing Simulators. Accessed: Mar. 16, 2023. [Online]. Available: https://quantum3d.com/uav-simulator/

[10] Stephen Kaputsos, Unmanned Aerial Vehicle (UAV) Pilot Simulator. MIT. Accessed: May 21, 2023. [Online]. Available: https://www.media.mit.edu/projects/cloud-uav-sim/overview/

[11] Coppelia robotics, Coppelia Sim from the creators of V-REP. Accessed: May 21, 2023. [Online]. Available: https://www.coppeliarobotics.com/

[12] A. Mairaj, A. I. Baba, and A. Y. Javaid, "Application specific drone simulators: Recent advances and challenges," Simul. Model. Pract. Theory, vol. 94, no. January, pp. 100–117, 2019, doi: 10.1016/j.simpat.2019.01.004.

[13] E. Capello, G. Guglieri, and F. B. Quagliotti, “UAVs and simulation: An experience on MAVs,” Aircr. Eng. Aerosp. Technol., vol. 81, no. 1, pp. 38–50, 2009, doi: 10.1108/00022660910926890.

[14] MATLAB, Math. Graphics. Programming. MathWorks. Accessed: May 21, 2023. [Online]. Available: https://uk.mathworks.com/products/matlab.html

[15] B. Kate, J. Waterman, K. Dantu, and M. Welsh, “Simbeeotic: A simulator and testbed for micro-aerial vehicle swarm experiments,” 2012 ACM/IEEE 11th Int. Conf. Inf. Process. Sens. Networks, pp. 49–60, 2014, doi: 10.1109/ipsn.2012.6920950.

[16] Flightgear, Flightgear Flight simulator. Accessed: May 21, 2023. [Online]. Available: https://home.flightgear.org/about/features/

[17] DroneSim Pro, droneSim Pro Drone Simulator. Accessed: May 21, 2023. [Online]. Available: https://www.dronesimpro.com/

[18] N. Horri and M. Pietraszko, “A Tutorial and Review on Flight Control Co-Simulation Using Matlab/Simulink and Flight Simulators,” Automation, vol. 3, no. 3, pp. 486–510, 2022, doi: 10.3390/automation3030025.

[19] J. Pinguet, P. Feyel, and G. Sandou, “A Neural Autopilot Training Platform based on a Matlab and X-Plane co-simulation,” 2021 Int. Conf. Unmanned Aircr. Syst. ICUAS 2021, pp. 1200–1209, 2021, doi: 10.1109/ICUAS51884.2021.9476679.

[20] L. I. Hatledal, R. Skulstad, G. Li, A. Styve, and H. Zhang, “Co-simulation as a Fundamental Technology for Twin Ships,” Model. Identif. Control, vol. 41, no. 4, pp. 297–311, 2020, doi: 10.4173/MIC.2020.4.2.

[21] G. White, A. Zink, L. Codecá, and S. Clarke, “A digital twin smart city for citizen feedback,” Cities, vol. 110, no. June 2020, 2021, doi: 10.1016/j.cities.2020.103064.

[22] H. Yi and K. Fan, “Co-simulation-based digital twin for thermal characteristics of motorized spindle,” Int. J. Adv. Manuf. Technol., pp. 4725–4737, 2023, doi: 10.1007/s00170-023-11060-6.

[23] Z. Szalay, “Next generation X-in-the-loop validation methodology for automated vehicle systems,” IEEE Access, vol. 9, pp. 35616–35632, 2021, doi: 10.1109/ACCESS.2021.3061732.

[24] J. Hoekstra, J. Ellerbroek, and J. M. Hoekstra, “BlueSky ATC Simulator Project: an Open Data and Open Source Approach Three-Dimensional Airborne Separation Assistance Displays View project BlueSky-Open source ATM simulator View project BlueSky ATC Simulator Project: an Open Data and Open Source Approach,” seventh Int. Conf. Res. Air Transp., no. June, 2016, [Online]. Available: https://www.researchgate.net/publication/304490055

[25] R. Fremond et al., “Demonstrating Advanced U-space Services for Urban Air Mobility in a Co-Simulation Environment”.

Co-simulation digital twin framework for testing future advanced air mobility concepts: a study with BlueSky and AirSim

Zhao, Junjie

IEEE

Zhao J, Conrad C, Fremond R, et al., (2023) Co-simulation digital twin framework for testing future advanced air mobility concepts: a study with BlueSky and AirSim. In: 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC), 1-5 October 2023, Barcelona, Spain
https://doi.org/10.1109/DASC58513.2023.10311124