ARL-TR-9465 ● May 2022

The Inclusion of Realistic Winds in a Simulated Environment for the Study of Wind-Unmanned Aircraft System (UAS) Interactions

by Kelly Kirk, Carl Lederman, Yansen Wang, and Brent Kraczek

NOTICES

Disclaimers

ARL-TR-9465 ● MAY 2022

The Inclusion of Realistic Winds in a Simulated Environment for the Study of Wind-Unmanned Aircraft System (UAS) Interactions

Kelly Kirk, Yansen Wang, and Brent Kraczek
DEVCOM Army Research Laboratory

Carl Lederman
Parsons Corporation

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* May 2022	**2. REPORT TYPE** Technical Report	**3. DATES COVERED (From - To)** 1 October 2020–31 August 2021
4. TITLE AND SUBTITLE The Inclusion of Realistic Winds in a Simulated Environment for the Study of Wind-Unmanned Aircraft System (UAS) Interactions		**5a. CONTRACT NUMBER** **5b. GRANT NUMBER** **5c. PROGRAM ELEMENT NUMBER**
6. AUTHOR(S) Kelly Kirk, Carl Lederman, Yansen Wang, and Brent Kraczek		**5d. PROJECT NUMBER** **5e. TASK NUMBER** **5f. WORK UNIT NUMBER**
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) DEVCOM Army Research Laboratory ATTN: FCDD-RLC-ED Aberdeen Proving Ground, MD 21005		**8. PERFORMING ORGANIZATION REPORT NUMBER** ARL-TR-9465
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		**10. SPONSOR/MONITOR'S ACRONYM(S)** **11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES

ORCID ID: Kelly Kirk, 0000-0002-8372-5749

14. ABSTRACT

Rapidly advancing technologies in graphics, gaming, and artificial intelligence are driving the development of synthetic environments (SEs). Accurate physics is a key component in SEs to ensure that simulations represent reality as closely as possible. This is particularly important for engineering systems, or for planning operations in regions of high complexity, such as urban environments. We have developed a simple simulation platform for testing interactions between unmanned aircraft systems (UASs) and winds in urban environments. This platform is built on Unreal Engine (i.e., a gaming engine) and cityscapes of real-world cities adapted from Open Street Map. The wind fields are simulated using lattice Boltzmann methods within a geometrically representative city model. From these wind fields, we can simulate wind-UAS interactions, visualize wind streamlines, and identify regions of likely high wind speed and/or turbulence. We employ remote procedure calls to handle the large size of volumetric wind data, storing and managing these data separately from the simulation environment. Within this simulation platform, we demonstrate how a hyper-local knowledge of winds can be used to model significant deviations in UAS trajectories for a preplanned flight through Chicago. The resulting information could be used to plan UAS operations—balancing safety, area coverage, power usage, location avoidance, and total operational time. Improved UAS-wind interaction models would also allow the testing of UAS suitability for specific locations and conditions.

15. SUBJECT TERMS

Military Information Sciences, Network and Computational Sciences, Synthetic Training Environment, unmanned aircraft system, UAS, simulation environments

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Kelly Kirk
a. REPORT Unclassified	**b. ABSTRACT** Unclassified	**c. THIS PAGE** Unclassified	UU	14	**19b. TELEPHONE NUMBER (Include area code)** (410) 278-9281

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

1. Introduction: Motivation and Overview

Simulation environments are a rapidly developing field in computation. Enabled by virtual reality (VR), augmented reality (AR), artificial intelligence (AI), and advanced gaming engines, they promise virtual worlds that may be the next major stage of the Internet. Commercially, many tech companies are positioning themselves to be leaders in this area, from the processor makers (i.e., NVIDIA, AMD, and Intel), to the game engine builders (especially Unreal Engine [UE] and Unity), and the large Internet companies (e.g., Facebook, which recently changed its name to Meta). From Meta's "Metaverse" to NVIDIA's "Omniverse," significant investments are currently being made. For example, multiple Forbes contributors have projected recently that NVIDIA's market value may eclipse that of Apple's within the next 5 years. These projections are based on the strength of NVIDIA's position in developing hardware and software tools for the Metaverse, as well as the importance of their hardware and software in AI applications.[1,2]

The US Army is currently developing its own VR/AR systems for training and development. Analysts have summarized the following "five ways that synthetic environments can benefit [military forces]"[3]:

1) Enhance collective experimentation, planning, and decision-making.
2) Transform training.
3) Strengthen interoperability across Multi-Domain Operations.
4) Test and investigate new technologies and technical concepts.
5) Better assess readiness across the force.

Under the leadership of the Synthetic Training Environment (STE) cross-functional team, the US Army is funding several projects. In 2019 and 2021, the Army awarded contracts for three related software development projects totaling between $300M and $400M: STE, the Common Synthetic Environment (CSE), and One World Terrain (OWT), with first deliverables expected in the coming months.[4–6] Collectively, these tools are being developed to assist with training and mission planning. For example, OWT will be a high-fidelity mapping tool, similar to commercial products, including Google Earth. OWT will generate realistic models of terrain and buildings so that Soldiers can become familiar with the physical environment prior to performing a mission.

Assuming the Metaverse, VR, and/or AR become widely adopted, all militaries will need to carefully consider how to use these tools to enhance operations and training. Technology can either level the competition or present new opportunities for overmatch. One potential area for overmatch will be physical realism, which is only as good as the underlying simulation technologies. In some cases (i.e., collision detection and rag-doll physics) gaming physics engines are well developed and reasonably accurate; however, in many cases the physics is not yet fully understood.

This technical report presents our efforts to build a simulation environment for windy-city UAS modeling. This targets the Army's fourth motivation objective—to test and investigate new technologies and technical concepts—but could ultimately be extended to the other four points. The city models are based on current OpenStreetMap (OSM) models (e.g., Chicago, Oklahoma City). We then employ lattice Boltzmann methods (LBMs) for simulation of the wind within the cities[7,8] based on historical conditions. State-of-the-art data methods are used to provide localized wind data and visualization within a virtual model of the city in UE, demonstrating how the wind affects the flight path of the UAS.

In Section 2, we describe the software components of our system and include frames taken from the simulations to demonstrate the data visualization. We then conclude with general observations of our approach and a brief discussion on future research.

2. Components of our Synthetic Environment (SE)

In this section, we describe the physics and data manipulation methods used within our simulation. These begin with base city models, obtained from OSM, and end with UAS-wind interactions within our UE-based simulation environment.

We use Python scripts to manipulate OSM city data for building locations, shapes, and heights for our simulation environment. These serve as input geometries of the Atmospheric Boundary Layer Environment (ABLE)–LBM to simulate turbulent air flows around buildings. These same geometries are reused within the simulation/visualization environment.

The ABLE-LBM is a fine-resolution (spatial: meters; temporal: seconds) atmospheric-environment computer model for urban and complex terrain applications. It provides fine-scale turbulent wind, temperature, and pollutant concentrations within an atmospheric boundary layer. Compared with older traditional Navier Stokes models, the ABLE-LBM is more accurate and hundreds of times faster for atmospheric variable predictions because it applies the most recent advances in atmospheric science and graphics processing unit (GPU) computation technology. The real-time prediction goal for a midsize city can be achieved using only a laptop computer. The ABLE-LBM model will also be versatile in taking atmospheric input data from nontraditional sources (e.g., retrieved from UAS or other remotely sensed data) and surface boundary condition data (e.g., cloud point terrain data) to increase agility and accuracy in data-denied or quickly changing situations.

The Climate Data Operators (CDOs) command line utilities were used to convert the simulated wind field FORTRAN binary data format to Network Common Data Form (NetCDF) format. The NetCDF format can be read by many tools (i.e., ParaView, VTK, and Python xarray). ParaView was used to calculate and generate the wind-shear areas and wind streamlines that were converted into 3-D polygons format, which were then imported to UE.

The drone wind visual simulation is built on Epic's UE version 4, an open real-time 3-D creation tool. We have also incorporated Microsoft's AirSim plug-in[9] to simulate the drone flight physics. See Fig. 1.

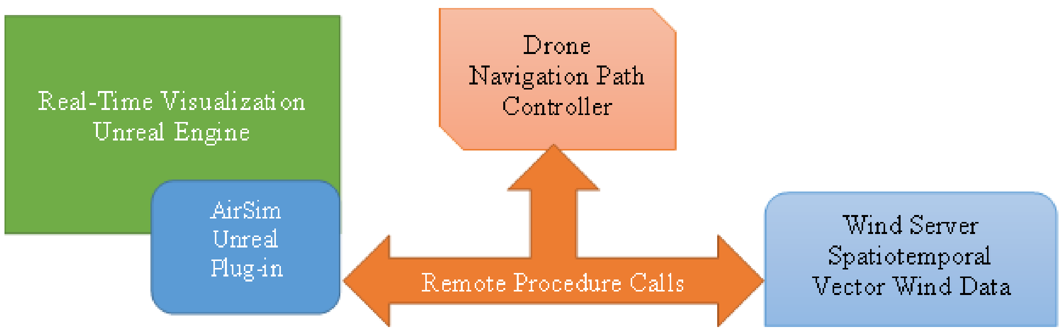

Fig. 1 Wind server components

The UE is started and connected via drone controller to provide the drone a flight path in the virtual environment. The drone spatial and/or temporal information is sent to the wind server and then to the return wind vectors. Next, the wind vectors are sent to the AirSim plug-in, which modifies the drone's physics behavior. The network commutation is handled by Remote Procedure Calls (RPCs) with Message Pack (a binary serialization method). We have also done this test with Real Time Publish Subscribe (RTPS) in the Robot Operating System (ROS2). The urban virtual environment 3-D models are generated from OSM data. An Unreal C++ Component was developed to allow Unreal Actors to communicate with the wind server without requiring the AirSim plug-in.

The wind server is built around a Python xarray[10] package that loads and manages the multidimensional array's wind vector field data. The data is stored in NetCDF[11] file format using the Climate and Forecast (CF) Metadata conventions. The server's design is intentionally modular and distributive to handle large data sets. The wind data grid size (at 3-m resolution) can get very large. A grid at 521 × 311 × 121 points with 100-time steps is around 22 GB. Currently, we use static wind fields based on averages over a set number of time steps or individual time steps. Distributed arrays could potentially be used to support this. A Python xarray can be distributed across many systems using Dask Arrays' spread-out memory requirements. Figures 2 and 3 show samples of the visualization output from the current state of this SE.

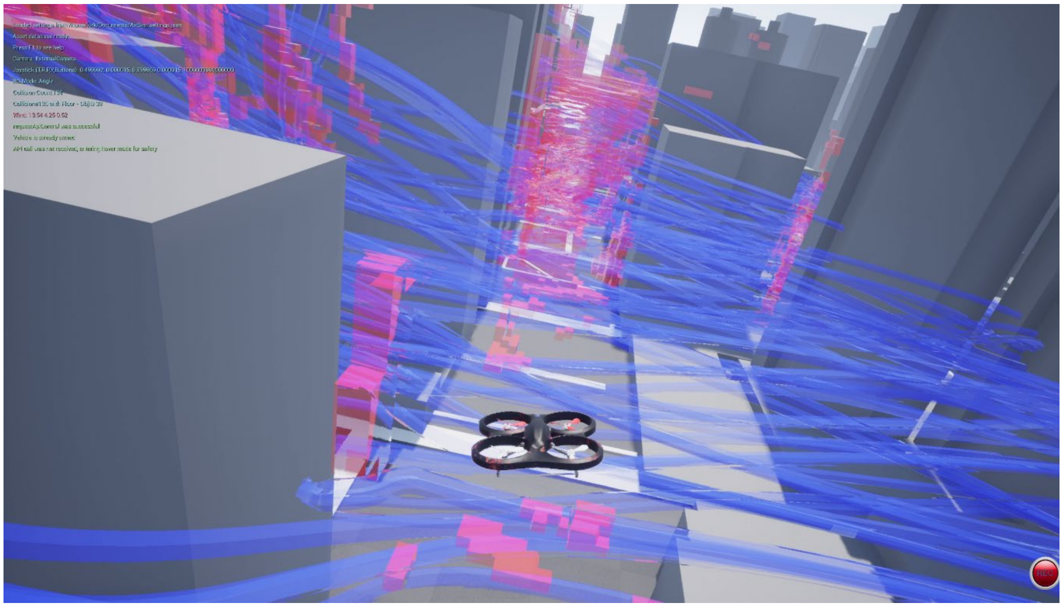

Fig. 2 Realistic wind in SE based on a section of Chicago, Illinois. Glassy stream tubes show wind field streamlines. Red volumetric fog represents regions of high turbulence, as identified by large z-values in the wind fields. UAS response to wind demonstrated by changes in flight path when UAS-wind interactions are toggled on and off.

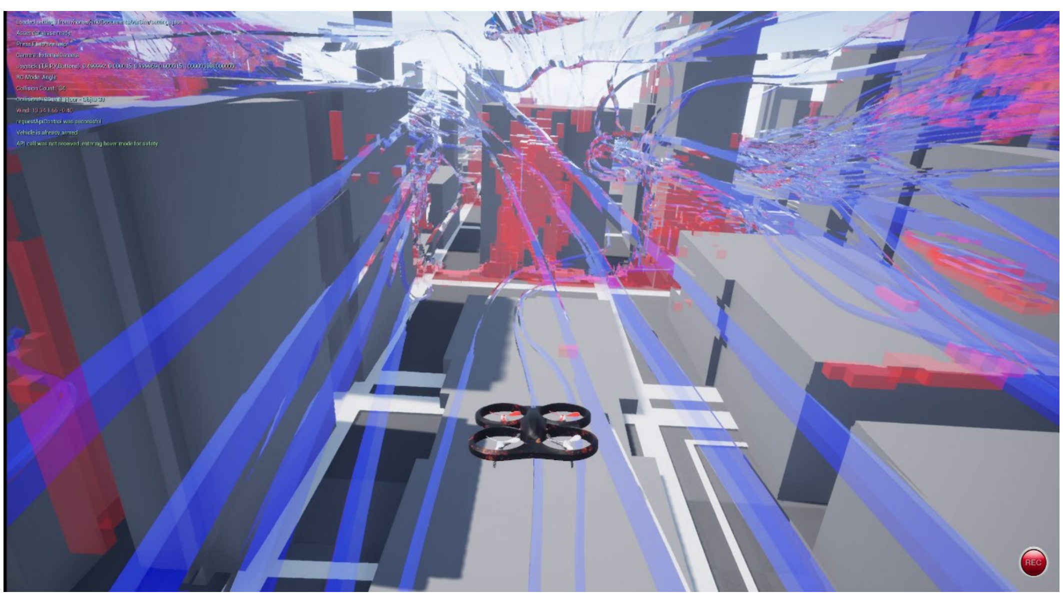

Fig. 3 Visualization wind streamlines and turbulence areas

3. Conclusions and Future Plans

We have demonstrated the use of open-source data and open-source and/or open-access computer tools to create a simulation for UAS-environment interactions in urban settings. Our test system has the capacity to be used in training, planning, and decision-making for UAS operations—balancing safety, area coverage, power usage, location avoidance, and total operational time.

There are several ways in which this system can be improved. Improved UAS-wind interaction models allow for UAS suitability testing of specific locations and conditions. We are currently investigating the integration of temporal 3-D wind fields data into UE as data inputs and graphical animations. Looking into the future, integration with other environments, such as NVIDIA's Omniverse, will require simplifying the connectors used to import data. Finally, multiple methods may be used to train for urban navigation in windy conditions, such as reinforcement learning and improved control methods.

4. References

1. Hackl C. The Metaverse is coming and it's a very big deal. Medium.com; 2020 July 5 [accessed 2022 May 5]. https://medium.com/@CathyHackl/the-metaverse-is-coming-and-its-a-very-big-deal.

2. Kindig B. The key to unlocking the Metaverse is NVIDIA's Omniverse. Forbes.com; 2021 Sep 2 [accessed 2022 May 5]. https://www.forbes.com/sites/bethkindig/2021/09/02/the-key-to-unlocking-the-metaverse-is-nvidias-omniverse.

3. Speranza L, McArdle J. Five ways synthetic environments can benefit NATO. DefenseNews. 2022 [accessed 2022 May 5]. https://www.defensenews.com/opinion/commentary/2022/02/03/five-ways-synthetic-environments-can-benefit-nato/.

4. NSTXL. Common Synthetic Environment (CSE) and One World Terrain (OWT). 2019 [accessed 2022 May 5]. https://nstxl.org/opportunity/common-synthetic-environment-cse-and-one-world-terrain-owt/.

5. Rozman J. The synthetic training environment. The Association of the United States Army (AUSA); 2020 Dec 10 [accessed 2022 May 5]. https://www.ausa.org/publications/synthetic-training-environment.

6. Cole engineering awarded $179M OTA by US Army. AP News. 2021 June 23.

7. Wang Y, MacCall BT, Hocut CM, Zeng X, Fernando HJS. Simulation of stratified flows over a ridge using a lattice Boltzmann model. Environ Fluid Mech. 2020a;20(5):1333–1355.

8. Wang Y, Decker J, Pardyjak ER. Large-eddy simulations of turbulent flows around buildings using the atmospheric boundary layer environment–lattice Boltzmann model (ABLE-LBM). J Appl Meteorol Climatol. 2020b;59(5):885–899.

9. Shah S, Dey D, Lovett C, Kapoor A. AirSim: High-fidelity visual and physical simulation for autonomous vehicles. 2017. arXiv:1705.05065.

10. Hoyer S, Hamman J. xarray: {N-D} labeled arrays and datasets in Python. J Open Res Software. 2017;5(3).

11. netCDF. UCAR/Unidata. http://doi.org/10.5065/D6H70CW6.

List of Symbols, Abbreviations, and Acronyms

3-D	three-dimensional
ABLE	Atmospheric Boundary Layer Environment
AI	artificial intelligence
AR	augmented reality
CDO	Climate Data Operator
CF	Climate and Forecast
CSE	Common Synthetic Environment
GPU	Graphical Processing Unit
LBM	lattice Boltzmann Method
NetCDF	Network Common Data Form
OSM	Open Street Map
OWT	One World Terrain
ROS2	Robot Operating System
RPC	Remote Procedure Call
RTPS	Real Time Publish Subscribe
SE	synthetic environment
STE	Synthetic Training Environment
UAS	unmanned aircraft system
UE	Unreal Engine
VR	virtual reality

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
1 (PDF)	DEVCOM ARL FCDD RLD DCI TECH LIB
3 (PDF)	DEVCOM ARL FCDD RLC ED K KIRK B KRACZEK FCDD RLC EM Y WANG