Mitigating the Effects of Jamming on Autonomous Vehicle Convoys with Behavior-Based Robotics

by

Calvin Cheung

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in the University of Michigan-Dearborn
2023

Doctoral Committee:

Assistant Professor Alireza Mohammadi, Co-Chair
Associate Professor Samir Rawashdeh, Co-Chair
Assistant Professor Youngki Kim
Assistant Professor Jaerock Kwon

Calvin M. Cheung

cmcheung@umich.edu

ORCID iD: 0000-0003-3876-1684

Dedication

To my wife, Cindy and children, Ethan and Avery. Thanks for putting up with all the claims of "I'll be done in five minutes," that rarely proved to be true. From here on out, we can depart for ice cream expeditiously.

Acknowledgements

This effort would not have been possible without the support of my Dissertation Committee. I am deeply grateful for the leadership and guidance provided by my advisors, Dr. Alireza Mohammadi and Dr. Samir Rawashdeh. The direction and focus they have provided for the research have been invaluable. In addition, I would like to express my gratitude to Dr. Jaerock Kwon and Dr. Youngki Kim for taking the time and effort to provide their perspectives. Their insight has greatly improved on the depth and quality of the work. Finally, I would like to extend a special thanks to Dr. Stanley Baek and Dr. Jonathon Smereka. Both played a significant part in establishing the foundation of my research and academic career. Their support set me on an interesting path with meaningful challenges and continues to propel me forward to this day.

Table of Contents

List of Tables

List of Figures

Abstract

Autonomous ground vehicle convoys are heavily reliant on radio communications when performing leader-follower operations. The lead vehicle sets the path and utilizes radio communications to send information such as path points, vehicle pose, vehicle speed, and other sensor data. Follower vehicles utilize this information to track the leader's path and mobilize to the proper positions. This reliance on radio communications makes autonomous ground vehicle convoys particularly vulnerable to network denial-of-service attacks, such as radio jamming. Jamming is a type of denial-of-service attack that attempts to disrupt or block wireless communications, which interferes with a radio's ability to transmit or receive data.

The contribution of this dissertation is to improve the performance of autonomous ground vehicle convoys when facing radio jamming attacks by utilizing a controls-oriented approach. To mitigate the effects of jamming attacks on autonomous convoys, we propose a behavior-based architecture named the Behavior Manager. The Behavior Manager utilizes layered costmaps and vector field histogram motion planning to implement motor schema behaviors. By utilizing the Behavior Manager, multiple behaviors can be created and combined to form a convoy controller capable of persisting with convoy operations while under a jamming attack. Based on a thorough review of relevant literature, this is the first time that techniques from behavioral robotics are being utilized to mitigate the effects of jamming attacks in any capacity. In addition, we propose a framework for comparative performance, named the Performance Metrics Framework, to gauge the performance of convoy systems. To develop the framework, we examined manned convoy requirements found in Army doctrine, along with common autonomous convoying

research metrics. By using the framework, we can categorize performance requirements into different priority areas and find relevant key metrics to use for performance comparison.

We conducted experiments to measure the performance of our Behavior Manager convoy controller in the face of radio jamming and utilized the Performance Metrics Framework in performing comparative analysis. In the experiments, simulated convoy runs were performed on multiple path plans under different types of jamming attacks. The experimental results showed that the Behavior Manager was able to improve the performance of autonomous convoys when faced with jamming attacks across all jammer types and path plans, ranging from 13.33% to 86.61% reductions in path error. These results show that a behavior-based robotics architecture approach can used to provide a controls-oriented layer of protection against radio jamming. When combined with common anti-jamming techniques, the Behavior Manager provides a robust, multifaceted defense against radio jamming.

Chapter 1 Introduction

1.1 Background

Ground vehicle convoys are widely employed in both commercial and military domains to reduce costs and increase transportation efficiency [1]. Fundamentally, a convoy is a group of two or more vehicles traveling from an origin point to an objective destination under a single leader [2]. The employment of ground vehicle convoys is an important part of an efficient supply logistics strategy. Despite many other transportation options, such as trains, aircrafts, and ships, being readily available, ground vehicle convoys still account for a significant portion of military supply distribution due to battlefield complexities and the need to include protective measures [3]. On the commercial side, ground vehicle convoys also play an important role in transportation due to the prevalence of paved roads and cost effectiveness when moving large or heavy materials [1].

In recent years, advances in autonomy have increased the viability of autonomous vehicles in transportation [4]. As the capabilities of autonomous vehicles has continued to grow, so too has the interest in leveraging autonomous vehicles to improve ground vehicle convoys. By incorporating autonomous vehicles in ground vehicle convoys, researchers seek to develop autonomous convoying capabilities that increase logistical efficiency, use less fuel, and reduce a convoy's carbon footprint [1]. There have been multiple efforts by civilian government organizations in this domain, such as the Netherlands' European Truck Platooning Challenge [5], the European Commission's Safe Road Trains for the Environment Project [6], and Singapore's "smart city" development [7]. These efforts, summarized in Table 1-1, focused on maturing autonomous convoy technology for improvements in safety, reduction in fuel consumption, and reduction of traffic congestion. In response to the growing demand, commercial entities have also focused on the development of autonomous convoys, with companies such as Peloton Technology, Daimler, Volvo, and Volkswagen [8] [9] researching and developing autonomous convoy solutions. Furthermore, there has been significant interest from the military in advancing the development of autonomous convoys systems, due to the risk reduction potential of decreasing the number of soldiers needed on the field through autonomy [10] [11].

Table 1-1. Summary of Various Governmental Autonomous Convoying Efforts.

Effort Name	Lead Organization	Description
European Truck Platooning Challenge 2016 [5]	Dutch Ministry of Infrastructure and the Environment	Drive automated platoons on publics roads across borders from production sites of European truck manufacturers to the Netherlands.
Safe Road Trains for the Environment (SARTRE) [6]	European Commission	Develop road trains that allow passenger cars to match movements to the distance, speed, and the direction of the car in front. Offload physical and cognitive duties from passenger cars to the platoon leader.
Singapore Full-Scale Autonomous Truck Platooning Trial [7]	Singapore Ministry of Transport	Autonomous truck platooning trials, in which fleets of trucks composed of three autonomous vehicles follow a manned vehicle to transport cargo between ports. Seeks to optimize road capacity.

At the most basic level, an autonomous ground vehicle convoy (AGVC) is composed of a lead vehicle and follower vehicles. The follower vehicles maintain formation and pace with the lead vehicle, per the overall convoy system requirements. This is typically accomplished through the sharing of sensor data and vehicle kinematics between the vehicles with inter-vehicle communications (IVC), which allows the separate vehicles to calculate and reach the desired speed and positions needed for the convoy [12]. The data is normally distributed through wireless networks, with different networking options being viable for information transmission, such as vehicular ad hoc networks (VANET) comprised of various connected vehicle technologies, such as Vehicle-to-Vehicle (V2V) communications and Vehicle-to-Infrastructure (V2I) communications [13], as seen in Figure 1-1. Many different standards, such as 3G/4G Long Term Evolution (LTE) cellular networks, dedicated short-range communications (DSRC) radios, or roadside wireless sensor networks, can be utilized to enable network communications. In a standard AGVC, reliable wireless communications are needed to perform convoy following, where disruption of communications can lead to a breakdown of the autonomous convoy. For this reason, it is critical to protect the wireless communications systems of autonomous convoys from potential cyber-physical attacks, such as radio jamming.

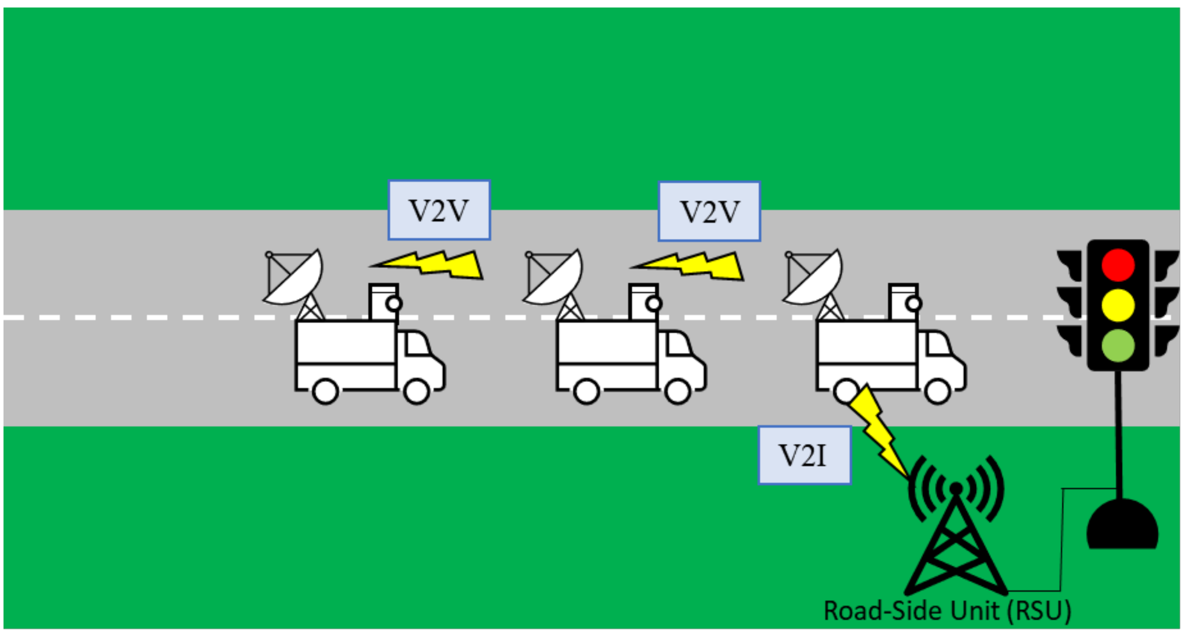


Figure 1-1 Simple example of VANET communications for an AGVC.

Radio jamming is a category of denial-of-service attacks in which the attacker disrupts a wireless communication system's ability to send or receive data. They are the most widely used denial-of-service attack against vehicular networks due to their simplicity and effectiveness [14] [15] [16]. Because AGVCs are reliant on IVC, radio jamming has the potential to cause complete failure of convoy following operations [11]. Due to the severity of consequences that wireless jamming can cause, it is vitally important to build in measures for fault tolerance and recoverability in autonomous convoy systems, to mitigate the damaging effects of denial-of-service attacks.

1.2 Objectives

The overall objective of this dissertation is to improve the performance of AGVCs when facing radio jamming attacks. We aim to accomplish this by focusing on mitigating the effects of IVC jamming with a controls-oriented approach via the creation a behavior-based robotics autonomous convoy system. The behavior-based approach will be robust to system error and loss of communications by using only the on-board sensors of each vehicle when under jamming attacks. Towards this objective, we establish a framework to compare convoy performance based on requirements found in military doctrine and common autonomous convoy performance metrics. This framework can be utilized to compare the performance of current convoy systems with future developments, ensuring measurable progress as we continue to iterate and improve on autonomous convoying technology. The collection of efforts described in this dissertation work towards the end goal of accelerating the development and adaptation of AGVCs so that the potential cost, safety, and environmental benefits made possible by AGVCs become fully actualized.

1.3 Thesis Organization

The remainder of this dissertation draws from an ensemble of peer-reviewed research efforts, summarized in Table 1-2, conducted during the course of my doctoral studies to explore the topic of jamming mitigation for AGVCs via behavior-based robotics.

Table 1-2. Peer-reviewed research efforts completed towards dissertation.

Paper Title	Dissertation Chapter	Major Contributions
Delivery of Healthcare Resources using Autonomous Ground Vehicle Convoy Systems: An Overview *[3]*	Chapter 2	• Survey of ground vehicle convoy requirements from publicly available Army doctrine • Survey of common AGVC metrics used in research literature • Framework for determining what metrics to use for AGVC performance comparison depending on the goals of the system
Jam Mitigation for Autonomous Convoys via Behavior-Based Robotics *[17]*	Chapter 3	• Controls-oriented countermeasures against jamming attacks on AGVCs • Robot Operating System (ROS) behavior-based robotics architecture that combines Motor Schema behavior-based robotics with the ROS navigation stack • New approach for motion planning in Motor Schema behavior-based robotics with integration of vector field histogram motion planning, which avoids the pitfalls of potential field motion planning.

Chapter 2 details the creation and application of a framework of comparison for autonomous convoy performance. We provide a brief historical exploration on the necessity of ground vehicle convoys, followed by an investigation of manned convoy requirements. With an understanding of the necessity and requirements, we surveyed the field of AGVC efforts to establish a generalized framework to compare the performance between AGVC systems, called the Performance Metrics Framework., We will leverage this framework to compare performance of various autonomous convoy systems given different jamming scenarios. Chapter 3 proposes a behavior-based autonomous vehicle convoy architecture, called the Behavior Manager, that utilizes layered costmaps and vector field histogram motion planning to implement a Motor Schema architecture that allows for convoy operations to persist in the presence of IVC radio jamming. In this chapter, we provide a detailed review of common anti-jamming measures and behavior-based robotics to fully contextualize the system architecture and experimental design, followed by a discussion about experimental results and potential paths forward. Chapter 4 contains final discussions and conclusions.

Chapter 2 Developing a Performance Metrics Framework for Autonomous Ground Vehicle Convoys

In order to properly gauge performance of autonomous ground vehicle convoy (AGVC) systems, a proper framework for comparative performance metrics needs to be established. Past efforts in this domain have had heavy focus on narrow and specialized areas of convoy performance such as human factors, trust metrics, or string stability analysis. This chapter, based on our previously published work [3], reviews available Army doctrine for manned convoy requirements and establishes a framework to compare performance of autonomous convoys. The framework, which we call the AGVC Performance Metrics Framework, looks at the requirements found in doctrine, categorizes them, and identifies key metrics that should be used for comparison based on the categories prioritized. After developing the Performance Metrics Framework, we utilize it to compare the performance of two autonomous convoys with unique convoy control strategies to demonstrate its application and utility.

2.1 Introduction

From a military perspective, a ground vehicle convoy is a column of two or more vehicles under a single leader, traveling from a set origin to an objective destination [18]. Military utilization of convoys has a long history, with doctrine on convoy utilization for the United States (U.S.) Army being described as early as 1847 in “An Elementary Treatise on Advanced-Guard, Out-Post, and Detachment Service of Troops” [19]. While the battlefield and vehicles have changed drastically throughout the years, the purpose of convoys have remained consistent: to control road movements in order to meet various logistical needs, such as movement of supplies, personnel, and equipment [20]. Even though the topic of convoys has been thoroughly dissected and studied by the Army [21] [22] [23], the advent of autonomous vehicles has led to modernization efforts to improve convoys through the addition of autonomy. These efforts aim to improve convoy efficiency and performance, reduce the risks to the Soldier, and decrease the overall cost of operations [10].

In addition to military research, there are various other civilian organizations looking to develop and utilize AGVC systems. Efforts such as the Netherlands' European Truck Platooning Challenge [5] and the European Commission's Safe Road Trains for the Environment Project [6] demonstrated the interest of civilian governments in maturing autonomous convoy technology for improvements in safety, reduction in fuel consumption, and reduction of traffic congestion. In support of these efforts and commercial development, many companies, such as Peloton Technology [8], Scania [7], Daimler, Volvo, and Volkswagen [9], are researching and developing autonomous convoy solutions.

At a high level, AGVCs have a lead vehicle and follower vehicles. Follower vehicles keep pace and formation with the lead per system requirements. This is normally done through the sharing of vehicle kinematics, intended maneuvers, or sensor data (cameras, GPS, LiDAR, wheel encoders, etc.) between the vehicles, which allows separate vehicles to actuate appropriately to meet the desired speed and formation [12]. The data is distributed wirelessly via a variety of different methods, such as vehicular ad hoc networks (VANET), Vehicle-to-Vehicle (V2V) communications, and Vehicle-to-Infrastructure (V2I) communications [13]. Several different standards and protocols are used for network communications, such as dedicated short-range communications (DSRC) radios, 3G/4G Long Term Evolution (LTE) cellular networks, and roadside wireless sensor networks, to improve network coverage and throughput.

In order to properly gauge the performance of AGVCs, a proper framework for comparative performance metrics needs to be established. Past efforts in this domain have had heavy focus on narrow and specialized areas of convoy performance without considering the complex requirements of convoys performing logistical operations, such as human factors, trust metrics [24], or string stability analysis [25]. In addition, broad assumptions and simplifications were used in the analysis, such as the removal of lateral position considerations for convoy member vehicles, or the consistent existence in an information flow topology for robust inter-vehicle communications [26]. While the constraints, assumptions, and narrow focus areas of performance metrics that have been previously discussed are highly valuable for their intended purposes, a more generalizable approach is needed to compare performance across a larger swathe of AGVC systems. The goal of this chapter is to establish a framework for performance metrics of AGVC systems by performing a review of AGVC literature and comparing the findings to requirements found in Army doctrine relating to manned convoy systems. Based on these two critical pieces of information, we propose metrics to use so that convoy performance can be compared. We will start with a brief historical exploration of the needs for autonomy in ground vehicle convoys, followed by an exploration of overall manned convoy requirements. From there, we will survey the field of AGVC efforts to determine common threads in performance metrics to establish a generalized AGVC performance metrics framework to be used to compare the performance of future efforts. Due to military and commercial efforts having separate sets of needs and requirements, this paper will focus on AGVCs in the military domain to be able to perform an in-depth look at the topic area.

2.2 Need for Autonomous Convoys

The use of ground convoys to perform supply operations has been codified as an important part of an efficient strategy for the U.S. military as early as 1847 [19]. While modern military operations include a plethora of transportation systems such as rail lines, aircrafts, and helicopters, ground vehicles still account for a significant portion of supply and equipment distribution. This reliance on ground vehicles was evident in Operation Iraqi Freedom, in which 98 percent of the military's supplies and equipment were delivered by ground transportation [10]. In linear warfare scenarios, convoy operations are not prone to being attacked [20], where linear warfare is defined by conflicts in which opposing enemy forces generally proceeded forward. The geometry of a linear warfare battlefield implies that there is a "front" in which direct contact between forces are made, two "side" flanks that are often protected, and a secure "rear" area. Advancement in linear warfare means that forces at the front advance forward to clear and secure land. As the front moves forward, the non-combat assets in the rear progress as well, pushing forward and extending the secure rear [27]. Because the convoys are used in the rear, convoy operations are viewed as low risk in linear battlefields. Convoys are used to bring supplies from the rear to a forward position, traveling through secured areas that are far away from enemy combatants at the front battlefields. Many of the largest conflicts in U.S. history, including World War I, World War II, and Desert Storm, were linear warfare campaigns. Despite the relative safety of convoy operations in linear battlefields, many modern conflicts and military operations in peacekeeping and humanitarian efforts are on a nonlinear battlefield. In contrast to linear battlefields, nonlinear battlefields do not have a defined front and secure rear area. The battlefield has a 360º area of operation with the center being a main operating base. In addition, many modern conflicts are against combatants that are using asymmetrical tactics.

Asymmetrical tactics are strategies designed to harm a military's assets without going up against the primary defenses and forces [27]. A prime example of this is the targeting of unarmored convoys during supply operations with improvised explosive devices (IEDs), snipers, and sudden ambushes on stopped vehicles [19] [20]. Past examples of U.S. military operations in nonlinear battlefields against combatants using asymmetrical tactics include the Vietnam War, humanitarian efforts on Bosnia and Somalia, and the wars and conflicts in Iraq and Afghanistan throughout the first quarter of the 21st century [20] [27]. U.S. Department of Defense studies project most future conflicts will be on nonlinear battlefields against combatants using asymmetrical tactics, indicating a continued threat to the personnel and resources needed for convoy operations. Given the threat of asymmetrical tactics on nonlinear battlefields, the U.S. Army is looking to leverage AGCV systems for strategic and logistical benefits including reduction of danger to personnel and reduction in the costs of logistics.

The greatest threat to the safety of personnel performing convoy operations in a nonlinear, asymmetrical battlefield are IEDs. IEDs are the main cause of battlefield casualties in Iraq and Afghanistan, accounting for 44% of the 36,000 casualties from 2005 to 2009 [10]. By leveraging autonomous convoys, vehicles can be operated with reduced direct human intervention, reducing the number of people needed in an operation, and thereby reducing the risk to human life by removing the personnel from dangerous situations. In addition to the lifesaving benefits, utilization of AGVCs would provide tremendous cost savings as well. The cost of deploying a Soldier is estimated to be $2.1M a year [28], which means reduction of personnel needed has a built-in financial benefit. Furthermore, the use of autonomy in a convoy allows for greater precision in vehicle spacing due to the removal of human error, allowing for decreased spacing between vehicles. This decrease in spacing would reduce overall convoy length and provide fuel savings, which had been previously estimated to be between eight to twelve percent depending on the separation distance [29].

In order to reduce the threat to personnel and reduce costs associated with convoy operations, the U.S. Army is looking to increase utilization of autonomy in future operational concepts. Precision logistics, which entails the use of robotic autonomous delivery, is highlighted as a required Army capability set for sustained support of multi-domain operations [30]. In addition, the U.S. Army Robotic and Autonomous System Strategy specifically calls out autonomous convoys as a tool to enhance soldier survivability and reduce their exposure to hazardous situations [31]. With the high-level needs being evident and understood by military leadership, a proper examination of the requirements is needed to be able define metrics of success for an AGVC system.

2.3 Manned Convoy Requirements

Military doctrine outlines the fundamental set of principles that guides military forces in support of meeting its objectives [32]. There exist four general types of military doctrine: Joint, Multinational, Multi-Service, and Service. While Joint, Multinational, and Multi-Service doctrine addresses processes common between multiple services (and nations, in the case of Multinational), every Service of the U.S. Armed Forces outlines Service specific doctrine defined to meet their idiosyncratic goals [33]. A thorough review of military doctrine can be performed to determine metrics and requirements of systems based on military needs. In this effort, we reviewed military doctrine to determine performance metrics for manned convoys. To limit the scope of the effort, we focused on Service specific doctrine from the U.S. Army due to their mission being most closely tied to the sustained utilization of ground vehicle convoys.

All Army doctrine fits into a hierarchical structure with one of three classifications: Army Doctrine Publications (ADP), Field Manuals (FM), and Army Techniques Publications (ATP) [33]. Each of these publications serve a distinct purpose. ADPs contain the fundamental principles and foundations that guide Army actions in support of its objectives. FMs contain the tactics, procedures, and other relevant information in the execution of the principles described in the ADP. ATPs detail the flexible, non-prescriptive techniques to be used to perform Army missions, functions, and tasks. The doctrine has a hierarchy, with ADPs on top, followed by FMs, followed by ATPs.

In addition to the doctrine, the Army also publishes training material, such as Training Circulars (TC) and Soldier Training Publications (STP). These documents can also contain information pertinent to the desired system performance and outcomes that are valuable in determining performance metrics. These documents, along with the aforementioned Army doctrine documents, are published from the Army Publishing Directorate [34].

Table 2-1 lists the Army publications found to be relevant to convoy performance. An important characteristic of DoD publications is the Distribution Statement. Publications that have a Distribution Statement A label have been reviewed through the DoD Operational Security process and have been approved for public release [35]. Any other Distribution Statements have restricted access and are not available to the general public. Owing to the limitation of availability in the information, the contents of those publications are not considered in this effort. However, they are included in Table 2-1 for the sake of completeness. The remaining Army publications that are approved for public release and pertinent to convoys are “ATP 4-11 Army Motor Transport Operations”, “STP 55-88M14-SM-TG Soldier’s Manual and Training Guide MOS 88 M MOTOR TRANSPORT OPERATOR, SKILL LEVELS 1, 2, 3, and 4”, and “TC 21-

305-20 Manual for the Wheeled Vehicle Operator". In the following sections, we will give a brief overview of the purpose of the publication, discuss its relationship to the convoy mission, and lay out the requirements that can be extracted toward the development of convoy performance metrics.

Table 2-1. Current Army publications relevant to convoy performance.

Publication Number	Publication Name	Distribution Statement
ATP 4-01.45	MULTI-SERVICE TACTICS, TECHNIQUES, AND PROCEDURES FOR TACTICAL CONVOY OPERATIONS	Distribution D
ATP 4-11	ARMY MOTOR TRANSPORT OPERATIONS	Distribution A
STP 55-88M14-SM-TG	SOLDIER`S MANUAL AND TRAINER`S GUIDE MOS 88M, MOS 88M MOTOR TRANSPORT OPERATOR, SKILL LEVELS 1, 2, 3, AND 4	Distribution A
TC 21-305-20	MANUAL FOR THE WHEELED VEHICLE OPERATOR	Distribution A
TC 4-11.46	CONVOY PROTECTION PLATFORM (CPP) COLLECTIVE LIVE FIRE EXERCISES	Distribution C

"ATP 4-11 Army Motor Transport Operations" details the Army's doctrine in the utilization of motor transportation in the support of operations [36]. This support includes the movement of personnel, units, supplies, and equipment by vehicle. By performing these functions, motor transports allow for essential distribution capabilities, force sustainment, and extended operational reach, making them an integral part of the Army's support and force sustainment. "ATP 4-11" has information on the fundamentals, operations, and unit elements that make up motor transport operations. While the doctrine itself explicitly states that it does not go into details about convoy operations and battle drills, it still contains relevant information on how convoys are utilized, since they are used for motor transport. In the document, a convoy is defined as “a group of vehicles moving from the same origin to a common destination and organized under a single commander for the purpose of control.” This definition of a convoy is important to note, since the statement gives the following high-level requirement, frequently referred to as autonomous rendezvous:

Requirement 1 - Two or more vehicles must be able to travel to a specified point.

In addition, the various types of hauling required of motor transports specifies the potential need for vehicles to make repeated trips, indicating the following requirement:

Requirement 2 - A convoy must have the ability return to the original location after initially reaching the destination.

From the perspective of overall convoy system parameters, “ATP 4-11” details multiple planning factors needed for convoy missions that shape the performance requirements of a convoy system. One important planning factor is the rate of march. The rate of march of a convoy mission is the average distance expected to be traveled by the convoy for a given period of time. The need to be able to set a rate of march parameter indicates the following requirement:

Requirement 3 - Convoy system must have an adjustable rate of march.

In addition to the rate of march, multiple planning factors related to convoy elements and associated gaps are discussed. A convoy can be broken down into smaller elements for organizational purposes. The smallest element is a march unit, which can have up to 25 vehicles. Next is a serial, which can consist of two to five march units. Following that is a column, which can consist of two or more serials. Figure 2-1 illustrates the breakdown of the described convoy elements.

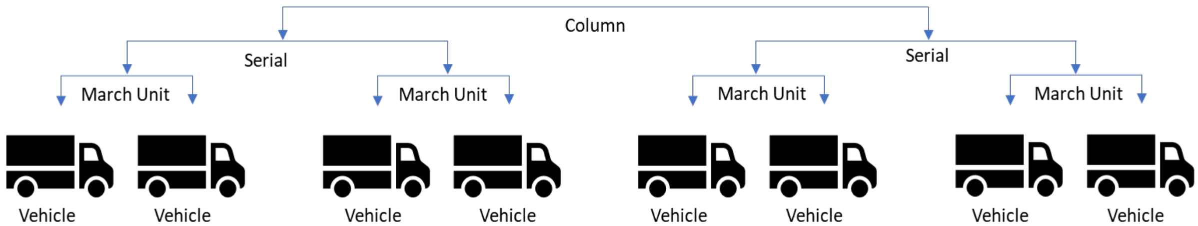


Figure 2-1. Convoy elements.

The proper gap spacing considerations of a convoy differs between vehicles and convoy elements. For vehicles, the gaps are defined by distance between vehicles, with the exact distance being set by a Convoy Commander. This indicates the following requirement:

Requirement 4 - Convoy system must have an adjustable gap distance between vehicles.

While convoy vehicles define the gap by distance, convoy element gaps are defined by a time gap. A time gap is the amount of time measured between convoy elements as they pass a specified point. Different convoy elements can have unique time gaps, such as march unit gaps and serial gaps, and are also set at the discretion of the Convoy Commander. This indicates the following requirement:

Requirement 5 - Convoy system must have an adjustable gap time between convoy elements.

Finally, "ATP 4-11" also indicates that if a vehicle in a convoy is involved in a motor accident, then only the affected vehicle and its immediate successor should stop. All other vehicles in the convoy should continue the path when possible. This gives the following requirement:

Requirement 6 - Convoy systems must be able to complete its route even in the event of one or more vehicles leaving the system.

“STP 55-88M14-SM-TG Soldier’s Manual and Trainer’s Guide MOS 88M” identifies the training requirements for Soldiers serving in the Military Occupational Specialty (MOS) of 88M, which is the designation for motor transport operators [37]. Rather than providing doctrinal guidance, STPs provide task summaries to help plan, conduct, and evaluate individual training in units. The task summaries provide information and instructions such as task conditions, task standards, performance steps, evaluation preparation, and performance measures. Much of the information covers the processes necessary in performing motor transport, such as mission preparation, transportation of cargo, and motor pool management. In reviewing the task summaries, certain portions of the text were found to reinforce the need of the requirements identified in “ATP 4-11”. Specific training tasks indicated a need for a convoy to increase transit speeds in kill-zones, reinforcing the adjustable rate of march in Requirement 3. In addition, “STP 55-88M14-SM-TG” described the need to situationally set gaps between convoy vehicles and elements depending on the desired convoy formation, reinforcing Requirement 4 and Requirement 5. Aside from the reinforcement of previously described requirements, “STP 55-88M14-SM-TG” also identifies a new requirement based on the responsibilities attributed to the Convoy Commander relating to catch-up speed. Convoy Commanders are to set a catch-up speed that convoy followers must abide by. This indicates the following requirements:

Requirement 7 - Convoy system must be able to specify a maximum catch-up speed for individual convoy vehicles that fall behind.

The final convoy related publication available for public release was “TC 21-305-20 Manual for the Wheeled Vehicle Operator”. This TC describes operating practices, procedures, and techniques to efficiently operate a wheeled vehicle, including a chapter devoted to motor marches and convoys [38]. In this chapter, proper gap and vehicle speeds are discussed. The catch-up speed referenced in Requirement 7 is further enforced, and a speed-based gap distance is suggested as follows:

$$g = m * s \qquad (2.1)$$

where g is the gap distance in yards, m is the speedometer multiplier, and s is the speed of the vehicle in miles per hour. The value of m is typically set at two but is variable as determined by the Convoy Commander. Note that equation 2.1 is presented in the doctrine as a general rule to follow to get an estimate for gap distance rather than a formal mathematical calculation, which is why the units in the equation do not match. This adjustable gap calculation further emphasizes the need for Requirement 4 and Requirement 5.

In addition to the publications available for public release, Table 2-1 shows two additional documents: "TC 4-11.46 Convoy Protection Platform (CPP) Collective Live Fire Exercises" and "ATP 4-01.45 Multi-Service Tactics, Techniques, and Procedures for Tactical Convoy Operations". "TC 4-11.46" deals primarily with gunnery and training in handling threats [39] [40], rather than topics pertaining to mobility performance requirements. "ATP 4-01.45 Multi-Service Tactics, Techniques, and Procedures for Tactical Convoy Operations," contains tactics, techniques, and procedures relevant to leading of troops, employment of gun trucks, battle drills and IED handling [41]. Both publications are restricted from public release to protect the information contained, and as such, are noted only for completeness.

2.4 Common Autonomous Convoy Performance Metrics

There exist numerous efforts in the development and improvement of AGVC systems that focus on a number of different areas, such as control objectives, VANET factors, and control strategies [13]. Each effort defines customized measures of performance and success based on the research goals, but there is not a standardized set of high-level metrics to be used across

AGVC systems. Despite the lack of standardization, there is nonetheless commonality between how AGVC efforts measure their system's performance due to the common problem space that is being explored. The most common subjects that AGVC efforts look to investigate are spacing policy and string stability; two closely related topic areas. By looking at metrics utilized in research efforts exploring these topics, we attempt to discern common threads in AGVC metrics that can be used as a performance metrics framework for AGVCs for military utilization.

Spacing policy is the collection of methods, actions, and plans by which a convoy sets the desired distance between the vehicles [42]. The two most widely used convoy spacing policies are constant spacing and variable spacing. In a constant spacing policy, the separation distance between convoy members is independent of the speed of the lead vehicle. The spacing error, $\varepsilon_j(t)$, is as follows [43]:

$$\varepsilon_j(t) = x_{j-1}(t) - x_j(t) - L_j \tag{2.2}$$

where j is the index of the vehicle in the convoy, x_j is the position of vehicle j, x_{j-1} is the position of the leader of vehicle j, and L_j is the following distance of vehicle j. In variable spacing, the spacing of the convoy vehicles are related to the vehicle's speed, typically using a constant time headway approach. The spacing error δ_j is defined as follows [44]:

$$\delta_j(t) = x_{j-1}(t) - x_j(t) - L_j - hv_j \tag{2.3}$$

where h is the time headway constant and v_j is the velocity of the vehicle j. Figure 2-2 illustrates the variables relating to spacing policy in the context of a three-vehicle convoy.

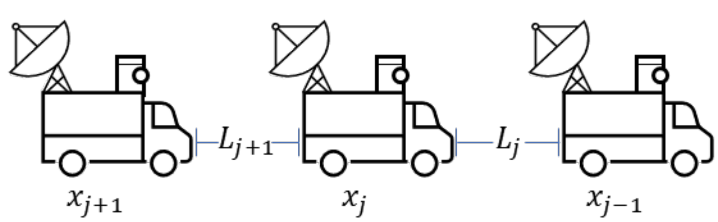


Figure 2-2. Spacing policy variables in the context of a convoy.

One of the primary goals of a convoy system is to reduce spacing error in accordance to the chosen spacing policy, which necessitates changes in control input to the follower vehicles. These changes and errors have the potential to propagate and amplify throughout the convoy, as each follower vehicle attempts to adjust their control parameters to reduce the error. A convoy system's reaction to this propagation of error is referred to as "string stability", with a convoy system being "string stable" if errors decrease, rather than increase, as they propagate through the convoy [45]. Intuitively, loss of string stability in a group of vehicles moving on a highway leads to undesired phenomenon such as the "accordion effect", which leads to accidents and/or traffic jams.

More formally, a system is "string stable" if the following constraints are satisfied [46]:

$$\|H(s)\|_\infty \leq 1 \tag{2.4}$$

$$h(t) > 0 \tag{2.5}$$

where *h(t)* represents the ratio of spacing error between two consecutive vehicles and *H(s)* represents the Laplace transform of this function as follows:

$$h(t) = \frac{\varepsilon_j(t)}{\varepsilon_{j-1}(t)} \tag{2.6}$$

$$H(s) = \mathcal{L}\big(h(t)\big) \tag{2.7}$$

presuming constant spacing, with $\delta_j(t)$ replacing $\varepsilon_j(t)$ for variable spacing.

It has been shown that string stability can be achieved in a convoy system with a variable spacing policy without any V2V communication, in contrast to constant spacing policy convoys which require some level of V2V communications to achieve stability [47]. Experimental verification of string stability and adherence to spacing policy is often performed to validate that

AGVCs are meeting the designed intent. The most commonly used metrics for experimentation can be split into separation distance, spacing error, velocity, and acceleration comparisons.

When separation distance metrics are used, it presumes that the convoy vehicles start off with the desired spacing distance in a stopped state and looks at how the separation distance changes as the convoy system progresses throughout time. Given that the separation distance is not static in a convoy system using a variable spacing policy, this metric is normally used when examining convoy controllers using a constant spacing policy [44] [47] [46]. Figure 2-3 shows a representative separation distance graph, with Figure 2-3a showing a string stable system, and Figure 2-3b showing a string unstable system.

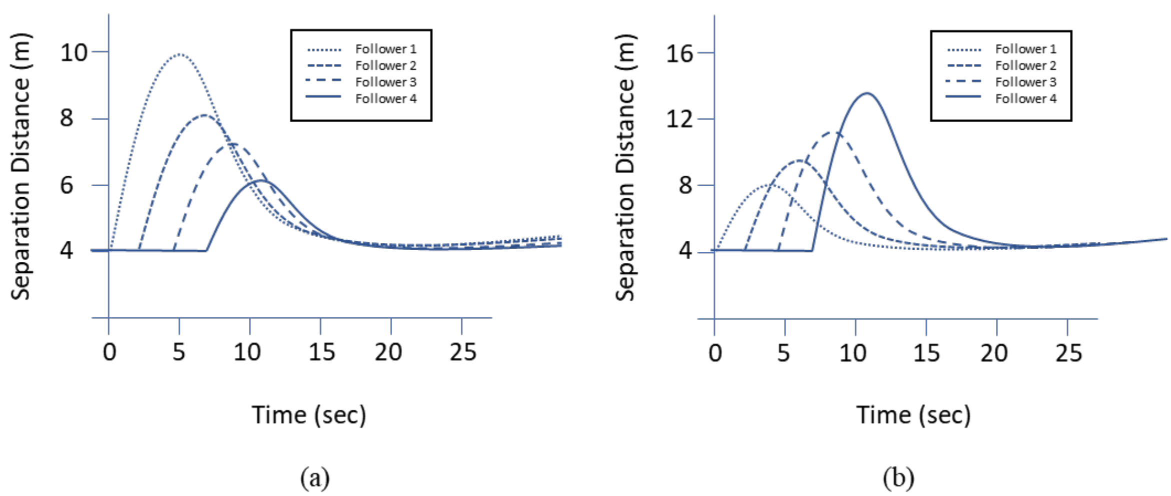


Figure 2-3. Sample graphs for separation distance in a (a) string stable system, and (b) string unstable system.

Another metric that is often used in gauging autonomous convoy performance with string stability and spacing policy is spacing error over time. Convoy systems that leverage variable spacing tend to use spacing errors as the experimental metric instead of separation distance, given the variable nature of separation distance between the member vehicles. Convoy systems that are string stable will show spacing errors that decrease along the follower vehicles [42] [48] [49] [50], as shown in Figure 2-4.

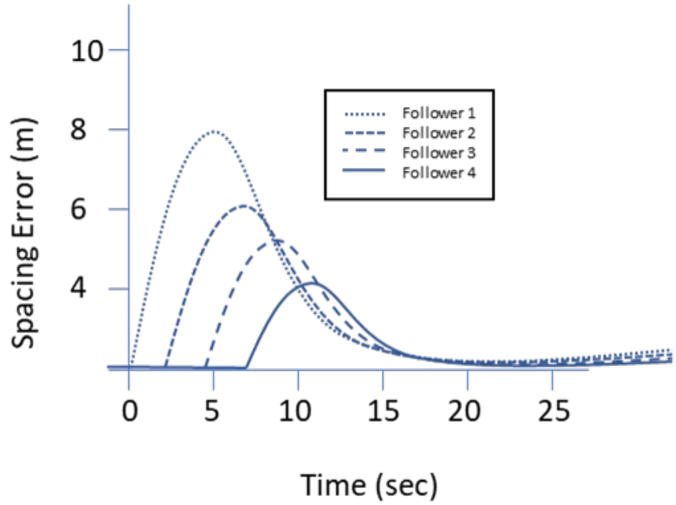


Figure 2-4. Sample graph for a string stable system regarding spacing error.

An additional common metric that was found when gauging autonomous convoy performance with string stability and spacing policy was vehicle velocity. Given that the primary goal of a convoy in motion is to have followers maintain a certain gap distance with a lead vehicle, followers will always be aiming to converge to a velocity that matches its leader [46]. As such, it is important to note if a convoy's ability to have the vehicles reach a desired velocity is string stable. Since disturbances that are exerted on an individual convoy member can adversely deteriorate the string stability of the whole convoy, it is important to ensure that disturbances to vehicle velocity are not amplified throughout a convoy's followers and to note the time to convergence at the desired velocity when examining performance [47] [48] [46] [49] [50]. Figure 2-5 shows a representative velocity graph for a system that is not string stable in terms of velocity.

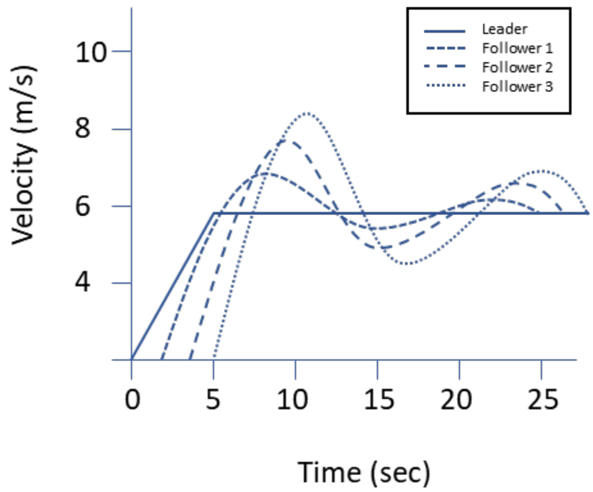


Figure 2-5. Sample graph for a string unstable system regarding vehicle velocity.

The final metric we reviewed in gauging autonomous convoy performance with string stability and spacing policy was control effort acceleration for the vehicles. If the control effort is not string stable, the reliability of vehicle operation can be put into jeopardy, as amplification of acceleration requests can exceed the limits of the vehicle's capabilities [51]. As such, string stability for acceleration is important for not only convoy performance, but also the overall safety and maintenance of the vehicles. Indeed, since the acceleration is proportional to exerted forces, the acceleration string stability metric can be directly used to study the effect of exerted disturbances on the convoy dynamics and its position/velocity string stability metrics. Ensuring disturbances to control effort are not amplified throughout a convoy's followers, and noting time to convergence at the desired acceleration, are important factors when examining performance [47] [48] [42] [49] [50]. Figure 2-6 shows a representative acceleration graph for a system that is not string stable in terms of acceleration.

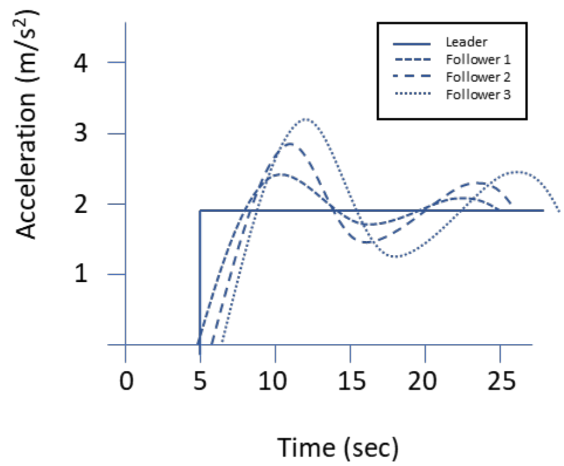


Figure 2-6. Sample graph for a string unstable system regarding vehicle acceleration.

In addition to string stability and spacing policy metrics, another metric that is commonly used in autonomous convoy performance is a convoy member vehicle's offset to the desired trajectory, which is known as path following error [45] [52] [53] [54], as shown in Figure 2-7. The path offsets between follower and leader vehicles are measured and used to calculate the path following errors based on the needs of the experiment, with techniques such as mean squared error or mean absolute error [55]. This is typically used when systems are looking to examine path replication, rather than string stability.

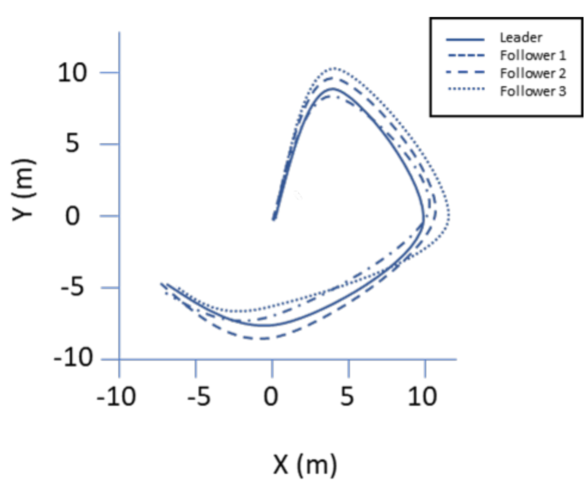


Figure 2-7. Sample graphs path following error.

2.5 Autonomous Ground Vehicle Performance Metrics Framework

Through analyzing Army doctrine, we have derived generalized requirements that can be leveraged to apply to AGVCs in assessing their ability to perform Army missions. By leveraging common autonomous convoy performance metrics to gauge how well the requirements are being met, we can develop a framework that can be used to assess AGVC performance across different systems. The framework will henceforth be referred to as the AGVC Performance Metrics Framework. For greater clarity, we will be classifying the manned convoy requirements into three categories of analysis as follows: Goal Specification, Spacing Policy, and System Parameters. Refer to Table 2-2 for an outline of the framework, with the specific categorization of requirements and metrics that should be used for comparison. The following sections will explain the choice of which metrics are best utilized for comparison for each different category of requirements. An example application of the framework will then be shown by applying it in an experiment to compare simulated AGVC controllers.

Table 2-2. AGVC Performance Metrics Framework.

Category	Requirement	Metrics to Use
Goal Specification	Req. 1 - Two or more vehicles must be able to travel to a specified point.	Path following error
	Req. 2 - A convoy must have the ability return to the original location after initially reaching the destination.	
	Req. 6 - Convoy systems must be able to complete its route even in the event of one or more vehicles leaving the system.	

Spacing Policy	Req. 4 - Convoy system must have an adjustable gap distance between vehicles.	String stability
Spacing Policy	Req. 5 - Convoy system must have an adjustable gap time between convoy elements.	String stability
System Parameters	Req. 3 - Convoy system must have an adjustable rate of march.	String stability
System Parameters	Req. 7 - Convoy system must be able to specify a maximum catch-up speed for individual convoy vehicles that fall behind.	String stability

2.5.1 Goal Specification

Per Army doctrine, a convoy system must be able to travel to a designated point (Requirement 1) and optionally return to the original point of departure (Requirement 2) as dictated by mission needs. This indicates that there is a desired path and goal that the AGVC is meant to follow as closely as possible, and deviation from said path is undesirable. Given these needs, path following error, as shown in in Figure 2-7, is the most appropriate metric to compare the performance of AGVC systems in terms of goal specification. The desired position of lead vehicles and the relative position of the follower vehicles can be used to calculate the offset between desired and actual positions. This metric can be used for both Requirement 1 and Requirement 2, since Requirement 2 can be considered an extension of Requirement 1 with multiple goal points. To evaluate overall path following error performance, we will adapt evaluation metrics for position tracking from the domain of computer vision [56] due to the similar goals between leader following and position tracking.

Another aspect of goal specification is that convoy systems must be able to complete their route even if one or more vehicles leave the convoy (Requirement 6). Once again, this looks at how well an AGVC follows the path of a lead vehicle, with the added complexity of having a convoy follower needing to modify which vehicle it is following to ensure that a disabled follower vehicle does not cause all followers to halt. This also can be examined by leveraging path following error, as shown in Figure 2-7, as a metric of comparison. Vehicles that are unable to continue with the convoy will produce a greater overall error in the system, giving a point of comparison between different AGVC implementations.

2.5.2 Spacing Policy

As previously defined, spacing policy is the collection of methods, actions, and plans by which a convoy sets the desired distance between the vehicles [42]. The two primary categories of spacing policies are constant spacing and variable spacing. The doctrinal requirements align with the two categories of spacing policy, with the need for an adjustable gap distance (Requirement 4) aligning with constant spacing, and the need for adjustable gap time (Requirement 5) aligning with variable spacing. The key areas of comparison for spacing policy performance are string stability and time to convergence. A string stable system will not propagate errors throughout a convoy, meaning that convoy followers will more closely adhere to the desired speed and position. In addition, string stability allows the overall convoy to reach its desired end state more rapidly, meaning the time needed for each follower vehicle to converge to the desired system parameter is lower. Therefore, string stability related metrics, such as amplification of the response of follower vehicles and overall time it takes for the follower vehicles to converge to the steady state [47] [48] [42] [49], are appropriate tools for comparison. Examples can be seen in Figures 2-2, 2-3, 2-4, and 2-5. The choice of which area to examine for string stability (separation distance, velocity, acceleration, etc.) is dependent on the mission goals that the AGVC is attempting to meet.

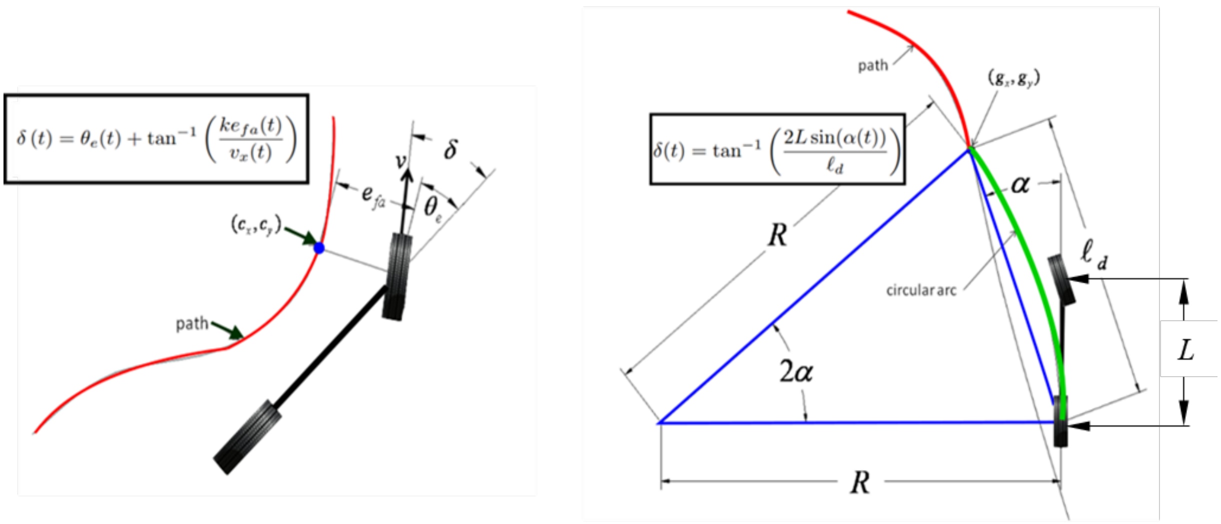
2.5.3 System Parameters

The requirements categorized under System Parameters deal with overall convoy system settings. Army doctrine defines the need for an adjustable rate of march (Requirement 3) and an ability to set a maximum catch-up speed for the convoy follower vehicles (Requirement 7). Overall, they impose qualifiers and restrictions to how the AGVC meets spacing policy requirements. These requirements can be analyzed with a binary success or failure by monitoring the overall convoy velocity and the speed of the individual vehicles. If comparisons of greater granularity are needed, those distinctions can be made by examining the string stability related metrics described in Spacing Policy at different rates of march and catch-up speeds. This would entail examining multiple runs of a convoy system and changing the system parameters to be evaluated between each run. The results of each run can be examined for string stability related metrics, such as amplification of the response of follower vehicles and overall time it takes for the follower vehicles to converge to the steady state. Examples of string stability can be seen in Figures 2-2, 2-3, 2-4, and 2-5. Changes in these measurements between the various runs can then be noted for an AGVC, which could then be compared with how changes affected performance for other AGVC systems.

2.6 Application of the Performance Metrics Framework

In order to apply the AGVC Performance Metrics Framework, we leveraged the Autonomous Navigation Virtual Environment Laboratory (ANVEL), "an interactive, real-time engineering modeling and simulation (M&S) software tool built specifically to assist in the research, design, testing, and evaluation of intelligent ground vehicles" [57]. ANVEL features Python application programmer interfaces to setup and control autonomous convoys in an M&S environment. In the simulations, a vehicular convoy of Palletized Load System (PLS) trucks was established with a lead vehicle who followed given path points, and three follower vehicles behind it. The network topology of the convoy was configured so that each vehicle only had information of its adjacent leader and follower. Each follower had the exact location of its leader and used that information for convoy following.

Two different convoy following controllers were used in our application of the framework. One convoy controller utilized a Pure Pursuit method for geometric path tracking, in which the center of the rear axle is used as the reference point on the vehicle to compute a steering angle towards a look-ahead point at a fixed distance [58]. The other convoy controller utilized the Stanley method for geometric path tracking, in which the front axle is used as the reference point, and both the heading error and cross-track error are used to find the proper steering angle [59]. These two control schemes represent the two ends of the spectrum of geometric/kinematic controllers in terms of dependency on the number of to-be-tuned parameters where the Pure Pursuit controller relies less on the system parameters while the Stanley controller, which was the winner of DARPA challenge 2005 [60], relies on more tunable parameters. Figure 2-8 depicts the schematic diagrams associated with the Stanley and Pure Pursuit control schemes. Some remarks are in order (see, e.g. Snider, 2009, for more detailed explanations [61]). Both Pure Pursuit and Stanley controllers belong to the family of path tracking algorithms, namely, algorithms that make a vehicle execute to a globally defined geometric path by applying appropriate steering commands that guide the vehicle along the path. The goal of any path tracking algorithm is to simultaneously minimize the lateral distance between the vehicle and the defined path, to constrain the steering control inputs to smooth input commands, and to minimize the heading of the vehicle and the defined path heading.



(a) (b)

Figure 2-8. Stanley (a) and Pure Pursuit (b) controller schematic diagrams, recreated from [61].

As it is demonstrated by Snider [61], Pure Pursuit controllers are essentially proportional controllers with a proportional gain of $2/l_d^2$ acting on the steering angle dynamics. A geometric interpretation of the parameter l_d in the gain $2/l_d^2$ is that it provides a kind of look-ahead distance. L is the As is customary in the Pure Pursuit control literature, the look-ahead distance is tuned to be stable at several constant speeds. If the look-ahead distance is a function of the speed of the vehicle, then gain-scheduling and linear parameter varying control (LPV) techniques can be used to analyze the stability of the resulting closed-loop dynamics [62]. The nonlinear feedback control law associated with the Stanley controller, on the other hand, relies on the cross-track error e_{fa} from the center of the front axle to the nearest path point, with a gain parameter of k. The intuition behind the Stanley control scheme is that the larger the cross-track error from the path, the further the steering of the wheels towards the path. Despite the demonstrated superiority of the Stanley controller in the DARPA challenge, in extremely rare maneuvers, Pure Pursuit controllers demonstrate more robustness with respect to sudden lane changes. On the other hand, Pure Pursuit controllers have shown failures under paths with fast varying curvatures [61].

By studying such extremes of Stanley and Pure Pursuit control schemes in our simulations, we highlight the results that can be expected for low-level control of military autonomous convoys across the spectrum of trajectory tracking control schemes. We compared the convoy controllers within the three categories defined by the framework: Goal Specification, Spacing Policy, and System Parameters. In Goal Specification, we are concerned with the amount of deviation of autonomous convoy members from a given specified path. In Spacing Policy, we are concerned with maintaining a desired distance between the autonomous convoy members. Finally, in System Parameters, we are concerned with the controller parameters that need to be tuned to achieve a given control objective. The results of the comparison are as follows.

2.6.1 Goal Specification

The two Goal Specification requirements that we will examine in this comparison are Requirement 1 ("Two or more vehicles must be able to travel to a specified point") and Requirement 6 ("Convoy systems must be able to complete its route even in the event of one or more vehicles leaving the system"). To compare performance in Goal Specification, a circular path plan was created. Circular paths provide proper test cases where one is interested in studying the effectiveness of the proposed controllers in minimizing the deviation of each autonomous convoy member from the desired paths. Per the framework detailed in Table 2-2, path following error from the desired path is the most appropriate metric to use for comparison between convoy controllers for the Goal Specification requirements.

To find the vehicle offset for Requirement 1, we recorded the position of each vehicle as it traveled from the beginning to the end of the route. 4,502 vehicle positions were recorded for each member of the convoy. For each vehicle, the Euclidian distance between every individual recorded vehicle position and the closest route path point was calculated to find the offset distance and calculate the path following error. The overall convoy vehicle positions for the two different convoy controllers applied to Requirement 1 are shown in Figure 2-9a and Figure 2-9b. To compare path following error performance of the convoy controllers, we leveraged metrics used in positional tracking [56] due to the similarities between vehicle path following and trajectory tracking in computer vision. The metrics and results for Requirement 1 are shown in Table 2-3. In addition, Figure 2-10 shows the path following error histograms for Requirement 1.

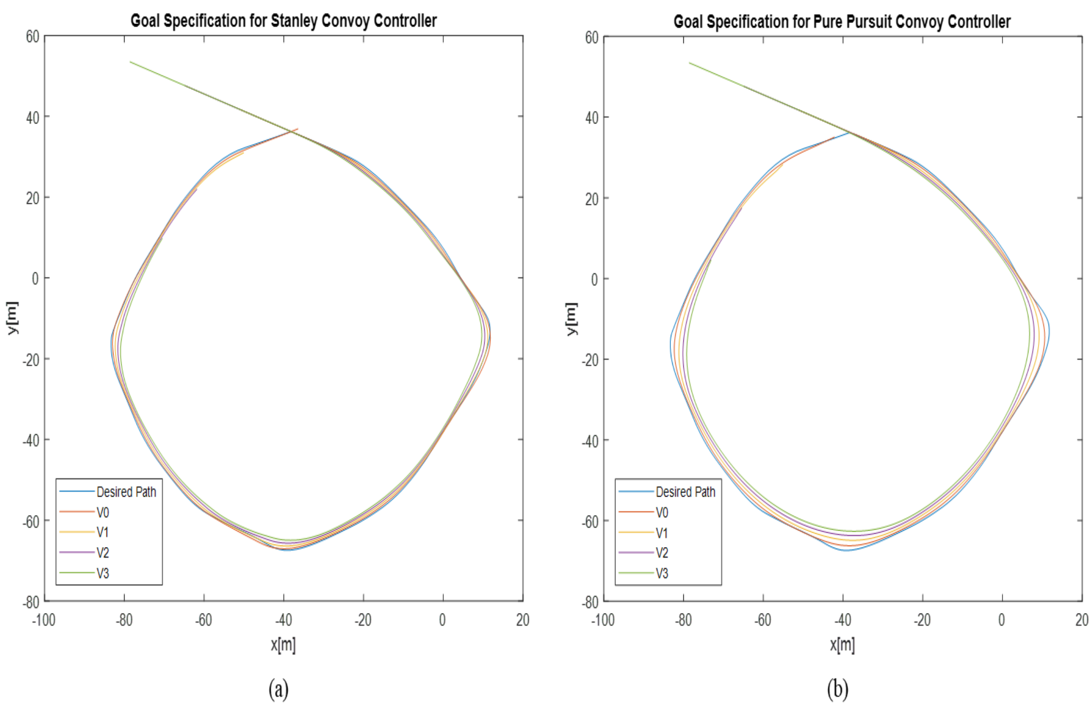


Figure 2-9. Planned path and taken path of convoy vehicles for Requirement 1 using (a) Stanley and (b) Pure Pursuit.

Table 2-3. Comparison metrics for path following error with Requirement 1.

	Vehicle 0 Path Following Error (m)		Vehicle 1 Path Following Error (m)		Vehicle 2 Path Following Error (m)		Vehicle 3 Path Following Error (m)	
	Stanley	Pure Pursuit	Stanley	Pure Pursuit	Stanley	Pure Pursuit	Stanley	Pure Pursuit
Mean	0.7149	0.5975	1.7203	2.0791	4.0645	4.5741	7.1026	7.9936
St. Dev	0.5568	0.3086	3.1583	3.0689	7.9315	7.7624	13.4847	13.1704

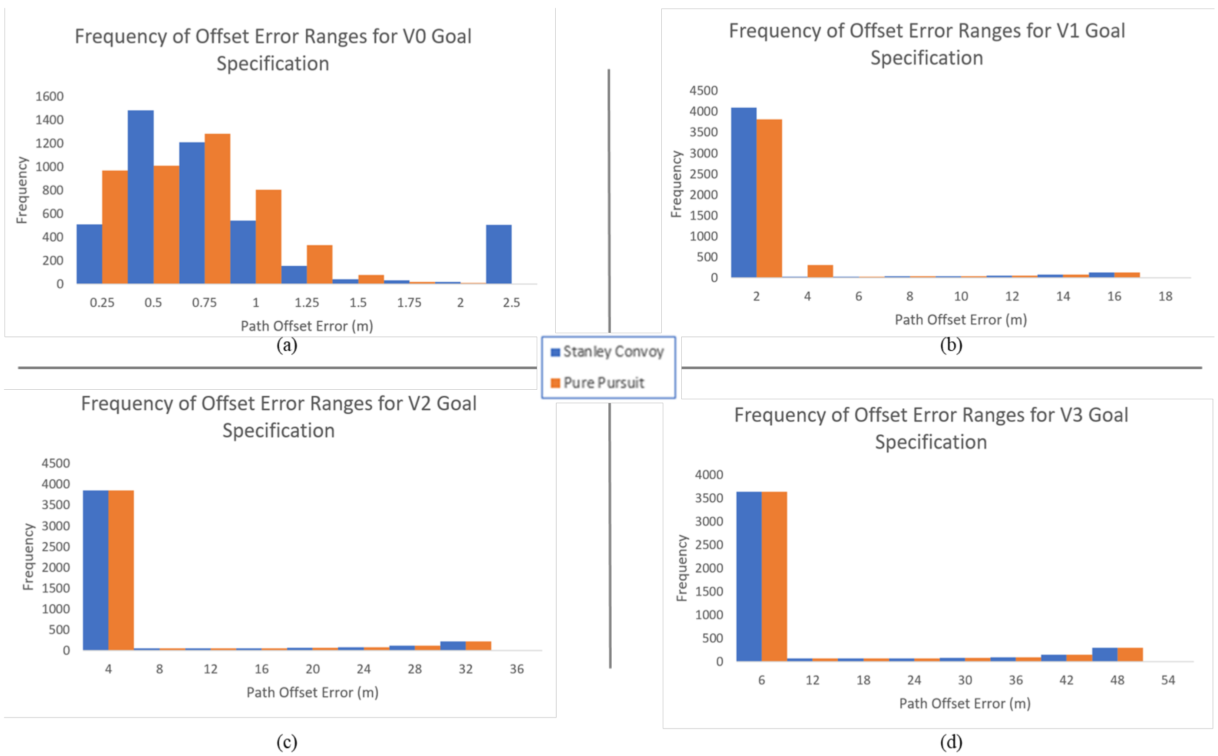


Figure 2-10. Histograms of path following error for Requirement 1 for (a) Vehicle 0, (b) Vehicle 1, (c) Vehicle 2, and (d) Vehicle 3.

As evident in Table 2-3, the lead convoy vehicle performed marginally better in adhering to the desired path for the Pure Pursuit controller as opposed to the Stanley controller, with a lower mean offset error of 0.1274 m. However, the Stanley convoy followers had a lower mean offset error when compared to Pure Pursuit, with the mean errors of V1, V2, and V3 respectively being lower by 0.3588 m, 0.5096 m, and 0.8910 m. Based on the histograms shown in Figure 2-10, it is evident that both controllers had similar distributions of error. This indicates that if

Requirement 1 ("Two or more vehicles must be able to travel to a specified point") is your most important factor of consideration, the convoy overall performs better utilizing a Stanley convoy controller. While the lead vehicle performed better with Pure Pursuit compared to Stanley, the total path following error was lower when the follower vehicles were taken into consideration.

To compare path following error performance of the convoy controllers for Requirement 6 ("Convoy systems must be able to complete its route even in the event of one or more vehicles leaving the system"), we utilized the same path and convoy controllers from Requirement 1 but modified the experiment so that Vehicle 1 stopped motion at 17.71 seconds into the run. At that time, Vehicle 2 modifies its leader to ignore Vehicle 1 and follow Vehicle 0 directly, while Vehicle 3 continues to follow Vehicle 2 per the initial setup. The overall convoy vehicle positions for the two different convoy controllers applied to Requirement 6 are shown in Figure 2-11a and Figure 2-11b. To compare path following error performance of the convoy controllers, we once again leveraged metrics used in positional tracking [55]. The metrics and results for Requirement 6 are shown in Table 2-4. In addition, Figure 2-12 shows the path following error histograms for Requirement 6.

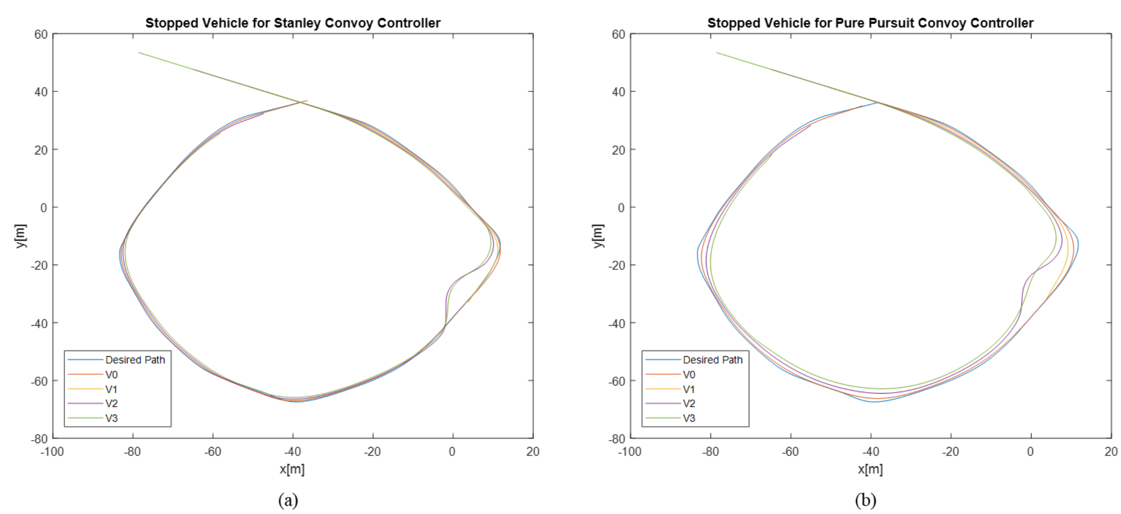


Figure 2-11. Planned path and taken path of convoy vehicles for Requirement 6 using (a) Stanley and (b) Pure Pursuit.

Table 2-4. Comparison metrics for path following error with Requirement 6.

	Vehicle 0 Path Following Error (m)		Vehicle 1 Path Following Error (m)		Vehicle 2 Path Following Error (m)		Vehicle 3 Path Following Error (m)	
	Stanley	Pure Pursuit	Stanley	Pure Pursuit	Stanley	Pure Pursuit	Stanley	Pure Pursuit
Mean	0.7156	0.5725	N/A	N/A	4.2375	4.7396	7.4087	8.2358
St. Dev	0.5571	0.3085	N/A	N/A	7.9525	7.7913	13.5224	13.1862

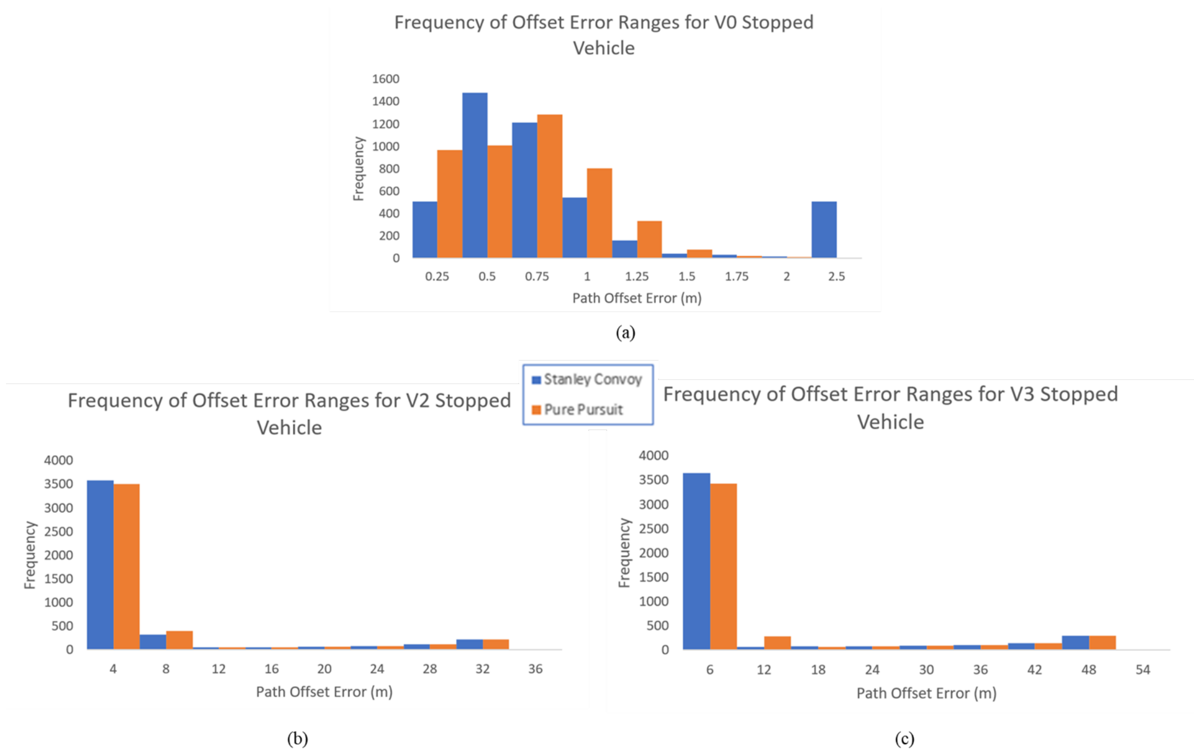


Figure 2-12. Histograms of path following error for Requirement 6 for (a) Vehicle 0, (b) Vehicle 2, and (c) Vehicle 3.

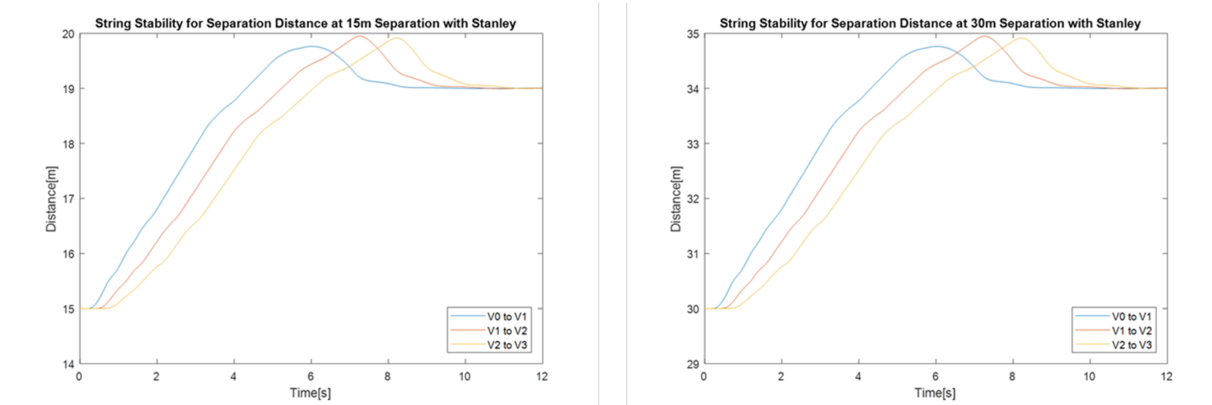
For the Stopped Vehicle experiment, the results of Vehicle 1 were omitted due to that vehicle leaving the convoy shortly after the experiment began. As evident in Table 2-4, the lead convoy vehicle once again performed marginally better in adhering to the desired path for the Pure Pursuit controller as opposed to the Stanley controller, with a lower mean offset error by 0.1431 m. Likewise, the subsequent Stanley convoy followers performed better in terms of mean offset error when compared to Pure Pursuit, with the mean errors of V2 and V3 respectively being lower by 0.5021 m and 0.8271 m. Based on the histograms shown in Figure 2-12, it is once again evident that both controllers had similar distributions of error. This indicates that if Requirement 6 (“Convoy systems must be able to complete its route even in the event of one or more vehicles leaving the system”) is your most important factor of consideration, the convoy overall once again performs better utilizing a Stanley convoy controller. Much like with Requirement 1, while the lead vehicle performed better with Pure Pursuit compared to Stanley, the total path following error was lower when the follower vehicles were taken into consideration.

Overall, by using the AGVC Performance Metrics Framework, it was determined that the Stanley convoy controller performed better for Goal Specification. This was based on our prioritization of Requirement 1 (“Two or more vehicles must be able to travel to a specified point”) and Requirement 6 (“Convoy systems must be able to complete its route even in the event of one or more vehicles leaving the system”) as the key areas of interest. Based on that prioritization, we selected path following error as the primary metric of comparison and determined that the Stanley convoy controller produced a lower overall convoy path following error when compared to the Pure Pursuit convoy controller.

2.6.2 Spacing Policy

To compare performance in Spacing Policy, a straight-line path was created in ANVEL. As previously described and shown in Table 2-2, a convoy system’s string stability is the most appropriate metric to use for comparison between convoy controllers for Spacing Policy requirements. The Spacing Policy requirement that we will examine in this comparison is Requirement 4 (“Convoy system must have an adjustable gap distance between vehicles”). To compare performance of this requirement between the convoy controllers, two gap distances were used: 15 m and 30 m. The 15 m distance was initially selected to have a gap distance that was nearly a full vehicle length long. Given that the PLS is 11.03 m, we opted to round up to 15 m rather than down, to ensure the gap was greater than a full vehicle's length, rather than shorter. The 30 m gap distance was selected by doubling the initially chosen gap distance, to ensure there was a significance difference between the gap distances. For both the Stanley convoy controller and Pure Pursuit convoy controller, a test run with a 15 m gap distance was recorded, followed by a run with a 30 m gap distance, both with a convoy speed set at 8 m/s. The position of each vehicle was recorded in regular intervals from start to finish, with a total of 2,000 samples taken per vehicle. Because the requirement is for adjustable gap distance, we will look at the string stability for both convoy controllers, and then compare how performance changes between the 15 m and 30 m gap distances to determine which one better handled adjusting of distances. Although the spacing policy simulations are being done along a straight-line path for the sake of brevity, the convoy controllers are general enough to regulate the distancing between the autonomous convoy members in more complex situations such as roads on curvy hills. In particular, one can use the longitude and latitude of the autonomous convoy members and then compute their distance from the Haversine equation [63].

Figure 2-13 shows separation distance between vehicles over time for the 15 m and 30 m settings using the Stanley convoy controller. Table 2-5 displays information about the separation distances between the vehicle pairs, the time when the peak distance was achieved, and how long it took the distances to reach a steady state.



(a) (b)

Figure 2-13. Separation distance between vehicles over time for a (a) 15 m and (b) 30 m gap distance with the Stanley controller.

Table 2-5. Separation distance information for different gap distance settings using the Stanley controller.

	15 m Gap			30 m Gap		
	V0 to V1	V1 to V2	V2 to V3	V0 to V1	V1 to V2	V2 to V3
Maximum Gap (m)	19.762	19.947	19.917	34.762	34.947	34.917
% Change from Previous Gap Distance	N/A	0.94%	-0.15%	N/A	0.53%	-0.09%
Peak Time (s)	6.02	7.26	8.22	6.02	7.26	8.22
Steady State Time (s)	9.26	10.33	11.1	9.26	10.33	11.1
Time to Steady State After Peak (s)	3.24	3.07	2.88	3.24	3.07	2.88

As seen in Table 2-5, both gap settings for the Stanley controller failed to achieve string stability. This can be seen by looking at the “% Change from Previous Gap Distance” row, where the initial percentage difference increased in gap distance for both the 15 m and 30 m settings. In a string stable system, the gap error would have decreased in each vehicle pair. It is notable however that the peak times, steady state times, and the time it took to reach steady state after the peak were the same between the two gap distance settings, with the steady state gap distances for each vehicle in 15 m and 30 m settings averaging 19.01 m and 34.01 m, respectively. Overall, this indicates that adjusting the gap distance did not delay how long it took the convoy system to reach steady state, meaning that doing so had no detrimental effect on performance regarding separation distance string stability.

Figure 2-14 shows separation distance between vehicles over time for the 15 m and 30 m settings using the Pure Pursuit convoy controller. Table 2-6 displays information about the separation distances between the vehicle pairs, the time when the peak distance was achieved, and how long it took the distances to reach a steady state.

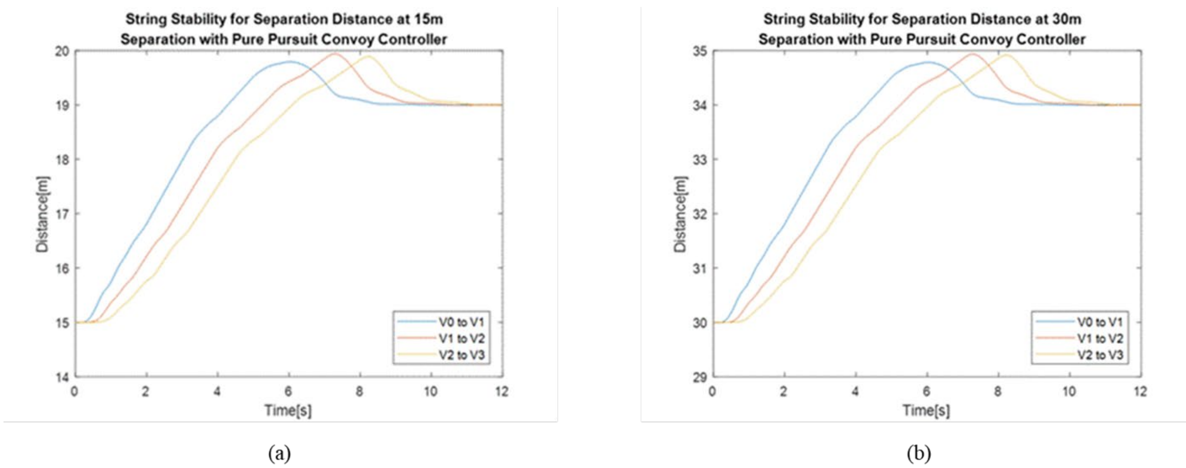


Figure 2-14. Separation distance between vehicles over time for (a) 15 m and (b) 30 m gap distance with the Pure Pursuit controller.

Table 2-6. Separation distance information for different gap distance settings using the Pure Pursuit controller.

	15 m Gap			30 m Gap		
	V0 to V1	V1 to V2	V2 to V3	V0 to V1	V1 to V2	V2 to V3
Maximum Gap (m)	19.762	19.947	19.917	34.762	34.947	34.917
% Change from Previous Gap Distance	N/A	0.94%	-0.15%	N/A	0.53%	-0.09%
Peak Time (s)	6.02	7.26	8.22	6.02	7.26	8.22
Steady State Time (s)	9.32	10.35	11.1	9.3	10.34	11.1
Time to Steady State After Peak (s)	3.3	3.06	2.88	3.28	3.06	2.88

As seen in Table 2-6, both gap settings for the Pure Pursuit controller failed to achieve string stability as well. Again, this can be seen by looking at the “% Change from Previous Gap Distance” row, where the initial percentage difference increased in distance for both the 15 m and 30 m settings. In a string stable system, the separation distance would have decreased in each vehicle pair. Like the Stanley convoy controller, it is once again notable that for the Pure Pursuit convoy controller, the peak times, steady state times, and the time it took to reach steady state after the peak were the same between the two gap distance settings, with the steady state gap distances for each vehicle in 15 m and 30 m settings averaging 19.01 m and 34.01 m, respectively. This indicates that adjusting the gap distance had no detrimental effect on performance regarding separation distance string stability.

When comparing the performance difference between the Stanley and Pure Pursuit convoy controllers when looking at Spacing Policy, adjusting the gap distance setting had very similar effects for separation distance string stability. For both the Stanley and Pure Pursuit convoy controllers, neither 15 m nor 30 m gap settings demonstrated a string stable separation distance. In addition, the steady state gap distance for both convoy controllers for each vehicle in 15 m and 30 m settings averaged 19.01 m and 34.01 m, respectively. Even with those similarities however, there were some minor variations that differentiated performance. The steady state time for "V0 to V1" and "V1 to V2" was earlier for the Stanley convoy controller when compared to the Pure Pursuit convoy controller for both gap distance settings. As seen in Table 2-7, using the Stanley convoy controller either maintained or hastened the time it took to reach steady state for every vehicle pair. This was true for both the 15 m gap setting and the 30 m gap setting. This indicates that using the Stanley convoy controller allowed the convoy to reach the desired gap distance more quickly when compared to Pure Pursuit.

Table 2-7. Comparison of When Steady State Gap Distance was Achieved for the Stanley and Pure Pursuit Convoy Controllers.

	15 m Gap Distance			30 m Gap Distance		
	V0 to V1	V1 to V2	V2 to V3	V0 to V1	V1 to V2	V2 to V3
Pure Pursuit Steady State Time (s)	9.32	10.35	11.1	9.3	10.34	11.1
Stanley Steady State Time (s)	9.26	10.33	11.1	9.26	10.33	11.1
% Difference from Pure Pursuit to Stanley	-0.64%	-0.19%	0.00%	-0.43%	-0.10%	0.00%

Overall, we utilized the AGVC Performance Metrics Framework to compare performance between the Stanley and Pure Pursuit convoy controllers based on Spacing Policy.

We prioritized Requirement 4 ("Convoy system must have an adjustable gap distance between vehicles") as the key area of interest. Based on that prioritization, we selected separation distance string stability as the primary metric of comparison and found that vehicles under the Stanley convoy controller reached the steady state gap distance more quickly for all gap distance settings, as compared to the Pure Pursuit convoy controller. This meant that by leveraging our framework, we determined that the Stanley convoy controller performed better than the Pure Pursuit convoy controller based on metrics relating to Spacing Policy.

2.6.3 System Parameters

To compare performance in System Parameters, we used the same straight-line path from the Spacing Policy experiments. As previously described and shown in Table 2-2, a convoy system's string stability is the most appropriate metric to use for comparison between convoy controllers for System Parameter requirements. The System Parameter requirement that we will examine in this comparison is Requirement 3 ("Convoy system must have an adjustable rate of march"). To compare performance of this requirement between the convoy controllers, two velocities were used: 10 m/s and 20 m/s. For both the Stanley convoy controller and Pure Pursuit convoy controller, an experiment was run with a desired convoy velocity of 10 m/s, followed by a run with a desired convoy velocity of 20 m/s. Both runs set the separation distance at 15 m. Once again, the position of each vehicle was recorded in regular intervals from start to finish, with a total of 4,000 samples taken per vehicle. Because the requirement is for adjustable rate of march, we will compare how performance changes between 10 m/s and 20 m/s for both controllers to determine which one better handled adjusting of rates of march.

Figure 2-15 shows the velocity of each vehicle over time for the 10 m/s and 20 m/s rates of march using the Stanley convoy controller. Table 2-8 displays information about the vehicle velocities, the time when the peak velocities were achieved, and how long it took the vehicle velocities to reach a steady state.

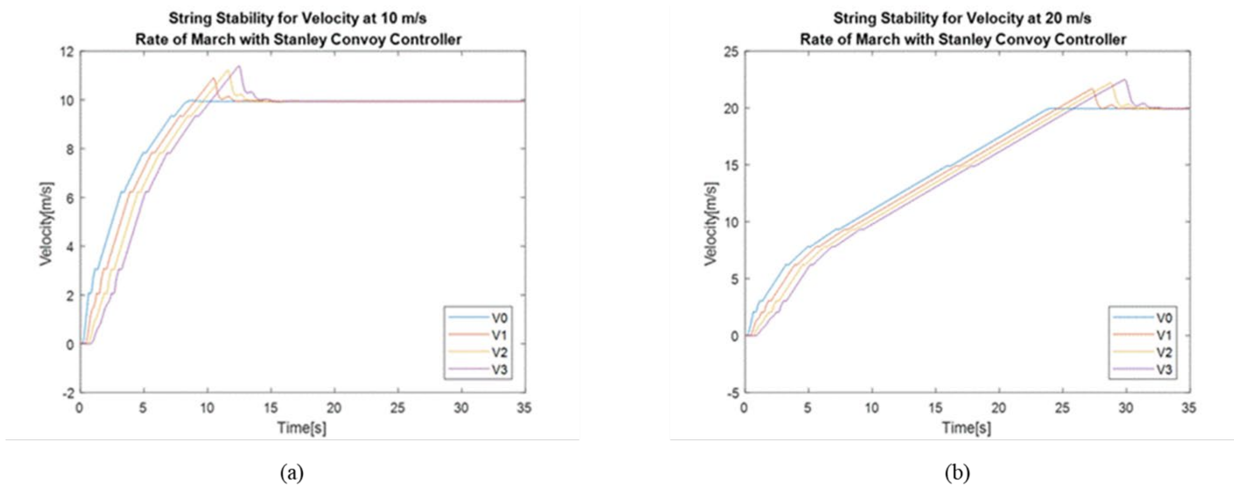


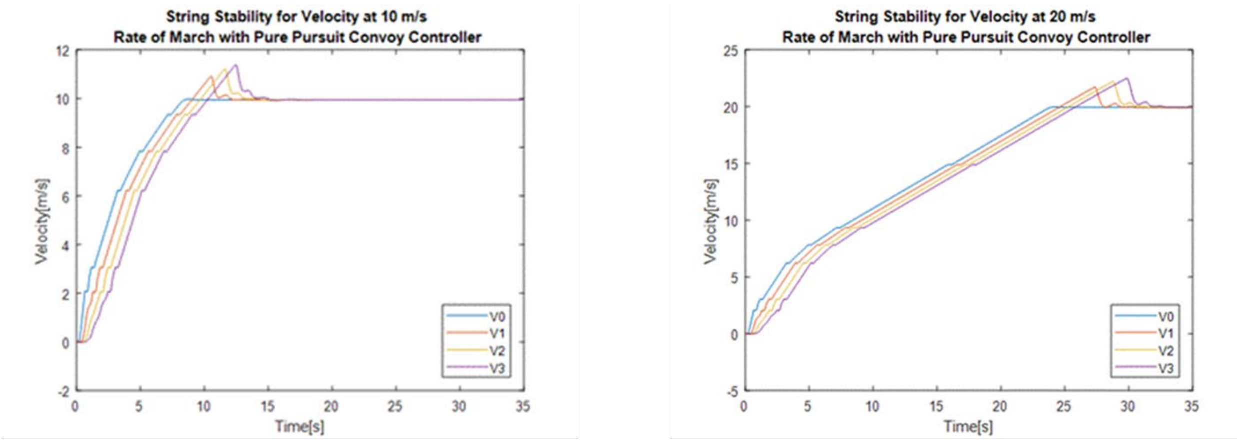
Figure 2-15. Velocity of each vehicle over time for (a) 10 m/s and (b) 20 m/s rates of march with the Stanley controller.

Table 2-8. Vehicle velocity information for different rate of march settings using the Stanley controller.

	10 m/s Rate of March				20 m/s Rate of March			
	V0	V1	V2	V3	V0	V1	V2	V3
Maximum Velocity (m/s)	9.9685	10.8875	11.2124	11.3903	19.9681	21.6799	22.1936	22.4943
% Change from Previous Velocity	N/A	9.22%	2.98%	1.59%	N/A	8.57%	2.37%	1.36%
Peak Time (s)	8.77	10.48	11.61	12.47	24.27	27.3	28.76	29.86
Steady State Time (s)	9.19	12.21	14.49	15.14	24.47	28.04	32.31	32.86
Time to Steady State After Peak (s)	0.42	1.73	2.88	2.67	0.2	0.74	3.55	3

As seen in Table 2-8, both rate of march settings for the Stanley controller failed to achieve string stability. This can be seen by looking at the "% Change from Previous Velocity" row, where the percentage differences increased from the desired 10 m/s setting and 20 m/s setting along the members of the convoy. In a string stable system, the percentage change in velocity would have decreased throughout the vehicles after passing the desired 10 m/s and 20 m/s settings. It is notable however that the percentage change in maximum velocity increased at a slower rate for the 20 m/s rate of march, compared to the 10 m/s rate of march. This can be seen with the smaller increases in percentage when comparing a vehicle's velocity to that of its leader. This indicates that changing the rate of march setting from a 10 m/s to 20 m/s setting does not cause an increase in string instability.

Figure 2-15 the shows velocity of each vehicle over time for the 10 m/s and 20 m/s rates of march using the Pure Pursuit convoy controller. Table 2-9 displays information about the vehicle velocities, the time when the peak velocities were achieved, and how long it took the vehicle velocities to reach a steady state.



(a) (b)

Figure 2-16. Velocity of each vehicle over time for (a) 10 m/s and (b) 20 m/s rates of march with the Pure Pursuit controller.

Table 2-9 Vehicle velocity information for different rate of march settings using the Pure Pursuit controller.

	10 m/s Rate of March				20 m/s Rate of March			
	V0	V1	V2	V3	V0	V1	V2	V3
Maximum Velocity (m/s)	9.9685	10.9084	11.2177	11.3822	19.9681	21.7086	22.1985	22.48796
% Change from Previous Velocity	N/A	9.43%	2.84%	1.47%	N/A	8.72%	2.26%	1.30%
Peak Time (s)	8.77	10.52	11.62	12.45	24.27	27.35	28.77	29.84
Steady State Time (s)	9.19	12.25	14.49	15.14	24.47	28.09	32.42	32.88
Time to Steady State After Peak (s)	0.42	1.73	2.87	2.69	0.2	0.74	3.65	3.04

As seen in Table 2-9, both rate of march settings for the Pure Pursuit controller failed to achieve string stability as well. This can once again be seen by looking at the "% Change from Previous Velocity" row, where the percentage differences increased from the desired 10 m/s setting and 20 m/s setting throughout the convoy. In a string stable system, the percentage change in velocity would have decreased throughout the vehicles after passing the desired 10 m/s and 20 m/s settings. Like the Stanley convoy control, it is once again notable that the percentage change in maximum velocity increased at a slower rate for the 20 m/s rate of march, compared to the 10 m/s rate of march. This also indicates that changing the rate of march setting from 10 m/s to 20 m/s setting does not cause in increase string instability.

When comparing the performance difference between the Stanley and Pure Pursuit convoy controllers in examining System Parameters, adjusting the rate of march setting had very similar effects for vehicle velocity string stability. For both the Stanley and Pure Pursuit convoy controllers, neither the 10 m/s nor 20 m/s rate of march settings demonstrated a string stable velocity. In addition, the steady state velocity for both convoy controllers for each vehicle in 10 m/s and 20 m/s settings averaged 9.95 m/s and 19.945 m/s, respectively. Even with those similarities however, there were some minor variances that differentiated performance. The steady state time for each vehicle was reached earlier for the Stanley convoy controller when compared to the Pure Pursuit convoy controller for both rate of march settings. As seen in Table 2-10, using the Stanley convoy controller either maintained or hastened the time it took to reach steady state for every vehicle. This was true for both the 10 m/s rate of march setting and the 20 m/s rate of march setting. This indicates that using the Stanley convoy controller allowed the convoy to reach the desired rates of march more quickly when compared to Pure Pursuit.

Table 2-10. Comparison of When Steady State Velocity was Achieved for the Stanley and Pure Pursuit Convoy Controllers.

	10 m/s Rate of March				20 m/s Rate of March			
	V0	V1	V2	V3	V0	V1	V2	V3
Pure Pursuit Steady State Time (s)	9.19	12.25	14.49	15.14	24.47	28.09	32.42	32.88
Stanley Steady State Time (s)	9.19	12.21	14.49	15.14	24.47	28.04	32.31	32.86
% Difference from Pure Pursuit to Stanley	0.00%	-0.33%	0.00%	0.00%	0.00%	-0.18%	-0.34%	-0.06%

Overall, we utilized the AGVC Performance Metrics Framework to compare performance between the Stanley and Pure Pursuit convoy controllers based on System Parameters policies. We prioritized Requirement 3 ("Convoy system must have an adjustable rate of march") as the key area of interest. Based on that prioritization, we selected velocity string stability as the primary metric of comparison and found that vehicles under the Stanley convoy controller reached the steady state velocity more quickly for all rate of march settings, as compared to the Pure Pursuit convoy controller. This meant that by leveraging our framework, we determined that the Stanley convoy controller performed better than the Pure Pursuit convoy controller based on metrics relating to System Parameters.

2.7 Discussion

By reviewing and analyzing both Army doctrine and the field of AGVC research, we were able to develop the AGVC Performance Metrics Framework. With the multitude of benefits provided by AGVCs, it is important to understand what autonomous convoy technology best serves the various commercial and military convoying needs. Even with the framework however, comparative performance is highly dependent on system requirement prioritization. No sole factor singularly defines the quality of a convoy, and considerations such as terrain, hostile forces, and size of the convoy elements may change what can be considered the best choice for an AGVC solution. The purpose of the framework is to provide a way to compare different AGVC efforts, but the user must have a strong understanding of the baseline needs and requirements to make a meaningful comparison. The intended outcome of this effort is to better understand how AGVC technologies perform relative to one another, and to have metrics to improve upon between the research and development of new systems.

2.8 Conclusion

In this effort, we performed a review of Army doctrine to derive requirements for convoy performance. Through examining publicly available doctrine, we identified seven key requirements to be met in developing AGVCs for a military context. By doing a survey of AGVC efforts, we found that metrics related to spacing policy, string stability, and path following error were commonly used and could be leveraged as the basis for a framework of performance comparison between different AGVC systems. With that framework in hand, we showed a sample application, comparing the performance of a Stanley convoy controller and a Pure Pursuit convoy controller. By creating this framework, we look to enable future AGVC development efforts to properly baseline and compare performance between existing systems, to find optimal solutions when utilizing AGVCs.

Chapter 3 Jam Mitigation for Autonomous Ground Vehicle Convoys via Behavior-Based Robotics

Autonomous ground vehicle convoys (AGVC) heavily rely on wireless communications to perform leader-follower operations, which make them particularly vulnerable to denial-of-service attacks such as jamming. To mitigate the effects of jamming on autonomous convoys, this chapter, based on our previously published work [17], proposes a behavior-based architecture, called the Behavior Manager, that utilizes layered costmaps and vector field histogram motion planning to implement motor schema behaviors. Using our proposed Behavior Manager, multiple behaviors can be created to form a convoy controller assemblage capable of continuing convoy operations while under a jamming attack. To measure the performance of our proposed solution to jammed autonomous convoying, simulated convoy runs are performed on multiple path plans under different types of jamming attacks, using both the assemblage and a basic delayed follower convoy controller. Extensive simulation results demonstrated that our proposed solution, the Behavior Manager, can be leveraged to dramatically improve the robustness of autonomous convoys when faced with jamming attacks and can be further extended due to its modular nature to combat other types of attacks through the development of additional behaviors and assemblages. When comparing the performance of the Behavior Manager convoy to that of the basic convoy controller, improvements were seen across all jammer types and path plans, ranging from 13.33% to 86.61% reductions in path error.

3.1 Introduction

Ground vehicle convoys are utilized in both commercial and military applications in order to reduce costs, increase vehicle efficiency, and improve safety for the transport personnel [1]. At the most basic level, a convoy is a group of two or more vehicles, traveling from a set origin to an objective destination under a single leader [2]. From a military perspective, the use of ground vehicle convoys to perform supply operations is an important part of an efficient logistics strategy. While several different transportation methods such as aircrafts, trains, and ships are available, ground vehicles still account for a significant portion of supply and equipment distribution due to battlefield complexities and the need to embed protective measures such as gun trucks [3]. Similarly, the usage of ground vehicle convoys is vital in the commercial sector, due to the prevalence of paved roads and cost effectiveness when moving large or heavy materials [1].

With the advent of autonomous driving, there has been heavy focus on leveraging autonomous vehicles to improve convoys. Much of the research and development towards autonomous vehicle convoys focus around three iterative employment concepts: minimally manned (MM), partially manned (PM), and fully autonomous (FA) [11]. In a minimally manned system, the lead vehicle in the convoy has a human driver, while the follower vehicles operate autonomously with a safety rider monitoring autonomous performance, ready to take over in the event a fault occurs. In a partially manned system, the lead vehicle convoy has a human driver, but all the follower vehicles are autonomous and unmanned. In a fully autonomous system, all vehicles in the convoy, leader and followers, are unmanned and autonomous. In all cases, there are numerous benefits in applying autonomy to ground vehicle convoys. From both the commercial and military perspective, the utilization of autonomous vehicle convoys can reduce fuel consumption, reduce traffic congestion, and improve safety [3].

In all the autonomous vehicle convoy employment concepts, the utilization of wireless communications has been paramount in their development. Two important applications of wireless communications include inter-vehicle communications (IVC) and Global Positioning Systems (GPS). IVC is used to exchange vital information between vehicles in the convoy, including pose, heading, speed, acceleration, and maneuver intentions [64]. This information is used by follower vehicles in maintaining the desired convoy formation along the desired path. In addition, GPS is often used as a vital tool in providing position, navigation, and timing information for each vehicle [11]. In many cases, proper wireless communications are a necessity in the operation of an autonomous convoy, where disruption of the wireless communications system can cause the autonomous convoy to fail. The criticality of proper wireless communications extends beyond autonomous convoys and applies to various other autonomous applications such as surveillance [65], mining, and agriculture [11]. This outsized importance of wireless communications makes it a critical piece of the system to protect from potential cyber-physical attacks, such as jamming.

Jamming is a type of denial-of-service attack that attempts to disrupt or block wireless communications. Jammers interfere with a radio’s ability to transmit or receive data, and are the most widely used denial-of-service attack for vehicular networks due to their effectiveness and simplicity [14] [15] [16]. The risk of jamming is especially prevalent in military applications, due to electronic warfare systems and strategies employed by adversaries to gain tactical advantages [15]. Given the importance of IVC in autonomous convoys, disruptions caused by jamming could cause instability in the leader-following functionality of the autonomous convoys, or even potentially cause a complete breakdown of the convoy [11]. Since jamming is one of the most simple and effective denial-of-service attacks [66], it is important to understand the attack and the proper ways to defend against it. There have been some limited efforts focused more narrowly on the topic of jamming as it relates to convoys, with a greater body of work focused on investigating the effects of jamming on vehicular ad-hoc networks (VANET) in general [67] [68] [69].

On the topic of jamming and its effects on vehicular convoys, there has been some work in developing a framework to measure the impact of the attack and a convoy's resilience. Van der Heijden et al. examined three Cooperative Adaptive Cruise Control convoy controllers and how various attacks such as jamming and data injection impacted their respective performance [16]. Both an attacker model and an evaluation framework were developed for a thorough examination of what sort of attack detection and attack resilience algorithms should be used to resist the effects of attacks on the VANETs. In another convoy specific effort, Hu et al. investigated how stealthy jamming attacks impacted convoy stability [70]. The stealthy jammer performed jamming against a simulated convoy's basic safety message (BSM), which contains information such as vehicle speed, position, and acceleration, over the control channel of dedicated short-range communications (DSRC), but only when the probability of being detected was low. The probability of being detected was determined based on how significantly jamming at a particular time would affect the packet loss ratio of the communications. To detect the stealthy jams, they analyzed received power and transmission delays of messages to differentiate between jamming attacks and normal interference. The effects of jamming on VANETs in general is a more thoroughly explored topic. Various efforts have focused on how jamming affects the transmission of BSMs over DSRC in relation to Forward Collision Warning capabilities [67] [68] [69]. These efforts create attack models to quantify the effects of the jamming attacks and come up with different mechanisms to mitigate their effects. Alturkostani et al. looked at vehicle distance with packet delivery ratio to detect jams, causing the system to enter a fail-safe mode when detected [67]. Serageldin et al. utilized redundant channels and data through dissimilar message types to reduce the effectiveness of jams [69]. In a different effort, Serageldin et al. looked to develop architectures resistant to jamming by using dual and triple-redundant channels at higher power ratings to enable BSM transmission [68]. Other efforts have leveraged more of the Wireless Access in Vehicular Environment standards in trying to detect jams, such as Malebary's work in looking at beacon frequency from roadside units to detect jams and broadcast warnings to other network members in its vicinity [14]. As a whole, prior efforts related to jamming of convoys and VANETS have had a heavy focus on detection, with a response of either broadcasting warnings to convoy vehicles or entering fail-safe modes. Rather than investigating the issue of jam detection, which has been the focus of the previous literature, our proposed solution instead focuses on controls-oriented countermeasures when jamming is unavoidable. We aim to develop methods and techniques to allow for continued convoy operation in the face of these denial-of-service attacks, mitigating the undesired outcome of jamming attacks, rather than the jamming itself.

There are a variety of control approaches and architectures utilized in multi-robot autonomous convoying. More broadly speaking, multi-robot convoying is a subset of formation control, and the proper choice for formation control design is dependent on which factors need to be prioritized, such as formation shape generation time [71], formation robustness during network congestion [72], and flexible prioritization between system performance and control effort [73]. One particularly robust formation control approach for dynamic environments is behavior-based robotics. In behavior-based robotics, robotic control is built as a collection of basic behaviors. These behaviors are modular and run in a concurrent and distributed manner to achieve desired system functionality. The utilization of behavior-based robotics architectures in multi-robot teaming has been previously explored in various research efforts seeking to take advantage of its benefits in dynamic environments. Early efforts include DARPA studies on the integration of navigational behaviors and formation behaviors to create human-led robotic teams to be used in different types of task environments [74]. This effort looked to take four robot teams and set them in a line, column, diamond, or wedge using a motor schema behavior approach. The behaviors used were move-to-goal, avoid-static-obstacle, avoid-robot, maintain-formation, and noise. Member vehicles in the robotic team transmitted position information to properly space out appropriately given the formation and method of formation centering. The path error and duration out of formation of the robots for each formation was compared to one another to determine which one had the best performance in different experimental scenarios. Another effort proposed a decentralized method utilizing a formation matrix [75]. In this effort, each robot leveraged a formation matrix, which defined leader-follower pairs. This formation matrix allocated robots to specific positions in a formation before the system began operation by using onboard sensors to detect the location of surrounding robots. Once the proper position of each member robot was allocated, the system was able to traverse to a destination avoiding obstacles, while maintaining the desired formation based on what was defined in the matrix. In this approach, a behavior network architecture was used, in which basic behaviors were linked together and either stimulated or suppressed. This stimulation/suppression was based on input vector values such as detected object location. The output vectors of the network defined the robot's path. While this effort did not use communications between robots in motion, initial allocation of vehicles in the specific starting positions was required for formation matrix setup. In addition, this effort focused on static robotic formations that maintained their shape, making this method less suitable for applications such as a convoy following defined roads. More recent efforts in behavior-based multi-robot teaming have sought to improve performance by optimizing inter-robot communications, rather than eliminating them. One such effort created a distributed framework for multi-robot behavior sequencing that allowed teams of robots to adjust their configuration to meet communication requirements for the different tasks [76]. The effort focused on the idea that coordinated behaviors between multiple robots must be sequenced together in a way that takes inter-robot distances into account to meet information flow constraints due to communications. The constraints translate into specific robot position configurations that need to be met within a finite time for the behaviors to be performed. The framework leveraged finite-time convergence control barrier functions to adjust configurations to meet the communication requirements for different sequences of behaviors. While these efforts focused on utilizing behavior-based control architectures for multi-vehicle teaming, they either heavily rely on IVC throughout their operations or require a manually arranged initial state. Furthermore, the focus on traveling in formation shapes makes the multi-vehicle movement less suitable for convoy path following operations, which is the outcome being sought by industry and military from autonomous convoying. Our efforts focus on the development of a more flexible framework towards convoy path following that utilizes IVC when available and having the additional capability to activate more robust behaviors not relying on communications when faced with jamming to continue convoy operations.

This chapter investigates developing an autonomous convoy system that mitigates the effects of jamming attacks. We aim to create an autonomous convoy that does not inherently rely on any Vehicle-to-Vehicle (V2V) or Vehicle-to-Infrastructure (V2I) network communications by design, creating a highly scalable system that is robust to system error and loss of communications by using only the on-board sensors of each vehicle. A behavior-based robotics architecture is used to facilitate the employment of simpler sensors and reduce the reliance on complex planning with world models.

The contributions of this chapter are as follows:

- This chapter develops controls-oriented countermeasures against jamming attacks on autonomous vehicle convoys that are independent of traditional radio anti-jamming techniques. This allows for layers of protection against jamming at both the network and convoy controller level, improving the robustness and performance of an overall convoy system when confronted with jamming.
- This chapter creates an architecture for behavior-based robotics that is uniquely integrated with the Robot Operating System (ROS) framework. This architecture combined a Motor Schema behavior-based robotics approach with the ROS navigation stack to a depth that had not previously been seen, paving the path for the greater ROS community to leverage behavior-based robotic architectures.
- The jamming mitigated motion planning in this chapter improves upon the basic Motor Schema approach by integrating it with vector field histogram motion planning, allowing us to avoid the pitfalls of potential field motion planning.

The remainder of the chapter is organized as follows: Section 3.2 provides background on AGVCs, jammers, and behavior-based robotics; Section 3.3 and Section 3.4 discusses the materials and methods used, including the behavior-based robotics architecture, algorithms, and design of experiment; Section 3.4 additionally shows the results to quantify the performance differences between AGVCs when jam mitigation is present; and Section 3.5 contains the final discussions and conclusions.

3.2 Background

The purpose of this effort is to develop an AGVC system that mitigates the effects of jamming by utilizing a behavior-based robotics approach. To enable a proper exploration of the effort, the following sections provide a brief background on three topics fundamental to our work: AGVCs, jammers, and behavior-based robotics.

3.2.1 Autonomous Ground Vehicle Convoys

AGVCs are composed of a leader vehicle and follower vehicles. Follower vehicles follow the path of the leader at a given offset distance. While there are various levels of autonomy employed by AGVCs, such as MM, PU, and FA [11], the fundamental design of autonomous convoys remains consistent. The leader vehicle sets the path, either through human driving or autonomous navigation, and utilizes V2V communications to send information to the follower vehicles. The information sent varies depending on system design, but typically includes path points, vehicle pose, vehicle speed, and other sensor information to help follower vehicles track their leader. The follower vehicle uses onboard sensors to estimate its own state, and produces a state estimate of its leader using the information received from V2V and the aforementioned sensors. A path planning system uses the state information and leader path points to generate a motion plan, and the vehicle is actuated accordingly [1]. Figure 3-1 lays out the high-level architecture as described.

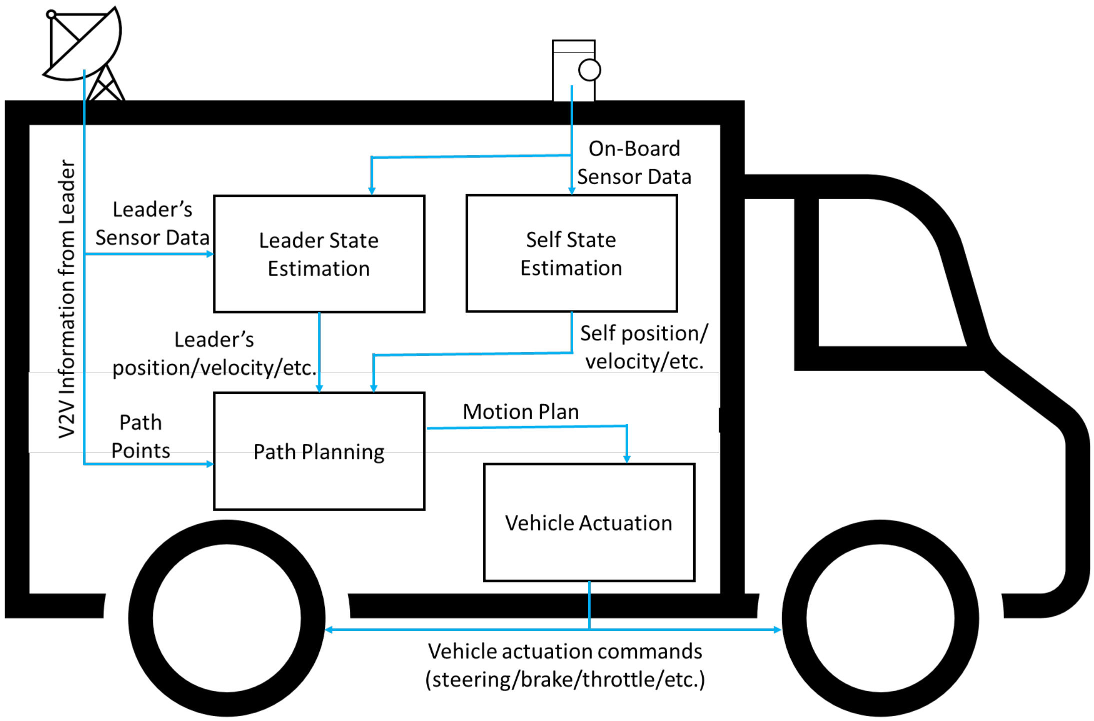


Figure 3-1. High-level architecture for a follower vehicle in an AGVC.

The V2V communications sent between vehicles can contain a broad range of information from a wide suite of sensors. Information relevant to the vehicle's speed and position, such as sensor data (cameras, GPS, Light Detection and Ranging [LiDAR], wheel encoders, etc.), vehicle kinematics, and intended maneuvers, would all need to go through the V2V communications to allow follower vehicles to reach the desired speed and formation for leader following [3]. Various V2V communication topologies can be utilized in AGVCs, depending on the needs of the system. Figure 3-2 shows some common leader-follower topologies used, including predecessor following, predecessor-leader following, bidirectional, and bidirectional-leader [77].

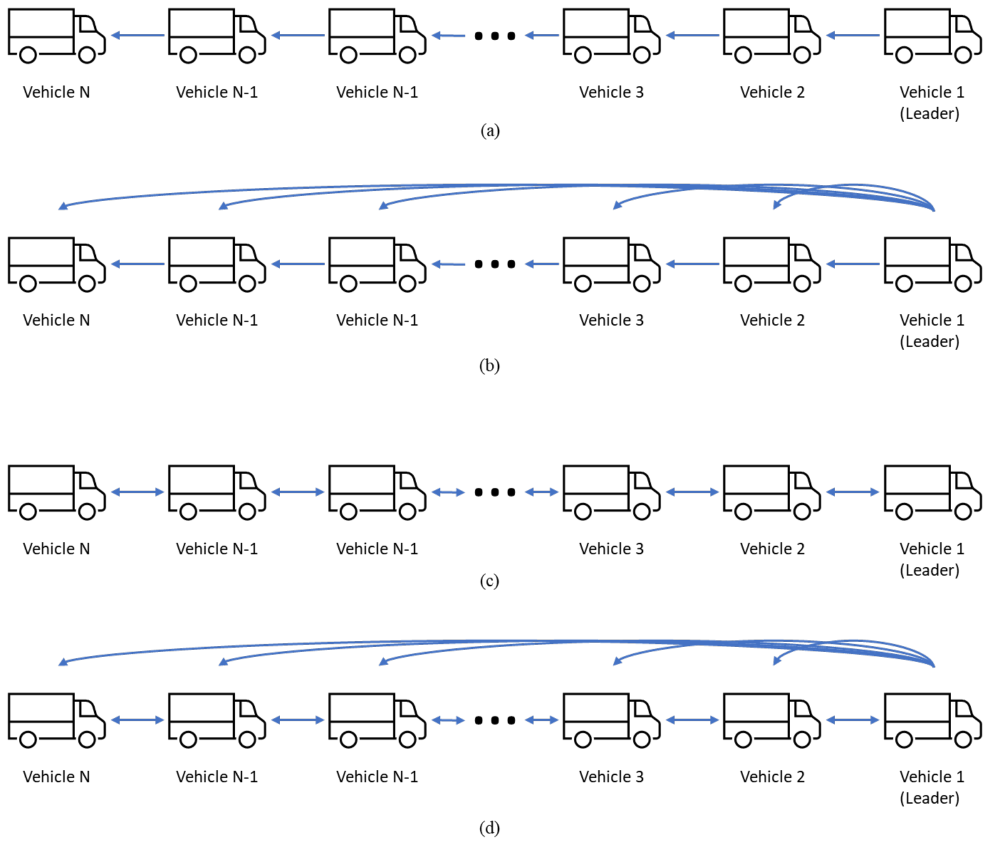


Figure 3-2. Common leader-follower communication topologies, adapted from [77]: (a) predecessor following; (b) predecessor-leader following; (c) bidirectional; (d) bidirectional-leader.

3.2.2 Jammers

Jamming is an effective and simple radio interference attack that can be used against wireless networks. In its most basic form, jammers emit radio frequency signals to fill a wireless channel to interfere with the sending and receiving of wireless communications. The jammer can work to either prevent the source from transmitting packets or disrupt the receiver from properly recognizing and processing legitimate packets. The end result in either case is the interference of the physical transmission and reception of wireless communications [66]. Different classifications of jammers utilize different techniques in jamming wireless networks. The five most commonly described techniques in literature are as follows in Table 3-1.

Table 3-1. Common jamming techniques.

Jammer Type	Description
Constant [14] [66] [67] [69] [78]	Continuously emits random noise over a wireless medium to interfere with legitimate communications.
Deceptive [66] [69] [78]	Periodically inject valid packets in their transmissions to deceive receivers into believing that legitimate messages are being sent.
Random [14] [66] [67] [69] [78]	Operates on jam and sleep periods. They will jam for a time duration of t_J, and sleep for a duration t_S before jamming again. Both t_J and t_S can be either fixed or values random.
Reactive [14] [66] [69] [78]	Continuously monitor a communications channel and only jams when it senses activity.
Intelligent [67] [78]	Uses knowledge of the communications protocols they are seeking to jam and analyze the transmissions to target specific messages or message types.

Despite the relative simplicity of basic jammers such as constant jamming and random jamming, jamming attacks are difficult to defend against. Conventional security mechanisms are ineffective, because attackers can disrupt the entire medium of communications itself rather than having to exploit the traditional areas of confidentiality, authentication, and integrity [66]. Furthermore, the deployment of jammers has a low barrier to entry. All radio operations on a wireless communications channel can be completely disrupted by a single device emitting noise at high power, regardless of what security measures are built into the underlying communications protocols. In addition, Software Defined Radios that can be used to launch jamming attacks are becoming increasingly affordable and easy to use [15]. In response to the vulnerabilities presented by jamming, several efforts throughout the years have proposed different anti-jamming methods. While many novel anti-jamming methods exist, the primary mechanisms to mitigate jamming can be broadly classified into larger groupings, with two prominent groupings being filtering and spread spectrum (SS) techniques. In filtering techniques, the radio receiver attempts to filter out the jammed signal by various means so that the intended signal can be properly parsed. One method of doing this used beamforming, in which a beamformer performed spatial filtering of spatial samples collected from propagating wave fields [79]. It then separated signals with overlapping frequency content that were delivered from different spatial locations. In a different filtering approach, an adaptive Gaussian filter was used to combat jamming caused by narrowband interference with gaussian white noise and pulsed noise [80]. The adaptive filter used optimal time-frequency localization and variable notch depth in conjunction with a fast Fourier-transform-based correlation to filter both continuous and time-varying narrowband interference. In addition to filtering techniques, SS techniques are often used to combat jamming. With SS techniques, a narrowband information signal uses data-independent, random sequences to spread the signal over a wide band of frequencies, in hopes that the jammer is unable to disrupt the entire frequency band [81]. The receiver can then correlate the signal it received with a copy of the random sequence to decode the data that was meant to be delivered. The two most commonly used SS techniques are direct sequence SS (DSSS) and frequency hopping SS (FHSS) [82]. In DSSS systems, the information signal is spread throughout the entire available bandwidth at the same time. This spreading of the signal causes the energy present at each particular frequency of the bandwidth to be very low and mistaken as noise by jammers that are actively listening for signals to jam. The intended receiver can then reconstruct the signal that it received from all the frequencies in the bandwidth. In FHSS systems, data is sent through a narrowband signal, but the frequency and channel being used to transmit signal rapidly changes. The frequency hopping pattern is known by the sender and intended receiver, allowing for messages to successfully be sent between them. In addition to filtering and spread spectrum techniques, various unique anti-jamming methods have been proposed to protect against the attacks, including exploiting reactive jammer reaction times [83], jammed node isolation [84], and directional antennas [85]. While all these methods can be effective at countering jamming attacks, no solution can offer absolute protection in all scenarios. For this reason, creating robust autonomous convoy solutions that can maintain operation in the face of a jamming attack is important for widespread commercial and military adoption.

3.2.3 Behavior-Based Robotics

The development of behavior-based robotics was a response to the prevalent control paradigms of the time, which were rooted in function-based, deliberative models [86], as shown in Figure 3-3a. Under the deliberative approach, robotic control relied heavily on world models, which is a model of the environment, relevant entities in the environment, and the state of the robot [87]. Updates to the world model would be determined in a sensing phase, plans based on the model were calculated in a planning phase, and action would then be taken in an acting phase. This approach responded poorly in dynamic environments due to the slow speed of model updates coupled with the need for re-planning [87].

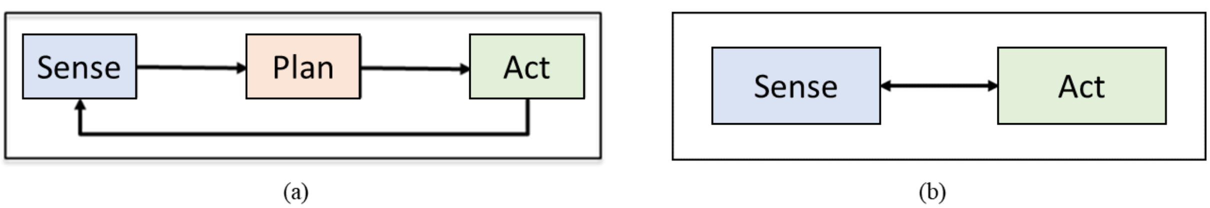


Figure 3-3. Robotic control approaches: (a) deliberative approach; and (b) reactive approach, adapted from [87].

To address the issues of deliberative control, researchers began to develop more reactive approaches with faster response times that would be more suited to the dynamic, real-world environments. In these reactive control approaches, there is a tight coupling between sensors and actuators. Control is performed via concurrent condition-action rules so that no complex computations or world models are needed, as shown in Figure 3-3b. A sensor reads the environment and acts upon the information given the condition-action rules in place. The tight coupling minimizes the need for heavy computation and complex world model [87]. Despite the benefits afforded by reactive control approaches, a purely reactive approach limits an autonomous system's ability to perform complex tasks. While there are various approaches that leverage reactive control with internal states and models, one approach that is particularly well suited to dynamic environments and multi-vehicle control is behavior-based robotics.

In a behavior-based robotics approach, robotic control is built as a collection of basic behaviors, with behaviors being defined as something that generates a motor response given some sensory stimulus. These behaviors are modular, and run in a concurrent and distributed manner to achieve desired system functionality. For instance, an autonomous vehicle that moves from its origin to some target location may be concurrently running behaviors such as Move-to-Goal, Avoid-Obstacles, and Stay-on-Path at all times in order to meet the system objective. These different behaviors may be purely reactive, in which they take sensor readings and use rule-based logic to immediately generate some desired motor response, or they can maintain their own state and memory to allow for more complex behaviors. With behaviors being run concurrently, an arbitration mechanism is used to decide on what actions to take, whether it be a single behavior's output or a fusion of multiple behaviors [88]. Arbitration techniques vary based on the specific behavior-based robotics architecture that is being used. Some examples include subsumption architectures [86], motor schemas [89], and circuit architectures [90]. Figure 3-4 shows an example of how a generalized behavioral-based robotics architecture can be laid out.

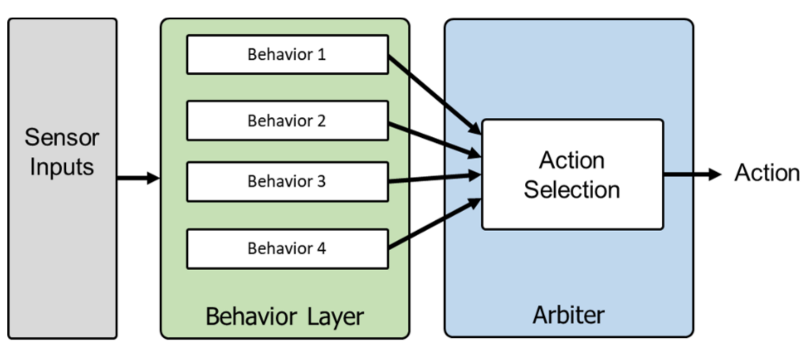


Figure 3-4. A generalized behavioral-based robotics architecture.

The behavior-based robotics architecture developed for this effort was based on Motor Schemas, a neurological-based approach towards behavioral robotics inspired by schema theory [89]. A schema is a unit of behavior that describes how an agent should react in a given situation and how the reaction can be performed [88]. The concurrent control of many different schemas is used to explain motor behavior. In the Motor Schema approach, each motor schema has an associated perceptual schema that provides the sensor information, which the motor schema uses to generate motor responses per its defined behavior. This is traditionally conducted through potential field response vectors around the robot. Coordination of motor schemas is achieved cooperatively, with no hierarchy of behaviors. Instead, weightings for the motor schemas are set based on a robot's current needs and combined through vector addition. The coordination between different schemas allows for the execution of complex, emergent actions with only a small number of primitive behaviors.

3.3 Materials and Methods

This following section details the development and testing of the proposed behavior-based AGVC system. A description of the tools leveraged, jamming attacker model, behavior-based robotics architecture, experimental design, and data analysis methods ensue for a thorough discussion on the quantifiable benefits of the proposed system.

3.3.1 Tools

Development of our AGVC system leveraged ROS. ROS is a flexible framework for the development of autonomous robotic software, leveraging a collection of libraries, tools, and conventions to facilitate and encourage collaborative software development and reuse [91]. In addition, Gazebo, an open-source 3D robotics simulator, was used in development, due to its integration with ROS and overall community support [92]. Utilizing ROS and Gazebo in our effort allowed us to build upon pre-existing models for our own developmental and simulation environment. To that end, the Clearpath Husky model was chosen as the robotic platform for development.

The Clearpath Husky is a robotic platform that is specialized for research and rapid prototyping [93]. The ROS libraries and Gazebo models are well supported with a large user base, making it an ideal platform to leverage for development [94]. The sensor suite used in this effort includes a GPS, inertial measurement unit, and a forward facing 270° field-of-view LMS1xx LiDAR system. Figure 3-5 shows a convoy of Clearpath Husky models in a column convoy formation, along with ROS sensor visualization tools showing the 270° field-of-view LMS1xx LiDAR for each vehicle.

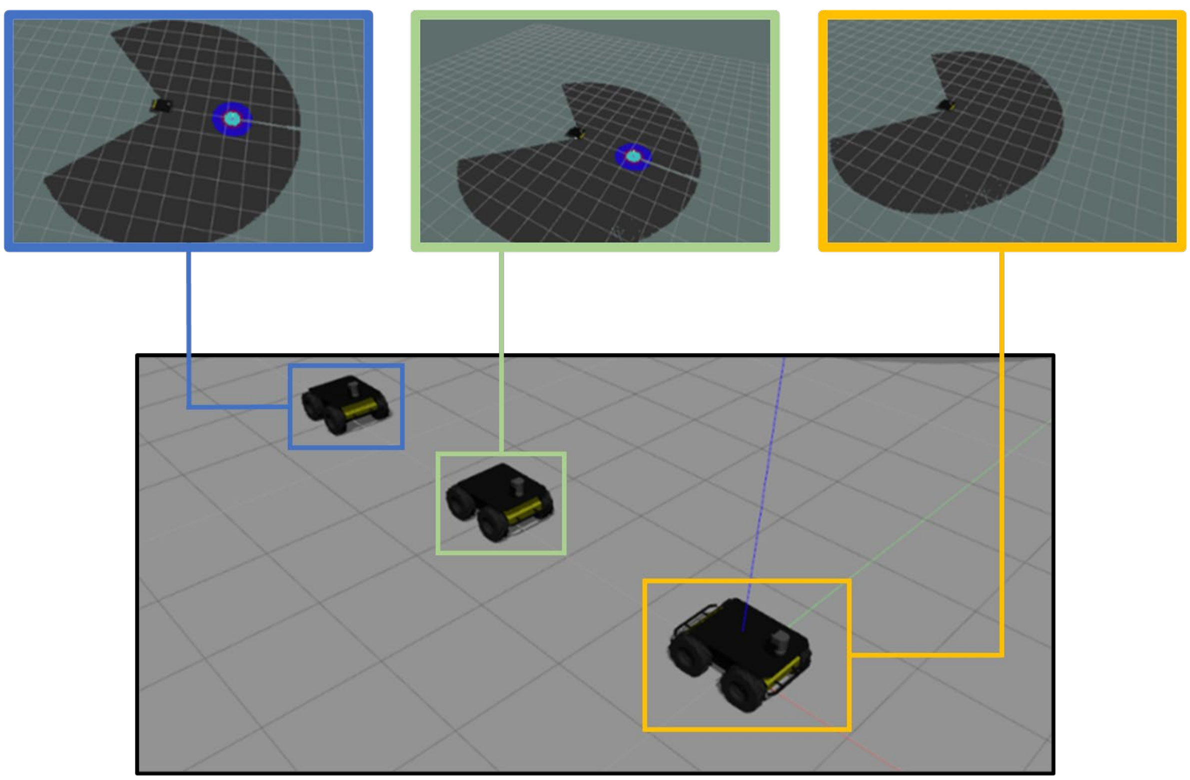


Figure 3-5. Convoy of Clearpath Husky robots in the Gazebo robotics simulator with each robot's sensor visualization in ROS tools.

3.3.2 Attacker Model

To develop a jamming attacker model, certain assumptions regarding the network configuration of the robots had to be established beforehand. We assumed that the three Clearpath Husky robots were equipped with wireless radios, allowing them to become mobile nodes. Each node would be able to transmit and receive data packets, forming a generic autonomous vehicular ad-hoc network, similar to what is described in an IEEE 802.11bd network. In addition, each node would be able to leave and rejoin the network without causing the ad-hoc network to fail. Due to the proximity of the robots, we simplified the overall model by assuming that no signal loss or degradation occurs from separation distances between the robots.

The jamming attacker model developed for this effort allows for a choice between simulating a constant jammer and a random jammer. The constant jammer type was chosen due to it being the worst-case jamming attack [69], while the random jammer type allows us to test a greater range of unique jamming scenarios in which follower vehicles end up being partially in the jamming zone. The other types of jammers described are focused on disguising jamming attacks. Since detecting disguised jamming attacks was not a focus of this effort, we focused on constant and random attacker models.

The jammer developed for this effort is configurable with the following inputs:

- lat, long – latitudinal and longitudinal coordinates for the center of the jammer;
- r_J – radius of the jamming area, centered at lat, long;
- type of jammer;
- t_J – jamming time for random jammer;
- t_S – sleep time for random jammer.

The latitudinal and longitudinal coordinates for the jammer's center and the jamming radius create a cylindrical jamming zone. Figure 3-6 illustrates the jamming attack model targeted at a robotic convoy.

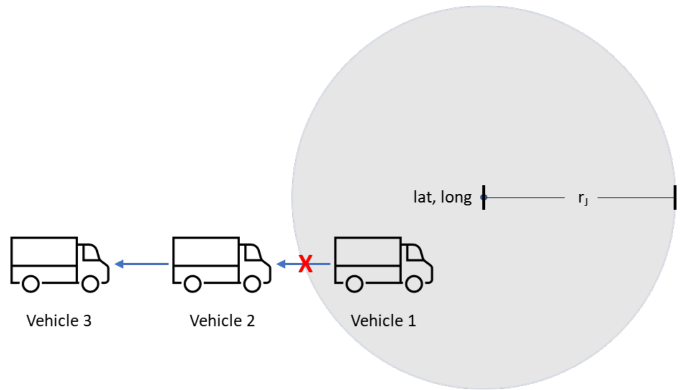


Figure 3-6. A convoy entering a jamming zone of radius r_J centered at lat, long.

3.3.3 Behavior Manager

As previously described, a behavior-based robotics architecture uses a collection of basic behaviors to perform robotic control, where behaviors are modular components that generate motor responses given some sensory stimulus, and are run in a concurrent and distributed manner. Behavior-based robotic architectures are particularly well-suited to multi-vehicle control in dynamic environments due to scalability, decentralization of control, and tight coupling of rapid sensor readings to real-time path planning [87] [95]. The advantages inherent to a behavior-based robotics architecture make it highly well-suited to provide robust protections against jamming for an autonomous convoy. Using the Motor Schema approach as a starting point, we developed a novel behavior-based robotics architecture that uniquely takes advantage of the layered costmap system implicit in the ROS navigation stack. Our behavior-based architecture, henceforth referred to as the Behavior Manager, encompasses behavioral costmaps, behaviors, assemblages, and how they holistically interact. The following sections provide details on the general design of the Behavior Manager, along with the specific implemented system we are proposing for jamming mitigation of autonomous convoys.

Behavioral Costmaps

A core component of the Behavior Manager is the collection of behavioral costmaps. Traditionally, costmaps are two-dimensional grids in which every cell represents a traversibility cost around a robot, based on sensor readings. These costs are used in calculating the optimal path when traversing to a goal by a path planner [87]. ROS utilizes a layered costmap system, in which multiple, separate costmaps are defined for different contexts. Each costmap is considered a layer that is combined to create a master costmap, which is utilized by the path planner [96].

The Behavior Manager leverages ROS' layered costmap system by having each layer represent the costs as defined by a perceptual schema behavior. In essence, every costmap layer exists due to a behavior dictating that a costmap is needed to perform some behavior. These costmap layers, which we call behavioral costmaps, are utilized both individually and as a combined master costmap by the different behaviors and by the Behavior Manager's path planning system. In our implementation of the behavioral costmaps, the costmaps are 200 × 200 grids, with every cell containing a value between 0 and 254. The autonomous vehicle is centered at the approximate center of the costmap, at the grid index (100,100). Figure 3-7 shows a simplified example of the behavioral costmaps.

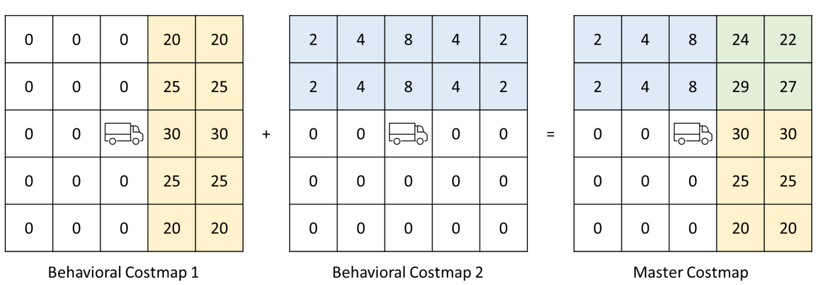


Figure 3-7. Simplified example of 5 × 5 behavioral costmaps, combined into a master costmap via addition.

Behaviors

In general, behaviors are modular components that generate motor responses given some sensory stimulus which are run in a concurrent and distributed manner [88]. In the Motor Schema approach, the definition of behaviors is expanded, and they are categorized as either motor schemas, which generate motor responses per its defined behavior, or perceptual schemas, which provide sensor information to the other behaviors [89]. The Behavior Manager is based on the Motor Schema approach, and accordingly, categorizes its behaviors as either perceptual schemas or motor schemas.

Behavior Manager perceptual schema behaviors utilize sensor data and generate behavioral costmaps based on the requirements of the behavior. For example, Figure 3-8 shows how a Keep-Standoff-Distance perceptual schema behavior would work. Figure 3-8a shows the world state, with the autonomous vehicle in the center, and an obstacle being detected in the upper right-hand corner. Figure 3-8b shows what the LiDAR readings would be, with a value of 253 being associated with the detected obstacle. Figure 3-8c shows the final behavioral costmap, with the perceptual schema increasing the costs in the area directly around the obstacle to meet the behavior's goal of keeping a standoff distance between the autonomous vehicle and objects detected by the sensors. Each perceptual schema has a gain that it assigns to its behavioral costmap to be used when combining the layers into a master costmap. Multiple perceptual schemas are run concurrently, creating multiple behavioral costmaps that are used by the greater Behavior Manager in individual behaviors and in overall path planning.

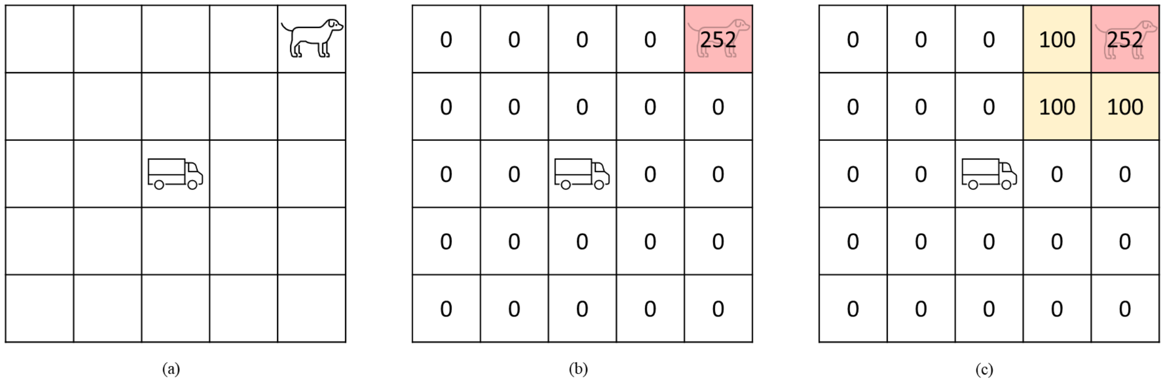


*Figure 3-8. A simplified example of a Keep-Standoff-Distance perceptual schema behavior: (**a**) the world state, with the autonomous vehicle in the center, and an obstacle being detected in the upper right-hand corner; (**b**) costmap with LiDAR readings; (**c**) final behavioral costmap, with the perceptual schema increasing the costs in the area directly around the obstacle.*

Behavior Manager motor schemas produce target goals, expressed as two-dimensional Cartesian coordinates **T**, based on the requirements of the behavior. For example, Figure 3-9 shows two different notional motor schema behaviors. Figure 3-9a represents a Stay-on-Road behavior. The target goal produced, represented by the star, causes the vehicle to stay on the road. Figure 3-9b on the other hand, represents a Follow-Leader behavior, with the leader being represented by a dog on the grass. The target goal produced directs the vehicle towards the leader. Each motor schema has a gain that it assigns to the goal that it produces. The goals and associated gains are utilized by the Behavior Manager to determine the final goal point for path planning.

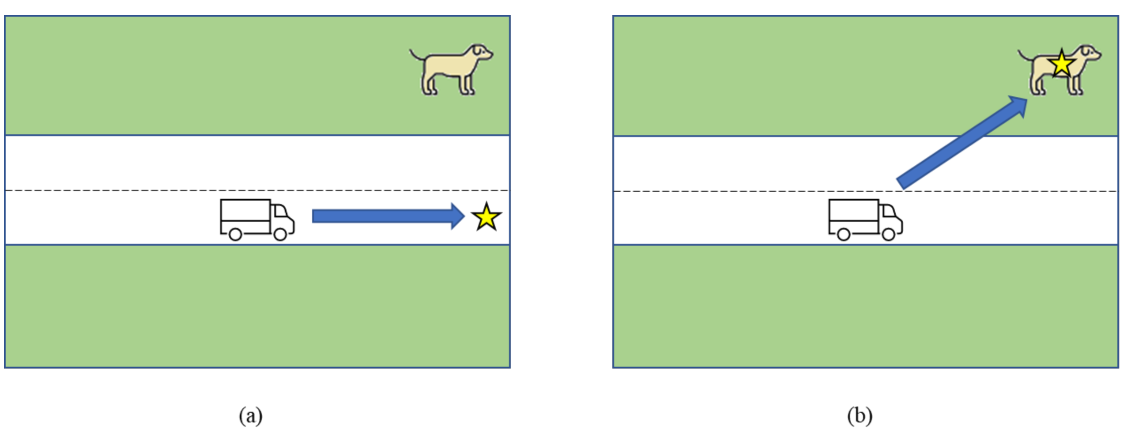


Figure 3-9. Two examples of motor schema behaviors: (a) Stay-of-Path; (b) Follow-Leader.

In the development of our autonomous convoy system, we created four behaviors to allow for leader following in the presence of jams: two motor schemas and two perceptual schemas. The behaviors are Move-to-Goal, Maintain-Formation, Avoid-Obstacle-Proximity, and Avoid-Leader-Zone, respectively.

Move-to-Goal

The Move-to-Goal motor schema behavior enables basic goal following. A high-level planner provides a coordinate **B** as an input to the Move-to-Goal behavior. The behavior then publishes the coordinates as the target goal. For our autonomous convoy following system, Move-to-Goal is used by the vehicles to move towards GPS breadcrumbs provided by their respective leaders for a predecessor-following convoy approach as follows:

$$\mathbf{T} = \mathbf{B} \tag{3.1}$$

where new GPS breadcrumb B coordinates are provided periodically.

Maintain-Formation

The Maintain-Formation motor schema behavior mitigates the effects of jamming attacks on autonomous convoy following by generating target goals **T** that are not dependent on a leader's GPS coordinates. Instead, the behavior uses the density-based spatial clustering of applications with noise (DBSCAN) algorithm to find a target goal nearest to the last known valid goal and to continue convoy operations during jamming attacks.

DBSCAN is a non-parametric clustering algorithm that, when given a set of points, groups together points that are closely packed together into different clusters [97]. It was chosen as the clustering algorithm to use due to its ability to discover cluster of arbitrary shapes and efficiency with large datasets. In the Maintain-Formation behavior, the points provided are from the behavioral costmaps generated by the Avoid-Obstacle-Proximity perceptual schema. For every cluster identified, the center of mass for each cluster is calculated for comparison to the last known valid **T** coordinates. To find the center of mass $\mathbf{C}_i$, where **C** are the coordinates of the center of mass for cluster i, and $i = 1, \ldots, n$ clusters, we considered each cluster as a system of weighted particles P_j, where $j = 1, \ldots, q$. The coordinates for each particle were given by the costmap cell coordinates c_j, with the mass m_j for each particle being defined as the cost for each given coordinate. The formula for $\mathbf{C}_i$ is as follows:

$$\mathbf{C}_i = \frac{1}{M_i} \sum_{j=1}^{q} m_j c_j, \quad (3.2)$$

where M_i is the total cost of all the cluster points in cluster i, given by:

$$M_i = \sum_{j=1}^{q} m_j, \quad (3.3)$$

When the autonomous vehicle fails to receive a GPS breadcrumb coordinate **B**, the Behavior Manager determines that a denial-of-service attack is occurring and that communications with the rest of the convoy have been disrupted. The Maintain-Formation behavior will then find the $\mathbf{C}_i$ nearest to the last valid target goal **T**, and set that as **T**. This process is repeated until a new valid **B** is provided, as shown in Algorithm 1.

Algorithm 1 Maintain-Formation
1: **if** no **B** received from leader
2: $\mathbf{C}_i$ = center of mass of clusters 1, ..., n from DBSCAN(costmap)
3: temp_distance = high value placeholder
4: **for** i = 1, …, n
5: cluster_distance = distance between **T** and $\mathbf{C}_i$
6: **if** cluster_distance < temp_distance

```
        T = C_i
        temp_distance = cluster_distance
    end if
  next i
end if
```

Avoid-Obstacle-Proximity

The Avoid-Obstacle-Proximity perceptual schema behavior produces a behavioral costmap that includes costs for objects detected by LiDAR, along with a buffer zone of costs around the objects in which the cost of the cell decreases as the distance from the object increases, up to a preset radius. The implementation of the buffer zone leverages the inflation layer provided by the ROS Navigation stack [98], which assigns a cost around objects with the following formula:

$$\text{cost} = e^{(-1 \times s \times (d - r))}(o - 1), \tag{3.4}$$

where s is a cost scaling factor, d is the distance from the obstacle, r is the robot's radius, and o is the cost to assign a cell that falls within the robot's radius. The ROS Navigation stack sets s to a default value of 10, which was kept in our experiments. Refer to Figure 3-10 for an example of how the buffer zone inflation is represented on a costmap. The buffer zone provided by the Avoid-Obstacle-Proximity behavior helps to prevent the vehicle from getting too close to obstacles, decreasing the chances of collisions.

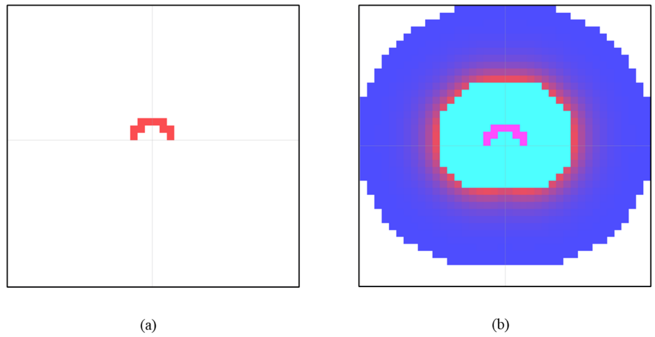


Figure 3-10. Costmap inflation in the Avoid-Obstacle-Proximity behavior: (a) the costmap produced from LiDAR sensing a construction cone; (b) the costmap produced from creating a buffer zone of inflated costs around the construction cone.

Avoid-Leader-Zone

The Avoid-Leader-Zone perceptual schema behavior produces a behavioral costmap that inserts a high-cost ring of a preset radius around a robot's leader. This ring creates a zone around the leader that the follower vehicles will not enter, in effect enforcing a following distance between the vehicles. To do this, the behavior takes the position of its leader as an input. If network communications are available, the position is provided wirelessly by the lead vehicle. If the convoy is under a jamming attack, the target goal **T**, as provided by the Maintain-Formation behavior, is used instead. The costmap coordinates that comprise the ring are calculated using basic trigonometric functions to obtain coordinates of a scaled unit circle with an offset center, as follows:

$$x_i = \text{floor}\left(\frac{a}{f}\cos\left(\frac{i}{n}2\pi\right)\right) + l_x, \tag{3.5}$$

$$y_i = \text{floor}\left(\frac{a}{f}\sin\left(\frac{i}{n}2\pi\right)\right) + l_y, \tag{3.6}$$

where $[x_i, y_i]$ are costmap indices for each point in the ring; $i = 1, \ldots, n$ costmap cells, with n representing the total number of costmap cells the ring should contain; a is the radius of the leader's following distance zone; f is the costmap cell resolution; and $[l_x, l_y]$ is the costmap index of the leader. The values used for the parameters were as follows:

- n = 100 cells
- a = 4 m
- f = 0.05 m/cell

Figure 3-11 shows a robot and the leader zone produced by the Avoid-Leader-Zone behavior.

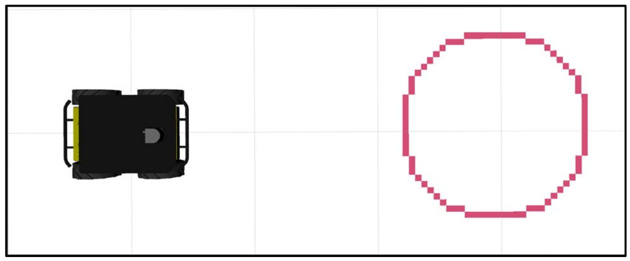


Figure 3-11. Behavioral costmap created by the Avoid-Leader-Zone perceptual schema behavior.

Assemblages

At a basic level, assemblages are the complex tasks that the robot is trying to accomplish. The assemblages are composed of multiple concurrently running behaviors. The behaviors can be classified as motor schemas, which produce goals to generate motor responses, or perceptual schemas, which generate behavioral costmaps. The behavioral costmaps are used by the motor schema behaviors in goal generation and by the assemblage for navigation. The assemblage acts as a coordinator of the behaviors, weighing and combining them as appropriate for the task at hand. Figure 3-12 shows the architecture of an assemblage.

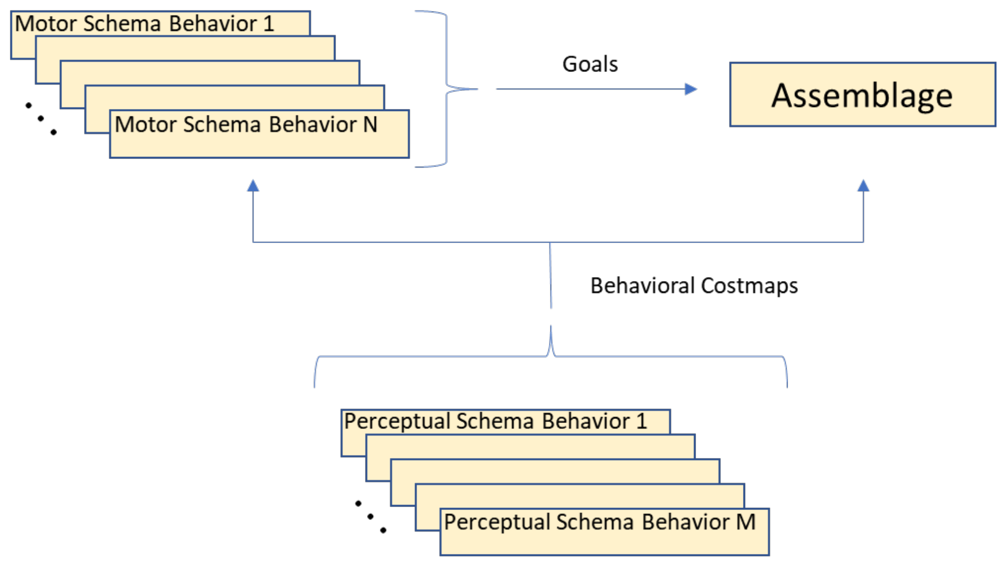


Figure 3-12. Architecture of an assemblage.

For our effort, we utilized the behaviors described above and developed a Follow-Leader-with-Jam-Mitigation (FLJM) assemblage that performs autonomous convoying with jam mitigation techniques. The architecture of the assemblage and the associated behaviors can be seen in Figure 3-13.

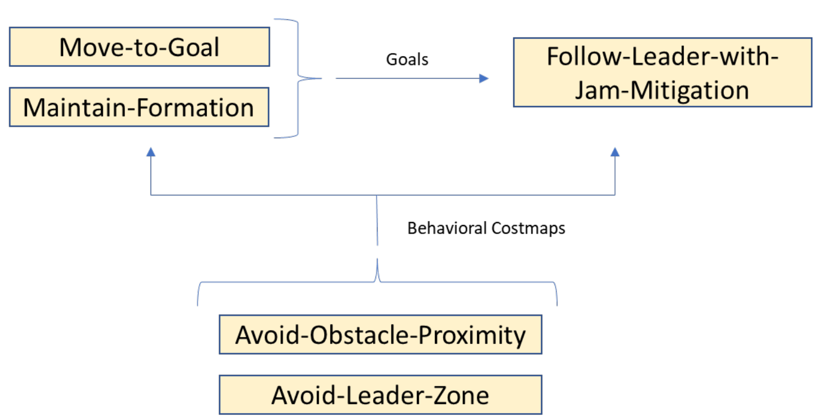


Figure 3-13. Architecture of the Follow-Leader-with-Jam-Mitigation assemblage.

The basic task of the FLJM assemblage is to perform leader following. An autonomous vehicle running the assemblage will take an assigned leader and await waypoints from them to follow. If the vehicle's network communications are unavailable, either due to a system failure or a denial-of-service attack, it will continue following the moving cluster of costmap points nearest to the last valid waypoint. Once network communications and waypoints are available again, the assemblage will revert to following waypoints. As with all assemblages, the FLJM assemblage is responsible for goal and behavioral costmap combination/selection for all the behaviors. The costmap gains for Avoid-Obstacle-Proximity and Avoid-Leader-Zone are set to be equal, meaning that combining their behavioral costmaps is performed through addition. Essentially, the costmaps are stacked on top of each other and the sum cost of every cell forms the master costmap. For goal combination, a selection method is utilized. If the system detects valid waypoints from network communications, the **T** provided by Move-to-Goal is used, allowing for waypoint following. If no network communications are received, the **T** provided by Maintain-Formation is used until network communications are restored and waypoints are provided again. Figure 3-14 characterizes the interactions between behaviors and waypoint transmissions through the network.

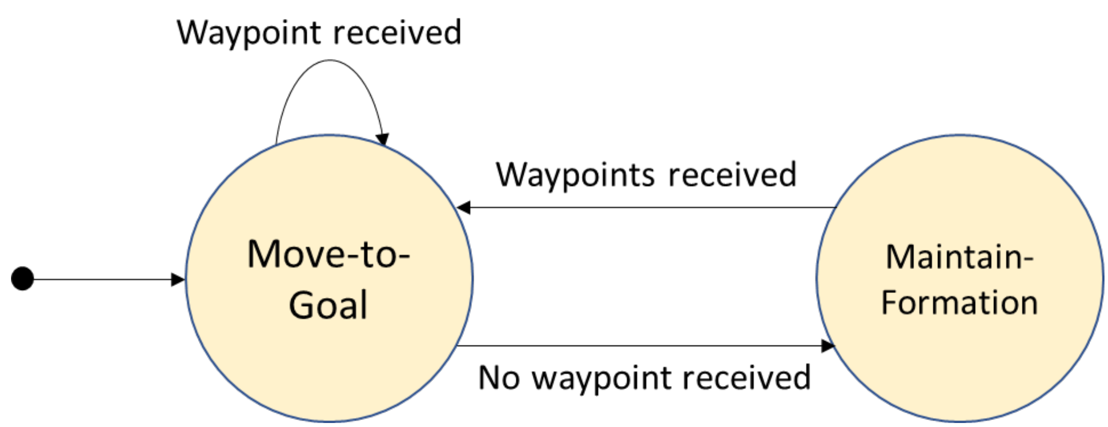


Figure 3-14. State diagram of the Follow-Leader-with-Jam-Mitigation assemblage.

To implement the FLJM assemblage, the behaviors and assemblage were developed as ROS nodes. In ROS, a node is a software module that performs computation [91]. ROS systems are composed of multiple, concurrently running nodes that communicate with each via a publisher/subscriber messaging system, leveraging the layered costmap system previously described with the behavioral costmaps. A node was developed for each motor schema behavior. The motor schema behavior nodes publish their respective **T** goals to the messaging systems. In addition, nodes were developed to manage the perceptual schema behaviors. These perceptual schema nodes ensured that the proper behavioral costmaps associated with each perceptual schema was active and in use. The FLJM assemblage itself is a separate node that takes in the various **T** goals, behavioral costmaps, and other information to perform the autonomous convoying as described.

Path Planning

A key difference between the Behavior Manager and the traditional Motor Schema approach is how motor responses are generated and acted upon. While the Motor Schema approach uses potential field response vectors and navigation, the Behavior Manager leverages layered costmaps with Vector Field Histogram (VFH) path planning. VFH is a path planning technique that was developed specifically to address the inherent limitations of potential field methods, such as U-shaped obstacle traps, oscillation in narrow corridors, or the inability to pass between closely spaced objects [99]. The VFH method represents the obstacles around a robot with certainty values in a Cartesian histogram grid. Data reduction over the histogram grid reduces it to a polar histogram of angular sectors around the robot with associated polar obstacle densities. Based on the desired goal and polar obstacle densities, drive and steering commands are given that best avoid obstacles while making progress towards the goal [100]. VFH was chosen as our path planning technique because of the aforementioned Cartesian histogram grid. By using the combined layered behavioral costmap as the Cartesian histogram grid, we naturally integrated costmap-based perceptual schemas with path planning, thereby avoiding potential field methods and their pitfalls.

3.3.4 Experimental Setup

To test the efficacy of the FLJM assemblage, we compared its performance to that of a basic convoy controller in a Gazebo simulation environment. The Gazebo world used in the experiment contained a flat ground pane without terrain or obstacles and a three-vehicle convoy of Clearpath Husky robots, a four-wheel skid-steer robotic vehicle. The setup was kept minimal to reduce the variables present when testing convoy performance. For the basic convoy controller, we implemented a leader-follower controller that performed "delayed following" with a predecessor following network topology. In a "delayed following" approach, a lead vehicle records the path it drove and relays that information to followers to repeat [101]. This approach is recognized as a common leader-follower convoying method [52], making it the ideal baseline for comparison. The implementation for the experimental setup had a 4 m separation distance between leaders and followers and leveraged the same VHF path planner that the FLJM assemblage used.

The high-level autonomous convoy setup was kept consistent between the basic convoy controller and the FLJM assemblage for an accurate comparison. In both cases, the autonomous convoy was FA and consisted of three vehicles: one convoy leader that drove along preset path plans, and two autonomous followers. The convoy leader precisely followed through given path points, allowing for repeatability in test runs, resulting in better comparisons between the basic convoy controller and the FLJM assemblage. Both unmanned followers would be running either the basic convoy controller or FLJM assemblage to perform autonomous following, depending on which system was being tested at the time. Each vehicle recorded GPS breadcrumbs of its position that it transmitted to its direct follower when wireless network communications were available.

The two preset path plans created for the test runs were a square loop and a roundabout turn, as seen in Figure 3-15. These paths were chosen to mimic common road formations and driving maneuvers. The square loop tests basic turns, while the roundabout tests the intersections of the same name. In addition, jamming zones were established along the paths to simulate the effects of denial-of-service attacks on the autonomous convoy. Figure 3-16 overlays the jamming zones along the paths.

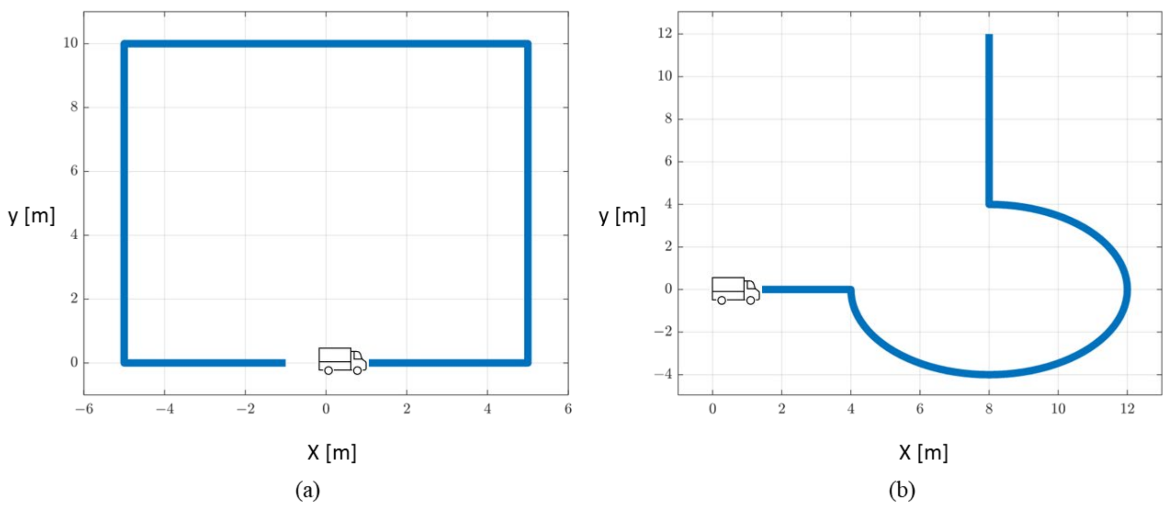


Figure 3-15. The preset path plans for the lead vehicle: (a) square loop and (b) roundabout.

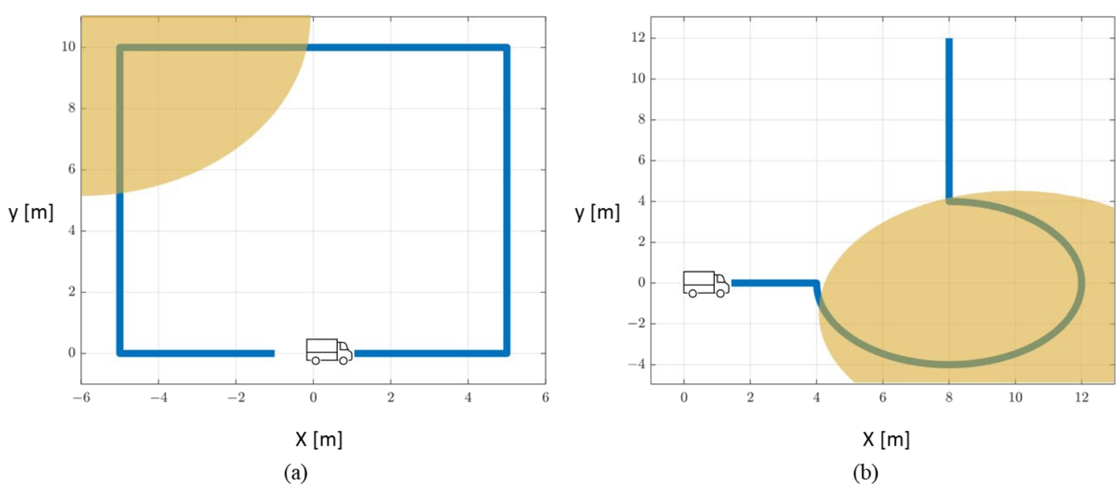


Figure 3-16. Jamming zones overlaid onto the preset plans for the lead vehicle: (a) square loop and (b) roundabout.

To comparatively test the performance of the convoy controllers, multiple test runs with the three vehicle convoy were performed for each jammed path shown in Figure 3-16. Ahead of each test run, the autonomous followers were configured to use either the basic convoy controller or the FLJM assemblage. Five runs were performed under constant jamming, and an additional five runs were performed under random jamming, with t_J, and t_S set to 10 s and 2 s, respectively. Both the basic convoy controller and the FLJM assemblage were tested in this fashion for each path plan.

3.4 Results

To measure the performance of the convoy controllers, we utilized the AGVC Performance Metrics Framework and determined that goal specification was the most relevant category for comparison. Based on that categorization, we chose path following error as the proper metric to use when comparing the convoy controllers, and compared the mean absolute error (MAE) between the follower vehicles' path and the convoy leader's path. The MAE was chosen as the measurement for evaluation as it has a direct interpretation to the real-world quantities being compared, namely the Euclidian distances between follower points and convoy leader points, in addition to not inflating the penalty for larger errors via squaring, which occurs in other commonly used performance measurements such as root mean squared error [55]. The paths of both the followers and the convoy leader were sampled at a rate of 1000 Hz, respectively creating arrays of position points for each vehicle. The Euclidean distance between the follower position points and corresponding convoy leader positions points were then used to calculate the MAE with the following formula:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |e_i|}{n}, \qquad (3.7)$$

where e is the Euclidean distance between the follower and convoy leader position points, and $i = 1, \ldots, n$, where n is the size of the array e. Table 3-2 shows the average MAE over five runs for each convoy controller for both the square path and roundabout path when under a constant jamming attack. Figure 3-17 shows the paths of the five runs taken by the followers, overlaid on the convoy leader's goal points for a visual comparison of performance.

Table 3-2. Average MAE under a constant jamming attack.

Convoy Configuration	Square (m)	Roundabout (m)
Follower 1 Using Basic Convoy Controller	1.2665	3.1334
Follower 2 Using Basic Convoy Controller	1.9226	3.5205
Follower 1 Using FLJM assemblage	0.4942	0.4197
Follower 2 Using FLJM assemblage	0.7321	0.8452

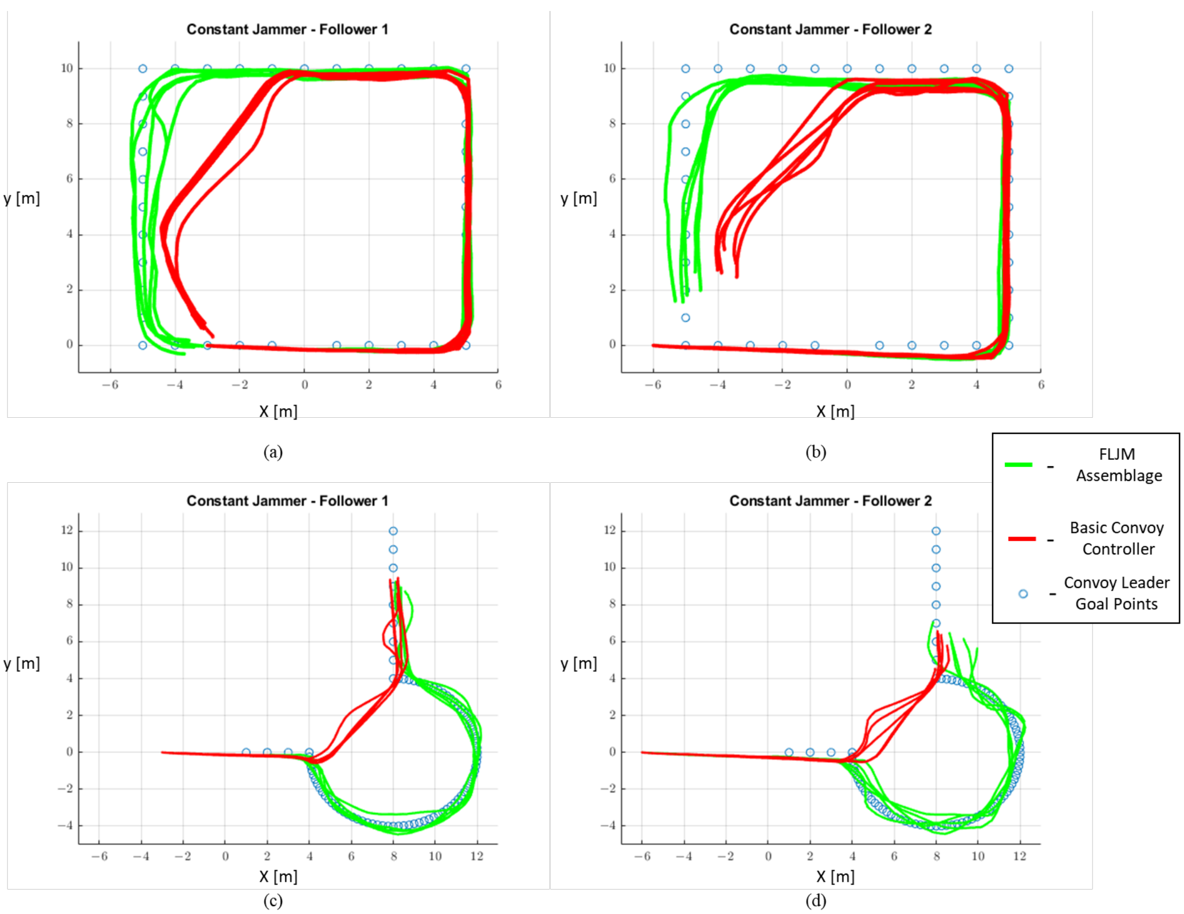


Figure 3-17. Plot of the paths taken by the followers under constant jamming on the square path for (a) follower 1 and (b) follower 2; and on the roundabout path for (c) follower 1 and (d) follower 2.

Likewise, Table 3-3 shows the average MAE while under a random jamming attack with t_J, and t_S set to 10 s and 2 s, while Figure 3-18 shows the paths taken for the five runs by the followers, overlaid on the convoy leader's goal points for a visual comparison of performance.

Table 3-3. Average MAE under a random jamming attack.

Convoy Configuration	Square (m)	Roundabout (m)
Follower 1 Using Basic Convoy Controller	0.6844	1.2637
Follower 2 Using Basic Convoy Controller	1.0028	1.0697
Follower 1 Using FLJM assemblage	0.4680	0.4821
Follower 2 Using FLJM assemblage	0.7654	0.9271

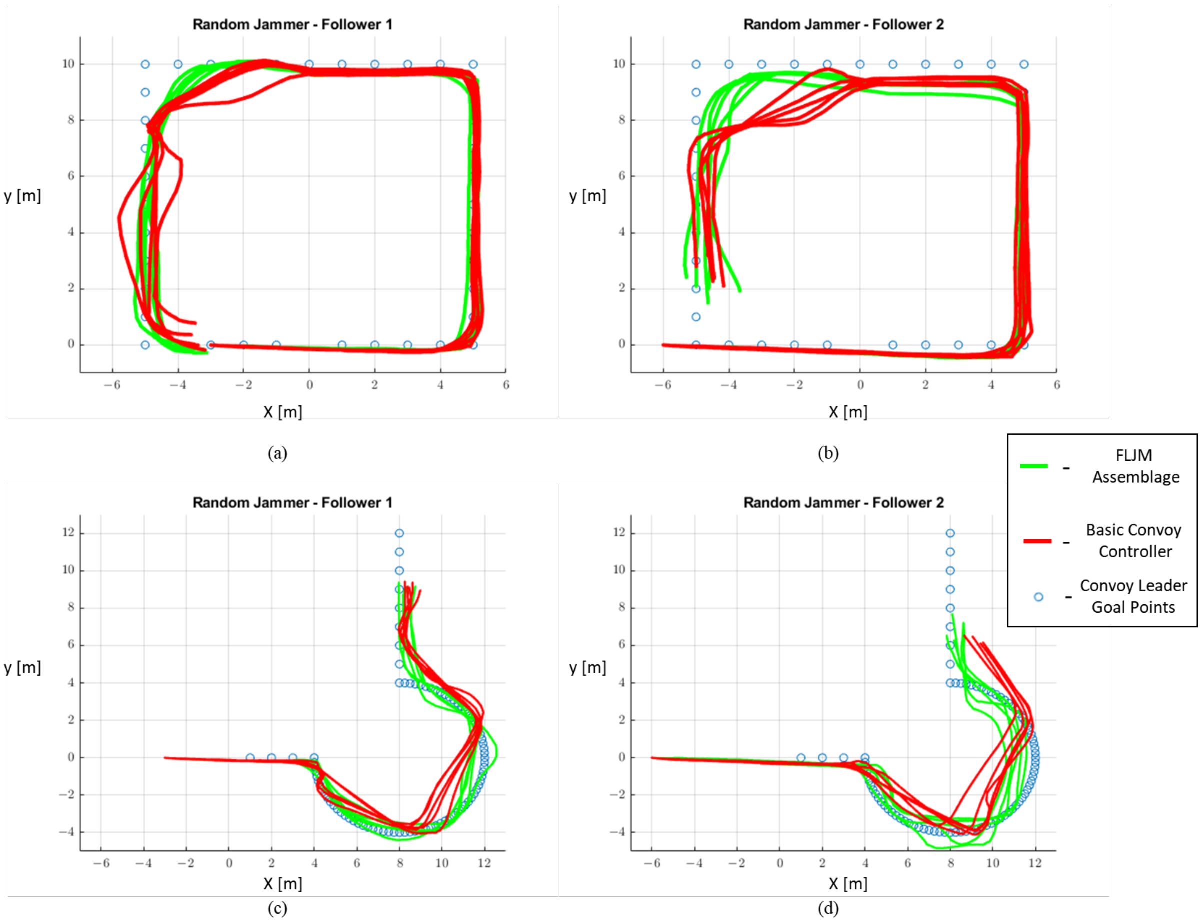


Figure 3-18. Plot of the paths taken by the followers under random jamming on the square path for (a) follower 1 and (b) follower 2; and on the roundabout path for (c) follower 1 and (d) follower 2.

As seen in Table 3-2, the utilization of the FLJM assemblage significantly improved convoy performance across both path plans when compared to the basic convoy controller under constant jamming. When following the squared loop path, using the FLJM assemblage decreased the average MAE by 60.98% for follower 1 and 61.92% for follower 2. The improvement was even more drastic with the roundabout path, where the average MAE was decreased by 86.61% and 75.99% for follower 1 and follower 2, respectively. These significant improvements were due to the basic convoy controller missing turns that the convoy leader performed in the jamming zones laid out in yellow in Figure 3-16. As seen in Figure 3-17, the FLJM assemblage allowed the followers to successfully continue leader-following inside the constant jamming zones, whereas the stoppage of communications prevented the followers from getting proper waypoints when relying solely on basic delayed following.

As seen in Table 3-3, the utilization of the FLJM assemblage also improved convoy performance across both path plans when compared to the basic convoy controller under random jamming. When following the squared loop path, using the FLJM assemblage decreased the average MAE by 31.62% for follower 1 and 23.67% for follower 2. Improvement was exhibited with the roundabout path as well, where the average MAE was decreased by 61.85% and 13.33% for follower 1 and follower 2, respectively.

While the utilization of the FLJM assemblage yielded similar performance across both the constant and random jammers, the performance improvements when compared to the basic convoy controller were less significant under random jamming attacks. This is due to the nature of random jammers, which alternate between a jamming and sleeping state. Rather than missing all the convoy leader's waypoints in a jamming zone, as in the case of constant jamming, the follower vehicles would periodically receive network transmissions from the jamming zone under random jamming, allowing for occasional transmission of waypoints inside the jammed area. This can be seen when comparing the basic convoy controller paths in Figure 3-17 and

Figure 3-18. In Figure 3-17, the basic convoy controller followers stop outside of the jamming zones and only restarts following when the convoy leader exits the constant jamming zone. In Figure 3-18 however, the followers are able to break into the random jamming zone and follow the occasional waypoints they receive when the jammer is sleeping. These results demonstrate that the value of using jamming mitigation techniques increases accordingly with the severity of the jamming attack. As previously stated, constant jamming is considered the worst-case jamming scenario with the most damaging impact, so the potential benefits afforded by using the FLJM assemblage is greater when compared to a random jamming attack, as seen in the experimental results described here.

3.5 Discussion

Jamming attacks are a simple, yet effective type of denial-of-service attack that have the potential to adversely degrade the performance of autonomous convoys. While many efforts have been undertaken to detect the presence of jams, we focused our efforts on finding ways to mitigate a jammer's effects to allow for continued convoy operation. The prevalence of behavior-based robotics approaches in multi-vehicle teaming made it a prime starting point for developing an approach to allow convoy following when confronted with jamming, and lead to the development of the Behavior Manager.

By utilizing the Behavior Manager and creating behaviors and assemblages to mitigate the effects of jamming, we were able to show improved convoy performance with up to an 86.61% reduction in average MAE. This demonstrates that a behavior-based robotics approach can be leveraged to dramatically improve the robustness of autonomous convoys when faced with jamming attacks. Furthermore, the modular nature of the Behavior Manager means that the capabilities can be extended to mitigate the effects of other types of attacks, such as LiDAR spoofing, replay attacks, and RADAR absorption [102], through the development of additional behaviors and assemblages.

A potential future area of research on the usage of behavior-based robotics towards mitigating the effects of attacks is dynamic behavioral weighting. While the usage of Q-learning towards behavior selection has been researched in the past [103], the application of modern learning techniques, such as deep reinforcement learning, towards multi-robot attack mitigation is a novel area that warrants further investigation. By training on datasets that demonstrate human reactions to attacks on manned convoys, the robustness of autonomous convoy systems utilizing behavior-based robotics could be greatly improved on, creating even more stable systems able to handle a broad spectrum of attacks.

Model predictive control (MPC) is comparable to behavior-based robotics, in that both methods are often applied to multi-vehicle formation control [1] [95]. In MPC, a model of a system is used to predict its future behavior. A cost function and online optimization algorithm are applied to the model to minimize the cost and find an optimal control action towards a desired reference over a prediction horizon. MPC can be used with multi-input and multi-output systems, and can also handle constraints on the inputs and outputs [104]. MPCs have been applied to various vehicular cyber-physical challenges, such as cooperative adaptive cruise control longitudinal motion [50] [105] and truck platoon fuel economy improvement [106] [107]. In terms of formation control, MPCs have been leveraged in a variety of ways to enable leader-follower capabilities and autonomous convoying. In one approach, an MPC controller was used to create a virtual structure, where the formation of the vehicles was defined by a virtual rigid structure that each vehicle tracked to for a relative positioning [108]. The MPC controller was used to generate reference trajectories for the horizontal and lateral offsets of the member vehicles, while a separate nonlinear MPC performed collision avoidance and lane-keeping for autonomous driving for the entire virtual structure. In another approach, a neural-dynamic optimization-based nonlinear MPC was used to enable leader-follower capabilities. This effort utilized a vision-based system, relying on a camera on follower vehicles to track their leaders, measuring state and velocity [109]. Another way in which MPCs were used for autonomous convoying was through a hybrid approach that combined potential fields with MPC. In this approach, the MPC's optimization algorithm was used to determine the motion control produced by the potential field, allowing for synchronization of the motion control and path planning [110].

Despite the variety of ways that MPCs can be used for formation control, there are difficulties in applying the approach [111]. Application of MPCs are mostly restricted to linear systems, with linear constraints and linear/quadratic cost. The usage of MPCs for general nonlinear systems and hybrid systems is a topic that is still under active research. In addition, analyzing and enforcing robustness and stability is difficult with MPC controllers. Doing so requires long prediction horizons, which drastically increase the computational requirements. While it is possible enforce robustness and stability without a long prediction horizon, that requires that the system is enforcing a control Lyapunov function, which is non-trivial. Also, deployment of MPCs on live autonomous vehicles can be impractical due to the computational time needed to solve the optimization problem [105]. The computational time required to solve the optimization problem increases with the number of autonomous vehicles and the prediction horizon. As the dimensionality of state and control input spaces increase, running the MPC can cause significant control delay, and even become intractable depending on the size of the convoy. Table 3-4 outlines the benefits and challenges when utilizing MPCs for the application of robotics [111].

Table 3-4. Benefits and challenges in utilizing MPCs for robotics.

Benefits	Challenges
• Allows for optimal or near optimal performance objective optimization. • Allows for multi-input/multi-objective systems. • Future information along the prediction horizon can be incorporated into system. • Enforces constraints on input and state.	• Difficult to apply to non-linear systems. • Difficult to analyze and enforce robustness and stability without long prediction horizons. • Computationally demanding. • Relatively difficult to tune and calibrate compared to other techniques such as proportional–integral–derivative controllers.

While incorporating Control Theory methods with behavior-based robotics can be done without MPC, such as application of Lyapunov analysis [112] and tracking controllers [113], there have been studies that directly incorporate MPCs with behavior-based robotics in an effort to take advantage of the benefits they provide. In one effort, researchers combined MPCs and behavior-based robotics towards control trajectory optimization. Different tunable feedback control laws were classified as behaviors to be used for state trajectory generation. The tunable parameters of the behaviors were optimized via MPC, allowing the behaviors to better accomplish different tasks relating to the control of a nonholonomic mobile robot [114]. Further efforts in this domain include application of the behavior-based MPC framework to virtual leader formation control [115] and improvements to safe obstacle avoidance in trajectory following [116]. Despite the benefits provided by MPC however, a desired outcome for our effort was to minimize the need for heavy computation and complex world models. MPCs would necessarily increase the computational complexity of the system when added to a behavior-based robotics approach. For this reason, we did not include behaviors that utilized MPCs in our effort.

Along with jamming, another important area of study in relation to autonomy and network communications is time delay in teleoperation. In a teleoperated autonomous vehicle, a human user remotely sends commands to the vehicle through a communications network while receiving sensor feedback such as video feed, GPS location, or vehicle status [117]. The delay inherent in network communications however can potentially cause issues if it is too large or inconsistent. While humans can adopt to small roundtrip time delay in teleoperation, larger delays cause an asynchrony between the desired and actual input. Experiments with teleoperated steering has shown that delays as low as 170 ms can cause for overcorrection in steering, leading to oscillations that degrade teleoperation performance, with the errors being exacerbated at higher speeds [118]. Various methods have been used to help overcome issues related to time delay, such as MPCs [119], model-free predictor frameworks [120], and blended model-based /model-free frameworks [117]. Despite both jamming and time delay being network communications problems however, they are fundamentally different and require different solutions. Jamming attacks aim to completely prevent the sending and reception of communications [66], while time delay, whether they be malicious or due to natural network conditions, will eventually result in delivered messages. The methods for handling blocked communications and delayed communications are necessarily different because one has to do with deciding on the appropriate action given missing information, while the other has to do with compensating for delays in information that will eventually arrive. These topics are not interchangeable and need to be appropriately handled independently.

Chapter 4 Conclusion

The maturation of autonomous ground vehicle convoy (AGVC) systems has been an important goal for both military and commercial spaces due to the improvements in fuel efficiency, traffic congestion, safety, and costs that the technology enables [5] [6] [8] [7] [9] [10]. Given the increasing importance of AGVC systems, it is vital that researchers aim to create robust solutions that are resilient against common cyber-physical attacks. The collection of efforts described in this dissertation work towards the end goal of measurably improving convoy performance in the face of radio jamming, creating safer and more secure AGVCs.

In developing our jamming mitigation system, we focused on controls-oriented mitigation techniques. Anti-jamming, which refers to direct methods to combat jamming, is a highly developed field of research that offers a large variety of solutions in trying to directly counteract radio jams [79] [80] [81] [82] [83] [84] [85]. Despite the level of development however, no solution can guarantee absolute protection against jamming attacks, making it important to have a layered approach in dealing with radio jams. By focusing on controls-oriented jamming mitigation, we can enable a layered security approach through using both anti-jamming systems and having jamming mitigation at the convoy controls level. We focused on a behavior-based robotics approach, the Behavior Manager, to minimize systems requirements and take advantage of the scalability afforded by behavior-based robotics [87] [95]. Jamming mitigation was made possible by developing behavioral assemblages that focused on uninterrupted performance of convoy operations when faced with radio jamming. When radio communications were disrupted, the convoy members relied solely on on-onboard sensors to perform following until communications were recovered. By using the Behavior Manager and its jamming mitigation assemblages, we saw performance improvements with multiple jammer types and path plans, ranging from 13.33% to 86.61% reductions in path error, when compared to a basic convoy controller. This level of improvement, when paired with traditional anti-jamming techniques, has the potential to greatly increase the security and robustness of AGVC systems across both military and commercial sectors.

To gauge the performance improvements afforded by the Behavior Manager, it was important to establish a framework for comparing AGVCs. From this need came the development of the AGVC Performance Metrics Framework. By looking at established convoy requirements found in Army doctrine and performance metrics common amongst AGVC research, we created a framework that linked requirements and metrics. Using the framework gives researchers a tool to determine what metrics to choose based on what convoy requirements their effort is focused on. By focusing on metrics of comparison, we enable continuous improvement as further research efforts continue to expand on AGVC performance.

With the Behavior Manager and Performance Metrics Framework established, a variety of directions can be taken to pursue further efforts in measurably improving radio jammed convoy performance using behavior-based robotics. From the perspective of the Behavior Manager, additional behaviors and behavioral assemblages can be developed for added layers of mitigation. While our efforts focused on minimizing sensor and computing requirements, removing those constraints give considerable space to grow. Future efforts could focus on increasing the breadth of behaviors and assemblages, wherein a large number of behaviors are developed to take advantage of different sensing modalities to mitigate the effects of jamming.

Some examples include behaviors to leverage monocular vision to track leader vehicles; stereovision to obtain distance information; and downward facing cameras to follow tire tracks. Additional behaviors give the Behavior Manager a larger pool of information to look at for behavioral combination, allowing for jamming mitigation that is more robust to single sensor failure. On the other hand, if computing requirements are not a constraint, the depth of each behavior can be increased, in that more computationally complex behaviors can be utilized. As previously mentioned, Q-learning has been applied to behavior-based robotics in the past [103]. This can naturally be extended to use more advanced types of learning for behavioral weighting and combining, such as Deep Q-networks [121] or Double Q-learning [122], which are more computationally expensive. Additionally, the individual behaviors themselves would be able to utilize more complex techniques as well, such as convolutional neural networks for image recognition [87]. From a holistic perspective, further research on utilizing the Behavior Manager on a convoy system running anti-jamming techniques would provide valuable insight into the interaction between the layers of defense again radio jamming attacks. Furthermore, physical deployment of the Behavior Manager against a physical jammer would also be beneficial in future studies to determine how more complex jamming patterns change the efficacy of the jamming mitigation, and what could be done to improve the assemblage.

Based on commercial and military interest, AGVCs will have a growing impact on transportation and logistics as we move forward into the future. The safety and security of these systems will be paramount to their widespread adaptation and are therefore important areas of research. Our work in this dissertation, which established the Behavior Manager and Performance Metrics Framework, adds another layer of defense against radio jamming, allowing for more robust protection against an entire category of cyber-physical attacks. With the

Behavior Manager and Performance Comparison Framework in hand, we hope to enable further developments in jamming mitigation for AGVCs, and for safer and more robust systems overall.

Bibliography

[1] S. Nahavandi, S. Mohamed, I. Hossain, D. Nahavandi, S. M. Salaken, M. Rokonuzzaman, R. Ayoub and R. Smith, "Autonomous Convoying: A Survey on Current Research and Development," *IEEE Access,* vol. 10, p. 13663–13683, 2022.

[2] C. T. J. Murphy, Convoy Operations in Afghanistan Handbook, Fort Leavenworth, KS: Center for Army Lessons Learned, 2010.

[3] C. Cheung, A. Mohammadi, S. Rawashdeh and S. Baek, "Delivery of Healthcare Resources using Autonomous Ground Vehicle Convoy Systems: An Overview," *Frontiers in Robotics and AI,* 2021.

[4] R. M. Gandia, F. Antonialli, B. H. Cavazza, A. M. Neto, D. A. d. Lima, J. Y. Sugano, I. Nicolai and A. L. Zambalde, "Autonomous vehicles: scientometric and bibliometric review," *Transport Reviews,* vol. 39, no. 1, pp. 9-28, 2019.

[5] Dutch Ministry of Infrastructure and the Environment, "European Truck Platooning Challenge 2016," European Truck Platooning Challenge 2016, 2016.

[6] M. Waibel, "SARTRE: Autonomous Car Platoons," *IEEE Spectrum,* 31 January 2011.

[7] M. Moon, "Singapore hosts first full-scale autonomous truck platoon trial," 9 November 2022. [Online]. Available: https://www.engadget.com/2017-01-25-singapore-full-scale-autonomous-truck-platooning-trial.html.

[8] I. King, "Peloton Technology Raises Cash for Automated Truck Convoy Technology - Bloomberg," *Bloomberg,* 5 April 2017.

[9] J. Vincent, "Self-driving truck convoy completes its first major journey across Europe - The Verge," The Verge, 7 April 2016. [Online]. Available: https://www.theverge.com/2016/4/7/11383392/self-driving-truck-platooning-europe. [Accessed 1 January 2019].

[10] D. A. Green, "The Future of Autonomous Ground Logistics: Convoys in the Department of Defense," United States Army Command and General Staff College School of Advanced Military Studies, Fort Leavenworth, KS, 2011.

[11] S. McKay, M. W. Lewis, N. M. Beyene and M. E. Boyer, "Automating Army Convoys: Technical and Tactical Risks and Opportunities," RAND Corporation, 2020.

[12] G. Nardini, A. Virdis, C. Campolo, A. Molinaro and G. Stea, "Cellular-V2X Communications for Platooning: Design and Evaluation," *Sensors,* vol. 18, no. 5, pp. 1527-1549, 2018.

[13] D. Jia, K. Lu, J. Wang, X. Zhang and X. Shen, "A Survey on Platoon-Based Vehicular Cyber-Physical Systems," *IEEE Communication Surveys & Tutorials,* vol. 18, no. 1, pp. 263-284, 2016.

[14] S. Malebary, "Real-Time Jamming Detection in Vehicular Network," *International Journal of Scientific Research and Innovative Technology,* vol. 3, pp. 159-166, 2016.

[15] S. Sciancalepore and R. Di Pietro, "Bittransfer: Mitigating Reactive Jamming in Electronic Warfare Scenarios," *IEEE Access,* vol. 7, p. 156175–156190, 2019.

[16] R. van der Heijden, T. Lukaseder and F. Kargl, "Analyzing attacks on cooperative adaptive cruise control (CACC)," in *2017 IEEE Vehicular Networking Conference (VNC)*, Torino, Italy, 2017.

[17] C. Cheung, S. Rawashdeh and A. Mohammadi, "Jam Mitigation for Autonomous Convoys via Behavior-Based Robotics," *Applied Sciences,* vol. 12, no. 19, p. 9863, 2022.

[18] Center for Army Lessons Learned, Convoy Operations in Afghanistan Handbook, Fort Leavenworth, KS: U. S. Army Combined Arms Center , 2010.

[19] R. E. Killblane, Circle the Wagons: The History of US Army Convoy Security, Fort Leavenworth, KS: Combat Studies Institute Press, 2005.

[20] B. R. Layer, "Some Principles of Convoy Operations in Operations Other Than War," School of Advanced Military Studies United Staets Army Command and General Staff College, Fort Leavenworth, KS, 1993.

[21] MPRI Kuwait Observer Controller Team, Convoy Leader Training Handbook, Doha, Kuwait: MPRI KOCT, 2003.

[22] R. E. Killblane, Convoy Ambush Case Studies Volume I - Korea and Vietnam, Fort Lee, VA: US Army Transportation School, 2013.

[23] R. E. Killblane, Convoy Ambush Case Studies Volume II - Iraq and Afghanistan, Fort Lee, VA: US Army Transportation School, 2015.

[24] J. Davis, A. Animashaun and E. Schoenherr, "Evaluation of Semi-Autonomous Convoy

Driving," *Journal of Field Robotics,* pp. 880-897, 2008.

[25] S. Feng, Y. Zhang, S. E. Li, Z. Cao, H. X. Liu and L. Li, "String Stability for Vehicular Platoon Control: Definitions and Analysis Methods," *Annual Reviews in Control,* pp. 81-97, 2019.

[26] S. Eben Li, Y. Zheng, Y. Wu, J. K. Hedrick, F. Gao, K. Li and H. Zhang, "Dynamical Modeling and Distributed Control of Connected and Automated Vehicles: Challenges and Opportunities," *IEEE Intelligent Transportation Systems Magazine,* pp. 46-58, 2019.

[27] M. C. Harrell, L. Werber, P. Schirmer, B. W. Hallmark, J. Kavanagh, D. Gershwin and P. S. Steinberg, "Assessing the Assignment Policy for Army Women," RAND Corporation, Santa Monica, CA, 2007.

[28] M. Krumboltz, "It costs $2.1 million per year for each soldier deployed in Afghanistan: Report," 25 October 2013. [Online]. Available: https://www.yahoo.com/news/it-costs--2-1-million-per-year-for-each-soldier-deployed-in-afghanistan--report-133150602.html. [Accessed 2 3 2020].

[29] F. Browand, J. McArthur and C. Radovich, "Fuel Saving Achieved in the Field Test," California Partners for Advanced Transit and Highways, Los Angeles, CA, 2004.

[30] U.S. Army Training and Doctrine Command, "The U.S. Army in Multi-Domain Operations 2028," U.S. Army Training and Doctrine Command, Fort Eustis, VA, 2018.

[31] Maneuver, Aviation, and Soldier Division Army Capabilities Integration Center, "The U.S. Army Robotic and Autonomous Systems Strategy," U.S. Army Training and Doctrine Command, Fort Eustis, VA, 2017.

[32] North Atlantic Treaty Organization Standardization Office, "AAP-06," North Atlantic Treaty Organization Standarization Office, Brussels, Belgium, 2019.

[33] U.S. Army Training and Doctrine Command, "ADP 1-01 Doctrine Primer," Department of the Army, Fort Belvoir, VA, 2019.

[34] Army Publishing Directorate, "https://armypubs.army.mil/," [Online]. Available: https://armypubs.army.mil/. [Accessed 16 March 2020].

[35] Office of the Under Secretary of Defense for Acquisition and Sustainment, *Department of Defense Instruction 5230.24 Distribution Statements on Technical Documents,* Washington, DC: Department of Defense, 2012.

[36] United States Army Combined Arms Support Command, *ATP 4-11 Army Motor Transport Operations,* Fort Lee, VA: Department of the Army, 2013.

[37] U.S. Army Training and Doctrine Command, *STP 55-88M14-SM-TG Soldier's Manual and Training Guide,* Arlington County, VA: United States Department of the Army, 2013.

[38] U.S. Army Training and Doctrine Command, *TC 21-305-20/AFMAN 24-306(I) Manual for the Wheeled Vehicle Operator,* Arlington County, VA: United States Department of the Army, 2016.

[39] T. N. Brooks, "Sustainment Battalion Convoy Protection Platform Gunnery," *Army Sustainment,* pp. 33-37, September 2012.

[40] J. B. Frembling, "Training and Certification for Convoy Escort Teams," *Army Sustainment,* pp. 41-43, September 2012.

[41] Air Land Sea Application Center, "Air Land See Application Center," [Online]. Available: https://www.alsa.mil/mttps/tco/. [Accessed 1 August 2020].

[42] G. Rödönyi, P. Gáspár, J. Bokor and L. Palkovics, "Experimental Verification of Robustness in a Semi-autonomous Heavy Vehicle Platoon," *Control Engineering Practice,* pp. 13-25, 2014.

[43] D. Swaroop, J. Hedrick, C. C. Chien and P. Ioannou, "A Comparision of Spacing and Headway Control Laws for Automatically Controlled Vehicles," *Vehicle System Dynamics,* pp. 597-625, 1994.

[44] A. Ali, G. Garcia and P. Martinet, "The Flatbed Platoon Towing Model for Safe and Dense Platooning on Highways," *IEEE Intelligent Transportation Systems Magazine,* pp. 58-68, 2015.

[45] G. Klančar, D. Matko and S. Blažič, "A Control Strategy for Platoons of Differential Drive Wheeled Mobile Robot," *Robotics and Autonomous Systems,* pp. 57-64, 2011.

[46] D. Lu, Z. Li and D. Huang, "Platooning as a Service of Autonomous Vehicles," in *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Maccau, 2017.

[47] G. Guo and W. Yue, "Autonomous Platoon Control Allowing Range-Limited Sensors," *IEEE Transactions on Vehicular Technology,* pp. 2901-2912, 2012.

[48] Y. Li and C. He, "Connected Autonomous Vehicle Platoon Control Considering Vehicle Dynamic Information," in *2018 37th Chinese Control Conference (CCC)*, Wuhan, 2018.

[49] Y. Liu, H. Gao, B. Xu, G. Liu and H. Cheng, "Autonomous Coordinated Control of a Platoon of Vehicles with Multiple Disturbances," *IET Control Theory & Applications,* pp. 2325-2335, 2014.

[50] S. Gong, J. Shen and L. Du, "Constrained Optimization and Distributed Computation Based Car Following Control of a Connected and Autonomous Vehicle," *Transportation Research Part B: Methodological,* pp. 314-334, 2016.

[51] S. Öncü, "String Stability of Interconnected Vehicles: Network-aware Modelling, Analysis and Experiments," Technische Universiteit Eindhoven, Eindhoven, 2013.

[52] X. Zhao, W. Yao, N. Li and Y. Wang, "Design of Leader's Path Following System for Multi-Vehicle Autonomous Convoy," in *2017 IEEE International Conference on Unmanned Systems (ICUS)*, Beijing, 2017.

[53] G. Klančar, D. Matko and S. Blažič, "Wheeled Mobile Robots Control in a Linear Platoon," *Journal of Intelligent and Robotic Systems,* p. 709–731, 2009.

[54] H. K. Goi, J. L. Giesbrecht, T. D. Barfoot and B. A. Francis, "Vision-Based Autonomous Convoying with Constant Time Delay," *Journal of Field Robotics,* pp. 430-449, 2010.

[55] X.-R. Li and Z. Zhao, "Measures of performance for evaluation of estimators and filters," in *Proceedings Volume 4473, Signal and Data Processing of Small Targets 2001*, San Diego, CA, 2001.

[56] C. J. Needham and R. D. Boyle, "Performance Evaluation Metrics and Statistics for Positional Tracker Evaluation," in *International Conference on Computer Vision Systems*, Graz, 2003.

[57] Quantum Signal, "ANVEL (Autonomous Navigation Virtual Environment Laboratory) from Quantum Signal LLC," 18 July 2018. [Online]. Available: https://www.youtube.com/watch?v=nd1P4KpBmTk.

[58] O. Amidi and C. E. Thorpe, "Integrated Mobile Robot Control," *Mobile Robots V,* 1991.

[59] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffman, K. Lau, C. Oakley, M. Palatucci, V. Pratt and P. Stang, "Stanley: The Robot That Won the DARPA Grand Challenge," *Journal of Field Robotics,* pp. 661-692, 2006.

[60] M. Buehler, K. Iagnemma and S. Singh, The 2005 DARPA Grand Challenge: The Great Robot Race, Springer, 2007.

[61] J. M. Snider, "Automatic Steering Methods for Autonomous Automobile Path Tracking," Robotics Institute, Pittsburgh, PA, 2009.

[62] D. Kapsalis, O. Sename, V. Milanes, J. Jairo and M. Molina, "Design and Experimental Validation of an LPV Pure Pursuit Automatic Steering Controller," in *16th IFAC*

Symposium on Control in Transportation Systems CTS 2021, Lille, France, 2012.

[63] N. H. Amer , K. Hudha, H. Zamzuri, V. R. Aparow, A. F. Z. Abidin, Z. A. Kadir and M. Murrad, "Adaptive modified Stanley controller with fuzzy supervisory system for trajectory tracking of an autonomous armoured vehicle," *Robotics and Autonomous Systems,* pp. 94-111, 2018.

[64] I. Llatser, A. Festag and G. Fettweis, "Vehicular communication performance in convoys of automated vehicles," in *2016 IEEE International Conference on Communications (ICC)*, 2016, 2016.

[65] H. J. M. Lopes and L. A. Danielli, "Cellular Automata in path planning navigation control applied in surveillance task using the e-Puck architecture," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Toronto, ON, Canada, 2020.

[66] W. Xu, W. Trappe, Y. Zhang and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, Urbana-Champaign. IL, 2005.

[67] H. Alturkostani, A. Chitrakar, R. Rinker and A. Robert, "On the Design of Jamming-Aware Safety Applications in VANETs," in *CISR '15: Proceedings of the 10th Annual Cyber and Information Security Research Conference*, Oak Ridge, TN, 2015.

[68] A. Serageldin and A. Krings, "The Impact of Dissimilarity and Redundancy on the Reliability of DSRC Safety Applications," in *2014 28th International Conference on Advanced Information Networking and Applications Workshops*, Victoria, BC, Canada, 2014.

[69] A. Serageldin, H. Alturkostani and A. Krings, "On the reliability of DSRC safety applications: A case of jamming," in *2013 International Conference on Connected Vehicles and Expo (ICCVE)*, Las Vegas, Nevada, 2013.

[70] Y. Hu, H. Shan, R. G. Dutta and Y. Jin, "Protecting Platoons from Stealthy Jamming Attack," in *2020 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, Kolkata, India, 2020.

[71] T. Gong, Y. Yang and J. Song, "Path Planning for Multiple Unmanned Vehicles (MUVs) Formation Shape Generation Based on Dual RRT Optimization," *Actuators,* vol. 11, no. 7, 2022.

[72] L. Yunhe, L. Bo and N. Xiaowei, "Research on Self-Organizing Behavior-Based UAV Formation Based on Distributed Control," in *ICVISP 2019: Proceedings of the 3rd International Conference on Vision, Image and Signal Processing*, Vancouver BC Canada, 2019.

[73] H. B. Jond, J. Platoš and Z. Sadreddini, "Autonomous Vehicle Convoy Formation Control with Size/Shape Switching for Automated Highways," *IJE TRANSACTIONS B: Applications,* vol. 33, no. 11, pp. 2174-2180, 2020.

[74] T. Balch and R. Arkin, "Behavior-Based Formation Control for Multirobot Teams," *IEEE Transactions on Robotics and Automation,* vol. 14, no. 6, pp. 926-939, 1998.

[75] G. Lee and D. Chwa, "Decentralized behavior-based formation control of multiple robots considering obstacle avoidance," *Intelligent Service Robotics,* pp. 127-138, 2018.

[76] P. Pierpaoli, A. Li, M. Srinivasan, X. Cai, S. Coogan and M. Egerstedt, "A Sequential Composition Framework for Coordinating Multirobot Behaviors," *IEEE Transactions on Robotics,* pp. 1-13, 2020.

[77] Y. Zheng, S. E. Li, J. Wang, D. Cao and K. Li, "Stability and Scalability of Homogeneous Vehicular Platoon: Study on the Influence of Information Flow Topologies," *IEEE Transactions on Intelligent Transportation Systems,* vol. 17, no. 1, pp. 14-26, 2016.

[78] K. Pelechrinis, M. Iliofotou and S. V. Krishnamurthy, "Denial of Service Attacks in Wireless Networks: The Case of Jammers," *IEEE Communications Surveys & Tutorials,* vol. 13, no. 2, pp. 245 - 257, 2011.

[79] B. D. Van Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE ASSP Magazine,* vol. 5, no. 2, pp. 4-24, 1988.

[80] G. Y. Luo, "On-line wavelet filtering of narrowband noise in signal detection of spread spectrum system for location tracking," *International Journal of Communication Systems,* vol. 25, no. 5, pp. 598-615, 2012.

[81] C. Popper, M. Strasser and S. Capkun, "Anti-jamming broadcast communication using uncoordinated spread spectrum techniques," *IEEE Journal on Selected Areas in Communications,* vol. 28, no. 5, pp. 703 - 715, 2010.

[82] R. Poisel, Modern communications jamming principles and techniques, second edition, Boston, MA: Artech House, 2011.

[83] Y. Liu and P. Ning, "BitTrickle: Defending against broadband and high-power reactive jamming attacks," in *2012 Proceedings IEEE INFOCOM*, Orlando, FL, 2012.

[84] Y. Xuan, Y. Shen, N. P. Nguyen and M. T. Thai, "A Trigger Identification Service for Defending Reactive Jammers in WSN," *IEEE Transactions on Mobile Computing,* vol. 11, no. 5, pp. 793 - 806, 2012.

[85] G. Noubir, "On Connectivity in Ad Hoc Networks under Jamming Using Directional Antennas and Mobility," in *International Conference on Wired/Wireless Internet*

Communications, Frankfurt, Germany, 2004.

[86] R. Brooks, "A Robust Layered Control System for a Mobile Robot," *IEEE Journal of Robotics and Automation,* vol. 2, no. 1, pp. 14-23, 1986.

[87] B. Siciliano and O. Khatib, Eds., Springer Handbook of Robotics, Springer Cham, 2016.

[88] R. C. Arkin, Behavior-Based Robotics, Cambridge, MA: MIT Press, 1998.

[89] R. C. Arkin, "Motor Schema - Based Mobile Robot Navigation," in *IEEE International Conference on Robotics and Automation*, Raleigh, North Carolina, 1987.

[90] L. P. Kaelbling, "An Architecture for Intelligent Reactive Systems," *Reasoning About Actions & Plans,* p. 395–410, 1987.

[91] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, Kobe, Japan, 2009.

[92] N. P. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, 2004.

[93] Clearpath Robotics, "Robots/Husky - ROS," [Online]. Available: http://wiki.ros.org/Robots/Husky.

[94] Clearpath Robotics, "Husky/husky: Common packages for the clearpath husky," [Online]. Available: https://github.com/husky/husky.

[95] A. Soni and H. Hu, "Formation Control for a Fleet of Autonomous Ground Vehicles: A Survey," *Robotics,* vol. 4, no. 67, 2018.

[96] D. V. Lu, D. Hershberger and W. D. Smart, "Layered costmaps for context-sensitive navigation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Chicago, IL, USA, 2014.

[97] M. Ester, H.-P. Kriegel, J. Sander and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *KDD'96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, Portland, OR, 1996.

[98] S. Macenski, "costmap_2d/hydro/inflation," [Online]. Available: http://wiki.ros.org/costmap_2d/hydro/inflation.

[99] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proceedings. 1991 IEEE International Conference on*

Robotics and Automation, Sacramento, CA, 1991.

[100] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation,* pp. 278 - 288, 1991.

[101] L. Vasseur, O. Lecointe, J. Dento, N. Cherfaoui, V. Marion and J. G. Morillon, "Leader-follower function for autonomous military convoys," in *Proceedings of SPIE*, Orlando, Florida, 2004.

[102] A. Chowdhury, G. Karmakar, J. Kamruzzaman, A. Jolfaei and R. Das, "Attacks on Self-Driving Cars and Their Countermeasures: A Survey," *IEEE Access,* vol. 8, pp. 207308-207342, 2020.

[103] E. Martinson, A. Stoytchev and R. C. Arkin, "Robot behavioral selection using Q-learning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, 2002.

[104] J. B. Rawlings, "Understanding Model Predictive Control," *Tutorial overview of model predictive control,* vol. 20, no. 3, pp. 38-52, 2000.

[105] J. Wang, S. Gong, S. Peeta and L. Lu, "A real-time deployable model predictive control-based cooperative platooning approach for connected and autonomous vehicles," *Transportation Research Part B: Methodological,* vol. 128, pp. 271-301, 2019.

[106] B. HomChaudhuri, A. Vahidi and P. Pisu, "A fuel economic model predictive control strategy for a group of connected vehicles in urban roads," in *2015 American Control Conference (ACC)*, Chicago, IL, 2015.

[107] T. Ard, B. Pattel, A. Vahidi and H. Borhan, "Considerate and Cooperative Model Predictive Control for Energy-Efficient Truck Platooning of Heterogeneous Fleets," in *2022 American Control Conference (ACC)*, Atlanta, GA, 2022.

[108] X. Qian, A. de La Fortelle and F. Moutarde, "A Hierarchical Model Predictive Control Framework for On-road Formation Control of Autonomous Vehicles," in *2016 IEEE Intelligent Vehicles Symposium*, Gothenburg, Sweden, 2016.

[109] Z. Li, W. Yuan, Y. Chen, F. Ke, X. Chu and C. L. P. Chen, "Neural-Dynamic Optimization-Based Model Predictive Control for Tracking and Formation of Nonholonomic Multirobot Systems," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 29, no. 12, pp. 6113 - 6122, 2018.

[110] Z. Huang, D. Chu, W. Chaozhong and Y. He, "Path Planning and Cooperative Control for Automated Vehicle Platoon Using Hybrid Automata," *IEEE Transactions on Intelligent Transportation Systems,* vol. 20, no. 3, pp. 959 - 974, 2019.

[111] S. Di Cairano, "An industry perspective on MPC in large volumes applications: Potential Benefits and Open Challenges," *IFAC Proceedings Volumes,* vol. 45, no. 17, pp. 52-59, 2012.

[112] P. Ogren, A. Backlund, T. Harryson, L. Kristensson and P. Stensson, "Autonomous UCAV Strike Missions using Behavior Control Lyapunov Functions," in *AIAA Guidance, Navigation, and Control Conference Proceedings*, Keystone, CO, 2006.

[113] M. Meza-Sánchez, E. Clemente, M. Rodríguez-Liñán and G. Olague, "Synthetic-analytic behavior-based control framework: Constraining velocity in tracking for nonholonomic wheeled mobile robots," *Information Sciences,* vol. 501, pp. 436-459, 2019.

[114] G. Droge, P. Kingston and M. Egerstedt, "Behavior-based switch-time MPC for mobile robots," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura-Algarve, Portugal, 2012.

[115] G. Droge, "Distributed virtual leader moving formation control using behavior-based MPC," in *2015 American Control Conference (ACC)*, Chicago, IL, 2015.

[116] R. Jacobson and G. Droge, "Approximate Filippov-based Switching for Behavior-based Trajectory Tracking," in *2021 American Control Conference (ACC)*, New Orleans, LA, 2021.

[117] Y. Zheng, M. J. Brudnak, J. Paramsothy , J. L. Stein and T. Ersal, "A Delay Compensation Framework for Predicting Heading in Teleoperated Ground Vehicles," *IEEE/ASME Transactions on Mechatronics,* vol. 24, no. 5, pp. 2365 - 2376, 2019.

[118] Y. Zheng, M. J. Brudnak, J. Paramsothy, J. L. Stein and T. Ersal, "Evaluation of a Predictor-Based Framework in High-Speed Teleoperated Military UGVs," *IEEE Transactions on Human-Machine Systems,* vol. 50, no. 6, pp. 561 - 572, 2020.

[119] A. Bemporad, "Predictive control of teleoperated constrained systems with unbounded communication delays," in *Proceedings of the 37th IEEE Conference on Decision and Control*, Tampa, FL, 1998.

[120] Y. Zheng, M. J. Brudnak, J. Paramsothy, J. L. Stein and T. Ersal, "A Predictor-Based Framework for Delay Compensation in Networked Closed-Loop Systems," *IEEE/ASME Transactions on Mechatronics,* vol. 23, no. 5, pp. 2482 - 2493, 2018.

[121] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, G. M. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, O. Georg, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg and D. Hassabis, "Human-level control through deep reinforcement learning," *nature,* vol. 518, no. 7540, pp. 529-533, 2015.

[122] H. van Hasselt, A. Guez and D. Silver, "Deep Reinforcement Learning with Double Q-learning," *Proceedings of the AAAI conference on artificial intelligence,* vol. 30, no. 1, 2016.