2019 NDIA GROUND VEHICLE SYSTEMS ENGINEERING and TECHNOLOGY SYMPOSIUM
Autonomous Ground Systems Technical Session
August 13-15, 2019 - Novi, Michigan

A practical application of decentralized UGV teaming: Autonomous Convoy Parking

Rich Mattes, Kurt Bruck, Anthony Cascone, Dave Martin

Neya Systems

ABSTRACT

This paper presents Neya's efforts in developing autonomous depot assembly and parking behaviors for the Ground Vehicle Systems Center's (GVSC) Autonomous Ground Re-supply (AGR) program. Convoys are a prime target for the enemy, and therefore GVSC is making efforts to remove the human operators and make them autonomous. However, humans still have to manually drive multiple convoy vehicles to and from their depot parking locations before and after autonomous convoy operations – a time-consuming and laborious process. Neya systems was responsible for the design, development, and testing of the autonomous depot assembly and disassembly behaviors, enabling end-to-end autonomy for convoy operations. Our solution to the problem, including the concept of operations, design, as well as approaches towards testing and validation are described in detail.

1. Introduction

The Autonomous Ground Resupply program (AGR) seeks to develop the capability for 10 or more Palletized Load System (PLS) vehicles to autonomously navigate in a convoy between Army operating bases. While autonomous convoy transit is the primary goal, an often-overlooked challenge within the operation of autonomous convoys is the significant manual effort required to assemble and disassemble the convoy before and after autonomous mission operations. Vehicles often start in a tightly packed parking depot area, and human operators must manually drive each vehicle to its starting position within the convoy. When a convoy arrives at its destination, each of the vehicles must be manually moved to its parking position. Automation of these currently manual processes would save soldier time and provide end-to-end autonomy to the robotic convoy problem.

Neya's task for AGR Increment 2 was to solve these last-mile problems and enable fully autonomous convoy operations. The first developmental priority during the project was the Depot Assembly capability: autonomously orchestrating an exit from a depot parking area to a convoy. The second priority was to develop a corresponding Depot Disassembly capability: autonomously parking the vehicles at the conclusion of a convoy mission.

The AGR program is designed to leverage a significant amount of simulation followed by testing on physical vehicles. Each AGR program increment culminates with a soldier experiment and

Distribution A. Approved for public release; distribution unlimited. OPSEC # 2719

refinements are made based on experiment feedback.

Neya has developed decentralized teaming behaviors that satisfy these program requirements and allow the AGR vehicles to autonomously navigate between a convoy formation and a depot parking area. Vehicles autonomously assemble on command from their parked locations to a convoy formation at the start of the convoy mission, and upon arrival at their destination the vehicles can then autonomously park in positions specified by a human operator. These capabilities have been implemented as assignments within the overall AGR Autonomy Kit system architecture[1] and are invoked at the start and end of a convoy mission.

Figure 1: An often-overlooked challenge: navigating from a Depot to a convoy formation.

A critical element in developing a practical application of decentralized teaming to the depot parking problem was to rely on limited input from the convoy operator to greatly reduce the size of the problem space. The AGR autonomous convoy always has a "human in the loop" in the lead vehicle of the convoy. By making a small tradeoff in the goal of 100% autonomous operations and requiring the operator to provide some additional inputs at the start and end of a mission, Neya was able to drastically simplify the problem and create practical, reliable, and predictable autonomous parking capabilities.

As these depot capabilities were developed, verifying them required extensive functional testing to ensure that the approach to the problem was robust to a variety of operational scenarios.

Demonstration and deployment of these advanced behaviors for soldier exercises requires integration into the AGR hardware/software platforms. Testing solely with the real vehicles is cost and resource prohibitive, so Neya developed a testing strategy that allowed for rigorous functional and integration testing in the lab, allowing us to maximize testing opportunities on-vehicle.

What follows is a discussion of the depot assembly and disassembly problem and strategies to reduce the design space in Section 2, Neya's solution including the Concept of Operations (CONOPS) in Section 3, design decisions made to support the CONOPS in Section 4, and the approach to and results of testing and validation in Section 5. Section 6 discusses the path forward for future development efforts.

2. Autonomous Convoy Parking: Reduction of the problem space

The problem space for the autonomous assembly of vehicles into a convoy is quite large. Achieving fully autonomous convoy assembly with minimal operator input (which vehicles are to be included in the convoy, and in what order) requires solving several open-ended technical problems.

First, vehicles must determine the proper convoy formation to assemble based on the commanded convoy order and the operating environment. This includes the positions of the vehicles relative to each other, and the overall size and shape of the convoy formation. The shape of the convoy formation may not necessarily always be a straight line – obstacles in the operating environment, or a lack of open space in the assembly area may dictate that the line of vehicles should go around corners or around obstacles. Determining the layout of the convoy formation in a congested depot quickly becomes a difficult geometric problem, compounded by the fact that the vehicles may not have an accurate map of the obstacles throughout the assembly area. Without accurate information about obstacles, the planned formation may need to change as vehicles encounter obstacles in the area.

It is also possible that a suitable staging area for a convoy just does not exist among the obstacles in the assembly area.

Once a convoy formation is determined, vehicles must navigate autonomously to achieve the convoy formation. During autonomous navigation, the vehicles must detect and avoid obstacles in their environment, both static and dynamic. When the environment is not known a-priori, the vehicles must discover obstacles in the area as they navigate to their goal position and react to obstacles that appear along the planned path of travel. Even if there is some a-priori information about the operating area, vehicles must continuously verify that the obstacle information is correct, and new obstacles have not appeared in their path of travel.

As vehicles navigate the area, they must not interfere with each other as they approach their goal points. If the last vehicle in a convoy assembles to its position first in a narrow corridor, it could block the other vehicles from getting to their convoy positions. Similarly, if two vehicles approach the goal formation at the same time, they must coordinate such that the vehicle further back in the formation yields to the vehicle further ahead in the formation. At minimum this requires each vehicle to maintain knowledge of the current and goal positions of the other vehicles, and requires predictive algorithms and/or data-sharing between vehicles so that the vehicles are all able to traverse to their goal positions without being blocked by other vehicles in the environment.

2.1. The leader's path: a practical implementation

Two of the primary issues in the problem space are the desired goal positions for the vehicles assembling to the convoy, and navigation through unknown space to that convoy formation. Solving both issues can be mostly avoided by having the human operator of the lead vehicle drive over the desired path for the rest of the convoy vehicles to assemble on.

The addition of a pre-defined path provides the exact shape of the final convoy formation to each of the following vehicles. Inter-vehicle gap distances and the convoy vehicle order commanded from the lead vehicle can then be used to determine the goal points along the pre-defined path for each of the follower vehicles.

A pre-defined leader path also aids in the obstacle detection and avoidance problem. Since the leader's path was driven by the lead vehicle, it can be assumed to generally be free of static obstacles and safe to navigate for each of the follower vehicles. Followers can limit the risk of encountering potentially unknown obstacles by navigating to the nearest point on the leader's path, and then following the leader's path to their final goal positions.

2.2. Multi-vehicle deconfliction

The motion of multiple vehicles in the same area needs to be coordinated to ensure that vehicles can navigate to achieve their goal positions without hindering the efforts of other vehicles to achieve their own goal positions. Coordinating the motion of several large vehicles with limited maneuverability and communications bandwidth requires trade-offs between the time it takes for the vehicles to achieve their goals, and the likelihood of the vehicles to mutually interfere as they reach their goal positions.

The most straightforward approach is to constrain the problem such that only one vehicle is moving at a time. This coordination approach minimizes the likelihood of interference and reduces vehicles to stationary obstacles as other vehicles are moving. The trade-off is that vehicles take longer to achieve their goal positions in this serial fashion, especially if vehicles have a long distance to travel.

Incorporating information about the leader's path, it is possible to make a different trade-off to allow for slightly more parallelism in motion. Vehicles know that when their immediate leaders are on the leader's path, their course of travel will be along that path. This additional information provides a method for a vehicle to plan its own motion in response to the observed motion of its immediate leader. Permitting a vehicle to move when its immediate leader is on the leader's path allows for multiple vehicles to move in the environment, while only one vehicle is planning through free space (off the leader's path) at a time. When a vehicle's immediate leader is on leader's path, the vehicle can use simplified logic to maintain its spacing relative to its immediate leader to plan and coordinate its motion.

3. Behavior Concept of Operations

The following discussion outlines the CONOPS for both assembly and disassembly. These CONOPS describe how the behaviors are intended to be used in the context of a full convoy mission, and are designed to provide inputs from the human operator to limit the problem space for autonomous operation as previously discussed.

3.1. Assembly

The decentralized planning system requires four critical inputs which are readily available within the AGR convoying operation. The first is a predefined vehicle convoy order which is provided by the operator at the start of the mission. Each vehicle is provided information on its immediate leader (the vehicle directly in front of it in the convoy) and the convoy's overall leader, which allows for each vehicle to implement a decision model to determine a coordinated exit taking in consideration the state of its immediate leader. Second, the location of each vehicle is known and is provided via the local mesh networking radio links paired with the GPS based localization systems on each vehicle. Third, local costmaps representing the immediate surroundings of each vehicle are acquired from the AGR Autonomy Kit's local sensing system, which are used in basic obstacle detection and avoidance when generating local path plans. And finally, the operator provides an initial "seed" trail through the parking area by driving the lead vehicle along the desired convoy assembly path. Upon initiation of the behavior, all the vehicles begin to simultaneously coordinate to position themselves as dictated by the initial order provided by the operator.

The expected sequence of events is shown in the following figure. While the implementation is robust enough to support different sequences, the notional operational use is broken into three steps:

1. A human operator configures a convoy on the Warfighter Machine Interface, assigning vehicles to their expected positions within the convoy.
2. The manned leader is driven through or near the parked unmanned vehicles. This creates a target path for each follower to join.
3. Once the manned leader is positioned at the head of the convoy, the assembly behavior is activated and followers plan paths to intercept the leaders trail, in the sequence specified in the configuration.

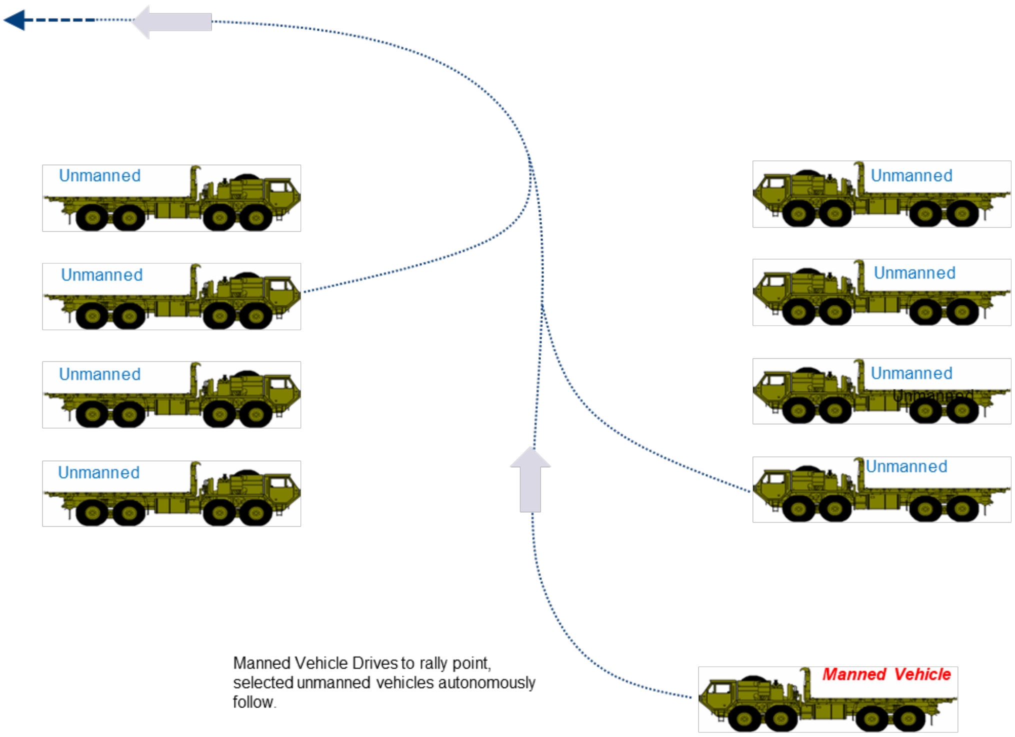

Figure 2: Depot assembly concept of operations: leveraging the human driver at the start greatly simplifies the problem space.

3.2. Disassembly

At the conclusion of a convoy mission, the vehicles must be positioned into a depot parking area. In this case, it would not be possible for the operator to provide a seed trail for each vehicle to achieve their parking positions, as the operator would often park as well, ultimately leaving an unreliable trail for the following vehicles to utilize. Instead, Neya developed a capability for the operator to provide a disassembly zone: a geo-fenced region of space where the vehicles are allowed to move autonomously and freely. Consequently, followers must remain in formation along the leader's path into the disassembly zone; once inside the zone, disassembly operations are executed. To further reduce the problem space, the operator provides a commanded parking location and orientation for each vehicle to park in, leading to a practical and reliable solution that employs a decentralized teaming technique.

Figure 3: Operator parking template for disassembly.

The notional disassembly sequence can be decomposed into four steps:

1. Prior to arriving at the depot, the human operator specifies the disassembly zone and assigned each vehicle is a parking location and orientation within the zone.

2. The manned leader is driven into the zone to establish a safe point of entry for followers. At that point, the manned leader is free to park or exit the zone, so long as it does not block the path of incoming followers.
3. The first follower will enter the zone along the leader's path but is then free to plan and navigate within the zone as needed to achieve the specified parking position. While the follower is parking, additional followers are permitted to approach the parking zone but may not enter it.
4. Once the follower is parked, the next follower in the convoy may enter and navigate within the zone. By limiting the parking zone to one moving vehicle at a time, the planning search space and risk of inter-vehicle interference is greatly reduced.

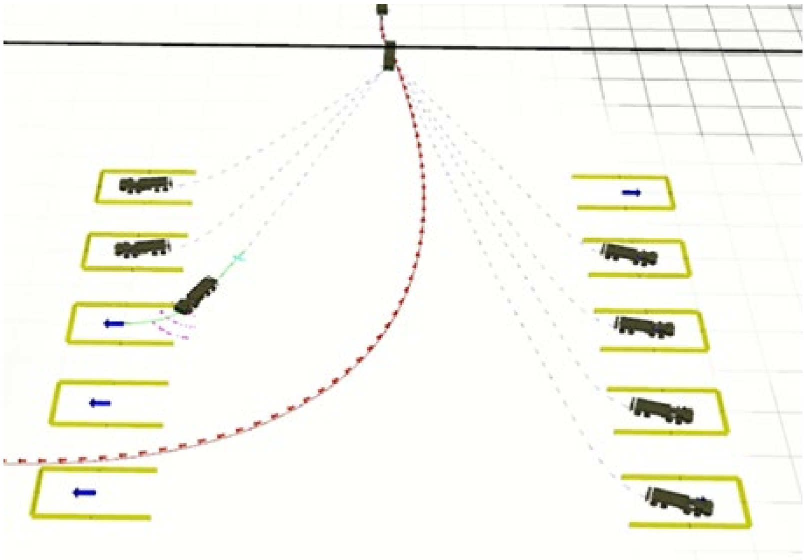

Figure 4: Simulation of Depot Disassembly

4. Software Design Decisions

This section discusses some of the key design decisions that were made to facilitate the operations described in Section 3. The strategies to reduce the design space discussed in Section 2 played a large role in shaping the overall design of the assembly and disassembly capabilities.

4.1. Inter-Vehicle Coordination

A major operational constraint on the AGR system is the limited communications available between vehicles. These vehicles operate in harsh conditions over many kilometers and must share the limited inter-vehicle communication bandwidth among the various inter-vehicle processes running in the Autonomy Kit. As such, the depot behaviors have been designed to operate in a distributed manner, leveraging the existing network communications where possible. The assembly and disassembly behaviors only need to provide minimal additional information for coordination outside of the existing AGR inter-vehicle information, such as the leader's path discussed in Section 2.1. This combination of information was sufficient to allow a serial motion coordination approach discussed in Section 2.2.

For the assembly behavior, we made the decision to limit movement such that only one vehicle may approach the leader's path at any time. By only letting one vehicle navigate through open terrain at a time, vehicles do not risk interfering with each other's movement, and do not need to coordinate to ensure they are not planning on traversing the same space at the same time. A vehicle only needs to monitor the position of its immediate leader in the convoy: once the immediate leader joins the leader's path, the vehicle is free to move towards the leader path, and may join the leader's path when it would result in the follower being behind its immediate leader.

Disassembly uses a similar approach to de-conflicting vehicle movement. When the vehicles are commanded to disassemble, the operator provides a geo-fence for the area in which vehicles are free to autonomously maneuver, and a list of all vehicles' assigned parking locations. Followers follow the leader path to the point at which it enters the parking geo-fence, after which each follower waits for its immediate leader to come to a stop in a pose near the assigned position and orientation before proceeding to its own parking position. Waiting for the immediate leader to achieve its parking position before maneuvering in parking area reduces the need for vehicles to coordinate their motion through the parking area.

4.2. Motion Planning

The Convoy Assembly and Disassembly behaviors are designed to extend the convoy functionality of the AGR Autonomy Kit, where the core design assumption that followers should utilize the convoy leader's path as much as possible.[1] This holds for normal leader-follower type convoy operations, however, with assembly and disassembly the followers will either start or end at a position off of the leader's path.

Given that the leader's path is considered the safest corridor of travel, the behaviors include path planning algorithms designed to minimize the distance travelled off the leader's path. In assembly, this means the planner generates as short a path as possible through free space to join the leader's path. In disassembly vehicles follow the leader's path for as long as possible before breaking off and parking in their assigned parking positions. Free motion in disassembly is limited to the geo-fenced parking area, which allows operators to constrain vehicle's motion planner and prevent the vehicle from traveling in undesired areas. This allows us to extend the idea of a "safe to maneuver" area beyond the leader's path and provides the planner with freedom to achieve the desired parking location within a space that can be safely monitored by the operator.

The path planner will only search for paths where the vehicle can drive forwards. This decision comes from the sensor configuration of the autonomy kit on the AGR PLS platform. The main perception system provides forward looking sensors only and does not provide adequate rear facing sensing due to the payload carrying mechanics. Since motor pools will have people walking throughout the parking area, the safest way to navigate through the environment is to drive in the direction with highest fidelity perception.

5. Software Testing and Validation

As the design was developed and implemented, a rigorous test strategy was required to ensure that the design was sound, and that the implementation was robust. The limited availability and prohibitive cost of testing on PLS vehicles required that the test approach included testing on a platform representative of the PLS, to verify our software as much as possible before integrating with the real vehicle. Our test strategy included static analysis, software unit and integration testing, system performance testing, system simulation testing with representative PLS model and Autonomy Kit software in the loop (SIL), and system integration testing on the PLS platforms.

5.1. Software Component Testing

Static analysis and linting was employed to enforce software development practices in a consistent manner and to help avoid common pitfalls. Each developer runs these tests on their own development system, and the tests are also executed automatically by a continuous integration build server that runs a full suite of tests on various development branches nightly. By using a common standard and peer review, we continuously improve our development practices and ensure that all lines of code have been developed in a consist manner.

Each piece of software was written with the ability to be unit tested to verify the implementation behaves as desired. Complex functions and algorithms are implemented to a specification that describes the operation of the algorithm, and tests are generated based on the specification. Integration tests are then added on top of the unit tests to verify the behavior of whole software components against their interface specification.

5.2. System Integration Testing

For integration testing with vehicle systems, the AGR project worked with contractors to develop a high-fidelity Software-in-the-Loop (SIL) simulator based on ANVEL. Contractors developing behaviors for AGR were provided with computers configured with the simulated representation of the PLS and virtual machine running Autonomy Kit software in a configuration mirroring the deployed systems. Each computer was configured to simulate one vehicle in real-time and can communicate with additional vehicles over ethernet, mimicking the mesh network of the real PLS vehicles.

The SIL simulation testbed proved to be an invaluable test fixture. It primarily served as a surrogate for integration with the AGR Autonomy Kit software, in lieu of running on real PLS hardware. Test scenarios reflecting assembly and disassembly operations were built and executed manually in simulation and provided valuable feedback on the interaction between our software and the Autonomy Kit software. These efforts allowed us to find and fix bugs in the interaction between our assignment and the Autonomy Kit software months before field testing took place.

The SIL tests also provided a way to simulate the execution of field tests before going to the field, allowing us to verify test setups and operating areas before arriving in the field.

5.3. System Performance Testing

Although these SIL simulation systems provided a highly accurate representation of the actual vehicles, they did not easily facilitate rigorous performance testing of the assembly and disassembly algorithms in a variety of scenarios. A key component of the test strategy was algorithm performance testing a 2D simulation environment based on the Stage simulator. This relatively light weight 2D simulation allowed for testing with as many vehicles as desired, in a large variety of starting geometries, leader paths, etc. Per the AGR Increment 2 program goals, we incorporated 10 vehicles (one leader and nine followers) for our simulation tests.

This 2D simulation testing was designed to evaluate mission success of the system independent of the real-time performance of the system, and of integration with the vehicle hardware and software. The simulation test framework consists of a test matrix that varied parameters such as starting and ending locations and orientations of the vehicles, leader path, map, convoy order, and gap distances. These combinations of scenarios create unique interactions that revealed edge cases, limitations, and overall capability of the behavior implementations. It also contains a test application that provided operator inputs at appropriate times according to the CONOPS and evaluates whether the vehicles achieved their mission within a specified timeframe.

Stage simulation test scenarios run in parallel, allowing a large suite of tests to be run in a fraction of the time it would take to execute them manually on the ANVEL simulation test bench. They were also automated to run with the latest development code each night to evaluate the performance of the software over time as it was developed, and to identify any regressions, and areas of improvement. The test suite was also run over proposed code changes before they could be merged into the main codebase, to identify any sources of regressions before code was incorporated into the mainline source branch.

5.4. Results

At present, with the over 400 automated unit, integration, and simulation tests we run, we have less than 4 simulation scenarios that fail. We are working to address these known issues while continuing to extend the scenarios in our test suite to further stress the system. This overall testing strategy led to the successful integration at Fort Bliss in May 2019, where we were able to demonstrate both assembly and disassembly with four PLS vehicles in a single week of on-vehicle field testing. Further, the extensive testing of the software in various configurations increased its reliability, such that we encountered no unexpected software faults or crashes during the Fort Bliss testing event.

6. Next Steps

As the base assembly and disassembly behaviors are finalized for soldier evaluation, there are still significant areas of expansion to improve the adaptability of the behaviors to more complex operating environments.

As it currently stands, each vehicle is responsible for maintaining its own map of obstacles in its immediate area, derived from the LIDAR and other sensors on the vehicle. Provided that there are numerous configurations of operating bases in which the AGR vehicles are used, there is likely to exist a scenario where the manned vehicle will not have the ability to drive near all of the vehicles that need to be assembled into a convoy, as shown in Figure 6. Vehicles assembling through large swaths of unknown space would benefit from sharing information about detected obstacles between each other, such that the first vehicles to navigate through an unknown space can map out the space for followers to generate better plans to their goal positions. Handling these cluttered operating base scenarios via advanced coordination strategies such as sharing world model maps between vehicles, and planning through large congested areas are areas of interest for further advancement of the assembly behavior.

Another area of interest is removing the requirement that the vehicles must always drive forward in the assembly and disassembly assignments. Integration of backing up and multi-point turns would expand the capabilities of the system in a number of ways. In assembly, multi-point turns can be used to further extend the ability of vehicles to navigate in congested areas. When a vehicle encounters an obstacle that it cannot avoid while driving forward, the ability of the vehicle to back up away from the obstacle would prevent the vehicle from getting stuck in place. In disassembly, there are scenarios where vehicles must back into their parking positions due to the presence of obstacles behind the parking position (e.g. a wall, fence, or another line of trucks.) Adding the capability to back into a parking position greatly expands the usability of the disassembly assignment in cluttered parking areas.

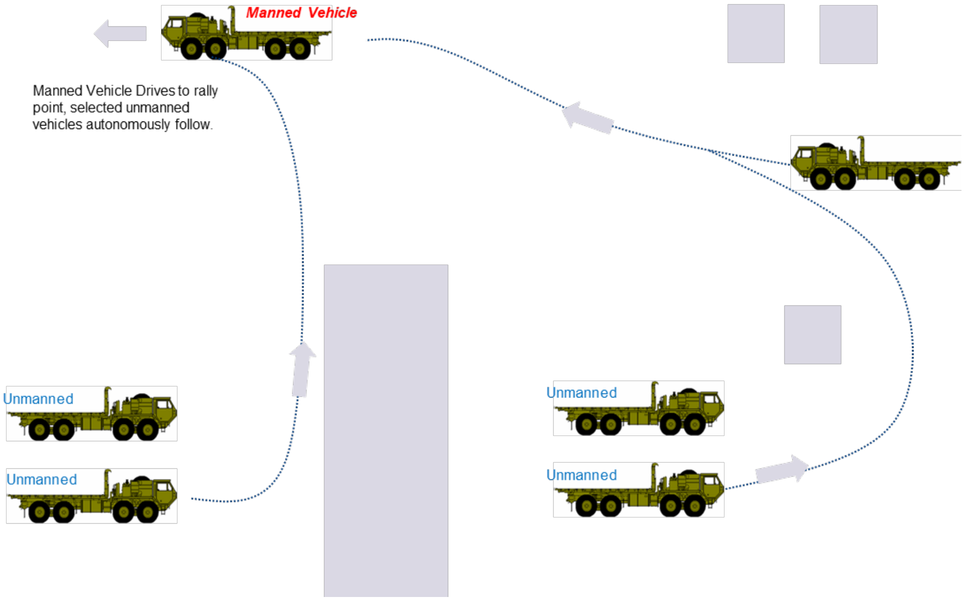

Figure 5: Advanced depot assembly in a tactical FOB.

7. References

[1] Anne Schneider, et al., “Autonomous Ground Resupply Autonomy Kit”, 2018 NDIA Ground Vehicle Systems Engineering and Technology Symposium, 2018.