**2019 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM**
**MODELING & SIMULATION, TEST & VALIDATION TECHNICAL SESSION**
**AUGUST 13-15, 2019 - NOVI, MICHIGAN**

# Are M&S Tools Ready for Assessing Off-road Mobility of Autonomous Vehicles?

**Michael Cole[1], Cesar Lucas[1], Kumar B Kulkarni[1], Daniel Carruth, PhD[2], Christopher Hudson[2], and Paramsothy Jayakumar, PhD[1]**

[1]US Army Combat Capabilities Development Command
Ground Vehicle Systems Center, Warren, MI
[2]Center for Advanced Vehicular Systems
Mississippi State University, Starkville, MS

## ABSTRACT

*Autonomous systems are the future of the Army and Ground Vehicle Systems Center has aligned itself accordingly to support unmanned ground vehicle (UGV) development. Physically testing autonomous algorithms and vehicle systems can be expensive and time consuming, a problem addressed by the use of modeling and simulation (M&S) tools. A multitude of both Government owned and Commercial Off-the-Shelf Tools (COTS) are widely available, all claim to virtually evaluate autonomous ground vehicles operating on various environments and scenarios. Most of the COTS tools primarily focus on the commercial automotive industry where vehicles are driven in a structured environment. In this paper two M&S tools, viz., Autonomous Navigation Virtual Environment Laboratory (ANVEL) and Rover Analysis Modeling and Simulation (ROAMS) are evaluated for military applications, where the demands for navigation include both on-road and off-road, as well as both structured and unstructured environments as a preliminary benchmark.*

**Citation:** Michael Cole, Cesar Lucas, Kumar B Kulkarni, Daniel Carruth, Christopher Hudson, P. Jayakumar; "Are M&S Tools Ready for Assessing Off-road Mobility of Autonomous Vehicles?", G*round Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 13-15, 2019.

## 1. INTRODUCTION

Autonomous systems are the future of the Army and Ground Vehicle Systems Center has aligned itself accordingly to support unmanned ground vehicle (UGV) development. Physically testing autonomous algorithms and vehicle systems can be expensive and time consuming, a problem addressed by the use of modeling and simulation (M&S) tools. A multitude of both Government owned and Commercial Off-the- Shelf Tools (COTS) are widely available, all claim to virtually evaluate autonomous ground vehicles operating on various environments and scenarios. Most of the COTS tools primarily focus on the commercial automotive industry where vehicles are driven in a structured environment. In this paper two M&S tools, viz., Autonomous Navigation Virtual Environment Laboratory (ANVEL) and Rover Analysis Modeling and Simulation (ROAMS) are evaluated for military applications, where the demands for navigation include both on-road and off-road, as well as both structured and unstructured environments as a preliminary benchmark.

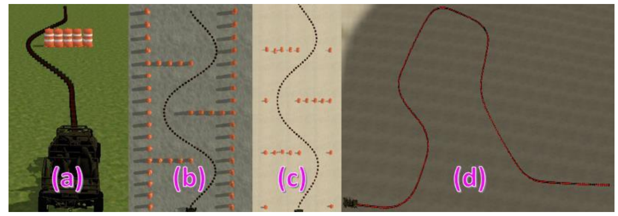
This paper is also in support of the ongoing NATO Science & Technology Organization (STO) Applied Vehicle Technology (AVT) Exploratory Team (AVT-ET-194) to develop mobility assessment methods of autonomous military ground vehicles leading to benchmark of software tools under the activity titled “Mobility Assessment Methods and Tools for Autonomous Military Ground Systems.” While the NATO effort develops capability requirements of M&S tools, this paper will compare a set of those capabilities that are essential to modeling and simulation of autonomous military vehicles. Examples include the fidelity of physics-based models for vehicle dynamics, for sensors, and for (off-road) terramechanics, availability of algorithms for perception, planning, and control, virtual environments, interoperability with open source tools, and real time performance.

A single autonomous platform, a Polaris MRZR4, was used across this benchmark for consistency. A set of three different autonomous navigation scenarios comprising of straight line driving around an obstacle, navigating through two (7m and 10m) slalom courses to a goal, and driving on a winding path with sharp turns (Figures 1a-d) on three different soil types were used for this benchmark. Experimental data collected on the vehicle in these off-road environments is used to perform verification and validation of M&S tools.

## 2. BACKGROUND

Especially in recent years, research related to modeling and simulation of autonomous ground vehicles has become more common. While commercial automotive simulation tools exist to simulate passenger vehicles, those tools are not situated well for military applications that include soft soil and uneven terrain found in unstructured environments.

*Figure 1 Autonomous navigation courses (a) Straight line (b) 7m Slalom (c) 10m Slalom (d) Complex traversal*




To address the shortcomings of commercial autonomous vehicle simulators, several military-specific tools have been developed such as ANVEL [1] and the Virtual Autonomous Navigation Environment (VANE) [2]. These tools support military vehicle models, sensor models such as cameras and LIDAR, and off-road terrain modeling capabilities. Outside these specific codes, many other tools exist that allow the user to build a simulation environment for UGVs using a modular approach such as Gazebo [3], Pre-Scan [4], and USARSim [5].

However, it is not clear which tool would be optimal for simulating autonomous military ground vehicles in an off-road environment due to a lack of comparisons of these types of tools in the literature. Past research has highlighted comparisons between tools for the M&S of mobile robots but without an emphasis on military applications.

For example, [6] compares eleven open source and commercially available robotic simulators such as Gazebo, MissionLab, and Webots. Since only one tool surveyed was exercised in simulation in the study using an example of trajectory planning and tracking as a case study, it is unclear if the tools presented are suitable for a military UGV application.

Another study [7], exercised three popular tools for mobile robot simulations. However, the presented case study simulated a small indoor robot travelling at very low speeds, so assessments of the tools capabilities to accurately simulate a medium sized military vehicle at medium or high speeds cannot easily be established.

Eleven robotic simulation development environments were studied in [8] looking at strengths and weaknesses of each tool. The emphasis of the paper was on high level features such as platform requirements, architecture of the tool, programming environment, usability, and impact on the robotics community. While these are important attributes, no mentions of military applications were observed in the work.

Two dozen autonomous vehicle simulation tools were examined in [9]. However, the focus of the study was on the capability of the software to model sensors such as GPS, LIDAR, and RADAR for autonomous vehicle operation and not military specific requirements like vehicle dynamics or off-road terrain modeling.

This paper begins to address the gap in the literature with a preliminary benchmark of two tools for autonomous military vehicle operations.

## 3. M&S TOOLS OVERVIEW

While many tools exist for modeling and simulation, few are capable of simulating autonomous vehicles with minimal dependencies on third party modules to conduct a closed loop simulation. The key features required for an M&S tool used in this study include the ability to accurately model vehicle dynamics, tire and soft soil interaction, sensor models for LIDAR, built in autonomous navigation algorithms, and interfaces to co-simulate with third party algorithms if the tool does not come bundled with them already. Two tools with very different purposes were used in this study as a preliminary autonomy M&S tool benchmark. ANVEL is based on game engine technology allowing users to quickly and easily model autonomous vehicles in various environments. ROAMS was developed for running very high-fidelity simulations to study how spacefaring rovers might perform on other planets. Both tools offer key features for M&S of autonomous vehicles.

### *3.1. ANVEL*

ANVEL is a customizable M&S tool specializing in autonomous ground vehicles technologies and applications. ANVEL allows modeling a wide collection of variables that are involved in vehicle operation, including mobility, detection and analyses of the surrounding environment through sensors, and controlling logic that guides a vehicle through a mission. It facilitates users to build virtual test environments, manipulate key parameters of sub-systems, terrains, sensors, and interactively test vehicle models under a variety of conditions. ANVEL leverages several key technologies not common to traditional M&S tools, including techniques from the commercial video-game industry, enabling users to easily add details to a terrain including grass, mud, sand, snow, trees, buildings, etc. to create a rich 3D environment. ANVEL was developed under contract for the US Army Engineer Research and Development Center (ERDC) to model and test autonomous navigation suites and algorithms specific robots and vehicles. This application is open source to the government agencies [10]. A desktop user interface is shown in Figure 2a.

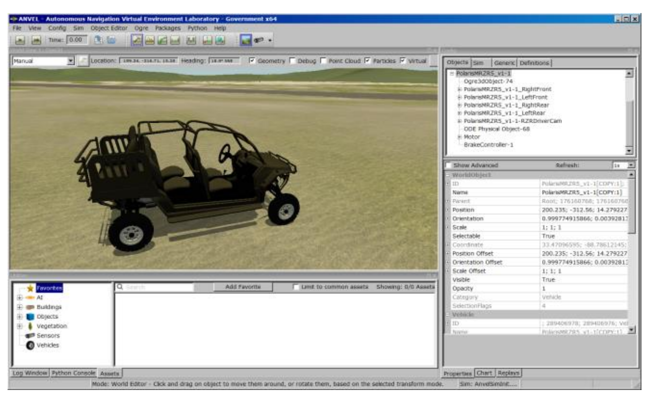


*Figure 2a. Desktop user interface of ANVEL showing various panels including MRZR model in the world view pane*

ANVEL relies on Open Dynamics Engine (ODE [11]), an open source high performance library for simulating rigid body dynamics. It is fully featured, stable, mature and platform independent with an easy to use C/C++ API. ODE not only enables definitions of vehicles, objects and environments within ANVEL but also collision detection. For graphics it relies on OGRE (Object-Oriented Graphics Rendering Engine) [12] an open source scene oriented flexible 3D engine. Albeit a game engine, it's class library abstracts all the details of using the underlying system libraries like Direct3D and OpenGL.

### *3.2. ROAMS*

ROAMS [13] is a high fidelity physics based simulation tool used to analyze, design, develop, test, and operate rovers for planetary surface exploration missions. ROAMS is a modular simulation framework for system engineering studies, technology development, and mission operations. ROAMS provides a simulation framework to facilitate its use by planetary exploration missions for studies in engineering, development of new technology, and for mission operation teams. ROAMS is an extension of the multi-mission dynamics engine DARTS (Dynamics Algorithms for Real-Time Simulation) and DSHELL spacecraft simulation toolkit, capable of modeling vehicle dynamics, sensors and actuators. Terrain modeling is handled by the SimScape module, allowing the user to define a custom terrain or import extremely large digital elevation models (DEM) into ROAMS.

ROAMS provides a number of high-fidelity models for various types of surface rovers. The modularity allows a user to configure the simulation and rover for various needs and fidelity. Different vehicles, sensors, terrain types, and navigation modes are available and can be configured for a specific simulation. A python interface is available which gives the user ability to customize the simulation tool and interface with external software. Graphical rendering is performed using OGRE. A desktop user interface of ROAMS is presented in Figure 2b.

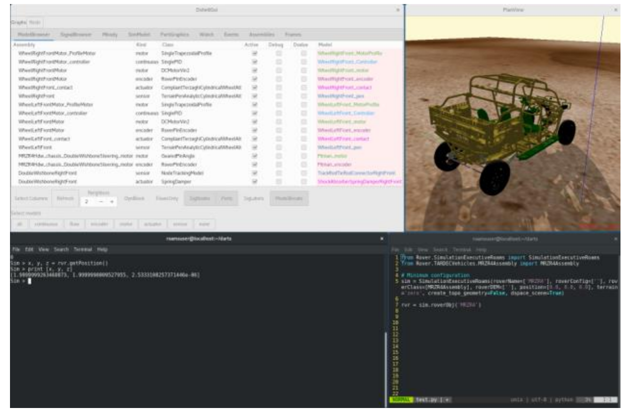


*Figure 2b Desktop user interface of ROAMS*

### *3.3. Tool Feature Comparison*

A brief comparison of high level tool features is presented in Table 1.

*Table 1 M&S Tool Features*

| # | M&S Tool Feature | ANVEL | ROAMS |
|---|---|---|---|
| 1 | API interface | Python, C++ | Python |
| 2 | Sensors | Built-in LIDAR, camera | Built-in LIDAR, camera |
| 3 | ROS compatibility | Supported | Supported |
| 4 | Dynamics engine | ODE | DARTS |
| 5 | Rendering engine | OGRE | OGRE |

## 4. MODELS OVERVIEW

A Polaris MRZR 4 was used in physical tests and this M&S benchmark for consistency. The MRZR is a military-grade vehicle with all-wheel drive, off-road suspension, and numerous configuration options for the Warfighter. As part of the benchmark study, it was desired to have the models built as accurately as possible and have these models validated against known physical test data. Vehicle mass and inertia, suspension and steering characteristics, suspension kinematics and compliance (K&C) measurements were conducted [14] to inform the dynamics models. A total of ten different tests were conducted, viz., front and rear bounce, front and rear roll, front and rear lateral compliance, front steering and rear aligning moment compliance, longitudinal lift/squat and steering ratio tests. This data was not only essential to build lower fidelity dynamic model in ANVEL but also was useful in correlating the higher fidelity ROAMS dynamic model's steering and suspension kinematics. Suitable tire characterization was also conducted to facilitate modeling of the DWT AT26x9-14 tire. A Pacejka tire model [15] constructed in ADAMS was suitably modified to be used in ANVEL and ROAMS models.

### 4.1. MRZR Properties for M&S

Mass, inertia and suspension properties were defined from the data obtained from static K&C tests [14]. Figure 3 shows the overall dimensions of the Polaris MRZR vehicle. Table 2-4 shows important properties used to construct the MRZR model.


*Figure 3 Polaris MRZR®4 showing overall dimensions*

*Table 2 Vehicle overall dimensions from the measured test data*

| # | Distance | (in) | (m) |
|---|---|---|---|
| 1 | Front tract width | 51.25 | 1.302 |
| 2 | Rear track width | 52.7 | 1.339 |
| 3 | Average track width | 52 | 1.32 |
| 4 | Wheel base | 107.2 | 2.723 |
| 5 | Longitudinal CG | 59.69 | 1.516 |
| 6 | Lateral CG | 0.76 | 0.019 |
| 7 | Vertical CG | 29.13 | 0.724 |

*Table 3 Weights included in the model from the measured test data*

| # | Weights | TOTAL | | Sprung | |
|---|---|---|---|---|---|
| | | (lb) | (kg) | (lb) | (kg) |
| 1 | Vehicle | 2816 | 1280 | 2503 | 1141 |
| 2 | Front Left | 619.3 | 281.5 | 551.9 | 250.9 |
| 3 | Front Right | 628.6 | 285.7 | 560.2 | 254.6 |
| 4 | Rear Left | 747.5 | 339.8 | 666.2 | 302.8 |
| 5 | Rear Right | 820.1 | 372.8 | 730.9 | 332.2 |
| 6 | Rim | 28.6 | 13.0 | | |
| 7 | Tire | 34.54 | 15.7 | | |

*Table 4 Suspension characteristics obtained from K&C test data*

| # | Susp. Stiffness | Left | | Right | |
|---|---|---|---|---|---|
| | | lb/in | kN/m | lb/in | kN/m |
| 1 | Front Vertical | 74.2 | 13.0 | 70.64 | 12.4 |
| 2 | Rear Vertical | 140.3 | 24.6 | 130.4 | 22.9 |

The suspension model is constructed using the stiffness values presented in Table 4, which are the slopes of second order polynomial curve fits from the quasi-static bounce and rebound tests conducted to fully characterize the suspension systems. Damping is modeled from the shock test data shown in Table 5. Table 6 contains the measured values from K&C tests, roll and auxiliary roll stiffness for the MRZR vehicle.

*Table 5 Test data obtained to model suspension damping*

| Shock absorber | Velocity | | Force | |
|---|---|---|---|---|
| | [m/s] | [mph] | [kN] | [lbf] |
| Jounce | -2.04 | -4.56 | -20.78 | -4672 |
| Rebound | 2.05 | 4.59 | 6.93 | 1558 |

*Table 6 Roll and Tire stiffness from test data*

| # | Stiffness | lb/in | kN/m |
|---|---|---|---|
| 1 | Front Roll | 1768.2 | 11.45 |
| 2 | Rear Roll | 4650.9 | 30.12 |
| 3 | Front Aux. Roll | 119.3 | 0.77 |
| 4 | Rear Aux. Roll | 2389.2 | 15.47 |
| 5 | Overall Roll | 8927.6 | 57.82 |
| 6 | Tire vertical | 983.2 | 172.25 |
| 7 | Ride stiffness (Frt) | 67.5 | 11.82 |
| 8 | Ride stiffness (Rr) | 119.0 | 20.85 |

Polaris supplied the Power and Torque vs. RPM as shown in Figure 4.

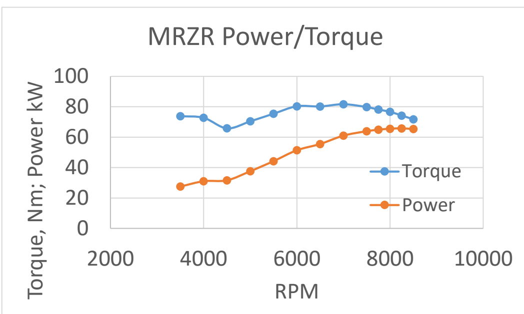



*Figure 4 Polaris MRZR RPM vs. Torque*

### 4.2. ANVEL Model

The ANVEL vehicle model is a low-fidelity generic vehicle with a main body and four wheels. Mass, inertia and suspension properties are defined from the data obtained from static tests [14]. Damping rate is suitably adjusted to wheel rates using measured motion ratios and CAD data. Vehicle center of gravity was adjusted such that its height above the ground matches what was measured in the test. This was accomplished by estimating the suspension travel by the body's own weight when the vehicle is set on the ground. Suspension travel at each of the four corners can simply be estimated as downward force due to sprung mass divided by the corresponding vertical stiffness.

ANVEL does not allow different values for roll stiffness to be defined for front and rear of a vehicle. However, a single value of overall roll stiffness is included by considering the measured roll and auxiliary roll stiffness for both front and rear to be in parallel. This was further tuned by matching the roll angle measured from the dynamic circle test.

A generic drivetrain model was used which included a transfer case, front and rear axle differentials. In lieu of top speed data, gear ratios were adjusted such that the vehicle model's maximum speed approximately matched the speed listed in the manufacturer's brochure (60mph) [16]. Table 7 contains the gear ratios of the drivetrain model.

*Table 7 Gear ratios of the drivetrain used in the model*

| # | Drivetrain component | Gear Ratio |
|---|---|---|
| 1 | Gear -Low | 2 |
| 2 | Gear - High | 1 |
| 3 | Shift Speed | 5 m/s |
| 4 | Front differential | 10.37 |
| 5 | Rear Differential | 10.37 |

### 4.3. ROAMS Model

The vehicle model in ROAMS is modeled as a multibody system. Individual bodies are modeled for the vehicle chassis, tires, and suspension components. Joints are used to connect each body to represent the double wishbone and trailing arm suspension setup of the vehicle on the front and rear axles, respectively. An MRZR4 CAD model was used to identify the joint locations of the vehicle model. Characterization data for the vehicle's mass, inertia, suspension, and tire properties in Section 4.1 was used to build the dynamics model. Component spring, damper, and anti-roll bar stiffnesses were used. Throttle and braking inputs are modeled as positive or negative torque applied at all four wheels, respectively.

### 4.4. Models Comparison

A brief comparison between the models from the two codes are presented in Table 8.

*Table 8 Comparison between ANVEL and ROAMS vehicle models*

| # | M&S Model Features | ANVEL | ROAMS |
|---|---|---|---|
| 1 | Degrees of freedom | 16 | 51 |
| 2 | Body definition | Lumped mass approach | Multibody approach |
| 3 | Suspension definition | Simplified vertical spring /damper at each corner | Planar double wishbone (front) Trailing arm (rear) Inclined spring/damper at each corner |
| 4 | Anti-roll bar definition | Combined aux roll stiffness | Multibody anti-roll bar model on rear axle |

## 5. MODEL VERIFICATION & VALIDATION

In addition to physical K&C tests, basic handling and stability tests such as constant radius, dropped throttle, constant steer, pulse steer, NATO double lane change and straight line braking were conducted [17]. All of the steering maneuver tests were conducted on dry flat pavement. Of the dynamic tests, three were run in both simulation tools to ensure accurate vehicle models were being used in this benchmark.

### *5.1. Constant Radius Test*

The constant radius (100ft) test was conducted to assess the steady-state handling performance of the vehicle. The physical test involved driving the test vehicle around a circular path of 100ft radius from a very low speed up to a speed needed to achieve a nominal lateral acceleration of about 0.5g. Both clockwise and counter-clockwise directions were tested. Metrics like vehicle speed, steering wheel angle, chassis roll angle, lateral acceleration and yaw rate were recorded during the testing. Figure 5-6 shows comparison plots of two measured quantities from physical tests as well as from ANVEL and ROAMS simulations (only clockwise data shown).

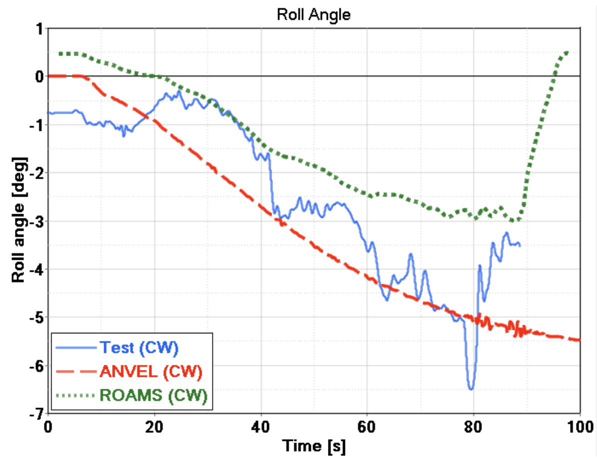



*Figure 5 Roll angle comparison for constant radius test (Only clockwise data shown)*

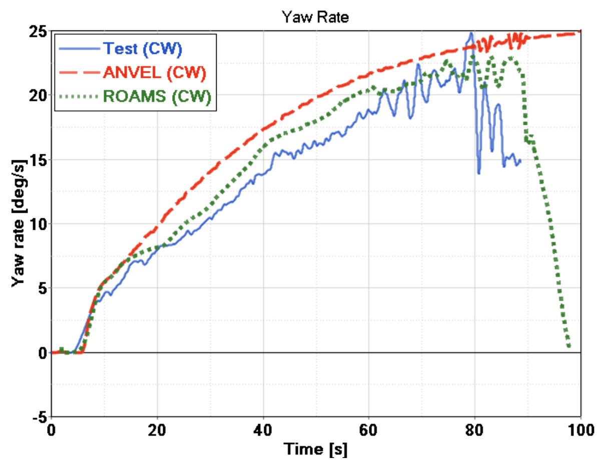



*Figure 6 Yaw rate comparison for constant radius test (Only clockwise data shown)*

### *5.2. Pulse Steer Test*

This test was conducted to identify any excessive vehicle response oscillations. Also, this test is a good indicator for frequency response of the vehicle for the steering input. A rapid triangular pulse steering angle input (maximum of 90 degrees) while the vehicle is driven along a straight path. The rate at which the steering inputs are provided for both ramp up and ramp down duration is 500 degrees/sec. This test was conducted at nominal vehicle speeds of 20 and 30 mph. Metrics like vehicle speed, steering wheel angle, roll angle, lateral acceleration and yaw rate were recorded during testing. Figure 7-8 shows comparison plots of two measured quantities from physical tests as well as from ANVEL and ROAMS simulations (only 30 mph, right turn data shown).

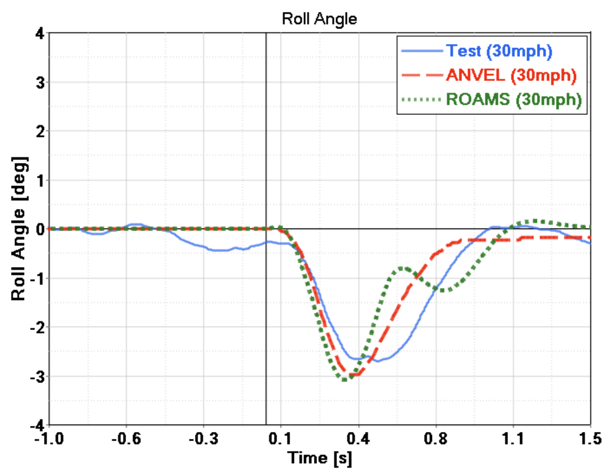



*Figure 7 Roll angle comparison for pulse steer test*

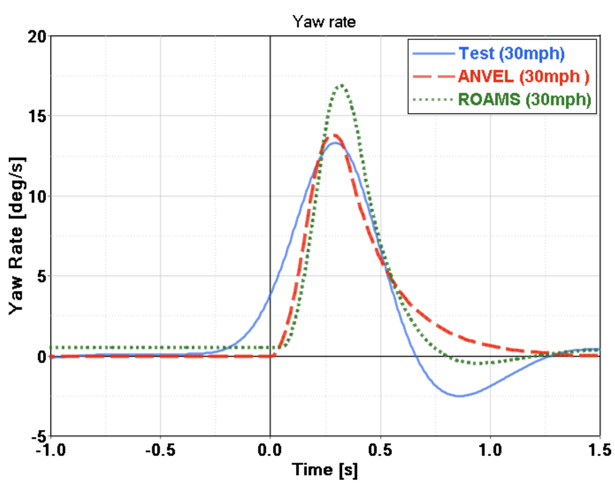



*Figure 8 Yaw rate comparison for pulse steer test*

### 5.3. NATO Double Lane Change Maneuver

This test was conducted to evaluate the vehicle's dynamic response to steering inputs at high speeds and simulates obstacle avoidance maneuvers. Figure 9a-b shows the lane track layout for the event based on NATO Allied Vehicle Testing Publication, AVTP:03-160 W [18]. Appropriate

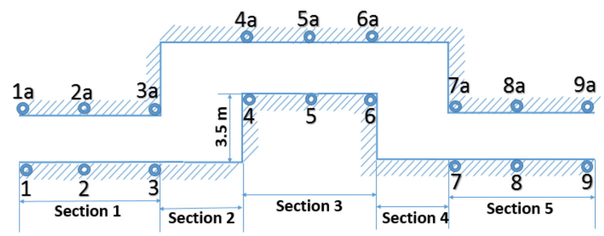



*Figure 9a NATO double lane change track layout*

| Section/s | Length, m | Width, m |
|---|---|---|
| 1, 5 | 15 | 1.1 x Vehicle width + 0.25 |
| 2, 4 | Overall length of vehicle* + 24 | 1.2 x Vehicle width + 3.5 |
| 3 | 25 | 1.2 x Vehicle width + 0.25 |
| * Overall length of vehicle is measured at 0.5m from the ground | | |

*Figure 9b NATO double lane change track dimensions*

vehicle length and width measurements were taken from the MRZR test vehicle and cone placements were made accordingly. Tests were conducted at 20, 30, and 40 mph vehicle speeds.

Metrics like vehicle speed, steering wheel angle, roll angle, lateral acceleration, and yaw rate were recorded during testing. Figure 10-11 shows comparison plots of two measured quantities from physical tests and as well as from ANVEL and ROAMS simulations (only 30mph data shown).

### 5.4. Model V&V Summary

From Figures 5-8, and Figures 10-11, it can be seen that simulation results from both software match closely with those from physical tests. For the constant radius test, roll angle and yaw rate correlate reasonably with the test, with the simulations results at the upper and lower bounds of the test data. For both the pulse steer and double lane change tests, the models' dynamic response to a transient steering input showed similar behavior to the test vehicle's response. For the pulse steer test, the models showed accurate roll angles and slightly higher yaw rates compared to test. The double lane change results for lateral acceleration and steering angle indicate properly modeled tire characteristics and steering systems, respectively.

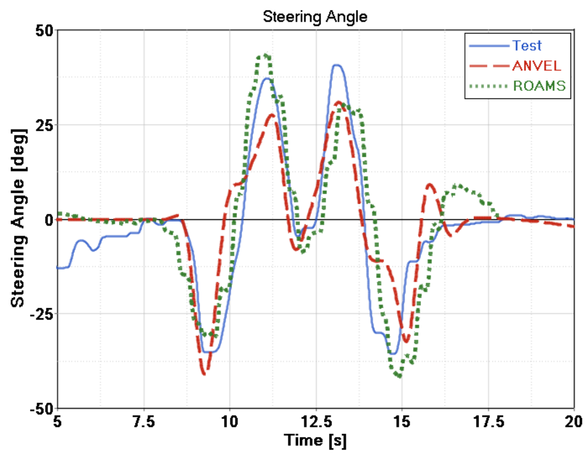



*Figure 10 Steering angle comparison between test and simulations for 30mph NATO double lane change maneuver*

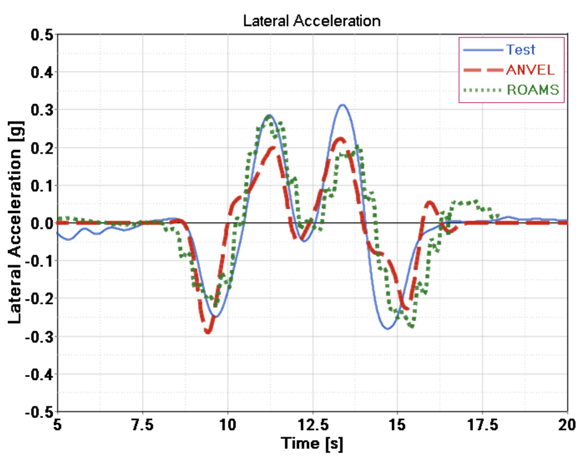



*Figure 11 Lateral acceleration comparison between test and simulations*

Table 9 is a summary comparison of various dynamic response quantities from these tests and simulations. It is a subjective comparison where engineering judgement has been used to determine the ratings as shown. A key to the rating is also presented. These tests identify how well the mass, suspension, steering system, and tire properties of the test vehicle are captured in the simulation models. This step to analyze the simulation tools' ability to model accurate vehicle dynamics was essential for our objective of benchmarking autonomy simulation capabilities.

*Table 9 Summary of dynamic test and simulation results subjective comparison*

| ANVEL | | | | |
|---|---|---|---|---|
| # | Test / Response | Const radius | Pulse steer | NATO double lane change |
| 1 | Steering wheel angle | ↑ | ↑ | ↑ |
| 2 | Vehicle speed | → | ↑ | ↑ |
| 3 | Roll angle | ↗ | ↗ | ↘ |
| 4 | Lateral accl. | ↗ | ↗ | ↑ |
| 5 | Yaw rate | → | ↑ | ↗ |
| 6 | Path followed | NA | | ↑ |

| ROAMS | | | | |
|---|---|---|---|---|
| # | Test / Response | Const radius | Pulse steer | NATO double lane change |
| 1 | Steering wheel angle | ↑ | ↑ | ↑ |
| 2 | Vehicle speed | → | ↑ | ↗ |
| 3 | Roll angle | → | → | ↘ |
| 4 | Lateral accl. | → | → | ↑ |
| 5 | Yaw rate | → | ↑ | ↗ |
| 6 | Path followed | NA | | ↗ |

| From | To | Rating |
|---|---|---|
| 80 | 100 | ↑ |
| 60 | 80 | ↗ |
| 40 | 60 | → |
| 20 | 40 | ↘ |
| 0 | 20 | ↓ |

## 6. OFF-ROAD SOIL MODELING

A key simulation component for military UGV applications is soft soil modeling for sand or clay based terrains. Effects like soil moisture content have a large impact on vehicle performance and must be captured in simulation when driving in soft soils. For both tools, off-road terrain properties are defined by the Bekker-Wong [19] soil model which represents the pressure-sinkage relationship of the tire and soil interaction using Equation 1

$$p = \left(\frac{k_c}{b} + k_\phi\right) z^n \qquad (1)$$

where p is the pressure, $k_c$ is the soil cohesion modulus, b is the width of the tire, $k_\phi$ is the friction modulus, z is the tire sinkage, and n is the exponent coefficient. Values for these parameters were determined from field tests.

While the simulation tools use the Bekker-Wong model, the soil property field measurements taken during physical testing were generated using the cone index system. Therefore, in order to produce the parameters required for the Bekker-Wong model, the cone index data was converted [20]. In order to generate six parameters from a cone index measurement, the test team (1) assumed typical values for some parameters based on historically known values, (2) used cohesion and friction modulus relationships with cone index based on empirical testing [21], and (3) iteratively solved equations by Janosi relating cone index to Bekker-Wong variables [22]. The various parameters for the soil model for the three different terrains used in simulation is shown in Table 11.

*Table 11 Bekker-Wong parameters for soft soils*

| Parameters | Sand | Clay | Grass | Unit |
|---|---|---|---|---|
| n | 1 | 0.046 | 0.355 | dimensionless |
| kc | 0 | 16.1 | 109.3 | lb/in^(n+1) |
| kϕ | 11.2 | 40.6 | 122.6 | lb/in^(n+2) |
| K | 0.8 | 0.5 | 0.3 | inches |
| c | 0.15 | 16.05 | 0.9 | psi |
| ϕ | 0.49 | 0.00 | 0.40 | degrees |

## 7. RTK PATH PLANNING ALGORITHMS

US Army's Ground Vehicle Systems Center (GVSC) has developed a suite of robotics tools to facilitate autonomous navigation of various types of ground systems. These tools, under the name Robotics Technology Kernel (RTK), are a library of tested, managed Robotics Operating Systems (ROS) packages which together establish a common robotics platform and can be combined to form parts or all of an "autonomy kit" (or A-kit) for simulation and testing of ground robots. ROS is a middleware software framework that allows a set of hardware or software devices, represented as "nodes" used on a robotic vehicle, to subscribe and publish messages among each other through the use of standardized "topics" which are common message passing interfaces agreed within the robotic framework [23]. All physical tests involving autonomous navigation employed hardware running RTK with the algorithms summarized in the next sections.

### *7.1. A* Path Planner*

The A* path planner is an algorithm [24] in RTK widely used in search and graph traversal, in which a vehicle finds an optimal path between multiple points called "nodes." It is seen as an addition to Dijkstra's algorithm, but uses heuristics for its search. A* is a best-first search algorithm, as it calculates the "cost" of the terrain from the starting node to the goal node, depending on the difficulty of travel on a particular node. The optimal path is the route that has the least amount of cost.

### *7.2. RRT Algorithm Based Path Planner*

RTK contains a path planner based on the Rapidly-exploring Random Tree (RRT) of feasible paths that is meant for unstructured and cluttered environments [25]. Its capabilities include: (1) planning to a single waypoint instead of multiple waypoints to represent the shape of the road, (2) planning on a persistent map, in which long distances and exploration can be used, (3) anytime planning, in which it can find a feasible path, then continuing to optimize as time allows, (4) the ability to find other paths and to re-plan as the vehicle executes the current path, and (5) finding the fastest path instead of the shortest.

### *7.3. Waypoint Follower*

A third RTK planner is a direct following route-translation method, operating on the assumption that the shape of a sequence of waypoints will match well with the shape of low-cost areas as observed by the perception system. It works by translating a pre-defined path about its geo-referenced coordinates to optimize the traversability cost.

## 8. INTEGRATION OF PATH PLANNERS IN M&S

### *8.1. RTK in Simulation*

Due to a limitation in time and resources, ROAMS and ANVEL with RTK was not used for this benchmark.

### *8.2. Path Planners for ROAMS simulations*

Path planners including the built-in waypoint follower within ROAMS and a model predictive control (MPC)-based algorithm were integrated into the framework and used in simulations across the various scenarios.

The ROAMS waypoint navigation is a stand-in navigation algorithm where the vehicle heads toward each individual waypoint as a goal, as a controller is used to maintain velocity and direction. This can be replaced with more sophisticated navigation algorithms.

The MPC algorithm is capability of obstacle detection and path planning. It uses a 2D planar LIDAR model and a user defined obstacle map to perform hazard detection. In order to plan dynamically safe and feasible paths, the algorithm uses an inertial 3 degree of freedom bicycle model of the plant vehicle. The user must specify the starting point, goal point, starting speed, LIDAR range, obstacle sizes and locations, and several vehicle properties for the 3DOF model such as mass, wheel size, and others in order to run a simulation with the MPC algorithm. See [26] for more details regarding this algorithm.

The MPC algorithm was integrated into the ROAMS simulation tool. ROAMS simulated the plant model, providing vehicle state information such as position, velocity, and heading to the autonomous controller. The MPC algorithm used the vehicle state information in its calculations for path planning and provided control inputs to the ROAMS model such as throttle, braking, and steering commands to reach a goal point while avoiding obstacles.

### *8.3. Path Planners for ANVEL simulations*

This study utilized the newest available version of ANVEL (v3.5) with limited ROS capability. At the time of the study, RTK integration for this particular version was still under development. Open source path planners from the ROS navigation package [27] such as A* and Dijkstra were used in ANVEL simulation, in addition to ANVEL's built-in waypoint follower based on Stanley [28].

ANVEL (v3.5) as an experimental feature includes ROS-integrated ANVEL plugins. The ROS plugins provide ANVEL the ability to initialize its own ROS node, start a ROS core, connect to a ROS core on a local or external machine, as well as create a number of new sensors which send and receive data using ROS topics. In order to use the ROS features, ANVEL must be installed on an Ubuntu 14.04 system alongside an installation of the ros-indigo-desktop-full and ros-indigo-navigation packages. These features have been used to work with the ROS 2D navigation package and other packages [27].

ROS's move_base [29] package provides an implementation of an action that, given a goal in the world, will attempt to reach it with a mobile base.

The move_base node links together a global and local planner to accomplish its global navigation task. The move_base node also maintains two costmaps, one for the global planner, and one for a local planner [29] that are used to accomplish navigation tasks. ROS navigation stack enables autonomous vehicles to move from place to place by providing a safe set of waypoints to follow. By processing data from the odometry, sensors and the map of the operating environment.

*Table 12 MSU AVS Physical Testing Matrix*

| Physical Testing Matrix | | | | |
|---|---|---|---|---|
| | Tele-operation | RTK – A* | RTK – RRT | RTK – Path Follower |
| Straight Line | SG | SG | SG | SG |
| 10m slalom | SG | SG | SG | SG |
| Complex traversal | M | M | M | M |
| S - Sand, G - Grass, M - Mud | | | | |

Maximizing the performance of this navigation stack requires some fine tuning of various navigation parameters. Both Djikstra and A* algorithms are graph search algorithms to find the shortest path from a source to a target. As there might be multiple paths with the same length in any grid map, the paths found in one run may slightly vary from the path found in a different run. Hence multiple (12) runs were made and mean and standard deviations are computed.

## 9. PHYSICAL TESTING

In order to perform verification and validation of the simulation framework developed in this effort, off-road test data was collected through an effort with the Mississippi State University (MSU) Center for Advanced Vehicular Systems (CAVS). The CAVS team was provided an instrumented Polaris MRZR4 vehicle outfitted with the GVSC RTK software suite, which included capability to test teleoperation and several autonomous algorithms.

The objective of the testing was to collect off-road, unmanned vehicle mobility performance data across several scenarios, soils types, and control modes. Table 12 details the collected test data for teleoperation and autonomy across the various course layouts and off-road soils.

Three soil types were used for testing: sand, grass on drained clay, and undrained clay (mud). In situ cone index measurements and lab testing was conducted to characterize the soil properties. The cone index measurements were then converted to Bekker-Wong soil model parameters for use in simulation, see Table 9 [20].

The straight line course was developed to test simple obstacle detection and avoidance. The scenario consisted of a starting point and goal point spaced 100 feet apart with an 8 foot wide obstacle placed 50 feet away from the starting location. The objective was to drive to and stop at the goal point as quickly as possible while avoiding collisions with the obstacle. The course layout is described in Figure 12.

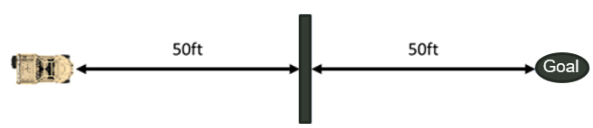



*Figure 12 Straight line course with an obstacle in the middle*

The 7m and 10m slalom courses were developed to test complex obstacle detection and avoidance with tight turns, representing real world situations where maneuverability in theater may be challenging. The scenario consisted of a starting point and goal point spaced 28m or 40m apart. Three 3m wide obstacles were placed every 7m or 10m thought out the course. The objective was to drive to and stop at the goal point as quickly as possible while avoiding collisions with the obstacles. The course layout is described in Figure 13.

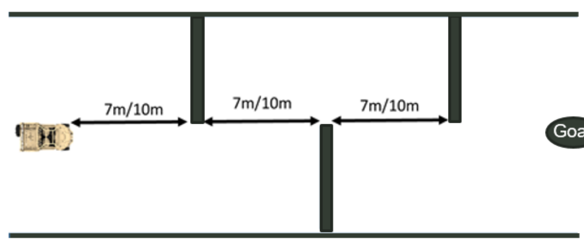



*Figure 13 Two slalom courses considered in this study*

The complex traversal course was developed to test the capability to follow a winding path. The scenario consisted of a starting point leading to a goal point by driving through "gates" along a predefined path. The driver or algorithm maneuvered the vehicle from the starting point to each gate until reaching the goal point. The objective was to drive to and stop at the goal point as quickly as possible while driving through all the gates without colliding with them. The course layout with notional gate locations is described in Figure 14.

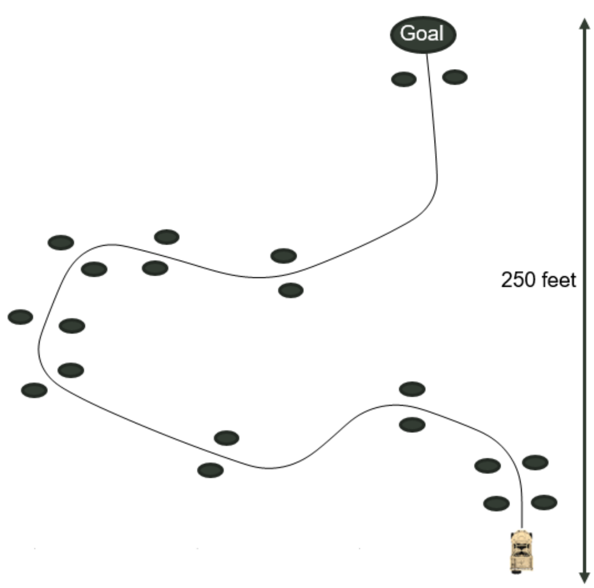



*Figure 14 A complex traversal course involving multiple turns on either side*

Two main driving modes were tested across the different soil types and scenarios: teleoperation with latency and autonomy.

Teleoperation by two different remote operators was used to get a baseline of performance of the UGV. The operators were located at a base station near the test site and used the video feed from the vehicle to the base station to conduct the teleoperation runs. The base latency of the system as delivered to the CAVS team was unknown, however estimates suggest approximately 500ms for the communication delay and 150ms for the control delay for a total roundtrip latency of approximately 650ms. An Ethernet delay simulator was used to inject additional latency into the teleoperation system at magnitudes of 100, 250, 500, 1000, and 1500ms.

The CAVS team tested three algorithms under the RTK software suite: A*, RRT, and a path follower. All algorithms used the vehicle's onboard sensors such as LIDAR, GPS/IMU, and wheel encoders for localization and obstacle detection, if capable. During a run, the environment is scanned using onboard LIDAR and cost maps are generated in real-time as inputs to the algorithms to use for autonomous navigation. The Human-Machine Interface (HMI) lets the user input waypoints for the autonomous algorithms to follow for each scenario while performing local obstacle avoidance as required.

## 10. RESULTS

### *10.1. Physical Test Results*

Physical test data is available for straight line and 10m slalom scenarios on sand and grass and for the complex traversal scenario on mud. Due to hardware recording issues and resource constraints, data for the other combinations of courses and soil types is not available.

Table 13 lists the average mean speeds (mph) for each course, soil type, and control mode broken down between teleoperation with latency and the autonomous algorithms [30]. A high level comparison of teleoperation vs. autonomous algorithm performance is plotted in Figure 15. For both teleoperation and autonomy, the mean speeds across all courses and soils types were averaged and plotted together. Since the algorithms ran on the vehicle in real time, the performance of autonomy was not affected by the latency as opposed to teleoperation performance, hence the constant value in the plot.

*Table 13 Average speeds from physical tests in MPH (Mean and standard deviations are shown)*

| Type | Course/Terrain | Straight Line | | 10m Slalom | | Complex Traversal |
|---|---|---|---|---|---|---|
| | Type/Latency | Grass | Sand | Grass | Sand | Mud |
| Teleop | +0 ms | 3.1 (0.3) | 2.8 (0.3) | 2.9 (0.8) | 1.8 ( * ) | 2.9 (0.6) |
| | +100 ms | 2.8 (0.3) | 2.9 (0.4) | 3.0 (0.4) | 1.8 (0.2) | 3.0 (0.2) |
| | +250 ms | 2.6 (0.4) | 2.9 (0.3) | 2.8 (0.5) | 2.4 (0.9) | 3.2 (0.1) |
| | +500 ms | 2.1 (0.6) | 1.7 (0.6) | 2.9 (0.5) | 2.0 (0.5) | 3.1 ( * ) |
| | +1000 ms | 2.3 (0.3) | 1.4 (0.4) | 1.9 (1.1) | 1.9 (0.4) | 2.9 (0.3) |
| | +1500 ms | 1.7 (0.5) | 1.3 (0.6) | 1.7 (1.5) | 0.4 ( * ) | 2.9 (0.4) |
| Auto | A* | 2.8 (0.1) | 2.6 (0.3) | 2.6 (0.1) | 2.6 (0.4) | 2.6 (0.01) |
| | RRT (single) | 1.8 (0.2) | 2.3 (0.4) | 1.6 (0.1) | 1.2 (0.4) | ** |
| | RRT (multiple) | ** | ** | 1.5 (0.6) | 1.7 (0.4) | 2.0 (0.2) |
| | Waypoint Following | 3.1 (0.5) | 2.8 (0.3) | 2.6 (0.01) | 2.6 (0.4) | 2.5 (0.3) |

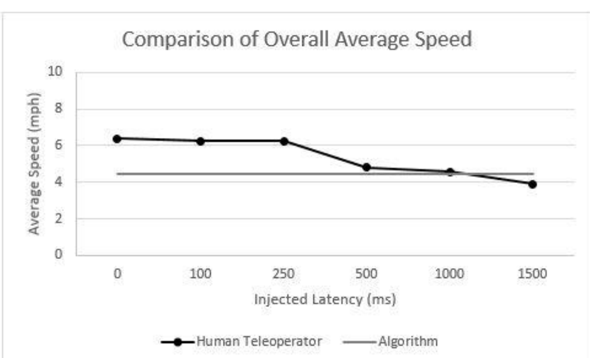



*Figure 15 Comparison of average speeds (m/s) between teleoperation and autonomous navigation from tests*

It was noted that the test speeds were lower than anticipated due to several reasons. For one, software and hardware governors were present on the vehicle for safety. However, the hardware governor was never activated during testing. The software governors may have restricted high speed driving. With that in mind, velocity time-history plots show that there was significant variation in maximum speed across driver, course, and soil type, suggesting the software governor did not limit performance across all tests. In addition, as seen in Figure 16, most speeds observed during testing fell well below the potentially-limited maximum speed observed, proving other factors such as course size limited performance.

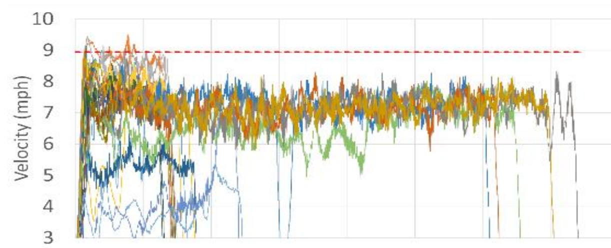



*Figure 16 A summary of average velocity from teleop tests*

Furthermore, the test team performed limited simulations of the test scenarios using a third party algorithm to analyze full autonomy and semi-autonomy. To represent the physical testing, a speed limit of 9 mph was set in the virtual model. Simulations were conducted for the straight line and slaloms scenarios. The speed limit was then increased to 18 mph and no significant change in average speeds were observed, indicating that mostly the course layout and autonomous algorithms constrained the vehicle's speed independent of the governor.

To confirm the hypothesis that the course layouts were limiting average speeds, an extra simulation was ran within this benchmark using the complex traversal course scaled in the longitudinal and lateral directions by a factor of six on mud in order to allow the vehicle to reach higher speeds. Results are presented in section 10.2.

### *10.2. Simulation results*

Simulations were carried out for all scenarios including those not addressed in physical testing.

It should be noted that the autonomous control algorithms can be significantly different between the physical testing, the simulations with ANVEL, and the simulations with ROAMS. Even within one type of algorithm, e.g. waypoint following, several tunable parameters defined within the algorithm can be altered to give different vehicle performance results. Therefore, comparisons between the test and simulation results should be performed in a general sense at a high level and not in detail. To that end, comparisons of average speeds are used in the analysis.

Table 14 (please see the following page) shows a draft summary of recorded average speeds (mph) from physical tests and simulations for straight line, 10m slalom and complex traversal courses. The 7m slalom proved to be too challenging during physical testing and in simulation as the vehicle collided into the barriers due to the tight course layout. Figures 17a-c show a graphical comparison between physical test and simulations from the two software for sand, grass and mud terrain respectively for the three different courses. Figure 17d show an aggregate comparison of average speeds from all three courses from physical test and simulation data from the two codes on three different terrains.

Results from simulating the complex traversal course scaled by a factor of six on mud are presented in Table 15. For three different autonomy algorithms, the average speed significantly increases as the course dimensions increase, proving that the course layout, in part, restricted performance of the UGV in test and simulation.

*Table 15 Average speeds (mph) for the complex traversal course on mud with original and 6x scaling*

| Tool | ANVEL | ROAMS | |
|---|---|---|---|
| Algorithm | Waypoint | Waypoint | MPC |
| Original scaling | 7.4 | 5.2 | 3.4 |
| 6x scaling | 21.8 | 14.4 | 16.6 |

## 11. CONCLUSIONS

Aggregating the test and simulation data together allows for comparisons to be performed between test and simulation data at a high level. Although RTK was not used in simulation, the types of algorithms it uses share similarities with those used in this simulation framework, such as waypoint followers and A*. Therefore, general comparisons can be made between the performance of autonomy in test and in simulation. To satisfy the objective of this benchmark, the two autonomous simulation tool results can be compared against one another and the test data as V&V of each of the tools' modeling capabilities. Lastly, combining all simulation results for each soil type shows the effect of soft soil on mobility for the autonomous vehicle.

### *11.1. Teleoperation vs. Autonomy - Test*

Figure 15 clearly indicates the trend in the test data once the mean of the average speeds are plotted across all courses and soil types. That is to say, teleoperation performed better than the autonomous algorithms tested within RTK until very high latencies were observed by the remote operators which lead to performance equalizing between the two control modes. One factor attributing to higher speeds during teleoperation is that the same operators were used for all teleoperation testing and gained experience as the testing continued which can lead to better performance.

*Table 14 Average speeds (mph) recorded from physical tests and simulations for the three different courses on three different terrains (Mean and standard deviations are shown when multiple simulations are performed; Non available results are shown as *)*

| # | | Event | Straight Line | | | 10m Slalom | | | Complex Traversal | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | **Sand** | **Grass** | **Mud** | **Sand** | **Grass** | **Mud** | **Sand** | **Grass** | **Mud** |
| 1 | TEST | Teleop #1 | 5.4 | 5.3 | * | 3.9 | 6.6 | * | * | * | 6.6 |
| 2 | TEST | Teleop #2 | 4.4 | 5.4 | * | 4.2 | 4.5 | * | * | * | 6.6 |
| 3 | TEST | A* | 5.8 | 6.3 | * | 5.7 | 5.8 | * | * | * | 5.9 |
| 4 | TEST | RRT (Single) | 5.2 | 3.9 | * | 2.7 | 3.5 | * | * | * | * |
| 5 | TEST | RRT (Multiple) | * | * | * | 3.7 | 3.4 | * | * | * | 4.5 |
| 6 | TEST | Waypoint follower | 6.2 | 7.0 | * | 5.8 | 5.8 | * | * | * | 5.5 |
| 7 | TEST | **Autonomy (Avg)** | **5.7** | **5.7** | * | **4.5** | **4.6** | * | * | * | **5.3** |
| 8 | ANVEL | Waypoint follower | 4.7 | 4.0 | 4.3 | 4.9 | 4.7 | 5.6 | 8.1 | 8.1 | 7.4 |
| 9 | ANVEL | Dijkstra | 4.0 (0.15) | 4.3 (0.22) | 5.1 (0.17) | 3.9 (0.19) | 4.1 (0.23) | 5.5 (0.28) | 4.0 (0.15) | 4.3 (0.22) | 4.4 (0.43) |
| 10 | ANVEL | A* | 4.0 (0.12) | 4.2 (0.08) | 5.1 (0.20) | 4.0 (0.22) | 4.1 (0.19) | 5.4 (0.34) | 4.0 (0.23) | 4.2 (0.08) | 4.7 (0.17) |
| 11 | ANVEL | **Autonoomy (Avg)** | **4.2** | **4.2** | **4.8** | **4.3** | **4.3** | **5.5** | **5.4** | **5.5** | **5.5** |
| 12 | ROAMS | Waypoint follower | 2.0 | 4.1 | 3.8 | 2.2 | 3.0 | 2.7 | 3.2 | 5.4 | 4.9 |
| 13 | ROAMS | MPC | 7.6 | 7.7 | 7.7 | 8.6 | 8.1 | 7.3 | 11.3 | 11.3 | 7.5 |
| 14 | ROAMS | **Autonomy (Avg)** | **4.8** | **5.9** | **5.8** | **5.4** | **5.6** | **5.0** | **7.3** | **8.4** | **6.2** |

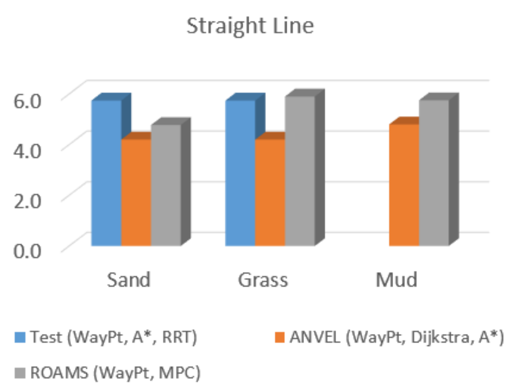



*Figure 17(a) Average speeds (mph) comparison for the straight line course*

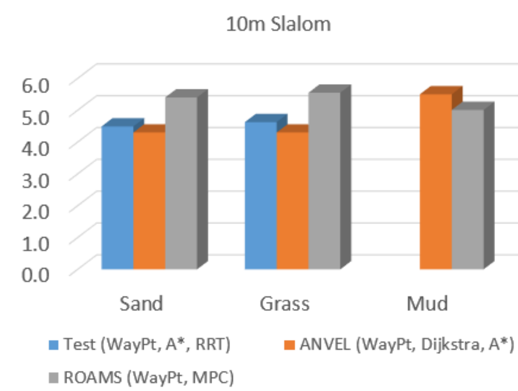



*Figure 17(b) Average speeds (mph) comparison for the 10m slalom course*

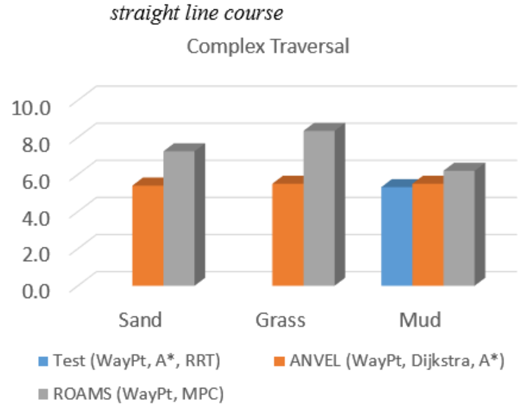



*Figure 17(c) Average speeds (mph) comparison for the complex traversal*

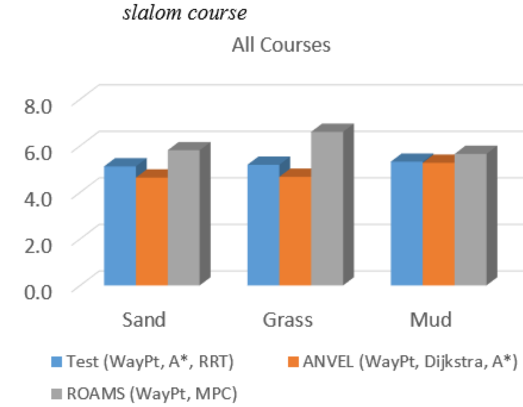



*Figure 17(d) Average speeds (mph) comparison for all courses run on autonomy*

Further statistical analysis using ANOVA was carried out by the CAVS test team. For the straight line and 10m slalom scenarios, the A* and waypoint following algorithm had higher average speeds than the human teleoperators who had higher speeds than the RRT based algorithm. However, for the complex traversal scenario, the human teleoperators were faster than all autonomy algorithms ($p < 0.001$). Additionally, latency did not affect the algorithms for any scenario nor the human teleoperators in the 10m slalom and complex traversal scenarios ($p = 0.166$ and $p = 0.813$, respectively) but latency did affect humans in the straight line scenario ($p < 0.001$).

### *11.2. Test vs. Simulation - Autonomy*

When averaging speeds across all soil types and courses, a comparison of test and simulation results for autonomy is possible. Figure 17a-d shows the difference in average speed between the test and simulation data. On the whole, the simulation average speeds are comparable to the test average speeds. Some observed variation can be attributed to differences in implementation between the algorithms on the test vehicle and in the simulation models. For example, high-level algorithm logic and methodology or gains in the low-level controller for actuating steering, throttle, and braking could be different. The level of fidelity and characteristics such as range of the LIDAR sensor models may also lead to differences in performance.

### *11.3. ANVEL vs. ROAMS – Autonomy*

Comparing the two tools ability to accurately represent autonomous vehicle navigation in soft soil was a key objective of this study. When comparing the average speeds of the simulations to the test data, both tools fare similarly well and match reasonably. Figure 17 shows how ROAMS is slightly closer to the test average speeds for the straight line course whereas ANVEL is better in the 10m slalom. Both perform similarly in mud. For the specific scenarios investigated in this benchmark, the low fidelity vehicle model in ANVEL did not negatively impact performance during the maneuvers, suggesting a high fidelity tool is not necessarily required at the speeds used in this study.

### *11.4. Effect of Soil Type on Average Speed*

The average speeds in test were not affected by changes in soil type as seen in Figure 17d when averaging results across all driving modes and courses. However, the test team concluded that differences can be seen when looking into the data. For example, autonomy was rarely affected by surface type whereas the human teleoperators did see performance differences on different terrains, particularly faster speeds in grass than sand. In simulation, when calculating the mean of average speeds across scenarios, there was little effect of soil type on performance for ANVEL and a minor affect in ROAMS. However, the effect of soil type may appear at speeds higher than what was used in this benchmark.

### *11.5. Overall Conclusions*

The goal of this paper was to determine if modern M&S tools are ready to assess autonomous military vehicles in off-road environments. After comparing the simulation results to the test data in this preliminary benchmark, it appears these two tools provide the key capabilities to model and simulate unmanned ground vehicles for military applications. The simulation results matched well with the test data for both the manned dynamic tests on pavement and unmanned off-road scenarios highlighted in Sections 5 and 10, respectively. Additional benchmarking studies should be conducted to include autonomous vehicle simulation tools that were not addressed in this analysis to get a broader understanding of M&S capabilities for military vehicle applications.

## 12. LESSONS LEARNED

Upon completion of this preliminary benchmark, several opportunities to improve the benchmarking process were identified.

First, higher speed physical testing would be pursued. This can be achieved on the MRZR hardware by disabling the speed governor and by using longer, more open course layouts in the mobility scenarios to allow the vehicle to reach higher speeds such as 30+ mph. Higher speeds make tasks such as path following or obstacle avoidance more difficult, highlighting the strengths and weaknesses of teleoperation and autonomy.

Secondly, full factorial physical testing on all soils and all courses would provide more data to compare with simulation results. Better coordination with the vehicle and test team would ensure complete test coverage.

Lastly, other metrics of mobility performance besides average speed could be investigated. Maximum speed, steering control effort, path tracking error, algorithm failures, and collisions with obstacles would provide additional feedback on remote operator and algorithm performance.

## 13. ACKNOWLEDGEMENTS

The authors thank Abhinandan Jain, PhD and Aaron Schutte, PhD (Jet Propulsion Laboratory), Tulga Ersal, PhD (University of Michigan), Jiechao Liu, PhD (Magna Autonomous Systems), Daniel Rice and Taylor Warden (Quantum Signal, LLC) for their modeling and simulation support during this study.

## 14. REFERENCES

[1] P. J. Durst, C. Goodin, C. Cummins, B. Gates, B. McKinley, T. George, M. M. Rohde, M. A. Toschlog and J. Crawford, "A real-time interactive simulation environment for unmanned ground vehicles: The autonomous navigation virtual environment laboratory (ANVEL)," *Information on Computing Science (ICIC) Fifth International Conference on,* pp. 7-10, 2012.

[2] C. Goodin, J. T. Carrillo, D. P. McInnis, C. L. Cummins, P. J. Durst, B. Q. Gates and B. S. Newell, "Unmanned ground vehicle simulation with the virtual autonomous navigation environment," *2017 International Conference on Military Technologies (ICMT),* pp. 160-165, MAy 2017.

[3] B. P. Gerkey, R. T. Vaughan and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," *in Proceedings of the International Conference on Advanced Robotics, Coimbra, Portugal,* pp. 317-323, Jul 2003.

[4] P. Jayakumar, W. Smith, B. A. Ross, R. Jategaonkar and K. Konarzewski, "Development of High Fidelity Mobility Simulation of an Autonomous Vehicle in an Off-Road Scenario Using Integrated Sensor, Controller, and Multi-Body Dynamics.," 2011.

[5] M. Lewis, J. Wang and S. Hughes, "Usarsim: Simulation for the study of human-robot interaction," *Journal of Cognitive Engineering and Decision Making,* pp. 98-120, 2007.

[6] A. Staranowicz and G. L. Mariottini, "A survey and comparison of commerical and open-source robotic simulator software," *Proceedings of the 4th International Conference on Pervasive Technologies Related to Assistive Environments,* May 25-27, 2011, Heraklion, Crete, Greece.

[7] P. Castillo-Pizarro, T. V. Arredondo and M. Torres-Torriti, "Introductory Survey to Open-Source Mobile Robot Simulation Software," *Robotics Symposium and Intelligent Robotic Meeting (LARS),* pp. 150-155, 2010, Latin America.

[8] J. Kramer and M. Scheutz, "Auton Robot," 2007. [Online]. Available: https://doi.org/10.1007/s10514-006-9013-8.

[9] F. Rosique, P. Navarro Lorente, C. Fernandez and A. Padilla, "A Systematic Review of Perception System and Simulators for Autonomous Vehicle Research," *Sensors,* vol. 19, no. 10.3390/s19030648, p. 648, 2019.

[10] US Army Engineer Research and Development Center, "VANE and ANVEL Training Manual", Vicksburg, MS, 2006.

[11] "https://www.ode.org," [Online].

[12] "https://www.ogre3d.org/about," [Online].

[13] NASA Jet Propulsion Laboratory, "ROAMS: Spacecraft Dynamics Simulator", https://dartslab.jpl.nasa.gov/ROAMS/index.php.

[14] SEA Ltd., "Suspension (Kinematics and Compliance) and Vehicle Inertia Measurements Facility (VIMF) Test Results on 2013 Polaris MRZR," July 8, 2016.

[15] Stackpole Engineering, "Tire Modeling report for DWT AT26x9-14(MF6.1 UM4)," Aug 17, 2016.

[16] "http://offroad.polaris.com".

[17] SEA Ltd., "Dynamic Handling and Stability Test Results on 2013 Polaris MRZR," July 22, 2016.

[18] NATO, ""NATO Allied Vehicle Testing Publication"," NATO, Sep 1991.

[19] J. Wong, "Theory of Ground Vehicles", 4th Edition ed., John WIley and Sons.

[20] G. L. Mason, J. E. Salmon, S. McLeod, P. Jayakumar, M. Cole and W. Smith, "An Overview of a Method to Convert Cone Index to Bevameter Parameters," *Journal of Terramechanics (Pending).*

[21] WES 1964, "Strength-Moisture-Desnity relations of fine grain soils in vehicle mobility research".

[22] Z. Janosi, "Prediction of "WES Cone Index" by means of Stress-Strain function of soils", Land Locomotion Laboratory, 1959.

[23] M. e. a. Quigley, "ROS: an open-source Robotic Operating System," *IRCRA workshop on open source software,* vol. 3, no. 3.2, 2009.

[24] https://en.wikipedia.org/wiki/A*_search_algorithm.

[25] https://en.wikipedia.org/wiki/Rapidly-exploring_random_tree, "Rapidly-exploring Random Tree".

[26] J. Liu, P. Jayakumar, J. Stein and T. Ersal, "Combined Speed and Steering Control in the High Speed Autonomous Ground Vehicles for Obstacle Avoidance Using Model Predictive Control," *IEEE Transactions on Vehicular Technology,* 2017 DOI 10.1109/TVT.2017.2707076.

[27] http://www.ros.org/navigation.

[28] S. Thrun and et.al., "Stanley: The robot that won the DARPA Grand Challenge," *Journal of Field Robotics*, vol. 23(9), pp. 661-692, 2006.

[29] http://www.ros.org/move_base.

[30] Mississippi State University, "MSU CAVS Intelligent Mobility Evaluation of an Autonomous Polaris MRZR4 - Final Report for W56HZV-17-C-0095-TO-001".

[31] T. D. Gillespie, "Fundamentals of Vehicle Dynamics", Society of Automotive Engineers, 1992.

[32] Society of Automotive Engineers, "Vehicle Dynamics Terminology SAE J670e", Warrendale, PA: SAE.