ERDC/CRREL MP-23-4

Cold Regions Research
and Engineering Laboratory

US Army Corps
of Engineers®
Engineer Research and
Development Center

ERDC
ENGINEER RESEARCH & DEVELOPMENT CENTER

Vehicle Modeling in Unreal Engine 4

F. Sapienza, M. Parker, M. Bodie, and S. A. Shoop

November 2023

Distribution Statement A. Approved for public release: distribution is unlimited.

ERDC/CRREL MP-23-4
November 2023

Vehicle Modeling in Unreal Engine 4

F. Sapienza, M. Parker, M. Bodie, and S. A. Shoop

US Army Engineer Research and Development Center (ERDC)
Cold Regions Research and Engineering Laboratory (CRREL)
72 Lyme Road
Hanover, NH 03755-1290

Final Report

Distribution Statement A. Approved for public release: distribution is unlimited.

Prepared for US Army Corps of Engineers
Washington, DC 20314-1000

Under Cold Regions Effects on Object Recognition Project under the ERDC 6.2 SBG202 Man/Unmanned Vehicle/Terrain Interaction (MUM Vehicle/Terrain) work package, and by the Vehicle Born IED Artificial Intelligence; and Machine Learning Analytics for Cold Mobility projects under the 6.2 Military Engineering Technology work package

Abstract

Vehicle modeling software has presented considerable challenges in properly representing vehicle mobility in extreme conditions. We have recently been developing new vehicle models and scenes in Unreal Engine. Unreal Engine is best known as a video game creation platform focused on graphics and has relatively few options for real world accurate physics modeling. UE4 allows for lots of customization internally or via supplemental C++ code, so this can be mitigated by the addition of various functions to account for different situations a vehicle might be in. We have successfully implemented the following: accurately functioning wheeled vehicles, tracked vehicles, and created simulated and real world environments, downloaded through Geowatch heightmaps. Each environment can have various terrain conditions including soil, rock, snow, and sand applied across its surface. Modeling snow in these environments is of particular interest and recent motion resistance and sinkage models have been integrated into the software to affect graphics and vehicle performance. This new model for vehicle mobility offers an opportunity to improve the physics and graphics of differing terrains especially for winter conditions. The new model also allows for features to be updated and added with ease in the future.

Preface

This study was conducted for the US Army Corps of Engineers Washington, DC, under the Cold Regions Effects on Object Recognition Project under the ERDC 6.2 SBG202 Man/Unmanned Vehicle/Terrain Interaction (MUM Vehicle/Terrain) work package, and by the Vehicle Born IED Artificial Intelligence, and Machine Learning Analytics for Cold Mobility projects under the 6.2 Military Engineering Technology work package.

The work was performed by the US Army Engineer Research and Development Center, Cold Regions Research and Engineering Laboratory (ERDC-CRREL). At the time of publication, the deputy director of ERDC-CRREL was Dr. Ivan P. Beckman, and the director was Dr. Joseph L. Corriveau.

This paper was originally published as F. Sapienza, M. Parker, M. Bodie, and S. A. Shoop, “Vehicle Modeling in Unreal Engine 4,” in *Proceedings of the ISTVS 20th International Conference, Montréal, Québec, Canada,* September 20–23, 2020.

COL Christian Patterson was commander of ERDC, and Dr. David W. Pittman was the director.

This page intentionally left blank.

VEHICLE MODELING IN UNREAL ENGINE 4

1. Background

1.1 MRZR4/D4 Vehicle

The Military is currently expanding its vehicle fleet to include smaller manned and unmanned vehicles. The MRZR is currently used within different branches of the military and is also being researched as a viable robotic combat vehicle. This work aims to expand the modeling of this vehicle in Unreal, the Army's new choice for vehicle modeling software. The D4 variant is a newer diesel powered model. These vehicles are manufactured for their versatility to be used in many different conditions on and off road and can be outfitted to better perform a specific task.

1.2 ANVEL Modeling

ANVEL is a vehicle modeling software developed by Quantum Signal and previously used by CRREL. Ford recently purchased Quantum Signal and the software is no longer available to the military. It offers excellent vehicle physics that allow for extremely accurate vehicle models. A goal for this project was to attain vehicle models at least as accurate or better than those achieved in ANVEL.

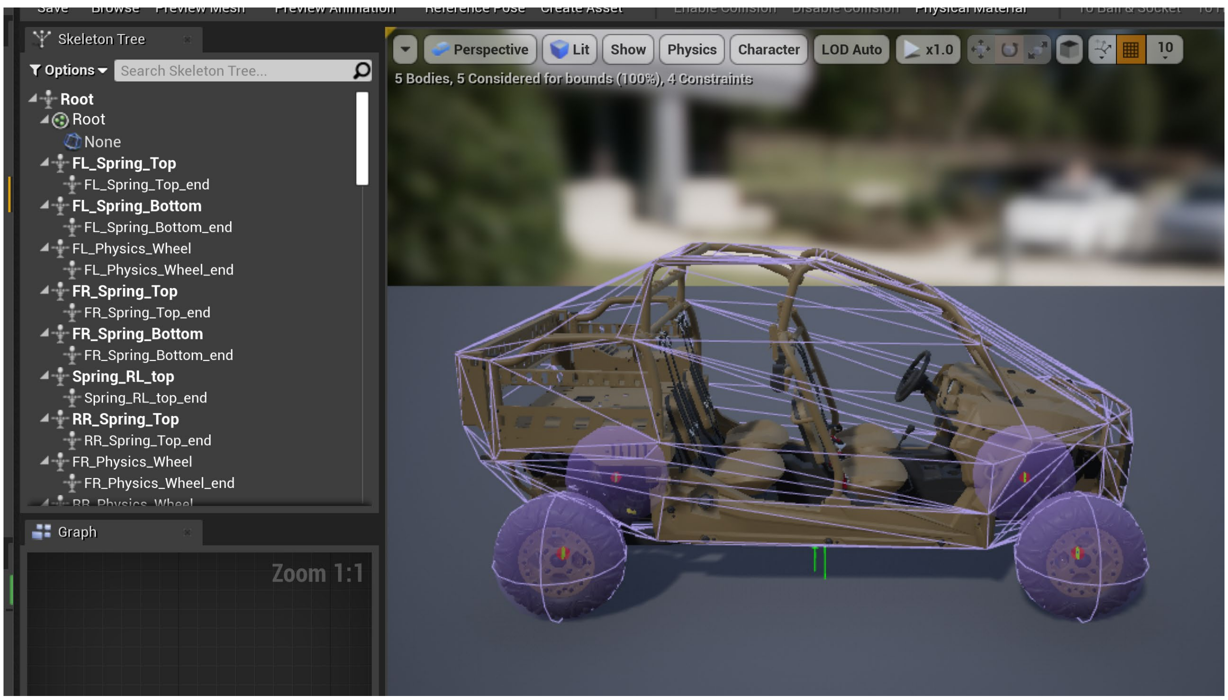
2. Unreal Engine Model

2.1 Modeling

Unreal Engine 4 is an open source platform traditionally used by game developers. As a result, the real world physics are not expansive and we are working to improve them to provide better simulations. Modeling the MRZR4/D4 in Unreal provided the ability to incorporate fine detail in things like vehicle graphics and landscape modeling. UE4 modeling begins with base classes of pawns, landscapes, actors, etc. these classes perform different functions in-game and are programmed with tasks or linked to one another. For vehicles it is useful to start with the basic vehicle class and expand on it to increase its functionality. UE4 uses blueprints to organize tasks to be carried out. There are many pre-programmed functions that allow the blueprints to function like traditional code. When these fail to produce the desired outcome one can write C++ code to create a new blueprint function or write an entire component in C++ if more changes are needed.

The vehicle mesh is built outside of UE4 in a 3D design program. This mesh serves to provide dimensions, visuals and eventually collision boundaries for the simulation. The mesh also functions as a skeleton indicating where forces are applied at various points, or bones, should produce resultant forces.

In the physics asset collision boundaries are generated based on a desired complexity. These boundaries limit component displacement when interacting with the environment. This allows the vehicle to resist obstacles and not sink through the landscape it is on. This is where constraints are set to limit the travel of the wheels to the max steering angle, and the suspension travel of the vehicle is limited by setting a spring constraint. A screen shot of the physics asset construction is shown in Figure 1.



5 Bodies, 5 Considered for bounds (100%), 4 Constraints

Figure 1: MRZR4 Physics Asset Showing Collision Geometry (Purple) and Suspension travel (Green and Red)

Many of the visuals in UE4 are handled by animations. One of these built-in animation functions for vehicles allows the wheels to turn and spin. Unfortunately, these animations do not interact with the environment, so the physical turning of the wheels has to be handled separately in a blueprint if deemed necessary. This proved especially important in the tracked vehicle model that will be explained later.

Landscapes can be either built in UE4 or downloaded in a variety of ways. In this project Geowatch heightmaps were converted to a gray-scale image, imported, and scaled to the proper elevation height (meters). This allowed for very detailed elevations on which different material properties could be assigned to make a realistic surface to drive on. One of these materials is snow which is added on top of the heightmap to a specified depth. The snow material can be seen in Figure 2

Figure 2: Model visual of MRZR4 in snow

Control of the vehicle can be handled either by the user in a simulation type environment allowing realtime adjustments *a priori*, or values for steering, braking, and throttle can be programmed to run a specific course. When programmed, the vehicle is controlled using a combination of throttle, brake and steering commands at various points in the terrain with continuous scale of input values. This means the user has the ability to independently prescribe various amounts of throttle brake and steering simultaneously.

In realtime operation, the debug command allows the user to see the vehicle inputs, vehicle acceleration, and the friction between the current surface and the wheels. In addition to the debug command the vehicle blueprint allows realtime output of roll, pitch, yaw, acceleration in three directions, vehicle center of gravity, and a visual representation of the vehicle collision boundaries (Figure 3).

Future improvements and additions can be made to the model project with little effort thanks to UE4's user interface. Functions can be added, or input values can simply be adjusted with little confusion involved.

2.2 Tracks

The MRZR4/D4 was also modeled with attached Mattracks M3 UR-HD Plus tracks. These tracks were recently purchased for use with the real vehicle and are constructed of a softer rubber compound to allow for better winter performance. The primary purpose of this modeling was to simulate mobility through the snow. UE4's built-in physics engine has many limitations, so this presented some challenges when trying to model the tracks. Most available UE4 tracked vehicle models are not very accurate to the real world and do not react to differing terrains.

In order to preserve accuracy, and implement the track-terrain interaction, the model uses the collision boundaries of the tracks but utilizes fourteen smaller physics wheels that provide locomotion, suspension, and ground interaction. The smaller wheels are positioned such that they provide force against the ground where the track would lie, and independent suspension of each wheel allows the crossing of obstacles in a realistic manner (Figure 3).

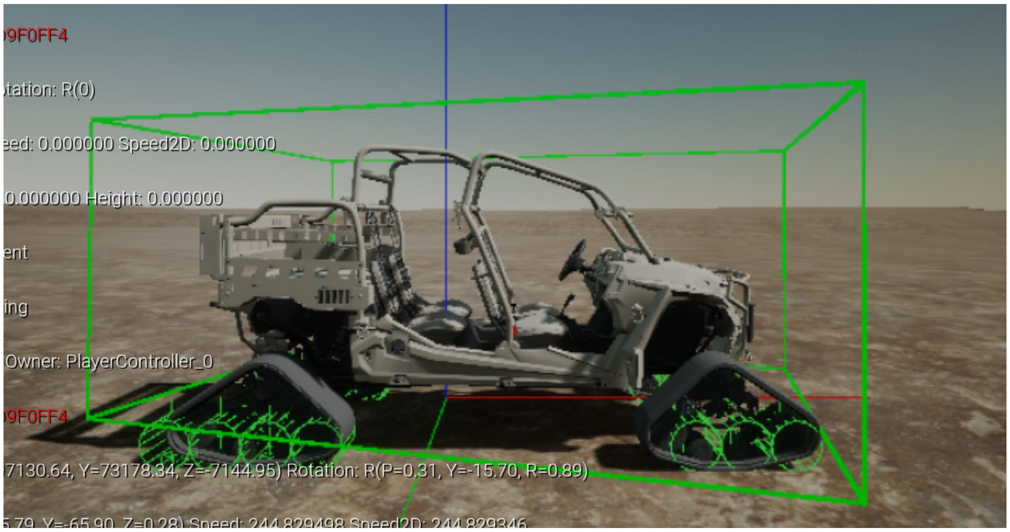


Figure 3: Tracked MRZR4 visual showing the track and wheel contact component and the collision bounding box

In addition to differing friction properties of surface materials the model also allows the changing of tire/track friction properties. The two friction values are multiplied to determine total friction. This is useful for the varying tire or track friction properties to simulate different rubber compounds and produce different vehicle behaviors without having to change the surface properties. For instance, it is expected the track would slip more on snow than asphalt so it should interact differently, and this can be modeled in the track friction asset, assigning different values for different physical materials.

The differential and torque output also had to be adjusted from the standard MRZR4 model. The gearing of the tracks resulted in a shaft output speed three times slower than that of the wheels which is what was used to adjust differential data. Torque data was continually modified until it provided realistic speeds and accelerations as experienced buy the vehicle with the original wheels on it.

2.3 Snow Resistance

Snow resistance is built into the vehicle blueprint and is enabled by triggers set on the landscape. These triggers take the form of standard UE4 trigger volumes. Once the vehicle enters a trigger volume the name is passed to the vehicle blueprint and based on its name a specific effect is enabled. When the vehicle exits the trigger volume the effect is taken away. This allows the application of multiple volumes that give different effects so different snow conditions and heights can be applied in different areas of a landscape.

The snow resistance is applied as a resistive force. The vehicle movement is normalized to get a direction vector and then the calculated force is applied opposite to this movement. If the vehicle is stationary no force is applied. This force can be applied to the main vehicle body or to separate points closer to the wheels to better simulate where the forces act in the field.

The force is calculated by Shoop et al. (2019) formulas for snow resistance and sinkage. Sinkage is calculated using snow height and densities. Using this, motion resistance is calculated using track or tire width and the length of tire/track in contact with the ground. This calculated resistance is applied to the vehicle. The snow resistance model approach can be modified to account for additional terrain conditions like mud or sand as long as those resistance equations are known.

3. Validation

In February of 2020 tests with a tracked MRZR2 were carried out at Keweenaw Research Center (KRC) located in Calumet, Michigan. Data was collected and has been used to validate the tracked MRZR4 UE4 model. The tract of land at KRC where tests were conducted was imported into UE4 and the model was tested in various ways on that landscape. Driver input data, throttle percentage, and steering angle, were used to control the model, while vehicle accelerations, velocity, roll, pitch, and yaw were used to compare the experimental and model vehicle response.

One test carried out at KRC was a traverse. This traverse was modeled in UE4 using measured snow conditions and then compared to the real data. In this model traverse two snow types were used, one for virgin snow and one for precompacted snow from previous runs. After tuning the model to better match the initial dataset, other traverses carried out at KRC were also tested and the model matched them well.

Another test carried out at KRC was on slopes with varied terrain conditions. These tests were used to help determine friction coefficients and torque values for the model. After adjustments from the traverse data the model required little adjustment to match other experimental runs collected at KRC

Figure 4 shows good agreement between the experimental and model vehicle paths when using experimental steering, throttle, and brake data as model inputs. Variations in the path taken could be due to ineffectiveness of the rigid landscape and variation in the snow not accounted for in the limited number of modeled snow types.

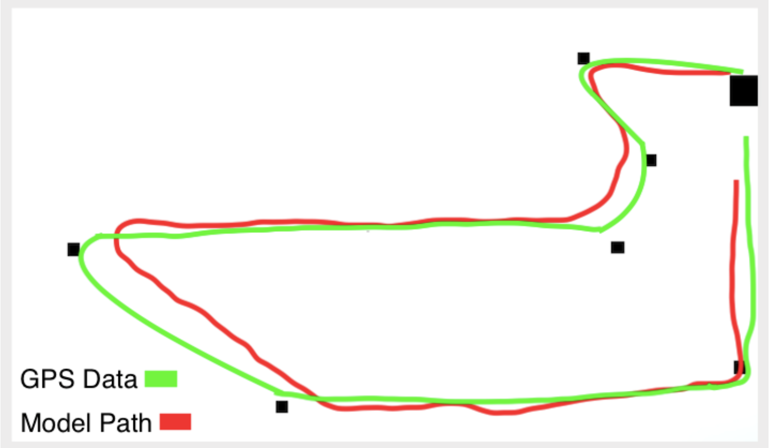


Figure 4: Path comparison between experimental tracked MRZR2 and modeled MRZR4 traverse (approx 130m x 70m)

Figure 5 shows the recorded accelerations of the MRZR and those simulated by the model. The model follows the trend of the measured accelerations but is victim to some spikes in accelerations on turns taken on this traverse. This could be due to issues with the rigid terrain mapping or problems with the suspension modeling.

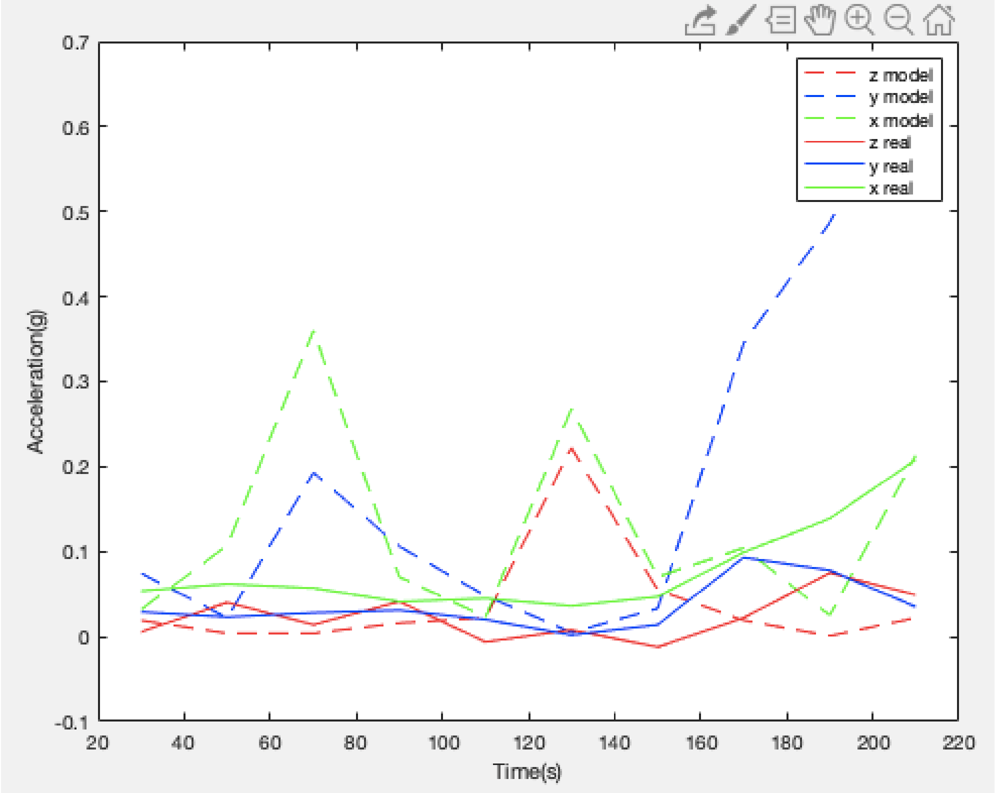


Figure 5: Aceeleration (g's) comparison between experimental and modeled MRZR traverse

In Figure 6 roll and pitch are compared between the real traverse and model. Yaw is excluded from this graph as it is reliant heavily on steering which was input into the model making the comparison almost identical. Roll and pitch react very similarly in the model but produce slightly dampened results for peak values. This could be due to variations in snow that were not modeled at a small scale, and will be explored further in future work.

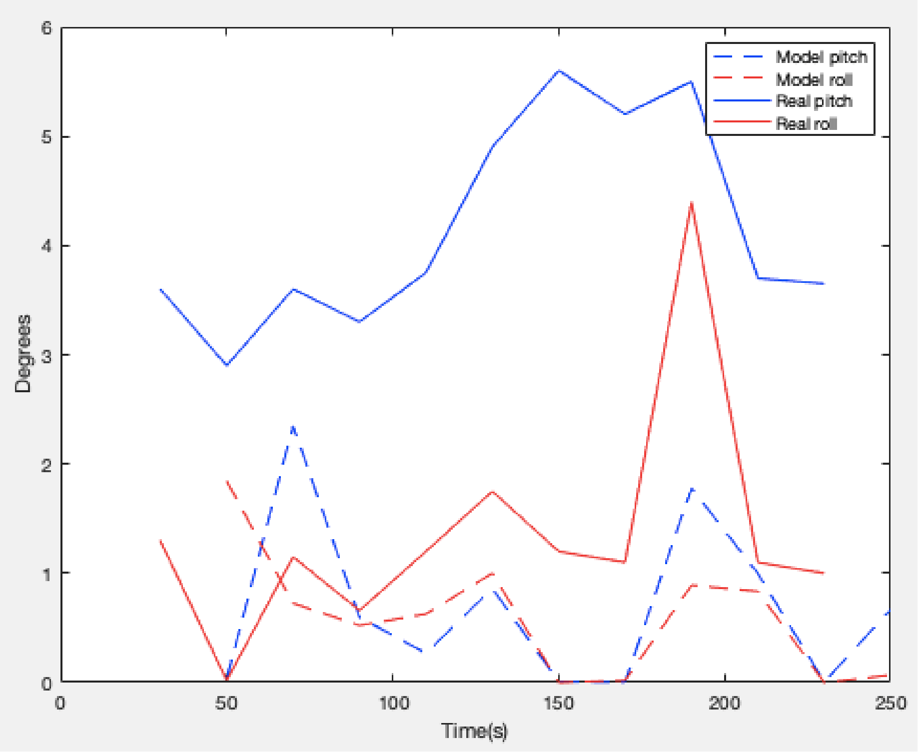


Figure 6: Roll, pitch (degrees) comparison between experimental and modeled tracked MRZR traverse.

4. Recommendations for Improvements

4.1 Proprietary information

The MRZR-D4 is a new military vehicle produced my Polaris. This means most of the information and testing done on it is proprietary and not available. As more testing and data is released for the vehicle the model can be modified to incorporate some of these specifics.

4.2 Testing

As more testing is done on the MRZR4/D4 the model can be compared and improved as well. A tremendous amount of testing has not yet been done on the MRZR-D4. This is because it is not as available to consumers as a standard consumer vehicle and it being a recently released vehicle. During the time of the making of this model the COVID-19 virus has prevented in person testing that can resume in the future.

In addition, most of the testing done with the tracked MRZR at KRC was done over a few days and consistently at lower rpms and speeds. Testing at the higher end of that spectrum could help with attaining an accurate range of values for traction and acceleration.

4.3 Tracks

The track model consisting of multiple wheels functions well, but it requires changes in torque and friction of the tire asset in order to produce realistic results. In a future iteration of the model, more accurate tracks that interface with the ground over a changing, realistic contact area could improve the model.

4.4 Simulator

There are plans to hook this model into a human-in-the-loop motion simulator utilizing a steering wheel, display, gas and brake pedals, etc. in order to have the ability to run built or imported courses in a virtual vehicle rather than having to bring the vehicle and personnel on sight.

5. Conclusions

The military is moving away from proprietary vehicle modeling to open source software such as Unreal Engine This move has allowed for the development of new features integrated into each vehicle model. The physics effects of UE4 are not great, however; this new model does a good job at replicating real world scenarios when properly modified with supplemental physics effects. Tracked and wheeled MRZR modes were achieved and tested well on the newly modeled snow surface. Additional testing and tweaking is needed to achieve a more accurate model. Then this model can be used in a predictive manner on new terrain. This model should be expanded to different vehicles and we are currently working with ERDC-GSL to improve the vehicle terrain interface in order to continue improving vehicle mobility modeling on different terrain conditions.

Nomenclature

a	Length of tire/track in contact with ground	[m]
b	Track or tire width	[m]
CRREL	Cold Regions Research and Engineering Laboratory	
ERDC	Engineer Research and Development Center	
GSL	Geotechnical and Structures Laboratory	
h	Initial snow depth	[m]
KRC	Keweenaw Research Center	
p_o	Initial snow density	[kg/m^3]
p_f	Final snow density	[kg/m^3]
UE4	Unreal Engine 4	
z	Sinkage	[m]

Acknowledgements

We would like to thank the Ground Vehicle Systems Center (GVCS) and the ERDC Geotechnical and Structures Laboratory for their collaboration and help with UE4 modeling. Funding support for this work was provided by Cold Regions Effects on Object Recognition Project under the ERDC 6.2 SBG202 Man/Unmanned Vehicle/Terrain Interaction (MUM Vehicle/Terrain) work package, and by the Vehicle Born IED Artificial Intelligence; and Machine Learning Analytics for Cold Mobility projects under the 6.2 Military Engineering Technology work package.

References

BoredEngineer, 2015. Tanks Tracks and N-Wheeled Vehicles,, https://forums.unrealengine.com/community/work-in-progress/54936-assets-open-source-tanks-tracks-and-n-wheeled-vehicles?view=thread

Milford M., 2016. UE4- How to Make Vehicles, Milford Tutorials

Offworld Defense Simulations, 2019. Enhanced Vehicle Plugin, Unreal Engine Marketplace

Shoop, S., M. Bodie, S. Frankenstein, M. Parker, and M. Bigl. 2019. Recent Improvements to Snow Mobility Algorithms. 15th ISTVS European-African Regional Conference, 13–15 September, Prague, Czech Republic (Paper 82).

REPORT DOCUMENTATION PAGE

1. REPORT DATE November 2023	**2. REPORT TYPE** Final	**3. DATES COVERED** **START DATE**	**END DATE**

4. TITLE AND SUBTITLE
Vehicle Modeling in Unreal Engine 4

5a. CONTRACT NUMBER	**5b. GRANT NUMBER**	**5c. PROGRAM ELEMENT**
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**

6. AUTHOR(S)
F. Sapienza, M. Parker, M. Bodie, and S. A. Shoop

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
US Army Engineer Research and Development Center (ERDC)
Cold Regions Research and Engineering Laboratory (CRREL)
72 Lyme Road
Hanover, NH 03755-1290

8. PERFORMING ORGANIZATION REPORT NUMBER
ERDC/CRREL MP-23-4

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
US Army Corps of Engineers
Washington, DC 20314-1000

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Statement A. Approved for public release: distribution is unlimited.

13. SUPPLEMENTARY NOTES
This paper was originally published as F. Sapienza, M. Parker, M. Bodie, and S. A. Shoop, “Vehicle Modeling in Unreal Engine 4,” in *Proceedings of the ISTVS 20th International Conference, Montréal, Québec, Canada,* September 20–23, 2020.
Funding by Cold Regions Effects on Object Recognition Project under the ERDC 6.2 SBG202 Man/Unmanned Vehicle/Terrain Interaction (MUM Vehicle/Terrain) work package, and by the Vehicle Born IED Artificial Intelligence; and Machine Learning Analytics for Cold Mobility projects under the 6.2 Military Engineering Technology work package.

14. ABSTRACT

Vehicle modeling software has presented considerable challenges in properly representing vehicle mobility in extreme conditions. We have recently been developing new vehicle models and scenes in Unreal Engine. Unreal Engine is best known as a video game creation platform focused on graphics and has relatively few options for real world accurate physics modeling. UE4 allows for lots of customization internally or via supplemental C++ code, so this can be mitigated by the addition of various functions to account for different situations a vehicle might be in. We have successfully implemented the following: accurately functioning wheeled vehicles, tracked vehicles, and created simulated and real world environments, downloaded through Geowatch heightmaps. Each environment can have various terrain conditions including soil, rock, snow, and sand applied across its surface. Modeling snow in these environments is of particular interest and recent motion resistance and sinkage models have been integrated into the software to affect graphics and vehicle performance. This new model for vehicle mobility offers an opportunity to improve the physics and graphics of differing terrains especially for winter conditions. The new model also allows for features to be updated and added with ease in the future.

15. SUBJECT TERMS
Terrain Mobility, Tracked Vehicle, Unreal Engine, Vehicle Modeling

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**
a. REPORT Unclassified	**b. ABSTRACT** Unclassified	**C. THIS PAGE** Unclassified	SAR	16

19a. NAME OF RESPONSIBLE PERSON	**19b. TELEPHONE NUMBER (include area code)**

STANDARD FORM 298 (REV. 5/2020)
PREVIOUS EDITION IS OBSOLETE. *Prescribed by ANSI Std. Z39.18*