2019 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
AUTONOMOUS GROUND SYSTEMS TECHNICAL SESSION
AUGUST 13-15, 2019 - NOVI, MICHIGAN

EVOLVING THE ROBOTIC TECHNOLOGY KERNEL TO EXPAND FUTURE FORCE AUTONOMOUS GROUND VEHICLE CAPABILITIES

MAJ Joseph Cymerman[1], MAJ Sang Yim[2], MAJ (R) Dominic Larkin[2], LTC Kathryn Pegues[3], 2LT Wyatt Gengler[2], 2LT Samuel Norman[2], 2LT Zachary Maxwell[1], 2LT Nicholas Gasparri[1], 2LT Mary Pollin[2], 2LT Clement Calderon[1], 2LT Jeremy Angle[3], 2LT Jaylen Collier[3]

[1]Department of Civil and Mechanical Engineering
[2]Department of Electrical Engineering and Computer Science
[3]Department of Systems Engineering
United States Military Academy, West Point, NY

ABSTRACT

This paper presents the conceptual design, development, and implementation of the Robotic Technology Kernel (RTK) in a Polaris GEM e2 by the United States Military Academy's autonomy research team. RTK is the autonomous software suite of the U.S. Army Combat Capabilities Development Command Ground Vehicles Systems Center and to this point has primarily been used within off-road environments. The research team's primary objectives were to verify RTK's platform-agnostic characteristic by implementing the control software on a small, low-speed electric vehicle and augmenting the software to provide the additional capability of operating within an established infrastructure rule set.

Citation: J. Cymerman, S. Yim, D. Larkin, K. Pegues, W. Gengler, S. Norman, Z. Maxwell, N. Gasparri, M. Pollin, C. Calderon, J. Angle, J. Collier, "Evolving the Robotic Technology Kernal to Expand Future Force Autonomous Ground Vehicle Capabilities," In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 13-15, 2019.

1. INTRODUCTION

Based on forecasted needs of the United States (U.S.) military force and its partners in future conflicts, current research efforts must quickly develop additional autonomous ground vehicle capabilities to retain battlefield superiority. The speed of development is a critical aspect of maintaining an asymmetric advantage.

To meet these forecasted requirements, a collaborative effort is underway to expand further a common autonomous vehicle framework for the U.S. Army, which consists of a collection of software procured over multiple robotics programs. The market place for autonomous robotic ground vehicles contains a broad array of available platforms. These autonomous systems vary in complexity, scope, and particular mission sets, each requiring unique capabilities. The potential for these systems in altering the battlefield has not been realized because of the narrow operational application. Additional work must increase the types of scenarios with which autonomous systems can effectively operate. A critical area still needing development, autonomous military vehicles must responsibly traverse a diverse domain while in motion. This traversal includes public roads in urban and rural terrain in tactical and especially non-tactical situations, where laws governing movement on public infrastructure would apply. Following stop signs and negotiating lane lines are not typical military functions, but a flexible, autonomous framework must include these and like capacities.

To address this problem set for the Army, Great Lakes Systems and Technology, LLC (GLS&T), through the U.S. Army Combat Capabilities Development Command (CCDC) Ground Vehicles Systems Center (GVSC), awarded funding to nine academic institutions to expand the Robotic Technology Kernal (RTK). RTK is GVSC's software package for autonomous navigation in off-road environments, to include the capability to navigate within a known infrastructure rule set.

This two-phased research effort, managed by GLS&T, focuses first on the development of RTK for use by a lone platform within an urban environment with the intent to then move toward multi-vehicle autonomous collaboration among individually implemented RTK solutions. In the first phase, the funded academic institutions adapted RTK for on-road logistics operations. From August 2018 to June 2019, each team developed an autonomous system by augmenting a commercially available vehicle with a sensor suite. The teams then expanded the RTK's capabilities to account for basic "road rules," such as the handling of a left-hand turn or coming to a halt at a stop sign. The teams tested their systems in June 2019 at the 27th Annual Intelligent Ground Vehicle Competition (IGVC).

This paper documents the research process of the autonomy research team at the United States Military Academy at West Point (USMA) during this first phase. The team consisted of undergraduate students and their advisors. The following pages describe the process of implementing RTK in a commercially available electric vehicle and documents the steps taken to evolve RTK for use with commercial vehicles in urban environments.

This research effort also collected the methods, techniques, challenges, and lessons learned from USMA's implementation. The effectiveness of RTK documentation and repository management will be assessed to improve future collaboration.

Lastly, a discussion of future work in the RTK space will explore the history of robotic development within the context of combat engineering. This section will motivate further decentralization of autonomy development, as USMA had performed here, and also articulate exciting opportunities for the rapid projection of combat power that could result from these efforts.

2. DESIGN PROCESS

The autonomy research team followed the Agile Design Process (ADP) as it developed an autonomous system utilizing an electric car and commercially available sensors. During the nine-month project, the team assessed RTK capabilities, framed the problem, created an execution plan, designed a system prototype, and developed packages to expand RTK capabilities in commercial vehicles. The team worked to adapt and extend the RTK software repository to account for on-road driving. After defining the problem, determining the functions that the autonomous system must perform was a critical step. The team carefully considered the characteristics of urban environments and reviewed the competition guidelines for IGVC to build a comprehensive list of functions. The primary function were the ability to sense the environment and determine a possible navigational path. Additional requirements determined that the final system must be affordable, reactive to changing environments, user-friendly, safe, durable, and expandable. The design functions evolved into a list of system requirements. The requirements were then used to assess proposed system alternatives and how well the final system met the stated needs of the stakeholders.

A Polaris GEM e2 (shown in figure 1) performed as the base for the autonomous system, which has a battery bank of six 12-volt batteries offering 72 volts at 75 amps equating to 5,400 kWh [1]. The motor operates via AC induction and is rated at 6.7 HP [1]. The vehicle's factory dimensions are 60-in. width, 115-in. length, and 75-in. height and weighs 1,200 lbs. [1].

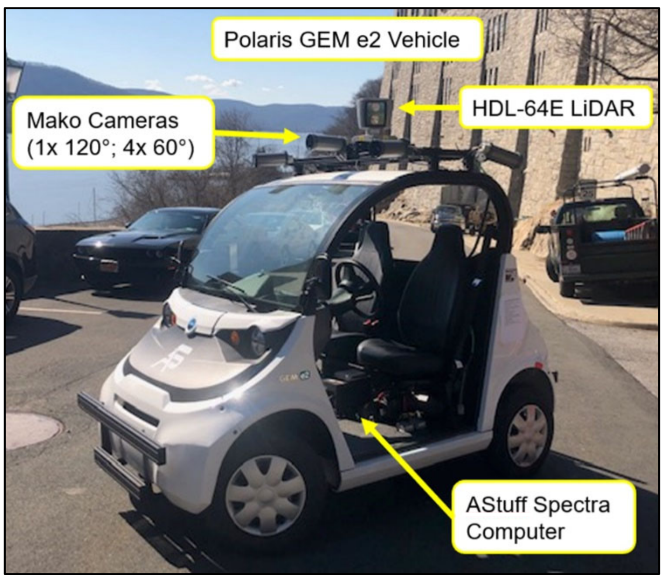

Figure 1: Polaris GEM e2.

2.1. Alternatives Generation

Designing an autonomous system, the team evaluated three different configurations and sensor suites which provide differing capabilities. All the alternatives met the system requirements for sensing the environment but varied in their ability to meet supplemental requirements such as expandability and affordability. The following paragraphs describe the three alternatives along with the advantages and disadvantages of each.

The first alternative was designed to be the most cost-effective because it leveraged existing equipment from previous projects and reduced procurement costs. This alternative uses infrared sensors and a Velodyne HDL-64E Light Detection and Ranging sensor (LiDAR) for obstacle detection, obstacle avoidance, lane detection, and lane following. Laptop computers with Intel i7 processors and NVIDIA GPUs operating Ubuntu 16.04 would provide processing capability. While this option was cost-effective, it had limited expandability for future implementations of object classification for road sign detection.

The second alternative focused on maximizing sensing capabilities. It included the implementation of five stereo cameras, two Velodyne HDL-32E LiDARs, ultrasonic sensors, and a custom-built computer with the stereo cameras providing depth to images. The two Velodyne HDL-32Es would be compatible with the current configuration of RTK with one placed in the front of the vehicle and the other in the rear. Ultrasonic sensors would enable close-range detection and self-parking solutions. While this alternative would simplify integration and adaptation to RTK, it was determined to be too costly.

Ultimately, the team selected an acceptable compromise between these two alternatives. This design included the HDL-64e LiDAR from the first option and the stereo cameras from the second option. After evaluating several vendors for a drive-by-wire solution, AutonomouStuff provided a drive-by-wire kit, the onboard computer, and cameras used for stereo vision. This solution met both design specifications by being cost-effective and expandable; however, this sensor selection had not yet been implemented in RTK and required further integration.

2.2. Conceptual Design

Approved by GLS&T in December 2018, USMA's design moved forward as the team awaited the delivery of the GEM e2 from AutonomouStuff. The Illinois-based company installed several items on the GEM before it arrived at USMA. One item was the Platform Actuation and Control Module (PACMod) drive-by-wire (DBW) system, which provided a Robot Operating System (ROS) interface to the CAN bus. Also, AutonomouStuff supplied a Multiplexed Vehicle Electrical Center (mVEC) and a control computer, an AutonomouStuff Spectra, which mounted to the floor beneath the seats. Once on hand, the team reformatted the storage drive on the computer and installed the operating system, ROS, RTK, and all supporting drivers and libraries. The GEM e2 also came with a mounting rack built from 80/20 T-slot aluminum bars on the roof, which support the five Mako G-319C cameras (three in front and two in back). The front middle camera employs a wide-angle lens. The rack also secures the Velodyne HDL-64E 3D LiDAR sensor, which is centered on the roof with a clear view to both the front and rear of the vehicle. An Xsens MTi-710 GPS/IMU and an 8-port Power over Ethernet (PoE) remain inside the vehicle. With the addition of the external sensors, the dimensions of the vehicle increased to 72 inches in width, 115 inches in length, and 89 inches in height.

Essential considerations in the design of the system included power availability and energy storage. For this research, the defined time needed for successful mission execution is the duration of the testing phase and obstacle execution at IGVC. The power analysis included individual assessments for distribution, quality, and consumption.

Power distribution within the GEM e2 (shown in figure 2) requires a regulated application of power to the sensors and components, the connectors of which need to be modified to draw power from the mVEC. The mVEC outputs 12 volts DC at a maximum of 20 amps [2]. A boost converter provides the required 24 volts at 10 amps [3] to the Spectra computer.

The power supplied from the mVEC to the sensors and components comes from the GEM e2's six 12-volt battery system. The delivery rate from the mVEC fluctuates because the available power of the batteries decreases as the vehicle continues to draw power over an extended period. As the GEM e2's battery system depletes, the voltage supplied through the mVEC also reduces. This drop in voltage is often referred to as an unintentional brownout, which causes irregular behaviors in elecrical systems. Voltage regulators eliminate this potential, and their output voltage was tested at various intervals of power consumption. The team found constant output from the voltage regulators even at changing levels of performance.

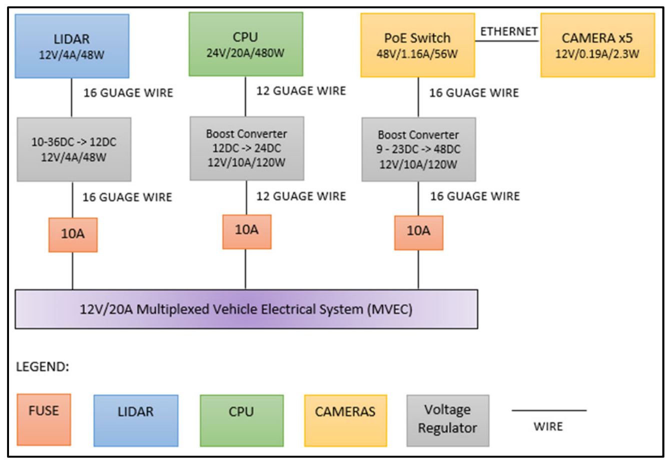

Figure 2: Power Distribution Diagram.

The primary objective in analyzing the GEM e2's power consumption was to determine the maximum run time at peak performance and under IGVC competition demands. The vehicle battery system provides power to the vehicle drive motors, its electrical systems (lights, dashboard, etc.) as well as the autonomous package components (the LiDAR, five cameras, and an onboard computer). The GEM e2's overall battery capacity is 5,400 watt-hours per the factory specifications [1]. The organic load of the vehicle, consisting of both the motor and standard electrical systems, utilizes

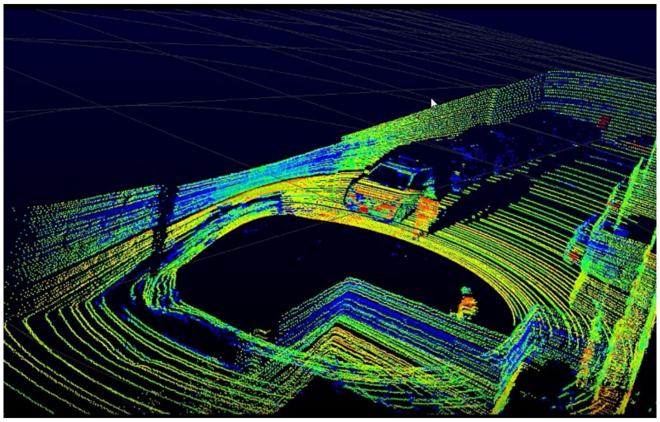

Figure 3: Image of a man and a parked car from the Velodyne HDL-64 LiDAR.

5,200 watts at maximum speed and electrical draw [1].

A power model based on battery capacity and the overall load was developed to determine the theoretical maximum endurance for the autonomous system. The model assumes that the total load is constant at peak performance and found by using the maximum voltage and amperage requirements consumed at the vehicle's top speed of 25-mph [1]. The vehicle's total runtime was determined by dividing the total battery capacity by the peak load values. Therefore, the theoretical endurance of the GEM e2 at peak performance is slightly greater than approximately one hour.

Given that the electric vehicle supplies additional power to sensors, the run time is expected to decrease. The augmented GEM e2, consisting of the vehicle equipped with an autonomous package, has a load of 5,627 watts. The additional power requirements cause an increase in demand by 7% and decreased the theoretical endurance, at constant maximum load, to about 57 minutes. This indicates that the additional hardware from the autonomy kit is not overtaxing the GEM compared to the demand of the electric motors.

PACMod relays system commands from RTK to the mechanical actuators on the GEM e2. PACMod exists as a ROS node on the onboard computer and provides an interface to the vehicle CAN bus, controlling steering, throttle (supplied motor voltage), braking, and gear selection (forward, neutral, and reverse). Additional software written by USMA translates the vehicle control messages from the RTK framework into PACMOD-compatible data types and values, which is discussed further in section 2.3.

The Velodyne HDL-64E LiDAR has 64 laser emitters and receivers to create a dense, three-dimensional point cloud with a 360-degree horizontal field of view [4]. The laser emitters are divided into four groups of 16 while the laser receivers are divided into two groups of 32 [4]. It is designed to have a 50-meter range for pavement and a 120-meter range for cars and foliage [4]. These limits are due to the differences in reflectivity of the materials. This LiDAR has a 26.8-degree vertical field of view, ranging from +2 degrees to -24.8 degrees [4]. The user can select a field of view update between 5 and 15 Hz. and outputs over 1.3 million points per second [4]. ROS provides several tools, such as ROS Visualization (rviz), that provide visualization of the 3D point cloud array, shown in figure 3. The visualization program offers representations of the point cloud array, which is made up of the intensities of object reflections within the environment. The array positions correspond to specific frame around the circumference of the LiDAR. Measuring intensity values allow the vehicle to determine whether a collection of points is an obstacle, a clear path, or a false return. The Velodyne HDL-64E LiDAR integrates directly with RTK like the 32E version of the Velodyne LiDAR. However, the 64E model provides a denser point-cloud output which allows for a more accurate representation of the environment.

Four of the five Mako G-319C provide a 60-degree field of view on each corner of the vehicle. The fifth camera, having a 120-degree field of view, faces forward and is centered left to right on the vehicle. The team designed a 3D representation of the GEM's field of view using the Solidworks CAD tool (figure 4). These cameras have a maximum frame rate of 37.5 frames per second and

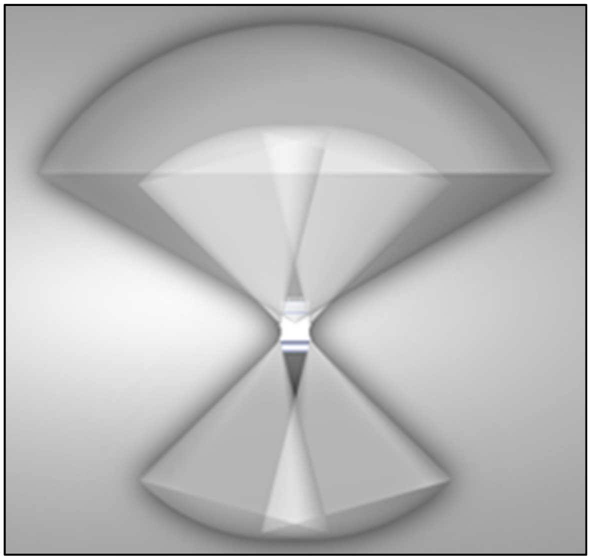

Figure 4: Field-of-View of the Autonomous with 5 functioning Cameras.

provided images with a resolution of 2064x1544 [5]. The availability of a weatherproof housing accessory was an important design choice (figure 5), and the cameras fulfilled the required data output and connectivity specifications.

Waypoint navigation depended heavily on the accuracy of the Xsens MTi-G-710 Series Global Positioning System/Inertial Measurment Unit (GPS/IMU). The Xsens was chosen for its familiarity and 2-meter accuracy [6], which is within the required 5-meters specified by the IGVC specification [7]. This sensor connected to the computer and received power through a single USB cable. The GPS/IMU was mounted in the center of the vehicle within the cab.

The Spectra computer used to control the vehicle was custom designed by AutonomouStuff and contains an Intel Xeon processor and an Nvidia GPU [3]. This computer can use up to 480 watts of power to operate at full capacity [3], and the mVEC can supply at most 240 watts of power to the computer. A benchmarking program from Phoronix Test Suite was used to present a sizeable computational load on both the computer's processor and GPU [8] to test the computer's performance at 240 watts. The team ran two

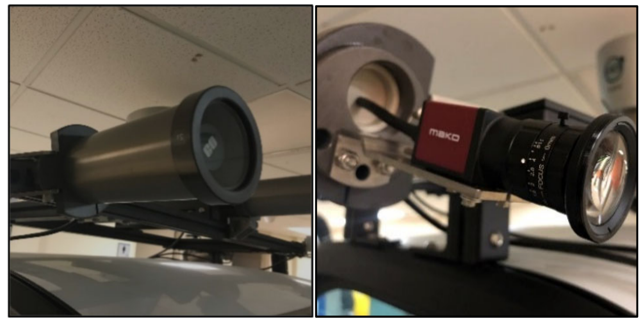

Figure 5: Mako Cameras with (left) and without (right) Waterproof Housing.

separate benchmarking tests. The first was a video encoding test, and the results demonstrated that at a marginal load, the computer consumed no more than 72 watts of power. To stress the computer, the team ran a program simulating running through an entire video game in less than 10 seconds. The results demonstrated that the computer only consumed 216 watts of power at high demand. Based on these results, the team determined that supplying the computer with 240 watts would be enough for the processing loads of the computer when executing autonomous functions. The cameras and the LiDAR connect to the computer through a switch via an ethernet cable. The GPS communicates with the computer via USB 2.0. The hardware architecture glass box diagram (figure 6)

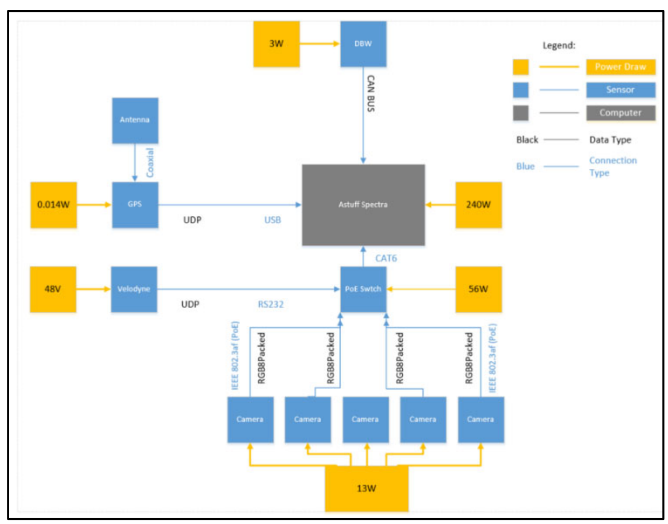

Figure 6: Hardware Architecture Glass Box Diagram

displays the component connection types and data message types.

2.3. Software Design (Vehicle Interface)

The software design consists of three sections: vehicle interface, sensor integration, and RTK enhancement. RTK initiates various systems via ROS launch files. These files can be nested inside each other, and this simplified the changing and starting up of the various subsystems. As a template, the team picked the Military RZR (MRZR) 8803 platform because of its similarity to the GEM e2 in size and functionality. Successful integration required modification of the basic control interface, the vehicle interface report, the gear status, the safety report, and the startup procedure.

In this model, RTK's Vehicle Management System (VMS) sends control commands and receives sensor inputs through PACMod, which is an interface to the Controlled Area Network (CAN) bus. Initial attempts to modify VMS to communicate with PACMod proved difficult and time-consuming. A new approach placed two custom nodes between them, acting as a bridge. These nodes called 'rtk_to_gem' and 'gem_to_rtk' successfully translated the data coming from one source into the expected type and format for the destination. This approach allowed for rapid development and testing. Figure 7 is a design diagram showing the flow from RTK to PACMod, and figure 8 illustrates the flow of information on the vehicle from a high level.

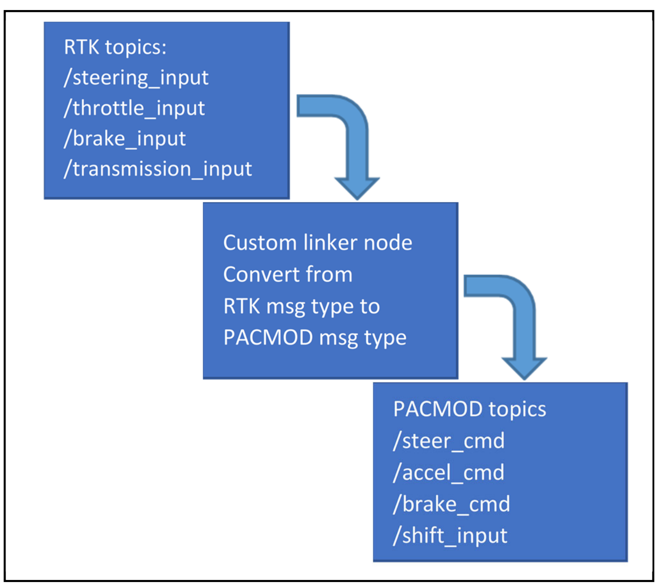

Figure 7: Vehicle Interface Software Flow.

Other changes were made to temporarily remediate issues that would prevent the vehicle from competing in IGVC. For example, the RTK 'gear_state_module' is responsible for checking for the proper transmission gear position. This electric vehicle does not have gears like the MRZR; instead, it has a selector with three positions: forward, neutral, and reverse. VMS was modified to prevent errors in RTK indicating a fault in the transmission, in such a way that the 'current_gear' is always equal to the value of the 'reported_gear.' For RTK to transition into autonomous driving mode, the transmission must be in 'park,' the engine status must be 'on,' and the vehicle must be stationary. Because the GEM-e2 did not have a transmission state of 'park,' this temporary hack allowed us to continue with the integration of the vehicle's sensors.

2.4. Software Design (Sensor Integration)

The launch files for RTK were configured for a Novatel GPS/GNSS and were modified to use the Xsens MTi-G-710 GNSS/INS. Since the Xsens already combines and filters the IMU and GPS readings, for initial testing, the RTK localization nodes were bypassed and replaced by the sensor readings directly from the Xsens. In later iterations, the RTK localization nodes were used; these also took into consideration the vehicle speed from PACmod and the transmission state.

Additional modifications to the launch files were made to remove the default RTK configuration of two Velodyne HDL-32Es and instead use the -64E. The Velodyne driver provided by RTK was modified and used with a single HDL-64E that provided both the front and rear sensor coverage. Using ROS rviz, we were able to verify that the

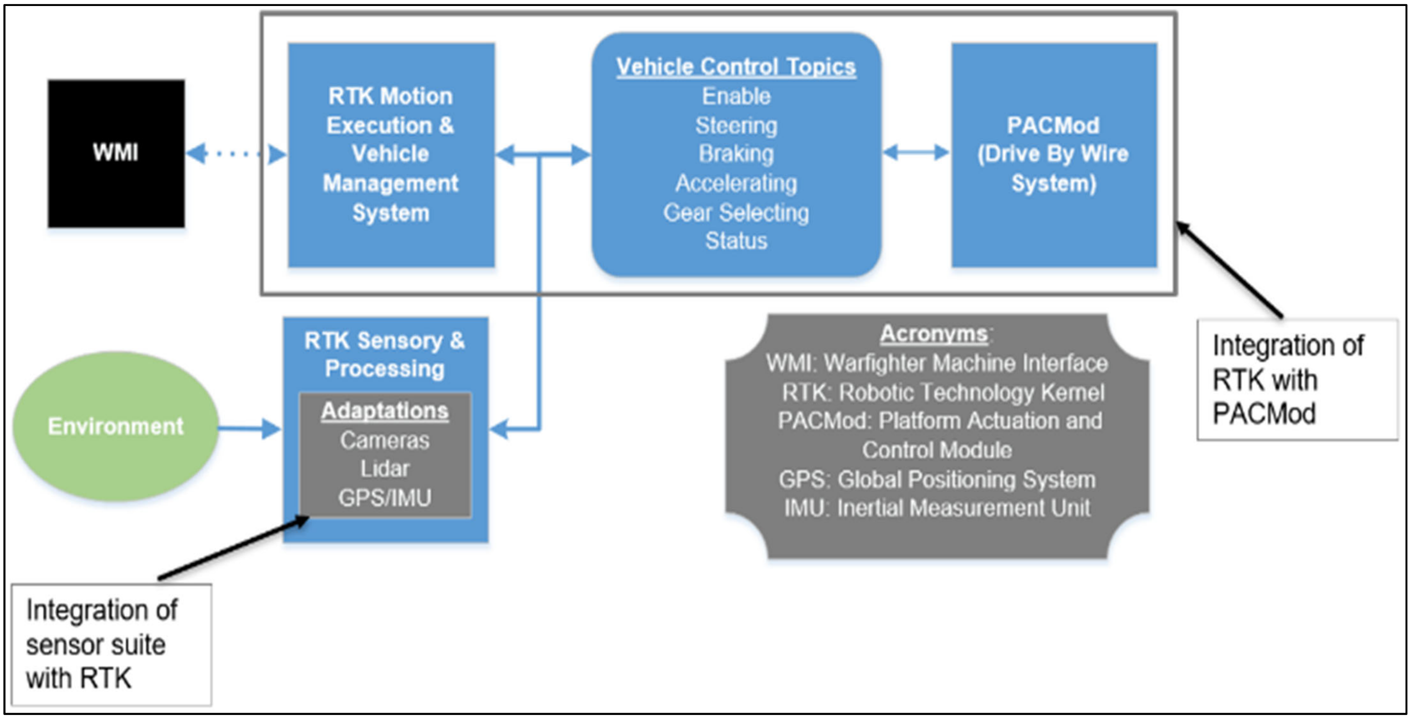

Figure 8: RTK Flow Diagram.

output provides 360 degrees of LiDAR coverage; however, initially, the RTK cost-map did not look calibrated. Additional changes to the LiDAR sensor transform parameters (yaw, pitch, and roll) were adjusted appropriately. When using the RTK Velodyne drivers, the transform values were taken from the Velodyne low-level launch file.

2.5. Software Design (RTK Enhancement)

Though the LiDAR successfully populated the cost-map in RTK, the vehicle experiences inconsistent reliability while interfacing with the Warfighter Machine Interface (WMI), which is GVSC's Interoperability Profile (IOP) compliant user interface software. During IGVC, the team identified that VMS needs to be further modified to manage the GEM platform robustly. The adapted MRZR configuration, used as a starting point, did not produce a consistent and reliable outcome. Further work is investigating the causes of irregular behaviors related to VMI and its interface with RTK.

USMA also implemented an extended version of lane-finding software developed by Udacity [9]. The first step consisted of calibrating the Mako cameras using the Open Source Computer Vision Library (OpenCV), which removed the distortion of the image caused by the curvature of the wide-angle lens. Next, as shown in the first quadrant of figure 9, a perspective transform takes a color image retrieved from the front-center camera (first quadrant) and reshapes it by following the red aiming lines, resulting in the picture in the second quadrant. Various thresholding techniques provide the contrast needed to identify lane lines (third quadrant), and a curve-fitting algorithm from OpenCV produces a superimposed trapezoid on the original image that defines the lane (fourth quadrant).

USMA's extension of this software comprised of a few aspects. First, the Udacity computer vision software became a ROS node that could retrieve images from the appropriate camera image topics. Secondly, the node casts the 2D lane lines, which

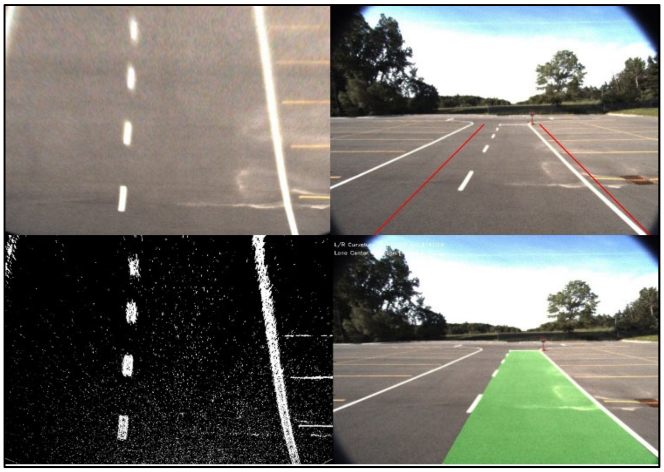

Figure 9: Image Processing for Lane-Finding.

were drawn on the final image using cartesian coordinates in units of pixels, to a three-dimensional point cloud that represents the location of the lane edges in real space relative to the vehicle. A coordinate transformation in conjunction with physical measurements of objects of various distances in the image resulted in a 3D point cloud that could then be retrieved by the RTK's path planning cost map (figure 10).

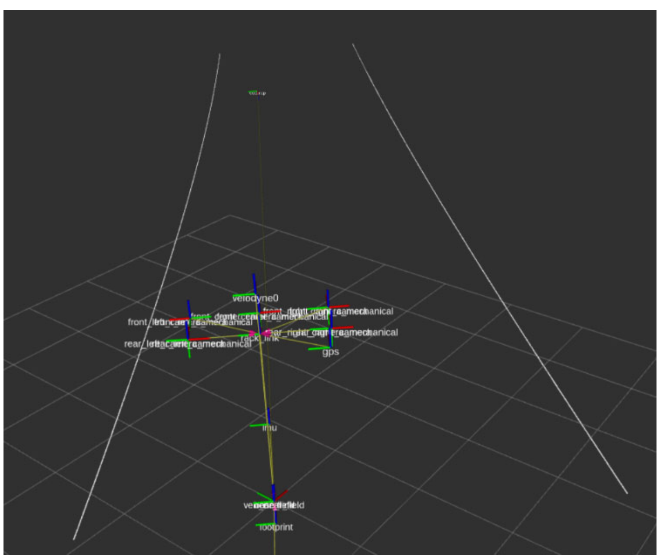

Figure 10: 3D Point Cloud of Lane Lines.

2.6. Simulation

The ANVEL simulation (figure 11), once linked with RTK, provided an MRZR vehicle model but did not include a model for the GEM e2; however, the team was able to utilize the MRZR for simulation purposes to test RTK because it closely resembled the GEM e2. The exact dimensions and specifications of the vehicle did not influence the success of the simulation, as the focus was on the integration of RTK rather than vehicle dynamics. Based on this assumption, it was not necessary to create a new model, though future work aims to create a dynamic GEM-specific model. During the design phase, it was necessary to use ANVEL to model the sensor placement and test using RTK to drive the vehicle and use waypoint navigation. After receiving the vehicle from AutonomouStuff, the team used ANVEL to integrate RTK while mounting the sensors installing the power supply.

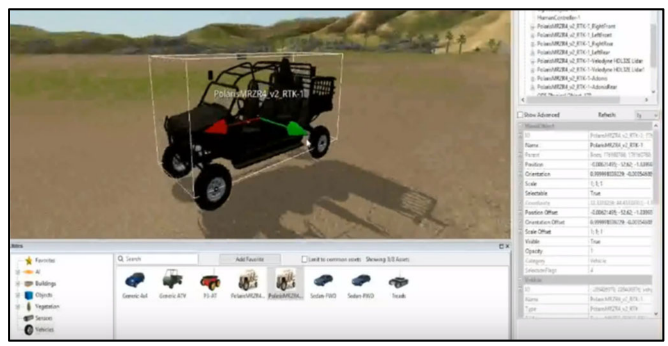

Figure 11: ANVEL Simulation with MRZR Model.

3. RTK Documentation & Lessons Learned

Finding documentation to help understand the interaction of RTK components remained one of the biggest challenges of this project. The amount and types of documentation for RTK vary amongst the different components. The types of documentation found were of three primary types, Markdown, inline annotations, and comments at the top of XML files. For example, much of the SUMET Platform Common nodes are documented using Markdown notation located in a 'doc' subfolder inside the 'platform_common' folder. This documentation is beneficial but lacks some of the details needed. Other sections of RTK had annotated source code, and documentation for this code could be generated by using tools such as Doxygen. The third type of documentation found were comments located inside of XML files such as the launch files or configuration files. The file 'localization.launch' located inside SUMET platform_common is an excellent example of this. Not all team members were aware of all three sources of documentation until late in the project. In hindsight, this challenge should have been solved earlier in the project.

An example of some of the challenges faced and eventually solved by finding documentation is the use of the terms 'far-field,' 'near-field,' and 'vehicle-near-field.' This caused some confusion amongst the students who were familiar with ROS. These terms describe a coordinate frame. Those students familiar with ROS new these as the frames 'map' (a global coordinate system), 'odom' (a world fixed frame based on the robots odometry) and 'base link' (a frame rigidly attached to the robot body) as described in ROS REP 105 [10]. This relationship became apparent when someone used the Linux grep tool to search for one of these terms in the RTK directory and found a note in the file CHANGELOG.rst located in 'dsat_mrzr' sub-directory. Using 'grep' to search for answers to questions became a common method to find documentation and again should have been taught to everyone earlier in the project.

4. IGVC Results

By June 2019, the USMA autonomy team demonstrated simple waypoint navigation at West Point but was unable to provide these navigation instructions reliably through WMI during the competition. The team successfully performed the emergency stop qualification task but was unable to qualify on the mobility tasks that required navigating the course.

As the only first-year competitor this year, the autonomy research team at USMA provided a unique perspective by using RTK as a true starting point for all aspects of the software design. This provided much value to GVSC's initiative to collect feedback on RTK and expand its capabilities.

5. AUTONOMOUS COMBAT ENGINEERING

RTK development within the context of combat engineering platforms represents an exciting opportunity to showcase the strengths of RTK while significantly enhancing the effectiveness of combat engineers, who are some of the most exposed soldiers on the battlefield.

Combat engineers perform three main functions: mobility, counter-mobility, and survivability—all of which require vehicular platforms to perform tasks related to earthmoving, gap crossing, and the emplacement or reduction of explosives and mines. Consistent with the recent prioritization of training and technological development to improve the Army's lethality in the offense and defense against a near-peer adversary, combat engineering functions play a critical role in the Army's ability to project and protect combat power.

The amount of combat engineer support organically assigned to combat units fluctuates according to need, but a look to the history of modern warfare provides some interesting insight relevant to the discussion today. As an example, during WWI, the United States sent 40 divisions of infantry or cavalry, each with its own engineer regiment, as more than 240,000 Army engineers served in Europe during the Great War [11]. The Corps of Engineers grew this enormous figure from only 256 officers and 11,175 enlisted at the war's beginning [11]. The Engineer Regiment again underwent rapid expansion during the Second World War, as the number of enlisted combat engineers grew from 6,000 to about 70,000 during the period from 1939 to 1941. Before America's participation, the peacetime engineer strength was one battalion per division. Over a year, by

December 1942, the number of engineer soldiers expanded once more from 93,109 to 333,209. At its peak, the engineer battalion retained the strength of 1,174 officers and enlisted, or about 8 percent of an infantry division [12]. Considering this history, a modern near-peer conflict might expect to see a similar wartime demand to expand engineer capabilities rapidly. Therefore, it would be the goal of autonomous engineer development to deliberately design a suite of software and platforms that could swiftly multiply engineer capability without a proportional growth in manpower.

5.1. Engineer Robotic Development

The Defense and Advanced Research Projects Agency (DARPA) shaped the autonomous research space indirectly through the Future Combat Systems (FCS) program, which was "the largest and most ambitious planned acquisition program in the Army's history," according to RAND's extensive history [13]. DARPA spearheaded the program from its inception upon receipt from the Chief of Staff of the Army, General Eric Shinseki. This was a $1B effort between DARPA and the Army, and "the goal was to develop truly state-of-the-art, revolutionary technologies. The approach was based at least partly on the assumption that extremely ambitious requirements would force engineers to develop innovative or breakthrough solutions, which, even if they fell short of formally established threshold targets, would presumably enable greater capabilities than if engineers were given less ambitious targets" [13]. Though canceled in 2009, the modernization effort envisioned a battlefield dominated by robotic vehicles and systems that enabled a very high level of situational awareness. The report goes on to assert very succinctly that, speaking of lessons learned relating to the testing of critical assumptions, "the Army did not have a clear grasp of which technologies were feasible and which were necessary and satisfactory to meet the needs of the future [13].

In 2000, Training and Doctrine Command (TRADOC) charged four industry teams to develop concepts and engineering models for brigade designs that would accomplish the vision set by DARPA and the Army, based on General Shinseki's original vision: "the force would consist of sub-20-ton vehicles and unmanned air and ground assets linked together by a seamless network of information that would enable information dominance and preemptive, decisive engagement with the adversary. It would be capable of direct and indirect fire, air defense, reconnaissance, troop transport, and would have nonlethal, *mobility/countermobility* and command and control capabilities" [13]. As a general assumption, protective armor could be substituted by superior mobility, connectivity, and situational awareness, and the initial design concepts were constrained by the transportability requirements of a C-130. One industry team, consisting of SAIC, Northrop Grumman, and Honeywell, proposed separate families of 16, 9, and 6-ton vehicles, with the 6-ton and 9-ton variants being unmanned and capable of deep insertion via sling load [13].

Though "robust countermine capability" was an original requirement for future combat vehicles, the surging IED threat ongoing in Iraq highlighted the reliance on unrealistic technologies that eventually doomed FCS. "The countermine [survivability] capabilities that would have been required to protect thinly armored vehicles reliably against most IEDs would have been impossibly high" [13].

In later iterations, the Unmanned Ground Vehicle (UGV) variant Multifunctional Utility/Logistics and Equipment (MULE), developed by Lockheed Martin, had three flavors in concept: transport, countermine, and light assault. All three depended on autonomous navigation and a common chassis. The countermine variant hosted the Ground Standoff Mine Detection System (GSTAMIDS) to detect mines with ground penetrating radar, and it could also mark and clear lanes for following ground vehicles [13]. By 2011, the Army canceled all three variants.

Not long after DARPA relinquished the lead of FCS, the agency organized a "legendary series of competitions...to bring out into the world technology that had been under development for decades in labs. There was military urgency at the time. The U.S. was fighting wars in Afghanistan and Iraq, and scores of soldiers were being killed by roadside bombs. Driverless vehicles could save lives on the front lines" [14]. Nevertheless, U.S. Army and Marine combat engineers persevered through more than 15 years of clearing roads and trails of explosive hazards in Iraq and Afghanistan. Often it was by means of regular patrolling on well-established routes with mine-resistant vehicles. In other instances, a combined-arms team of mounted and dismounted engineers and infantry would deliberately clear areas during named operations on marginally trafficable trails. Occasionally, route clearance packages would employ robotic systems like the DOK-ING MV-4, but most explosive reductions occurred through manual detection and the use of small, remote-controlled robots (like the TALON) to place a detonating charge from a safe distance.

In light of the heavily manual process of breaching or route clearance still today, an exciting development has been the Robotic Breach Concept emanating from GVSC, resulting in demonstrations of autonomous breaching and obscuration vehicles at the Yakima Training Center, WA in May 2019 [15] and previously at Grafenwoehr, Germany in April 2018 [16]. These efforts support the U.S. Army Strategy for Robotic and Autonomous Systems (RAS), one goal of which is to “facilitate movement with improved route clearance” [17]. The strategy clearly outlines that is important that the Army “invests in capabilities for route clearance, breaching, and C-IED,” and mentions objectives for protection that include Explosive Ordinance Disposal (EOD) operations [17]. In the mid-term (2021-2030), the strategy foresees “the first increments of RAS enabled combat platforms will have optionally-manned, teleoperated or semiautonomous technology.” And later in the future (2031-2040), it describes a situation where “autonomous systems, fully integrated into the force, allow Soldiers and leaders to focus on the execution of the mission rather than the manipulation and direct task control of robots” [17].

While the RAS strategy is critical, it may not adequately anticipate the full scope of mobility, counter-mobility, and survivability requirements that would likely be necessary to win a near-peer engagement involving offensive and defensive operations. With the history of the modern conventional war as a guide, the RAS and the Army's research efforts, in general, should recognize the historical tendency of rapidly increasing engineer support during intense armed conflicts.

5.2. Vision: Engineer Employment and RTK

The historical fluctuation of demand for engineer assets motivates a framework for robotic or semi-autonomous engineering vehicles that would provide commanders the flexibility to rapidly increase engineer capabilities without the overhead of training new soldiers. For example, consider a deliberate defense by an armored brigade combat team (ABCT) against another armored force that outnumbers the defenders by a typical 3:1 ratio. In this situation today, an ABCT in the U.S. Army would heavily rely on temporarily assigned engineer assets that would come from “echelons above brigade” (EAB) units. Current engineer doctrine admits that the organic engineer capability in an ABCT could not support the intensive requirements of a deliberate defense [18]. Fighting limited conflicts seems feasible under this model; however, the eventual exhaustion of EAB assets would presumably become inevitable within the context of sustained conflict. Therefore, the ability to rapidly employ and integrate unmanned engineer assets could effectively bridge the capability gap that would otherwise prevent an ABCT from winning while unsupported.

Turning a piece of terrain into a defendable landscape can involve dozens of mine-laying and earthmoving machines, and while additional unmanned assets would increase the overall work capacity of the unit, they would also increase the efficiency of operations by removing a significant contributing factor to delays – fatigue of human operators. Within a limited span of time, perhaps a few hours or possibly days, an ABCT's digging and mine-emplacing assets would have to be pushed to their limits to accomplish maximal effects (e.g. blocking or turning) on the battlefield before the engagement begins, while also fulfilling friendly survivability demands (fighting positions for tanks, soldiers, artillery pieces, etc.). Under the best circumstances, this is a highly complex operation that could gain a significant advantage from a framework that leverages autonomy.

Engineer officers must design, wargame, and then manage the execution of a construction effort to shape the engagement area (EA). These three aspects are conducive to the implementation of an intelligent robotics solution that synchronizes and optimizes engineering assets, both in the design of the EA but also in the operation and sequencing of tasks. In a vision, an engineer officer would design several EA courses of action by drawing tactical or protective obstacles and fighting positions on a digital map with terrain information, and then select attributes to each (e.g. density of mines, depth of tank ditch, type of survivability position, etc.). The engineer could then leverage intelligence information on the enemy and wargame the designs for the commander, who could then select the course of action to execute. Lastly, the engineer would then use this framework to deliver instructions to the engineer assets assigned to the overall effort and employ the fleet with an optimized solution based on real and changing conditions (workload, travel time, terrain conditions, etc.). The endstate could be one of these two possibilities: the ABCT constructs the most robust defense possible in the given amount of time, or it builds the fewest amount of defensive structures required to repel the enemy in the shortest amount of time.

A vision like this is certainly beyond the scope of RTK; however, a common framework for the control and synchronization of a system of vehicles would be a critical piece of the effort – one that GVSC has demonstrated using RTK in the Wingman and Leader-Follower demonstrations [19].

Furthermore, USMA's effort in expanding and implementing RTK provides a model for other stakeholders throughout the Army. In about eleven months, eight cadets and four active duty officers (two of which are still serving in an operational capacity) sourced a vehicle platform and deployed RTK to perform teleoperation and waypoint navigation. An extension of this accomplishment, though requiring support from the CCDCs and especially GVSC, is that the development and expansion of RTK could be accessible to operational units, especially because of the framework's platform-agnostic design and basis in ROS. Selecting soldiers and officers with an aptitude for robotics or other technical skills, and providing them with additional training, operational commanders can install an autonomy kit and implement RTK on any vehicle within their motor pools to develop the functions and behaviors that they deem to be most valuable. As part of the previously discussed vision, this would be the entry point for RTK-integrated combat engineering vehicles with supplementary modules for tactical mine emplacement, earthmoving, and obstacle breaching.

1. REFERENCES

[1] Global Electric Motor Cars. "GEM e2, GEM e4, GEM e6, GEM eS, GEM eL, GEM eLXD, SERVICE MANUAL." Polaris [2011 - 2012], 1 - 189.

[2] "Multiplexed Vehicle Electrical Center (mVEC)", Eaton.com, 2019. [Online].

[3] "AStuff Spectra, Computing-AutonomouStuff, LLC", AutonomouStuff LLC, 2018. [Online].

[4] HDL-64E S2 and S2.1 User's Manual, 4th ed. 2018.

[5] Detail, - Allied Vision, 18-Oct-2016. [Online].

[6] "MTi-G-710 - Products - Xsens 3D motion tracking", Xsens 3D motion tracking, 2018. [Online].

[7] "The 27th Annual Intelligent Ground Vehicle Competition (IGVC) Self-Drive", IGVC, version 26, November, 2018.

[8] "Open-Source Automated Benchmarking," Phoronix Media, 2019.

[9] "Advanced Lane Finding", Self-Driving Car Engineer Nanodegree Program, Udacity, July, 2018. [Online]. Available: https://github.com/udacity/CarND-Advanced-Lane-Lines. [Accessed July 2018].

[10] "Coordinate Frames for Mobile Platforms", ROS.org, October, 2010. [Online]. Available: https://www.ros.org/reps/rep-0105.html. [Accessed 25 June 2019].

[11] D. Wilson, "An Unprecedented Effort: Army Engineers in France During the Great War", The Military Engineer, Vol. 110, No. 718, Alexandria, VA, 2018.

[12] B. Coll et. al., "The Corps of Engineers: Troops and Equipment", United States Army in World War II, The Technical Services, Center of Military History, U.S. Army, Washington D.C. 1988.

[13] C. Pernin, et. al., "Lessons from the Army's Future Combat Systems Program", RAND, 2012.

[14] D. Hull, "The PayPal Mafia of Self-Driving Cars Has Been at It a Decade", October 2017. [Online]. Available: https://www.bloomberg.com/news/features/2017-10-30/it-s-been-10-years-since-robots-proved-they-could-drive. [Accessed 18 May 2019].

[15] "GVSC Lead Integrator at Robotic Complex Breach Concept", Defense Visual Information Distribution Service, May, 2019. [Online]. Available: https://www.dvidshub.net/video/680291/gvsc-lead-integrator-robotic-complex-breach-concept. [Accessed 18 May 2019].

[16] "Multinational Exercise combines New Technology, Soldiers", Defense Visual Information Distribution Service, April, 2018. [Online]. Available: https://www.dvidshub.net/news/272109/multinational-exercise-combines-new-technology-soldiers. [Accessed 18 May 2019].

[17] "The U.S. Army Robotic and Autonomous Systems Strategy", U.S. Army Training and Doctrine Command, March, 2017.

[18] "Engineer Operations – Brigade Combat Team and Below", Army Techniques Publication No. 3-34.22, December, 2014, 3-11.

[19] K. Mills, "Army's S&T Investment in Ground Vehicle Robotics", TARDEC Ground Vehicle Robotics, April, 2018. [Online]. Available: https://ndiastorage.blob.core.usgovcloudapi.net/ndia/2018/groundrobot/MillsPT1.pdf. [Accessed 25 June 2019].