MOTION PLANNING FOR AUTONOMOUS GRAIN CARTS

A Dissertation

by

LANTIAN SHANGGUAN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	J. Alex Thomasson
Co-Chair of Committee,	Swaminathan Gopalswamy
Committee Members,	Stephen W. Searcy
	Sivakumar Rathinam
Head of Department,	Stephen W. Searcy

May 2020

Major Subject: Biological and Agricultural Engineering

ABSTRACT

In crop harvesting, a combine travels around in the field to collect grain while a grain cart commutes between the combine and a semi-trailer by the roadside to transport the grain. There are several problems associated with human-operated grain carts: labor shortage and increasing labor cost, operational imprecision and inefficiency as well as safety hazards. All of these problems can potentially be addressed if grain carts were autonomous. To facilitate full autonomy of grain carts, this study develops a motion planning algorithm, featuring a novel integration of Artificial Potential Field (APF) with Fuzzy Logic Control (FLC). In addition, this study proposes a high-level software and hardware solution to building the navigation systems for implementing the developed motion planning algorithm on autonomous grain carts, covering sensor selection, communication options, control technique and actuation plan.

A set of simulation tests featuring the comparison between the proposed APF+FLC planner and a simple APF planner were carried out in MatLab Simulink. The simulation tests demonstrated that the proposed motion planning algorithm and the associated task scheduling strategy could promptly direct an autonomous grain cart to intelligently perform the logistical tasks in harvesting operations where unharvested crops were the only obstacles as well as when random static or dynamic obstacles existed, outperforming the simple APF planner in trajectory length and smoothness by roughly 15% to 20%. In addition, another set of simulation tests comparing the proposed APF+FLC planner with a Vector- Field-Histogram (VFH) planner were conducted to

further evaluate the performance of the proposed algorithm. It was shown that although the VFH planner tended to plan smoother paths, the APF+FLC planner was superior in terms of generating shorter paths with less computational cost (shorter and less both by as much as 60%). Results of the two sets of simulation tests verified the effectiveness, robustness, efficiency and computational ease of the proposed motion planning algorithm. Following the simulation tests, a set of mobile robot tests implementing the proposed navigation solution were conducted, in which the proposed algorithm was effective in directing the grain cart to intelligently accomplish the logistical tasks in harvest operations. Additionally, the mobile robot tests included a variety of more general obstacle avoidance cases, in which the proposed algorithm was always effective in leading the robot to efficiently accomplish the navigation tasks, outperforming a simple APF planner in trajectory length by as much as 25% and in smoothness by as much as three times. The mobile robot tests verified the effectiveness and practicality of the proposed navigation solution as well as the effectiveness, robustness, and especially efficiency of the proposed motion planning algorithm.

DEDICATION

To my father

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by a dissertation committee consisting of Professors J. Alex Thomasson [advisor] and Stephen W. Searcy of the Department of Biological and Agricultural Engineering and Professors Swaminathan Gopalswamy [co-advisor] and Sivakumar Rathinam of the Department of Mechanical Engineering.

The conduct of this research work as well as the organization and writing of the dissertation was closely supervised and extensively instructed by Professor J. Alex Thomasson. A trip to a farm for learning about and experiencing a real wheat harvest involving the cooperation between a combine and a grain cart was arranged by Professor J. Alex Thomasson. The field data adopted for the simulation tests and the robots employed for the mobile robot tests were provided by Professor J. Alex Thomasson. The space for the mobile robot tests was made available by Professor J. Alex Thomasson and Mr. Richard Epting. Professor J. Alex Thomasson also sponsored the student for attending the 2019 ASABE Annual International Meeting to communicate with and learn from other researchers in related research areas. The 2D lidar, tracking cameras, power banks, mini PC and one of the laptops for the mobile robot tests were provided by Professor Swaminathan Gopalswamy. Determination of the scope of the study and design of the mobile robot tests were instructed by Professor Stephen W. Searcy at the early stage of the study. Professor Stephen W. Searcy also provided in-depth instructions on writing the dissertation. A great amount of knowledge, insight, skill and experience

on autonomous vehicle technologies was gained through the student's experience of working with Professors Swaminathan Gopalswamy, Sivakumar Rathinam as well as P. R. Kumar from the Department of Electrical and Computer Engineering on multiple research projects. The conduct of the mobile robot tests was supported by Aditya Narayanan and Hussein Gharakhani. The following postdoctoral researcher and students also provided constructive ideas/advice during the conduct of this study: Dr. Xiongzhe Han, Tianyi Wang, Sai Vemprala, Akshay Sarvesh, Kenny Chour, David Franklin, Tianqi Li, Mengke Liu and Yutong Li.

All other work conducted for the dissertation was completed by the student independently.

Funding Sources

Graduate study was supported by a four-year scholarship from China Scholarship Council, two semesters of Graduate Assistantship Teaching from the Department of Mechanical Engineering and four years of Graduate Student Scholarship from the Department of Biological and Agricultural Engineering.

This work was also made possible in part by (i) Tama Plastic Industry and (ii) Deere and Company under Project "Fate and Transport Study On Cotton", (iii) Fiber Initiative (State of Texas) under Project "Detection and Removal of Plastic Contaminants", (iv) USDA Agricultural Research Service under Project "Detection and Removal of Plastic Contamination of Cotton and Development of Cotton Gin Process Models" with Grant Number CFDA # 10.001, (v) the U.S. Army Research Laboratory under Contract Number W911NF-19-2-0033, (vi) the U.S. Army Research Office under

Contract Number W911NF-18-10331, (vii) the U.S. ONR under Contract Number N00014-18-1-2048, (viii) NSF Science & Technology Center Grant CCF-0939370, and (ix) Denso Corporation (Japan) under Contract Number 406770-00001. The contents of the dissertation are solely the responsibility of the author and do not represent the official views of the above awarding offices.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

Page

Page

Page

Page

Page

Page

Page

LIST OF TABLES

Page

1. INTRODUCTION

1.1. Background

1.1.1. Autonomous Agricultural Vehicles

1.1.1.1. Motivations

With global population predicted to exceed nine billion by 2050, the demand for food and other agricultural products is expected to increase by 50% compared with that in 2012 [1]. Meanwhile, this production should ideally consume fewer resources, including arable farmland, labor, water and energy as their availability is limited. The concept of producing more with less provides a catalyst for development and deployment of autonomous agricultural vehicles (AAVs) [1]. AAVs have the potential to increase agricultural productivity by addressing the issues associated with human operated machines, including labor shortages, increasing labor costs, operational imprecision and inefficiency as well as safety hazards. Below is a more detailed description of the benefits of AAVs.

First, finding and keeping skilled farm laborers is becoming a greater challenge, and the labor costs are increasing. From 2002 to 2014, the number of available field and crop workers in the U.S. dropped by more than 20% (i.e., at least 146,000 people) while their wages increased by 12% [2]. Over a longer span, the real (i.e., inflation-adjusted) wage for nonsupervisory farm workers in the U.S. increased by approximately 30% from 1989 to 2017 [3]. Even with higher wages, farm work is still less attractive than the expanding number of well-paid jobs in other industries that are less physically intensive.

The average age of agricultural producers in the U.S. has increased from 53.2 in 2002 to 57.5 in 2017 [4]. When the older farmers exit, there will be fewer producers as not many young farmers are entering. The labor shortage, cost growth and farmers' aging all contribute to the dearth of inexpensive operators that can reliably and efficiently operate the large and technologically-advanced farming machines in use today. However, if the agricultural vehicles were autonomous, they could potentially operate without an operator in the cab, addressing the labor issues. Theoretically, AAVs could operate all day every day. Unlike human workers, AAVs would not need rest to maintain high performance as long as fuel/energy was resupplied, which would be a particularly valuable advantage during busy seasons like planting and harvesting. With AAVs, farm workers with computer skills could scan multiple monitors remotely in a control room or in the field to supervise and/or control the farming operations, running large farms more efficiently [5].

Second, if equipped with a system of sophisticated sensors, well-designed controllers and high-precision actuators, AAVs could potentially "see", "think" and "act" with high precision and efficiency. With the help of navigation technology, AAVs could self-locate with only centimeter-level errors. High-accuracy localization can not only ensure all crop rows are straight and precisely plowed or planted, but also prevent the frequent problem of either covering the same area twice or missing patches, improving the uniformity and effectiveness of activities like spraying and planting. With optimized path planning, nonoverlapping coverage saves both time and fuel (by as much as 40% in some cases) [6]. If coupled with other technologies such as Geographic

Information System (GIS), AAVs could apply just the right amount of water, fertilizer, pesticides, etc., using resources appropriately and responsibly [7]. In addition, AAVs could support real-time communications with farmers and other machines, which would promote more effective coordination and more efficient cooperation. In fact, it has been estimated that advances in agricultural technology, primarily ground and air AAVs that would facilitate precision farming and help farmers make more informed decisions, could lead to a more than 70% rise in farm yields globally by the year 2050 [8].

Last but not least, farming is considered one of the most hazardous occupations in the U.S. With transportation incidents (including tractor overturns) being the leading cause, there were 417 farm work-related deaths in 2016, resulting in a fatality rate of 21.4 per 100,000 workers [9]. While humans have inherent limitations (e.g., reaction time, vision, fatigue, potential impairments such as alcohol) and could choose to operate improperly (e.g., take a shortcut or ignore a warning), AAVs have the potential to enhance operational safety of farming machines as they could be equipped with blind-spot-free sensors and responsive actuators and, most importantly, they could be programmed to never take risky actions.

Providing nonstop operations with reduced labor and improved precision, efficiency and safety, AAVs could potentially bring about higher productivity and great economic benefits to agriculture.

1.1.1.2. Status

Tractors equipped with automated steering systems that can follow pre-programmed routes are already being utilized on large farms around the world [10],

while agricultural robots are beginning to be deployed to automate various agricultural processes, including harvesting, fruit picking, ploughing, soil maintenance, weeding, planting and irrigation [11]. GPS guidance, auto-steering systems and GIS software, along with many other technologies, are being employed to make farming machines operate with less human involvement. It is estimated that guidance systems are now supporting 60% to 70% of the crop harvest in North America and more than 90% in Australia [12]. Given this technological foundation, leveraging commodity components and advanced software, the transition to full autonomy in agriculture may happen faster than it is happening in other domains [13]. That being said, most semi-autonomous and autonomous farming machines are still under development and/or testing. Some researchers [5] believe the automated technology in agriculture is now in the second of four stages: (i) automation of individual functions and (ii) auto-steering under operator control have been achieved, but (iii) unguided operation and (iv) fully autonomous farming are yet to be mastered. Although systems like auto steering have been very successful in terms of increasing productivity, operators are still required to be behind the wheel to handle complex situations, e.g., trash buildup on an implement or a mechanical failure that cannot be handled by an automated system [14]. In fact, it is one of the biggest challenges in AAV technology to develop systems that can promptly identify and properly react to complex and unpredictable situations in farming operations.

AAV technology, though still at an early stage, has a large market that is rapidly increasing [15]: valued at USD 55 billion in 2016, the autonomous farm equipment

market is predicted to surpass USD 180 billion by 2024. The market growth is primarily driven by the wide use of tractors and harvesters, the labor shortage and demand for higher productivity; meanwhile, the market growth is supported by the implementation of Internet-of-Things (IoT) and precision farming technologies like autosteering. While tractors are anticipated to contribute over 50% of the revenue share by 2024, harvesters and UAVs are also predicted to exhibit significant growth in the market share. Regionally, North America dominates the global AAV market mainly due to the strong presence of well-established industry players. Europe is estimated to grow by about 20% by 2024, growth that is propelled by the use of products that offer better fuel efficiency and higher productivity. Asia Pacific, led by China, is anticipated to grow by over 25% by 2024. Case IH, Agribotix, Agrobot, AGCO, John Deere, Harvest Automation, Autonomous Tractor Corporation, CLAAS and Mahindra are among the prominent manufacturers and service providers in the AAV arena.

1.1.1.3. Challenges

1.1.1.3.1. Human Factor

Handing over the steering wheel is a major difficulty for farmers in the transitioning through the stages of autonomy in agriculture. It may take them years to feel comfortable stepping out of the cab and letting the technology take control. Additionally, farmers' ability to interact with these advanced yet complex computational systems is limited and needs improvement [14]. Operators need to be taught how to work with highly automated systems, just like pilots need to be trained for flying airplanes. Requirements for such training may slow down the penetration of AAVs.

1.1.1.3.2. Regulations

The new automated technology brings about unpredictability of potential legal changes. For most farm activities, including tillage and harvesting, no current law would apply restrictions on an AAV when it is performing farming tasks out in the field. Besides, having a farmer remotely monitor the machine in a control room may or may not qualify as working under a licensed or certified applicator [16]. Moreover, regulation and licensing for AAVs will probably not be available until the liability issue has been addressed. Some researchers believe that wide adoption of AAVs will depend on new laws that determine the liability in accidents involving AAVs. AAVs may sometimes be required to travel on the highways between fields due to the widely dispersed nature of the blocks of land farmed by many individuals; an AAV may also deviate from the planned track and accidentally gets onto the road nearby due to a malfunction of the control software [14]. In either case, if the AAV hits a car, is it the liability on the owner of the machine, the manufacturer that developed the software, or some other entity?

1.1.1.3.3. Reliability

While AAVs have the potential to dramatically change and benefit agriculture, it is a great challenge to design AAVs that can perform safely in every situation of every farming task [14]. Any malfunction of hardware or software could result in performance degradation or even a severe accident. Hence it is essential for the equipment manufacturers to ensure that AAVs and the software controlling them are reliable. Software companies are exploiting various techniques, such as formal code inspection,

unit testing, regression testing, and large-scale simulation to validate their software before real-world applications [13].

1.1.1.3.4. Technical Issues

First, drive trains present a problem. Traditional drive train vendors have generally not provided the key precondition for autonomous driving functions, drive-by-wire. Therefore, some auto tech companies have turned to electric drive trains, as Google and Tesla have for their autonomous cars [5]. A traditional drive train is based on cables, mechanical transfer of forces, hydraulic pressure and other ways to provide direct physical control, while drive-by-wire consists of servomotors or electromechanical actuators that are controlled through electric wires. Therefore, in the case of retrofitting an old farming machine into self-driving equipment, the big difference between traditional mechanical connections and drive-by-wire mechanism could be an issue.

Second, sensing presents a problem. Many driverless cars use cameras, radar, and lidar to sense the surrounding environment. Yet in agricultural fields, fewer recognizable objects or structures exist to provide obvious cues for decisions. Besides, compared to laboratories, agricultural environments have more uncertainties. Algorithms for computer vision and radar or lidar sensing are commonly developed on highly structured data, like a camera focusing on stationary, readily identifiable objects in a building. However, in the field, leaves fly around, lighting conditions vary, dust and dirt cover camera lenses, which can all affect sensing algorithms. Thus, algorithms designed for and tested in a laboratory environment may not work as well in a much more

unstructured farming environment [5]. There are some navigation systems that are exclusively reliant on GPS [17][18], but it cannot be ignored that GPS signals tend to be affected by sunspots, reflections from objects, tree coverage and ionospheric interference [16]. To increase the accuracy of conventional GPS navigation systems, additional technologies such as Differential GPS (DGPS) and Real-Time Kinematic GPS (RTK-GPS) have been employed; yet both require a base station to be located within the receiving range of the rover receiver mounted on the AAV [19].

Last, field surveys present a problem. Although some studies have looked at autonomous navigation based on ad-hoc localization and online motion planning, a large number of navigation solutions [20] rely on a priori knowledge of the field for offline path planning before motion, which requires a high-precision field survey in advance. This may require a great amount of work as the survey results need to include the accurate shape and coordinates of the field boundaries, longitude and latitude of the rows as well as the division between headlands and arable lands.

1.1.2. Autonomous Grain Carts

1.1.2.1. Overview

The combine harvester is the principal machine for harvesting grain crops. As shown in Figure 1.1, crops are first gathered in by the header at the front, then travel up the grain conveyor to feed into the threshing cylinder, following which the separating cylinder separates the grain from the straw. The grain, after being sieved, will be temporarily stored in the grain tank while the waste straw passes along the straw walkers

towards the back of the machine, where a straw spreader throws the straw over a wide area of ground behind the combine.

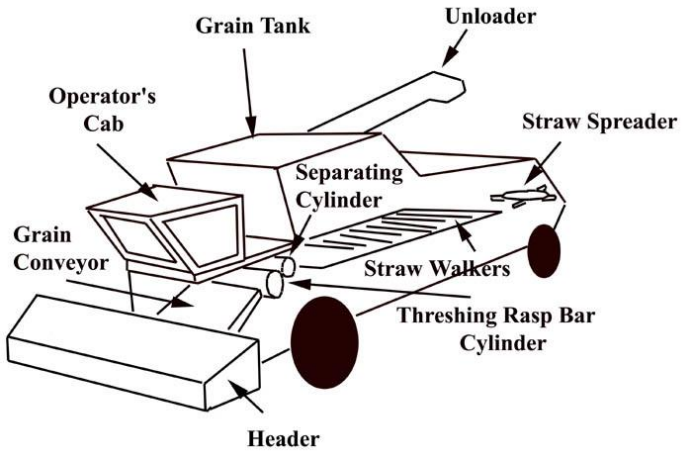


Figure 1.1 Combine harvester. Reprinted from [21].

Cooperating with the combine, a grain cart is a trailer towed by a tractor, and the term "grain cart" is used herein to encompass the system including both the tractor and the trailer, whose large capacity allows it to service single or, although not discussed in this work, multiple combines [22][23] in harvest operations. When harvesting crops such as wheat, soy and corn, a combine follows a specific route to cover the field while a grain cart performs a series of supporting logistical tasks. When the combine's tank fills up, the grain cart will approach and drive alongside the combine to unload the grain without interrupting the harvesting (Figure 1.2). Then the grain cart will travel to an in-field storage station, which is typically a semi-trailer parked by the roadside for later road transport. After transferring the grain to the semi-trailer, the grain cart will either stand by or go to the combine for the next work cycle.

Figure 1.2 Grain cart unloading grain from combine. Reprinted from [24].

If the grain cart is autonomous, it can perform all the above logistical tasks without an operator, which would solve the aforementioned problems associated with human-operated machines; i.e., labor shortage and increasing labor cost, operational imprecision and inefficiency as well as safety hazards. To develop an autonomous grain cart that can intelligently navigate in the field and efficiently cooperate with the combine, the following challenges need to be addressed.

1.1.2.2. Challenges

1.1.2.2.1. Temporal Constraints

The objective in harvest is to maximize crop quantity and quality while minimizing the inputs of fuel, time and labor. Doing so requires the combine to operate nearly nonstop throughout the harvest, requiring the grain cart to meet the combine when

the combine reaches full capacity. These temporal constraints mean that the grain cart operator must carefully schedule and sequence the operations. Otherwise, the combine may fill up and be forced to stop before the grain cart arrives, interrupting the harvesting; or the grain cart may arrive too early and have to follow the combine before unloading starts, resulting in extra fuel consumption and nonproductive time [25]. This important meeting timing is currently determined by the combine's operator based on experience, which can be inconsistent and often suboptimal.

1.1.2.2.2. Spatial Constraints

The cooperation of the combine and grain cart must account for the dynamic nature of the harvesting environment. As the combine travels along the rows cutting the crop plants, harvested areas that are traversable for the grain cart are constantly changing. Routes planned in the last work cycle could have become suboptimal and need updating. Additionally, when the combine needs to be unloaded, only the track to the left of the combine's path can be used by the grain cart, as almost all combines have the unloading auger on the left [26].

1.1.2.2.3. Real-time Adaptation

Besides the two major challenges above, unexpected obstacles and/or events may occur during harvesting, such as the varying yield of different field areas that can change the combine's speed and fill rate, or another vehicle's passing through the field and interfering with the harvest. Thus, motion planning for grain carts is inherently a dynamic problem that needs real-time adaptation, raising the requirement for computational efficiency [27].

1.1.2.2.4. Numerous Parameters

The cooperation of the combine and the grain cart is made particularly difficult by the inherent complexity of harvest operations, primarily stemming from the numerous parameters to take into account [27] such as dimensions, pose (i.e., position and orientation), speed, grain tank capacity and unloading rate of the combine; geometry and yield of the field; locations of obstacles or restricted areas in the field; and dimensions, pose, speed, unloading rate and kinematic constraints of the grain cart.

1.2. Literature Review

1.2.1. Industry

1.2.1.1. Kinze

Iowa-based Kinze, a manufacturer of grain carts and planters, has been working with Jaybridge Robotics on the Kinze Autonomous Grain Harvesting System (Figure 1.3). During the multiyear project, Kinze has improved from the first system in which a combine controlled a single grain cart to the most recent system in which a combine can control multiple grain carts. The latest system keeps track of all vehicles in operation and orchestrates their paths to ensure the grain carts always take the safest and most efficient route to the combine [14].

The software by Jaybridge Robotics runs on the embedded computer onboard the tractor and is capable of performing key tasks including path planning, control, collision-free navigation and vehicle-to-vehicle (V2V) communication, which all take place in real-time during harvest. Via a touch-screen tablet, the user interface (Figure 1.4) allows operators to remotely coordinate the vehicles in four modes. Offload drives the grain cart

alongside and in sync with the combine to unload the grain; Follow keeps the grain cart following along behind the combine; Park orders the grain cart to return to a designated location where it transports the grain to a semi-trailer; Idle simply makes the grain cart stop and await further instruction [13].

Figure 1.3 Kinze testing Autonomous Grain Harvesting System. Reprinted from [13].

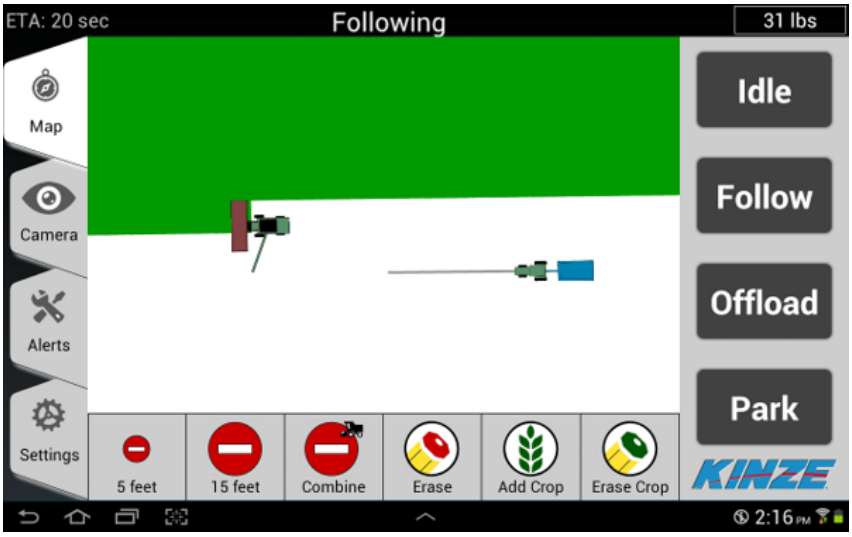


Figure 1.4 User interface of Jaybridge Robotics' software. Reprinted from [13].

The Kinze Autonomous Grain Harvesting system was first unveiled to the public in 2011, soon after which the system was tested in actual harvests in Illinois. In the following years, Kinze has continued their work on enhancing the capabilities and robustness of the system, preparing for future commercialization [13].

1.2.1.2. Smart Ag

The Iowa-based software company, Smart Ag, released the AutoCart system in 2018. This cloud-based software application is designed for automating tractors pulling grain carts, allowing farmers to use tablets to set unloading locations, adjust speeds, monitor locations and command the grain cart to sync motions with the combine. This technology is expected to be able to automate existing equipment regardless of the manufacturer. Using AutoCart in a harvest operation, the combine can signal the grain cart to approach and unload the grain. Once the unloading is finished, the system automatically directs the grain cart to the grain transfer point in the field [28]. The software calculates in real time everywhere the grain cart can go before planning the most efficient path. The software can detect crops, humans, vehicles, etc., and will stop the grain cart before it hits an obstacle [29].

1.2.1.3. Autonomous Tractor Corporation

Autonomous Tractor Corporation (ATC) has engineered retrofit kits named eDrive and AutoDrive. While the former equips tractors with diesel-electric technology using diesel generator sets and wheel motors, the latter upgrades tractors to be fail-safe and autonomous by providing navigation functions, safety features and implement management. ATC's retrofit kits allow farmers to "train" the tractor and the implement

by manually performing the tasks first, after which, the tractor will replicate and extrapolate the training carefully. ATC has released a series of application packages, including tilling, grain cart shuttling (between combine and semi-trailer), manure spreading, spraying, land leveling and planting. As a supplement, ATC also developed and patented its Laser-Radio Navigation System, which incorporates multiple sensors to achieve accurate and reliable navigation, better supporting autonomous and safe movements and operations [30].

1.2.1.4. CNH Industrial

In 2016 CNH Industrial unveiled an autonomous concept tractor (Figure 1.5) under their brand Case IH. As CNH continues to improve the design, the autonomous tractor, which is fully functional and can already work in the field, is scheduled to be commercially available as early as 2020 [8], assuming legislation issues for autonomous vehicles do not hinder progress.

Figure 1.5 Case IH concept autonomous tractor. Reprinted from [31].

New Holland Agriculture, another subsidiary of CNH Industrial, released their autonomous tractor NHDrive (Figure 1.6) at the same time. Carrying the same technology from the same external company (i.e., Autonomous Solutions Incorporated), NHDrive differs from the Case IH's autonomous tractor by maintaining a fully equipped cab so a human operator can handle those tasks too challenging for complete autonomy.

Similar to Kinze's case, operators can use a tablet to plan and adjust paths for the CNH Industrial tractors, and even manage multiple tractors to run various operations in separate fields [32].

Figure 1.6 NHDrive autonomous tractor. Reprinted from [33].

Before the concept autonomous tractor, Case IH introduced in 2011 a V2V system with which a combine could request a grain cart to approach for unloading. Once close enough, the combine could take control of the grain cart, coordinating both

machines for more precise and efficient unloading. Besides increased efficiency, adoption of the V2V system lowers the required skill level of operators as well [34].

1.2.1.5. John Deere

John Deere's AutoTrac system integrates multiple sensors to provide an assistive system that automatically steers the machine through the field. AutoTrac offers benefits including less operator fatigue, reduced overlaps and misses, less wear and tear on equipment, and fuel savings [35].

John Deere's Machine Sync connects up to ten machines within a three-mile radius, and it can not only synchronize the movements of the combine and grain cart for grain transfer, but also improve overall harvest logistics. Installing Machine Sync requires a set of communication hardware and software, which all work together to provide real-time communication capability between farm workers so that they are aware of each other's locations as well as the fill levels of the combines [34].

1.2.1.6. AGCO

AGCO in 2011 unveiled a prototype of Fendt's GuideConnect, a system that pairs vehicles as a single unit to improve productivity in farming operations. The leading tractor is driven by a human operator, and the following tractor is unmanned and directed and monitored via satellite navigation and radio communications. Using the GuideConnect system when two tractors are supposed to perform an operation together, the unmanned tractor can follow the leading tractor at a predefined distance and offset, not only in straight lines, but also for turns at the end of the field. When navigating around obstacles in the field, the human driver can command the following tractor to

travel straight-behind until the obstacles are passed. When the satellite signal is interrupted, the following vehicle stops automatically. AGCO is continuing to improve the system via additional field tests [34].

1.2.1.7. Yanmar

In 2018, Yanmar launched two tractors with autonomous features in Japan, namely the semi-autonomous Auto Tractor and the unmanned Robot Tractor. Both tractors rely on Yanmar's unique Information and Communications Technology (ICT) that combines advances in industrial robotics with precise positioning data provided by the onboard RTK-GPS. A waterproof, ruggedized tablet is employed for the human operator to program driving routes and track the machines. In operation, two modes are available to control the tractor: in "Auto" mode, the tractor handles all driving operations (i.e., forward, reverse, turn and stop), while in "Linear" mode, the tractor only handles moving back and forth across the field during cultivation, offering more flexibility so that human operators can take over to deal with more difficult terrain and/or complex situations. The ICT system also gathers data through a sensor network to facilitate farm management [36][37].

1.2.2. Academia

1.2.2.1. Coordination/Cooperation

Numerous academic studies have been carried out to develop leader-follower systems for the purposes of higher working efficiency and lower labor requirement. For example [38], two robot tractors were used to form such a system in which they could either work separately or together in a certain spatial arrangement during the operation.

When turning on the headland, they coordinated without keeping the spatial arrangement to make the best use of available space and avoid collisions. Field experiment results showed that the two robot tractors could work safely together with minor lateral navigation errors and improved efficiency. A number of other studies have focused on only one automated tractor that would follow a human-driven tractor. One study implemented an Extended Kalman Filter that fused an odometer and a laser range finder (LRF) for stable tracking in noisy conditions [39], while another study involved designing a monocular vision-based sensing system that worked with a PID controller for maintaining the required distance between two vehicles [40]. Simulation results of the former technique verified high accuracy and stability, while field test results of the latter technique indicated satisfactory accuracy and the potential for application in grain harvest operations. Another study [41] focused on the development of two motion control algorithms named GOTO and FOLLOW and had the similar objective of coordinating two farming vehicles. The specific concern was that if there were a potential collision due to path overlap, the slave (follower) would automatically slow down or change paths to yield to the master (leader). The FOLLOW algorithm incorporated a nonlinear sliding mode controller that guided the slave to follow, regardless of the traveling speed and direction, the master at the specified distance and angle. This study further validated that the sliding mode controller could outperform PD controllers with lower lateral offset and better spacing controls.

Some studies considered coordinating more than two farming machines. For example, one study focused on a distributed control framework to coordinate teams of

agricultural vehicles to operate in proximity [42]. This framework supported peer-to-peer operation mode in addition to master-slave mode. Equipped with a nonlinear model predictive controller, each vehicle incorporated the other vehicles' motion plans in the computation of optimal control actions for collision-free path tracking. Simulation tests showed that the minimization of the tracking error enabled the controller to accurately track all paths, even those with sharp turns. A system architecture presented in another study [43], also designed to synthesize a fleet of agricultural robots, integrated software from different developers. For validation, the architecture was applied in a fleet of ground-based mobile units based on a commercial tractor chassis. In general, the proposed architecture was determined to be efficient in terms of integrating new sensors, implements, and innovative algorithms in a fleet of agricultural robots.

Another study [44] took a very different approach, as no vehicle was automated, and a multi-agent coordination and control system for a combine and a grain cart involved in a crop harvesting process was deployed. The system generated detailed instructions and guidance for the operator of each vehicle via a graphical user interface (GUI). The actual execution of the operations was monitored in case of any variations or disturbances, and the control instructions were updated to remain valid and effective throughout the process.

The above studies focused on connecting two or more agricultural vehicles to enable them to achieve collision-free cooperation when performing farming tasks together. They placed more importance on the efficient and safe local coordination of the

vehicle motions, while the complex navigation and scheduling tasks in the large-scale dynamic farming environment were not addressed.

1.2.2.2. Motion Planning

As described previously, motion planning in complex and dynamic agricultural environments is a difficult task, which a few studies have addressed. One study [45] considered an optimized coverage planning approach for farming operations in the fields where obstacles were present. The approach featured four planning steps: (i) generation of field tracks, (ii) clusterization of the tracks into blocks considering the obstacle areas, (iii) generation of headland paths, and (iv) sequence optimization for farming vehicles to visit the blocks. This approach was tested and proved to be able to provide necessary information for navigation of farming machines in complex fields with obstacles. Another study [46] that also considered obstacles proposed an algorithmic approach for optimal path generation for service units. The approach abstracted a field as a two-dimensional grid. The grid cells had different states in both the representation domain (i.e., obstacle, free, initial, or goal region) and action domain (i.e., service unit to move one cell up, down, left, or right). Based on this concept, a discrete transition graph was created, and finally the optimal path was generated by a graph search algorithm. According to the simulation results, the proposed approach increased the overall efficiency of the cooperative operations, reduced soil compaction, and could be applied in large scale cooperative operations, thanks to its low computational requirements. Using a different method, another study [47] that developed a randomized motion planner for mobile robots in agricultural environments also achieved high computational

efficiency. The highlight of the proposed approach was splitting planning into two phases to reduce computational time. Phase I was primary path planning with Rapidly-exploring Random Tree (RRT), and Phase II was path smoothening and refining with B-spline curves. Results of the numerical experiments validated the practicality and effectiveness of the planner.

Although the techniques proposed in the above studies showed promise for motion planning in complex environments, they were not feasible for online applications. They were able to generate efficient paths for the vehicle to cover an entire field with obstacles, but the computations relied on the full knowledge of the field and were performed before the vehicle started any motion. However, to enable a grain cart to navigate autonomously, the motion planning needs to be conducted in real-time while the grain cart is moving around and the environment is changing.

1.2.2.3. Task Scheduling

A number of studies have approached the problem of optimal task scheduling in farming operations from different perspectives. One study [48] addressed the dynamic vehicle routing problem considering time constraints on servicing demands. The main contribution of this work was that it provided insight into how temporal information on location of demands could be converted into a reachability graph, based on which optimal routing policies could be designed for a service vehicle. A performance analysis of the optimal policy was conducted, and the results justified the analytic claims presented. An earlier study [49] proposed two optimization methods for vehicle routing problems under time constraints. The first method featured a K-Tree relaxation and the

second a Lagrangian decomposition. Both algorithms could provide optimal solutions to problems with up to 100 visiting points. Another study [50] did not focus on algorithms yet contributed to this research field by proposing a novel taxonomy that categorized problems with temporal and ordering constraints. The taxonomy emphasized the differences between temporal and ordering constraints, and it organized the reviewed literature accordingly. A variety of topics related to task allocation were covered by this study, including vehicle routing and scheduling problems.

The task scheduling techniques presented above could address various servicing demands at many locations. However, the scheduling problems they dealt with were static, as the servicing demands and visiting points were invariant over time. In harvest operations, the task scheduling problem is dynamic, as the combine's fill rate and speed vary with time, and unexpected events/obstacles occur, which affect the temporal constraints in unpredictable ways.

1.2.3. Research Gaps

From the standpoint of industry, numerous farm equipment manufacturers and startups have ongoing projects that involve AAVs. Most AAV systems being developed or marketed (e.g., remote control with tablet computers, leader-follower systems, and driving assistance) may be practical for today's farming operations, but they are better described as remote-controlled or semi-autonomous instead of fully autonomous, as human supervision and/or control is still required.

From the standpoint of academic research, a number of studies involving planning for AAVs have been published. However, the majority of AAV studies have

tried to solve simplified generic problems without focusing on specific farming operations that are often of higher complexity. The interactions between combines and automated/autonomous grain carts that are characterized by tight interleaving of temporal and spatial constraints have been scarcely investigated [51]. Most production-level harvesting with combines and grain carts is still performed without detailed planning, and harvesting efficiency heavily relies on the operators' experience [52].

Therefore, the existing motion-planning approaches, although shown to be effective and efficient in many applications, cannot be used without modifications or improvements for autonomous grain carts that service combines in harvest operations. Ideally, a good planning algorithm for an autonomous grain cart can promptly generate the most efficient motions to direct the grain cart to reach the combine at the right time in the right pose, taking into account the pose, speed and fill level of the combine, the dynamic environment, and the kinematic constraints of the grain cart. To the best of the author's knowledge, such motion planning algorithms are still to be developed.

1.3. Objectives

To address the research gaps noted above, this study attempts to develop a motion planning algorithm that can facilitate full autonomy of grain carts for intelligent navigation in crop fields and efficient service to combines in harvest operations. This overall goal can be divided into the following two specific objectives.

The first objective is to develop a motion planning algorithm and an associated task scheduling strategy for autonomous grain carts to intelligently navigate in the field and accomplish the logistical tasks in harvest operations. This objective takes into

account the spatial constraints (i.e., environmental dynamics caused by crop rows and other static and dynamic obstacles) and temporal constraints (i.e., meeting timing between grain cart and combine). The solution procedure involves performing simulation tests to verify the effectiveness (i.e., accomplishment of navigation tasks), robustness (i.e., effectiveness in simple and complex test cases), efficiency (i.e., generation of short and smooth paths) and computational ease (i.e., low consumption of CPU time) of the motion planning algorithm and the associated task scheduling strategy.

The second objective is to provide a high-level software and hardware solution to building navigation systems on autonomous grain carts for implementing the proposed motion planning algorithm. This objective covers sensor selection, communication options, control technique, and actuation plan. The solution procedure involves carrying out mobile robot tests to first verify the effectiveness and practicality (i.e., effectiveness in real-world setups) of the navigation solution, then further verify the effectiveness, robustness, and especially efficiency of the proposed motion planning algorithm.

1.4. References

[1] Food and Agriculture Organization of the United Nations. (2017). *The future of food and agriculture: trends and challenges*. Rome.

[2] Bronars, S. (2015). A Vanishing Breed: How the Decline in US Farm Laborers Over the Last Decade Has Hurt the US Economy and Slowed Production on American Farms. *Partnership for a New American Economy*, *2*.

[3] Zahniser, S., Taylor, J. E., Hertz, T., & Charlton, D. (2018). *Farm Labor Markets in the United States and Mexico Pose Challenges for US Agriculture* (No. 1476-2018-8188).

[4] United States Department of Agriculture, National Agricultural Statistics Service. (2019). *2017 Census of Agriculture*. Washington, D.C.

[5] Geller, Tom. "Farm automation gets smarter." *Communications of the ACM* 59.11 (2016): 18-19.

[6] The Economist Newspaper. (2016, May 11). The Future of Agriculture. Retrieved from https://www.economist.com/technology-quarterly/2016-06-09/factory-fresh?fsrc=scn/fb/te/bl/ed/thefutureofagriculture

[7] Anonymous. (2016, October 26). Agriculture Technology Spotlight: Self-Driving Tractors. Retrieved from https://www.firstfurrow.com/agriculture-technology-spotlight-self-driving-tractors/

[8] Daniel, J. (2016, September 16). Future of Farming: Driverless Tractors, Ag Robots. Retrieved from https://www.cnbc.com/2016/09/16/future-of-farming-driverless-tractors-ag-robots.html

[9] Centers for Disease Control and Prevention. (2019, October 9). Agricultural Safety. Retrieved from https://www.cdc.gov/niosh/topics/aginjury/

[10] Belton, P. (2016, November 25). In the Future, Will Farming Be Fully Automated? Retrieved from https://www.bbc.com/news/business-38089984

[11] Canada, H., & Michell Zappa. (2014, May 5). 15 Emerging Agriculture Technologies that will change the world. Retrieved from https://www.businessinsider.com.au/15-emerging-agriculture-technologies-2014-4

[12] Mayfield, B. (2016, March 29). John Deere Hands-Free Guidance System Continues its Evolution. Retrieved from https://johndeerejournal.com/2016/03/terry-picket-first-gps-unit/

[13] Brown, H. (2013, November 11). From Precision Farming to Autonomous Farming: How Commodity Technologies Enable Revolutionary Impact. Retrieved from https://robohub.org/from-precision-farming-to-autonomous-farming-how-commodity-technologies-enable-revolutionary-impact/

[14] Bedord, L. (2016, April 19). Fully Autonomous Vehicles Are Coming. Retrieved from https://www.agriculture.com/content/fully-autonomous-vehicles-are-coming

[15] Bhutani, A. (2018, February). Autonomous Farm Equipment Market Share - Industry Size Report 2024. Retrieved from https://www.gminsights.com/industry-analysis/autonomous-farm-equipment-market

[16] Farmingmag. (2019, May 3). What You Need to Know About Autonomous Vehicles. Retrieved from https://www.farmingmagazine.com/bits-and-pieces/tractors/autonomous-vehicles/

[17] Stoll, Albert, and Heinz Dieter Kutzbach. "Guidance of a forage harvester with GPS." *Precision Agriculture* 2.3 (2000): 281-291.

[18] Thuilot, Benoit, et al. "Automatic guidance of a farm tractor along curved paths, using a unique CP-DGPS." *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*. Vol. 2. IEEE, 2001.

[19] Shalal, N., et al. "A review of autonomous navigation systems in agricultural environments." *2013 Society for Engineering in Agriculture Conference: Innovative Agricultural Technologies for a Sustainable Future*. Engineers Australia, 2013.

[20] Hameed, Ibrahim Abdel Fattah Abdel. "Intelligent Behavior of Autonomous Vehicles in Outdoor Environment." *Technical Reports Mechanical Engineering* 1.2 (2012).

[21] Roberts, C. C. (2020, March 13). Combine Harvester Fires. Retrieved from https://www.croberts.com/harvester.htm

[22] Zhang, Mengzhe, and Sourabh Bhattacharya. "Scheduling and motion planning for autonomous grain carts." *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015.

[23] Hansen, Alan C., Robert H. Hornbaker, and Qin Zhang. "Monitoring and analysis of in-field grain handling operations." *International conference on crop harvesting and processing*. American Society of Agricultural and Biological Engineers, 2003.

[24] Anonymous. (2020, January 28). Import of Agricultural Machinery Fell in 2019. Retrieved from https://www.productivacm.com/importacion-de-maquinarias-agricolas-cayo-en-2019/

[25] Ali, Osman, Bart Verlinden, and Dirk Van Oudheusden. "Infield logistics planning for crop-harvesting operations." *Engineering Optimization* 41.2 (2009): 183-197.

[26] Scheuren, Stephan, et al. "The problem of spatio-temporally constrained motion planning for cooperative vehicles." *Proceedings of the 26th Workshop "Planen, Scheduling und Konfigurieren, Entwerfen"(PuK 2011)*. 2011.

[27] Bochtis, Dionysis, et al. "Field operations planning for agricultural vehicles: A hierarchical modeling framework." *Agricultural Engineering International: CIGR Journal* (2007).

[28] Flammini, D. (2018, January 12). Iowa Software Company Releases Autonomous Grain Cart Software. Retrieved from https://www.farms.com/ag-industry-news/iowa-software-company-releases-autonomous-grain-cart-software-107.aspx

[29] Anonymous. (2018, April 3). Remarkable New Smart Grain Cart. Retrieved from https://www.agnook.com/news/crops/remarkable-new-smart-grain-cart/

[30] Anonymous. (2015, November 13). Autonomous Agricultural Vehicles. Retrieved from https://www.idtechex.com/en/research-article/autonomous-agricultural-vehicles/8678

[31] AgroProfi. (2016, September 3). Case IH Super Autonomous Concept Tractor of the Future. Retrieved from https://www.youtube.com/watch?v=Mb8d-bqlOU8&t

[32] Dormehl, L. (2016, September 2). Self-Driving Tractors Are the Terminators of the Farming World. Retrieved from https://www.digitaltrends.com/cool-tech/self-driving-tractors/

[33] Anonymous. (2017, August 29). CNH Industrial Presents a Prototype of a Tractor Fueled with Methane. Retrieved from https://agriculture.newholland.com/lar/es-ar/NewsSite/Pages/news/2017/CNH-Industrial-presenta-un-prototipo-de-tractor-alimentado-con-metano.aspx

[34] Hest, D. (2018, December 8). New Driverless Tractor, Grain Cart Systems Coming This Year. Retrieved from https://www.farmprogress.com/precision-guidance/new-driverless-tractor-grain-cart-systems-coming-year

[35] Anonymous. (2012, May 21). John Deere AutoTrac Controller Widens the Net. Retrieved from http://www.thecropsite.com/news/11070/john-deere-autotrac-controller-widens-the-net/

[36] Szondy, D. (2018, July 22). Yanmar's Robotic Tractors Bring Autonomous Vehicles Out to Farm. Retrieved from https://newatlas.com/yanmar-robotic-tractors/55558/

[37] Allen, J. (2018, October 26). Yanmar Launches Autonomous Tractors. Retrieved from https://www.ivtinternational.com/news/autonomous-vehicles/yanmar-releases-market-ready-robot-tractors.html

[38] Zhang, Chi, Noboru Noguchi, and Liangliang Yang. "Leader–follower system using two robot tractors to improve work efficiency." *Computers and Electronics in Agriculture* 121 (2016): 269-281.

[39] Huan, Zhang Lin, Takigawa Tomohiro, and Ahamed Tofael. "Leader-Follower Tracking System for Agricultural Vehicles: Fusion of Laser and Odometry

Positioning Using Extended Kalman Filter." *IAES International Journal of Robotics and Automation* 4.1 (2015): 1.

[40] Zhang, Linhuan, et al. "Vision-based leader vehicle trajectory tracking for multiple agricultural vehicles." *Sensors* 16.4 (2016): 578.

[41] Noguchi, Noboru, et al. "Development of a master–slave robot system for farm operations." *Computers and Electronics in agriculture* 44.1 (2004): 1-19.

[42] Vougioukas, Stavros G. "A distributed control framework for motion coordination of teams of autonomous agricultural vehicles." *Biosystems engineering* 113.3 (2012): 284-297.

[43] Emmi, Luis, et al. "New trends in robotics for agriculture: integration and assessment of a real fleet of robots." *The Scientific World Journal* 2014 (2014).

[44] Ali, Osman, et al. "Multi-agent coordination and control system for multi-vehicle agricultural operations." *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[45] Hameed, Ibrahim A., Dionysis Bochtis, and Claus A. Sørensen. "An optimized field coverage planning approach for navigation of agricultural robots in fields involving obstacle areas." *International Journal of Advanced Robotic Systems* 10.5 (2013): 231.

[46] Bochtis, D. D., C. G. Sørensen, and S. G. Vougioukas. "Path planning for in-field navigation-aiding of service units." *Computers and Electronics in Agriculture* 74.1 (2010): 80-90.

[47] Elbanhawi, Mohamed, and Milan Simic. "Randomised kinodynamic motion planning for an autonomous vehicle in semi-structured agricultural areas." *Biosystems engineering* 126 (2014): 30-44.

[48] Bopardikar, Shaunak D., Stephen L. Smith, and Francesco Bullo. "On dynamic vehicle routing with time constraints." *IEEE Transactions on Robotics* 30.6 (2014): 1524-1532.

[49] Kolen, Antoon WJ, A. H. G. Rinnooy Kan, and Harry WJM Trienekens. "Vehicle routing with time windows." *Operations Research* 35.2 (1987): 266-273.

[50] Nunes, Ernesto, et al. "A taxonomy for task allocation problems with temporal and ordering constraints." *Robotics and Autonomous Systems* 90 (2017): 55-70.

[51] Scheuren, S., Stiene, S., Hartanto, R., Hertzberg, J., & Reinecke, M. (2013). Spatio-Temporally Constrained Planning for Cooperative Vehicles in a Harvesting Scenario. *KI - Künstliche Intelligenz*, *27*(4), 341–346. doi: 10.1007/s13218-013-0267-y

[52] Ali, O., Verlinden, B., & Oudheusden, D. V. (2009). Infield logistics planning for crop-harvesting operations. *Engineering Optimization*, *41*(2), 183–197. doi: 10.1080/03052150802406540

2. SOLUTION PROCEDURE

This study accounts for the following technology components, which are indispensable to building an AAV: sensors that perceive the environment and measure the vehicle's states; communication between the vehicles for information sharing; decision-making algorithms for planning and control; and actuators for implementing the control commands [1]. Figure 2.1 illustrates a general architecture of navigation systems on AAVs that can be extracted and summarized from other studies [2][3][4][5][6]. Following this architecture, this section explores the solution procedure to address the objectives noted in the previous section. Specifically, this section focuses on the development of the motion planning algorithm and provides a high-level solution to building the associated navigation system, covering sensor selection, communication options, controller technique and actuation plan.

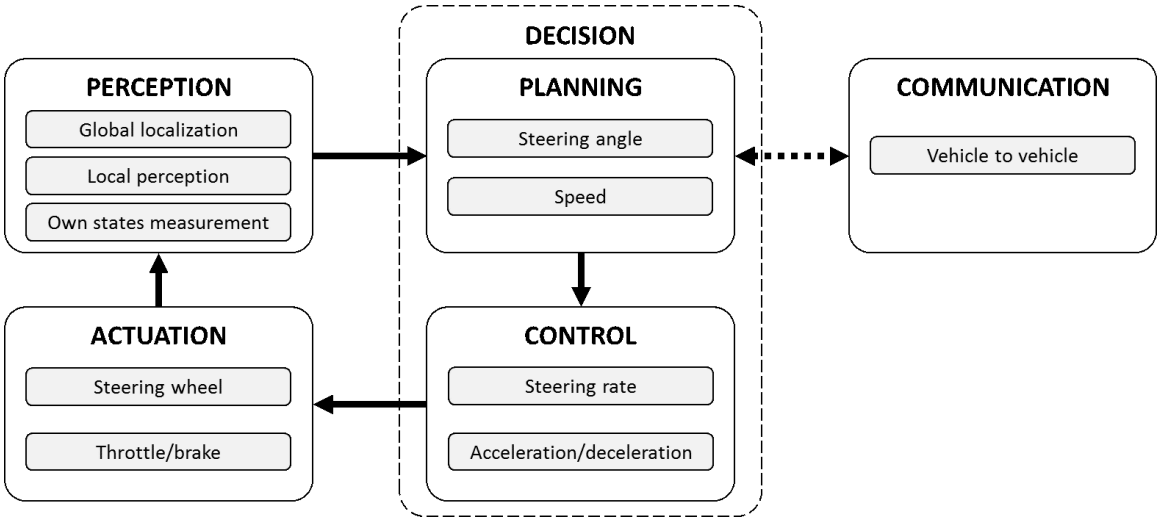


Figure 2.1 General architecture for navigation systems on AAVs.

2.1. Sensor Selection

As the first link of the technology chain of the navigation system, sensors collect essential data on environmental variables and vehicle states and feed them to the planner. To replace the sophisticated human sensing system based on the eyes, ears and most importantly, the brain, sensing systems on autonomous grain carts need to be accurate, reliable and robust. Popular sensors used for automating agricultural vehicles include Real-Time Kinematic Global Positioning System (RTK-GPS) receivers, cameras, Inertial Measurement Units (IMUs), Geomagnetic Direction Sensors (GDSs), radar systems, Fiber Optic Gyroscopes (FOGs), Light Detection and Ranging (LIDAR) sensors, optical encoders, potentiometers, Radio Frequency (RF) receivers, piezoelectric yaw rate sensors and Near Infra-Red (NIR) sensors [6].

Among these, GPS based systems are the most extensively adopted. Having been in practical use for many years though, GPS based systems are primarily employed only for global localization and cannot provide any information about the environmental dynamics caused by nature or humans. Therefore, more research has been devoted recently to guiding agricultural vehicles with local sensors that observe the vehicles' surroundings in real time [7], in which the major challenge is the reliable obstacle detection and traversability assessment in the complex unstructured agricultural environment. Traversability assessment mainly differentiates soil surfaces covered by crop plants from traversable land where crop plants have been harvested. Meanwhile, a variety of other types of obstacles can be encountered in the field, including static obstacles (e.g., trees, utility poles and storage buildings) and dynamic obstacles (e.g.,

people, animals and other farming machines). In this subsection, a number of sensors are compared, and some of them are selected to form the perception system of autonomous grain carts, whose functions include global localization, local perception and vehicle states measurement [1][8].

2.1.1. Global Localization

Global Navigation Satellite Systems (GNSS) refer to all satellite-based global localization systems that can be publicly used for vehicle positioning (e.g., GPS, GLONASS, Galileo, Beidou). Including its applications in precision farming and agricultural automation, GPS has been extensively used in modern agricultural production. Knowing the accurate location of a farming machine in the field can facilitate a variety of activities, including yield monitoring, variable rate application and automated navigation.

To improve localization accuracy of basic GPS, Differential GPS (DGPS) has been devised, which corrects original GPS signals with signals emitted by a reference receiver at a well-known location. Differential corrections improve localization data considerably by cancelling satellite ephemeris and clock errors, reaching a centimeter-level accuracy with commercial differential signal providers in case of the best implementations. Global localization of an autonomous grain cart requires centimeter-level accuracy as it needs to carefully cooperate with the combine in close proximity to unload the grain evenly and precisely into the cart. More importantly, the localization needs to be performed in real-time as an autonomous grain cart navigates in the field during a harvest operation. Fortunately, this can be achieved with RTK-GPS. An RTK-

GPS set consists of two receivers, namely a base and a rover, a radio link for data transmission between receivers, and computer software that calculates differential corrections from the base receiver, which is placed in proximity to the operating location of the rover receiver. Allowing the vehicle to be as far as ten kilometers away from the base, RTK-GPS can provide accuracy as high as two centimeters, which fulfills the requirement for global localization of autonomous grain carts.

2.1.2. Local Perception

Considering the large scales of crop fields and the parking locations of semi-trailers, autonomous navigation of grain carts in complex harvest operations in large crop fields would require a sensing range of hundreds of meters to detect distant unharvested crop rows and other random static or dynamic obstacles. Meanwhile, centimeter-level accuracy and resolution are needed so that obstacles like individual crop plants and utility poles would not be missed in the obstacle detection. Different alternatives for local perception are available, but each has advantages and disadvantages.

Monocular or stereo computer vision provides high resolution, high sampling rate and rich content (e.g., color, texture and range), yet the sensing range is relatively short (i.e., 15 to 30 m) and the required processing algorithms tend to be complex. Besides, cameras are sensitive to lighting and visibility conditions (e.g., rain, fog, smoke, etc.). Radar is more robust to diverse environmental conditions and has a longer detection range (i.e., 100 to 150 m), but its low resolution produces difficulties in signal processing and interpretation. Lidar has drawn much attention recently for sensing on

autonomous vehicles. Although it is somewhat sensitive to dust, fog and rain, lidar provides high accuracy, high resolution and fast detection over a reasonable range (up to 250 m). Lidar employs a laser optical scanner that emits pulsed laser beams and receives the beams reflected from the detected object to determine the distance by measuring the time interval between the emission and reception of the laser beams. Reports from several studies [1][5][6] have discussed the superiority of lidar over other widely used sensors in robustness, range, response speed and cost in AAV applications.

Considering that (i) most harvests take place in weather conditions without rain or fog, and (ii) the primary task of the grain cart is to navigate through crop rows (i.e., obstacle avoidance with no need for identification of traffic lights or signs), 2D lidar should be able to deliver sufficient local perception for an autonomous grain cart.

2.1.3. Vehicle States Measurement

Real-time measurement of grain carts' states of motion is also essential to autonomous navigation. While RTK-GPS can provide an estimate of global location of a grain cart, IMUs (i.e., electronic devices that combine accelerometers and gyroscopes) can capture a grain cart's instantaneous linear acceleration and velocity as well as yaw rate and heading. While accelerometers detect accelerations along the three orthogonal Cartesian axes (which can be integrated to obtain linear velocities), the gyroscopes measure instantaneous angular rates experienced by the vehicle around the Cartesian axes (which can be integrated to obtain attitude angles yaw, pitch, and roll).

The most notable problem with IMUs is the error accumulation over time, i.e., sensor drift, which can be addressed by a variety of sensor fusion methods. For example,

fluxgate compasses can be incorporated complementarily to IMUs for more robust heading detection; meanwhile, magnetic counters (which count the number of rotations of a driven wheel) and radar sensors can be employed to support more accurate linear velocity measurement.

2.2. Communication Options

Establishment of V2V communication is imperative for efficient cooperation between a combine and grain cart. Specifically, the grain cart needs to know the combine's pose, speed and fill level for planning when and where to go for meeting. Cell Global System for Mobile communications (GSM), Wireless Local Area Network (WLAN), Bluetooth and ZigBee are popular means used for V2V communications in agricultural operations involving two or more vehicles [9][10]. These technologies compete with each other in bit rate, signal quality, coverage area, cost and power consumption.

GSM networks are known for their broad coverage and good signal quality, but they are far more expensive to establish and maintain than WLANs. WLAN is a flexible data communication protocol implemented to extend or substitute for a wired local area network. Operating in the 2.4 GHz frequency band, WLAN consumes much less energy than GSM while providing a signaling rate (11Mbps) high enough for fast transfers of video, audio, graphics and files. Although the distance between the transmitter and receiver has a great influence on WLAN's signal quality and bit rate, the typical transmission range of 100 m can be extended to ten times longer with the technology of long-range Wi-Fi. Bluetooth is a wireless protocol that uses the 2.4 GHz, 915 and 868

MHz radio bands to communicate among relatively simple devices, transferring audio, graphics and files at 1 Mbps but only within very short distances (i.e., typically less than 10 m). Bluetooth consumes relatively minimal power. Finally, ZigBee, featuring low bit rate (250 kbps), power consumption and latency as well as a simple network configuration, is suitable for small data packet transfer in wireless control and monitoring applications. The ZigBee transmission distance can also be enlarged to thousands of meters by establishing a mesh network with intermediate devices.

The above comparison shows that WLAN and ZigBee meet the following communication requirements for an autonomous grain cart: (i) capability for real-time transmit of pose, velocity and fill level, (ii) low energy consumption, (iii) 1000-meter-level range to cover larger fields, and (iv) inexpensive and simple networking configuration.

2.3. Planner Development

2.3.1. Existing Planning Techniques

Motion (path) planning has been a popular subject of study in mobile robotics. Researchers have categorized planning techniques in different ways. For example, some compared conventional algorithms with heuristic algorithms [11][12]. Conventional algorithms (e.g., Artificial Potential Field, Roadmaps) have solved a wide variety of planning problems, yet they all have drawbacks like computational complexity, sub-optimality and path infeasibility under certain constraints. Consequently, many heuristic algorithms (e.g., Fuzzy Logic, Neural Networks) have been developed to address these issues, aiming at higher efficiency and enhanced intelligence. Although heuristic

algorithms do not guarantee a solution, whether utilized independently or as a complement to conventional approaches, when they do find the solution they are normally significantly faster than conventional methods. Some researchers have differentiated offline approaches from online approaches [13][14][15]. While offline planning is completed before the motion based on a priori information about the field and operation, online planning takes place incrementally during motion based on the real-time sensing information. Some of the most widely used planning techniques are discussed below, noting whether they are conventional or heuristic. For the conventional techniques, offline or online is also noted. Heuristic techniques can be both online and offline as they are often employed as a complement to conventional techniques.

2.3.1.1. The Bug Algorithm (conventional, online)

The Bug Algorithms are among the earliest and simplest sensor-based planners that address online planning in unknown environments. The basic Bug Algorithm (Figure 2.2) navigates an area and controls a robot equipped with touch sensors such that it circumvents a detected obstacle clockwise or counterclockwise until reaching a straight line to the goal, which is then followed by the robot to either reach the goal or another obstacle. The basic Bug Algorithm can be upgraded to the Tangent Bug Algorithm when the robot is equipped with long range sensors, which allows the robot to follow the tangent line to the next obstacle lying on the straight line to the goal. The Bug Algorithms are known for simplicity and low memory consumption. However, they have drawbacks such as long and suboptimal paths as well as potential collisions.

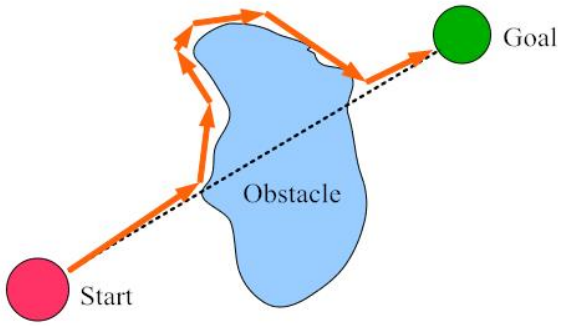


Figure 2.2 Robot circumventing obstacle with Bug Algorithm. Reprinted from [16].

2.3.1.2. Artificial Potential Field (conventional, online)

Artificial Potential Field (APF) refers to the potential field constructed by the potential functions artificially assigned to the goal and the obstacles in a navigation task. The potential functions are specifically designed such that taking the gradient of them generates attractive forces towards the goal and repulsive forces from the obstacles. The farther from the goal, the greater the attractive forces; the closer to the obstacles, the greater the repulsive forces. The vehicle is assumed to navigate under the influence of the resultant forces, which will lead the vehicle to the goal while keeping it away from the obstacles (Figure 2.3). APF is known for its mathematical elegance and simplicity as well as its computational efficiency, which makes it suitable for online feedback planning and control. However, this technique suffers from a major drawback that often compromises the plan's rationality and efficiency; when there is a cancellation of attractive and repulsive forces, the vehicle may become stagnant at or wander around points other than the goal, a situation known as being trapped in local minima (Figure

2.4). Due to the force cancellation or contradiction, the robot may also experience oscillations in long narrow corridors and not be able to pass between closely spaced obstacles [17].

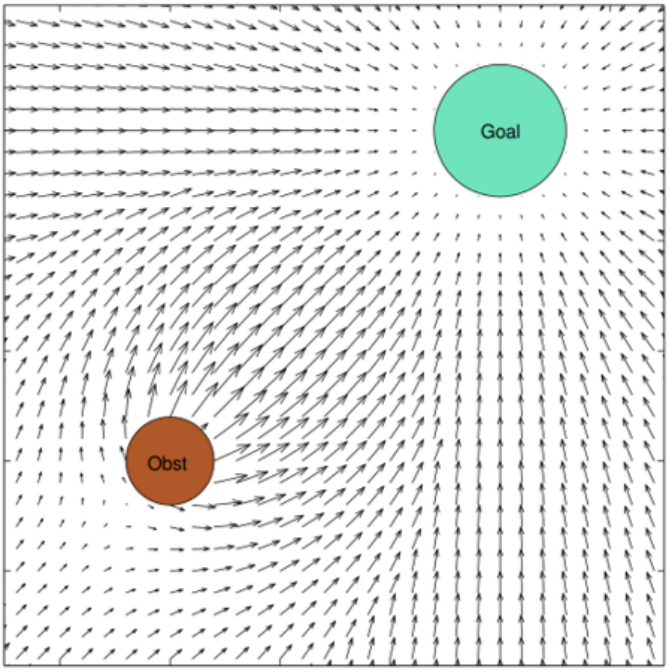


Figure 2.3 Artificial Potential Field with repulsive and attractive forces. Reprinted from [18].

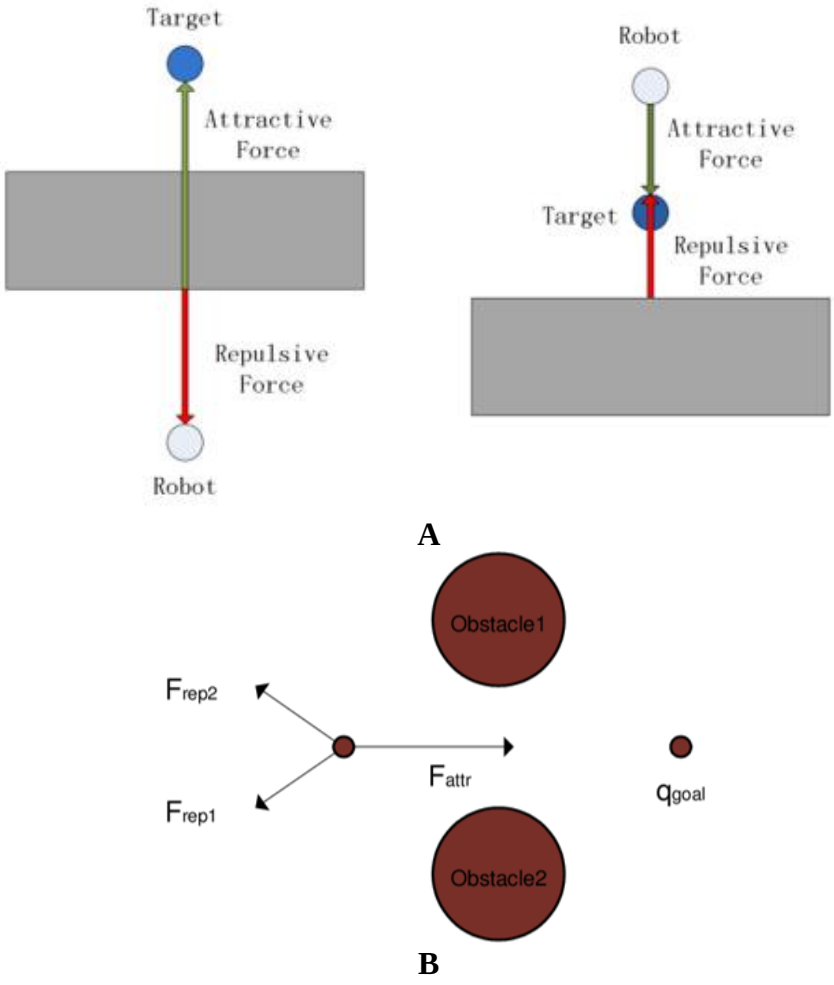


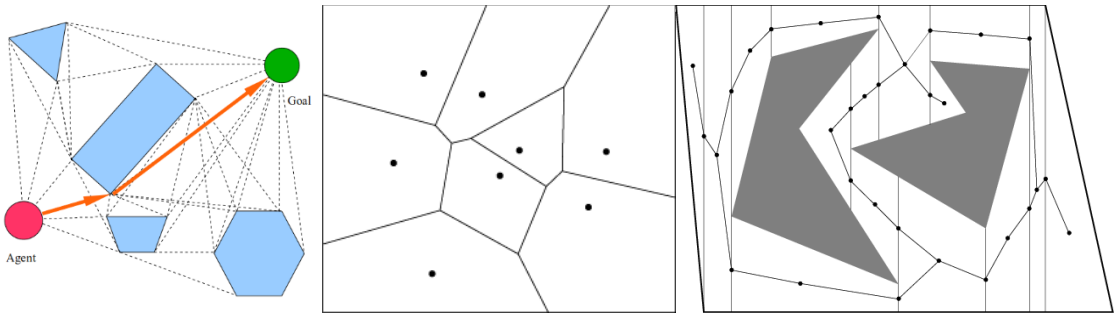
Figure 2.4 Local minima caused by force cancellation and contradiction. (A) Obstacle in between and obstacle behind goal. Reprinted from [19]. (B) No pass through narrow gap. Reprinted from [20].

2.3.1.3. Roadmaps (conventional, offline)

The roadmap approach is built on the concept of configuration space (i.e., C-space, the set of all positions the robot can attain). While physical obstacles are represented by forbidden regions in the C-space, the process of searching for a collision-free path is reduced to constructing a continuous route connecting start and target points in the free region. In other words, with the geometric representation of the environment,

the planning task becomes a graph-searching problem, which enables many offline planning techniques to produce a globally optimal solution. Well-known roadmaps include the visibility graph, Voronoi diagram and cell decomposition, which capture topology of the free space in different ways (Figure 2.5).

A visibility graph is drawn by connecting two vertices of mutually visible polygonal obstacles. The shortest path is then identified through the connecting lines in the visibility graph. Because the number of lines is determined by the number of obstacles and their edges, this approach is more efficient in sparse environments. The Voronoi diagram is constructed by connecting points equidistant from two or more adjacent obstacles. Consequently, the path is safer but normally longer. The cell decomposition approach decomposes the C-space into cells and then searches for a route in the cell graph. The main difficulty lies in the determination of size of the cells. The smaller the size, the more accurately the environment is modeled. However, smaller sizes consume more memory and exponentially raise search complexity.



A **B** **C**

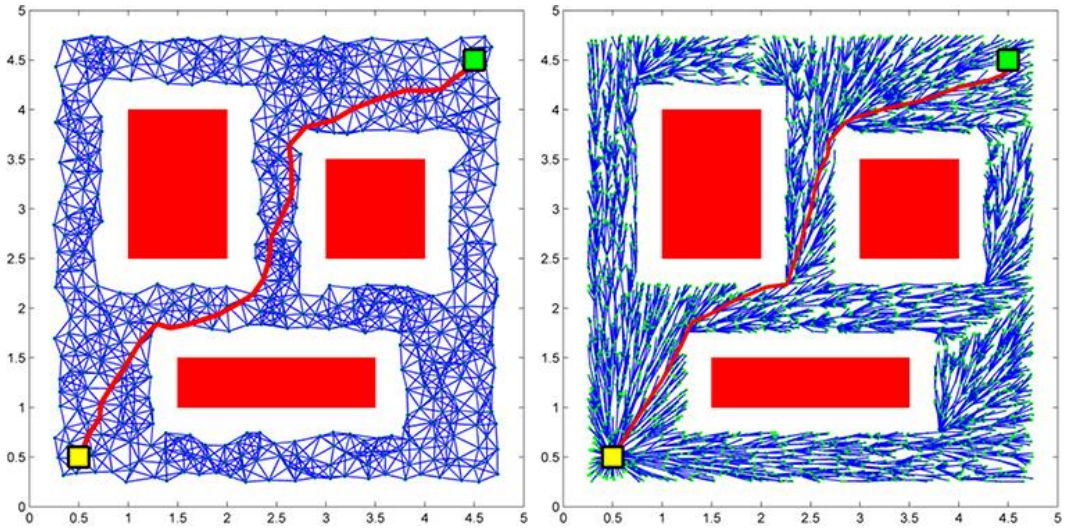
Figure 2.5 Popular roadmap approaches. (A) Visibility graph. Reprinted from [16]. (B) Voronoi diagram. Reprinted from [21]. (C) Cell decomposition. Reprinted from [22].

Once the roadmap is constructed, standard graph search techniques such as the well-known Dijkstra's Algorithm and A-Star (A*) Algorithm family can be used to solve for the shortest path. Dijkstra's Algorithm is a graph searching algorithm that finds the single-source shortest path in the graph. In automated driving, it has been implemented in multi-vehicle simulations as well as in the 2007 DARPA Urban Challenge. The A* Algorithm, as an extension of Dijkstra's Algorithm, is a graph searching algorithm that enables a fast node search with the implementation of heuristics. The algorithm is suitable for searching spaces mostly known a priori by the vehicle and is costly in terms of memory and speed for vast areas. The A* Algorithm has been modified in several applications, and some of the improved versions are D, Field D*, Theta*, ARA* and AD*. The A* Algorithm family has been used for planning in unstructured spaces and parking lots and was adopted by the teams of Stanford University and Karlsruhe Institute of Technology (KIT) for their vehicles in the 2007 DARPA Urban Challenge, where the winning vehicle "Boss" used AD*.

2.3.1.4. Sampling-Based Methods (conventional, online/offline)

The basic idea behind sampling-based approaches is to randomly sample the C-space while looking for connectivity inside it. The most well-known methods are Probabilistic Roadmap (PRM) and Rapidly-exploring Random Tree (RRT). The PRM method first uses coarse sampling to obtain nodes of the roadmap and then fine sampling for the edges, which are the collision-free paths between nodes. Once the roadmap is constructed, planning queries are answered by connecting the start and goal. Basic PRM uses a uniform random distribution for node sampling and works well for many

problems. However, PRM is a multi-query planner and may not be able to solve single-query problems sufficiently fast, in which case, other sampling-based planners such as RRT and Expansive-Spaces Tree (EST) can be adopted. The RRT method searches the route to the goal by randomly probing the C-space and incrementally expanding a "tree" from the start point while avoiding obstacles. Without any orientational preference, the planner explores the search space uniformly. This method is efficient, because while extending, the "tree" does not attempt to cover all nodes of the entire free space, but rather terminates the search immediately when the goal is reached. Able to solve very complex problems (e.g., MIT used RRT to navigate their vehicle at the DARPA Urban Challenge), the sampling-based methods suffer from a major drawback: the generated path is not optimal and quite jerky. An extension of RRT called RRT* was developed by adding a lower bound estimate to the RRT search, with which asymptotic optimality may be achieved with greater computational costs. The same technique has also been applied to develop the improved form of PRM, namely PRM*, which is illustrated along with RRT* in Figure 2.6. In addition to automotive applications, the sampling-based planners are known to outperform many other planners when solving high dimensional problems, such as motion planning for humanoids and robotic arms of many degrees of freedom.



A **B**

Figure 2.6 Path searching with sampling-based methods. Reprinted from [23]. (A) PRT* takes random samples from search space, tests occupancy, and connects free configurations. (B) RRT* incrementally constructs space-filling tree towards unsearched space by connecting random samples from search space.

2.3.1.5. Fuzzy Logic (heuristic)

Fuzzy Logic (FL) resembles the human decision-making methodology and is known for its simplicity, efficiency, and robustness in spite of uncertainty and imprecise information. Any event, process, or function that evolves continuously cannot be characterized simply in terms of either true or false conditions; therefore, compared with binary logic that only looks at absolute truth (1) or falseness (0), FL deals with membership values in the range of [0, 1]; closer to 0 means a higher tendency towards falseness; closer to 1 means a higher tendency towards truth. Thus, FL can often be more practical in solving real-world problems. As a matter of fact, FL is not logic that is fuzzy but logic that deals with fuzziness, uncertainty and imprecision. FL solves problems in a perception-reaction fashion in the sense in which it first breaks down the problem into a number of simple tasks and then addresses each task, applying the associated set of

fuzzy rules developed based on human experts' heuristic knowledge, experience, or intuition, which normally take the form of IF...THEN... When an input does not precisely correspond to any predefined rule, partial matching of the input is used to interpolate an answer.

2.3.1.6. Neural Network (heuristic)

Inspired by biological nervous systems (e.g., human brain), the Neural Network (NN) is an information-processing paradigm whose procedures can be expressed as first receiving inputs, then multiplying each input by a predefined weight and finally applying an activation function to the sum of the results. During information processing, the weights are regularly updated according to the experience learned by the numerous highly interconnected collaborating neurons, namely the processing elements. NNs may sound complicated, but the technique requires fairly simple calculations, which normally do not involve searching for free C-space, optimization of any cost function, or local collision checking procedures.

2.3.1.7. Genetic Algorithm (heuristic)

As a problem-solving strategy, the Genetic Algorithm (GA) mimics biological evolution. The evolution typically starts from a set of randomly generated candidate solutions, whose properties, or "genes", can be employed to produce better solutions. By applying genetic operators such as mutation and crossover, new generations of solutions are created, each of which is evaluated by a metric called a fitness function specified by the objective of the computation. In this iterative process, only parent solutions of high fitness are selected to produce new generations of solutions, thus, optimal solution is

reached when the iteration is completed. In planning applications, GA is often utilized along with NN techniques.

2.3.1.8. Ant Colony Optimization (heuristic)

Ant Colony Optimization (ACO) is a probabilistic computational technique that was based on the behavior of ants searching a path from their colony to a food source. When ants wander around for food, they lay down pheromone trails, which other ants tend to follow. The denser the pheromone, the more attractive the trails. When the ants find a short path between their colony and the food source, the pheromone density on this path becomes higher as the ants commute along it more frequently, leaving more pheromone each time. On the other hand, pheromone on longer paths tends to evaporate before being marched over again. Thus, shorter paths are followed by more and more ants while longer paths become less and less attractive, which eventually leads all the ants to follow the shortest path. Figure 2.7 illustrates this process.

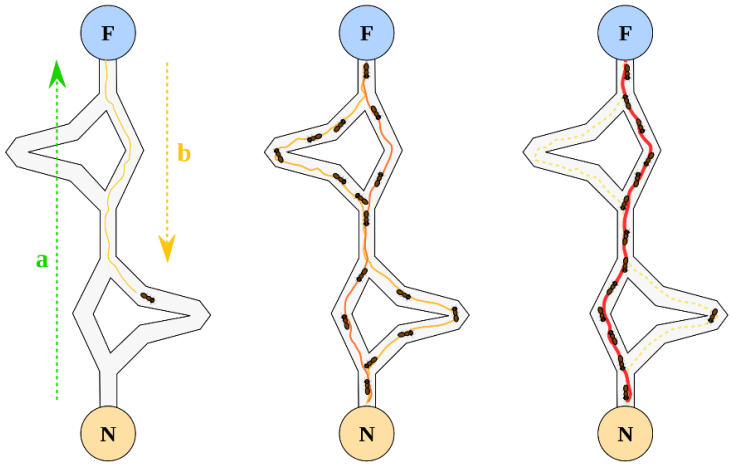


Figure 2.7 Shortest path between food (F) and nest (N) found by Ant Colony Optimization. Reprinted from [24].

2.3.1.9. Particle Swarm Optimization (heuristic)

Particle Swarm Optimization (PSO) attempts to solve an optimization problem in the C-space by iteratively improving the quality of a population of candidate solutions, namely the particles (because these candidate solutions, or in other words, the potential optimal position points, look like particles wandering around the C-space). The improvement is achieved by moving the particles around in the search-space, whose directions are simultaneously influenced by their local best-known positions and guided towards better positions found by other particles. Thus, the iteration leads the swarm to the optimal solution. PSO techniques have proven beneficial for avoiding undesirable situations such as congestion and deadlocks.

2.3.1.10. Simulated Annealing (heuristic)

Simulated Annealing (SA) models the physical process of heating and cooling a material to alter its physical properties by changing its internal structure and minimizing its internal energy. Instead of energy, SA aims at minimizing (or maximizing) the objective function of the optimization problem. To simulate the physical annealing, a temperature variable is adopted, which is first set high and then slowly decreased as the algorithm executes. At each iteration, a new point is randomly generated whose distance from the previous point (i.e., the search scope) is to some extent proportional to the temperature variable. When searching for the shortest path, new points that lower the value of the objective function are accepted, but with a certain probability, points that raise the value are adopted as well, for the purpose of preventing the computation from being trapped in local minima. The probability of accepting “bad” points is higher when

the temperature variable is high in the beginning and gradually reduced as the temperature cools down later in the process, by which time, ideally, the search has converged to a small area containing the optimal solution. SA's principal benefit is that the gradual cooling process makes it remarkably effective when planning in large C-spaces with many local minima.

2.3.1.11. Interpolation

Whether using conventional or heuristic planning techniques, the generated path, which is normally composed of a set of way points, could be infeasible for vehicles to follow under certain kinematic and/or dynamic constraints. Therefore, interpolations are sometimes adopted as path smoothing solutions, which take into account the feasibility, comfort, vehicle dynamics and other parameters for improving the trajectory. Specifically, interpolation constructs and inserts a new set of data within the range of the given set of way points, generating a new path to improve the trajectory continuity and smoothness. The interpolation implements different curves and/or shapes to smooth the path, among which the most commonly used are straight and circular shapes, clothoid curves, polynomial curves, Bézier curves and spline curves. A broad range of interpolation examples can be found in [15].

2.3.2. Planning for Autonomous Grain Carts

2.3.2.1. Theoretical Basis and Rationale

After exploring and comparing the features of existing planning techniques, Artificial Potential Field (APF) and Fuzzy Logic Control (FLC) were selected to form the basis of the planning algorithm for autonomous grain carts. As previously described,

APF features mathematical elegance and simplicity as well as computational efficiency that supports online feedback planning. However, the potential force cancellation or contradiction can cause serious problems like local minima traps and oscillations, which degrade the rationality and efficiency of the generated plans. To fully exploit the advantages of APF and overcome its limitations, the following two improvement strategies were proposed and implemented together to construct a new algorithm with upgraded performance. (i) To reduce the probability for local minima traps, the influence range of the repulsive forces from the obstacles was to be set to be small so that they would only be effective when the grain cart traveled dangerously close to the obstacles; when the grain cart was at safe distances from the obstacles, another algorithm should be able to generate rational and efficient motion plans, taking into account the total travel distance. (ii) Force cancellation or contradiction that degrades APF's performance is a simple mathematical problem not necessarily arising from complex environments or situations. A human operator can expertly navigate a grain cart to commute between the semi-trailer and the combine while avoiding different types of obstacles, so their operational knowledge, experience, and intuition can be leveraged to perform efficient motion planning. FL is a simple, efficient and robust approach to solving problems with fuzziness, uncertainty and imprecision, and it resembles the human decision-making methodology, leveraging human knowledge, experience, and intuition. Navigation in dynamic agricultural environments requires fast reactions to uncertainties, while the behaviors of agricultural vehicles are often too complex to be modeled accurately, Fuzzy Logic Control (FLC) can directly leverage knowledge and experience and intuition as

well as the imprecise reasoning and decision-making mechanism of human operators to provide planning solutions quite efficiently. FLC is thus a good choice for overcoming the deficiencies of APF.

As popular planning techniques, both APF and FLC have been investigated for decades, during which a number of studies employed FLC to address the drawbacks of APF for upgraded performance. While conventional APF depends only on the distance between the robot (vehicle) and obstacle, the scheme proposed in [25] introduced a fuzzy variable that specified the level of influence each obstacle had over the robot's future path; an obstacle was more influential when it was closer to the robot, in front of the robot and when the robot was moving fast. This variable improved the planning by scaling the APF produced by the obstacle (a similar concept is found in [26]). Satisfactory results were obtained in simple simulation tests. The APF+FLC planner introduced in [27] enabled navigation of a two-wheel mobile robot in unknown environments. APF generated the initial path, which was temporarily ignored when the FLC detected a potential collision and took reactive actions. When collisions were no longer possible, APF took back the control. Taking into account the distances to obstacles, as well as the heading and velocity errors (i.e., difference between the current and desired values), 15 fuzzy rules estimated the collision probability and derived the FLC output, namely the rotating speed of each wheel. Efficiency of this method was verified via simulations in various static and dynamic environments. Similarly, another study [28] combined planned and reactive behaviors for real-time navigation of a mobile robot by associating an Electrostatic Potential Field (EPF) path planner with a two-

layered FL inference engine. While the first layer estimated collision probabilities, the second layer guaranteed collision avoidance by taking corrective maneuvers, temporarily overriding the initial EPF-generated path. Satisfactory results were obtained both in simulations and real-world experiments with a mobile robot. In [29], the APF and FLC concepts were integrated into a common framework termed Potential Fuzzy Controller. The intelligence and robustness of the approach came from the 15 collision-avoidance fuzzy-rules, as well as the synthesis of both heuristic knowledge and the sampled sensor-input-actuation-output data pairs. Ten tests were conducted, which demonstrated that the proposed hybrid algorithm outperformed FL alone and the conventional APF+FL method. Also integrating APF with FLC for path planning, another study [30] compared two navigation schemes: while the first one had an APF planner and a PID controller, the second one had a Fuzzy-APF planner and a Fuzzy controller. Although both schemes produced good results, the latter supported more smooth and efficient motions, as well as faster reactions to obstacles. The effectiveness and robustness of the proposed method was validated in simulations and physical tests with a small simple robot. The APF + FLC planner developed in [31] made use of an FLC expert system to guide a mobile robot with the most appropriate heading towards a target. The key highlight of the approach was specifying APF repulsive forces with IF-THEN fuzzy rules. Simulation results verified the feasibility and computational efficiency of this approach in both static and dynamic environments.

It has been shown that a number of studies have incorporated APF with FLC for motion (path) planning of vehicles or mobile robots in various simulated and/or physical

environments. Satisfactory test results demonstrated the great potential of this approach, including effectiveness, robustness and efficiency. However, among all these studies, none has investigated agricultural environments, not to mention harvest operations, whose planning task is distinctive and complex due to the challenges described previously: temporal constraints for meeting timing of the grain cart and combine, spatial constraints from the dynamic environment, need for real-time adaptation to unexpected events and obstacles, and numerous parameters of the combine, grain cart and crop field. In this study, APF and FLC are integrated in an innovative way for the development of the motion planning algorithm for autonomous navigation of grain carts in crop harvest operations, addressing the aforementioned challenges.

2.3.2.2. Construction of Artificial Potential Field

In real-time implementations, APF-based planning is an iterative process in which the resultant artificial force is computed at the current pose of the vehicle, then the vehicle takes a small step forward in the direction of this force. Incremental computation and movement are repeated until the goal is reached. The scope of this study is limited to a plain field (i.e., no significant slopes or curves) that can be represented by a 2D C-space. The vehicle position is symbolized by $q = (x, y)$. Correspondingly, the attractive and repulsive potentials at q as well as their summation are $U_{att}(q)$, $U_{rep}(q)$, and $U(q)$, respectively. With $U(q)$ being differentiable for every q in the C-space, the resultant artificial force is

$$\vec{F}(q) = -\nabla U(q) = -\nabla U_{att}(q) - \nabla U_{rep}(q),$$

where $\nabla U(q)$ denotes gradient of U at q, namely a vector pointing at the direction of fastest change of U at (x, y). In a 2D C-space, the gradient can be calculated as

$$\nabla U(q) = \begin{bmatrix} \frac{\partial U}{\partial x} \\ \frac{\partial U}{\partial y} \end{bmatrix}.$$

Looking at the attractive potential first, the farther away the vehicle from the goal, the greater the $U_{att}(q)$. Therefore, a possible form of the attractive potential is

$$U_{att}(q) = \frac{1}{2}\xi\rho^2(q_{goal}),$$

where ξ is a constant scaling factor and $\rho(\cdot)$ represents the distance between the vehicle and something. $\rho(q_{goal})$ is the distance between the vehicle and the goal. In a general C-space which is not necessarily 2D,

$$\rho(q_{goal}) = \|q - q_{goal}\| = \sqrt{\sum(x_i - x_{gi})^2},$$

where $q = (x_1, x_2, \dots, x_n)$ and $q_{goal} = (x_{g1}, x_{g2}, \dots, x_{gn})$ are the coordinates of the vehicle and the goal, respectively. Therefore, the attractive force can be derived as follows.

$$\vec{F}_{att}(q) = -\nabla U_{att}(q) = -\nabla\frac{1}{2}\xi\rho^2(q_{goal}) = -\xi\rho(q_{goal})\nabla\rho(q_{goal}),$$

where

$$\nabla\rho(q_{goal}) = \nabla\sqrt{\sum(x_i - x_{gi})^2} = \frac{\sum(x_i - x_{gi})}{\sqrt{\sum(x_i - x_{gi})^2}} = \frac{q - q_{goal}}{\|q - q_{goal}\|}.$$

Therefore, we obtain the attractive force as a vector whose direction is from q to q_{goal}, attracting the vehicle to the goal:

$$\vec{F}_{att}(q) = -\xi\|q - q_{goal}\|\frac{q - q_{goal}}{\|q - q_{goal}\|} = \xi(q_{goal} - q).$$

This equation indicates that the magnitude of the attractive force is proportional to the distance between the goal and the vehicle; when the vehicle approaches the goal, $\vec{F}_{att}(q)$ converges to zero, which is reasonable, yet when the vehicle moves away from the goal, $\vec{F}_{att}(q)$ grows without a bound, which could be unstable. To keep $\vec{F}_{att}(q)$ bounded, a hybrid concept can be adopted that replaces $\frac{1}{2}\xi\rho^2(q_{goal})$ with $\xi d\rho(q_{goal})$ for $U_{att}(q)$ when the distance between the vehicle and the goal is greater than the predefined threshold d, meaning

$$\vec{F}_{att}(q) = -\nabla U_{att}(q) = -\xi d\frac{q - q_{goal}}{\|q - q_{goal}\|}.$$

Thus, the complete representations of the attractive potential and force are as follows.

$$U_{att}(q) = \begin{cases} \frac{1}{2}\xi\rho^2(q_{goal}) & \text{if } \rho(q_{goal}) \le d \\ d\xi\rho(q_{goal}) & \text{if } \rho(q_{goal}) > d \end{cases},$$

$$\vec{F}_{att}(q) = \begin{cases} -\xi\|q - q_{goal}\|\frac{q - q_{goal}}{\|q - q_{goal}\|} & \text{if } \|q - q_{goal}\| \le d \\ -\xi d\frac{q - q_{goal}}{\|q - q_{goal}\|} & \text{if } \|q - q_{goal}\| > d \end{cases}.$$

In contrast to the attractive potential, the repulsive potentials are expected to increase as the vehicle moves closer to the obstacles. Consequently, a repulsive potential can be simply structured as

$$U_{rep}(q) = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{\rho(q_{obs})} - \frac{1}{\rho_0}\right)^2 & \text{if } \rho(q_{obs}) \le \rho_0 \\ 0 & \text{if } \rho(q_{obs}) > \rho_0 \end{cases},$$

where η is a scaling factor, $\rho(q_{obs})$ is the distance between the vehicle and the detected obstacle, and ρ_0 is a positive constant standing for the obstacles' influence range. Note that ρ_0 should be properly small (based on specification of safety distance) for the crops, combine and semi-trailer as the grain cart sometimes needs to travel extremely close to them; meanwhile, ρ_0 can be large for other random static and dynamic obstacles that the grain cart should stay safely away from. Similar to the attractive potential, the repulsive potentials are formed in a hybrid fashion so that the influences of the distant obstacles are neglected. Taking gradient of $U_{rep}(q)$ when it is not 0, the repulsive force is obtained as below.

$$\begin{aligned} \vec{F}_{rep}(q) &= -\nabla U_{rep}(q) = -\nabla\frac{1}{2}\eta\left(\frac{1}{\rho(q_{obs})} - \frac{1}{\rho_0}\right)^2 \\ &= \eta\left(\frac{1}{\rho(q_{obs})} - \frac{1}{\rho_0}\right)\frac{1}{\rho^2(q_{obs})}\nabla\rho(q_{obs}). \end{aligned}$$

By analogy with the derivation of $\nabla\rho(q_{goal})$,

$$\nabla\rho(q_{obs}) = \frac{q - q_{obs}}{\|q - q_{obs}\|},$$

where q_{obs} is the position of the obstacle. Thus, $\nabla\rho(q_{obs})$ is a unit vector directed from q_{obs} to q, "pushing" the vehicle away from the obstacles. The complete representation of the repulsive forces is

$$\vec{F}_{rep}(q) = \begin{cases} \eta\left(\frac{1}{\rho(q_{obs})} - \frac{1}{\rho_0}\right)\left(\frac{1}{\rho^2(q_{obs})}\right)\frac{q - q_{obs}}{\|q - q_{obs}\|} & \text{if } \rho(q_{obs}) \le \rho_0 \\ 0 & \text{if } \rho(q_{obs}) > \rho_0 \end{cases}.$$

In this study, the resultant artificial potential force is converted to motion planning via one of the most popular conversion techniques, force to velocity [32], where the magnitude and direction of the resultant force, respectively, determine the desired linear velocity and heading for the grain cart.

2.3.2.3. Formulation of Fuzzy Logic Control

A more detailed description of FL [2][33][34] is presented here, followed by the formulation of the FLC part of the proposed motion planning algorithm. In most cases, traditional planning and control requires a set of model equations derived from physical laws to generate feedback control rules to regulate the behavior of the controlled system. This method works well for simple cases, but struggles as the system becomes complex. Meanwhile, FL is capable of dealing with complex nonlinear processes involving noisy input data. It provides intuitive solutions by drawing conclusions from vague, incomplete or imprecise information. Construction of an FL planner or controller is normally faster than that of a traditional one, and in general its operation requires less memory and computational power.

To use FLC, the behavior of the system to be controlled is described linguistically with ambiguous statements such as very hot, fairly fast and slightly left,

which can be easily dealt with based on fuzzy reasoning with the concept of degrees of membership. As mentioned previously, traditional logic based on classical set theory works with variables whose values are either TRUE or FALSE (i.e., a given element is either a member of a set or not a member of a set). FLC based on fuzzy set theory allows elements to be partial members of a set, and the degree of membership ranges from 0 to 1 (i.e., from no membership to full membership).

Figure 2.8 illustrates how a typical closed-loop FLC process is formulated. In response to actuation, the process to be controlled generates the output, which is sampled and measured by the sensors. This measurement is usually a precise quantity and is first "fuzzified" into a fuzzy quantity that, with different degrees of membership, belongs to a number of fuzzy sets. Following a set of predefined linguistic fuzzy rules, the fuzzy controller then makes decisions based on the fuzzified measurement. The fuzzy decision is also in the form of a number of fuzzy sets (with different degrees of membership) and is eventually converted into a precise quantity with defuzzification methods. Finally, the defuzzified control command is carried out by the actuation system to regulate the behavior of the process, starting a new control cycle.

In this study, the sensors take measurements of the linear velocity and heading error. The FLC part of the proposed planner uses the measurements to calculate the desired steering angle as the planner output, based on which the controller will determine control commands. Last, the actuators will carry out the control to regulate the behavior of the autonomous grain cart.

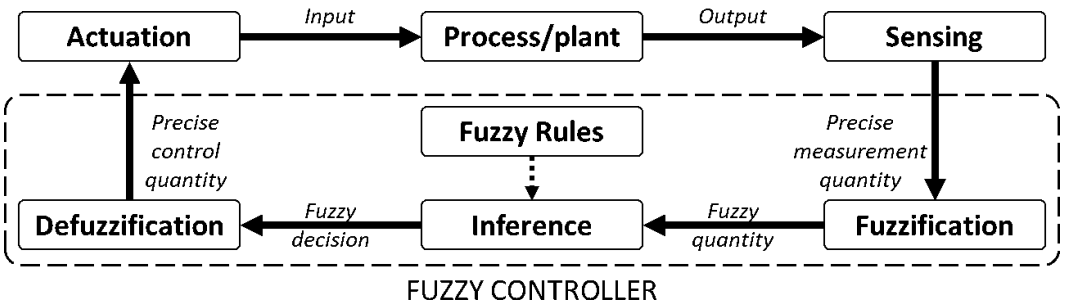


Figure 2.8 Typical closed-loop FLC process.

Fuzzy rules are the core of FLC and leverage human operators' knowledge, experience, and/or intuition. In this study, it makes intuitive sense that the grain cart steers fast to avoid the obstacles efficiently; yet, it should not steer so fast that the stability of the tractor as well as the comfort and safety of the human supervisor in the cab, if any, can be ensured. Considering the complexity and nonlinearity of the grain cart's behavior traveling in the field over unpredictably uneven and slippery terrain, finding the optimal balance between steering efficiency and stability (as well as safety and comfort) is challenging. However, from the human operator's point of view, the solution to this planning problem is simple and clear; the larger the difference between the desired and current headings (i.e., heading error), the greater the steering angle; meanwhile, the higher the linear velocity, the smaller the steering angle.

Compared with the measurement of linear velocity, the determination of heading error is complex (Figure 2.9); when a single obstacle blocks the straight path to the goal, the vehicle should go around it from the closer side (i.e., the side closer to the goal); when multiple obstacles block the straight path to the goal, the vehicle should deal with

them one by one (starting from the closest) from the closer side, exploring the openings between the obstacles.

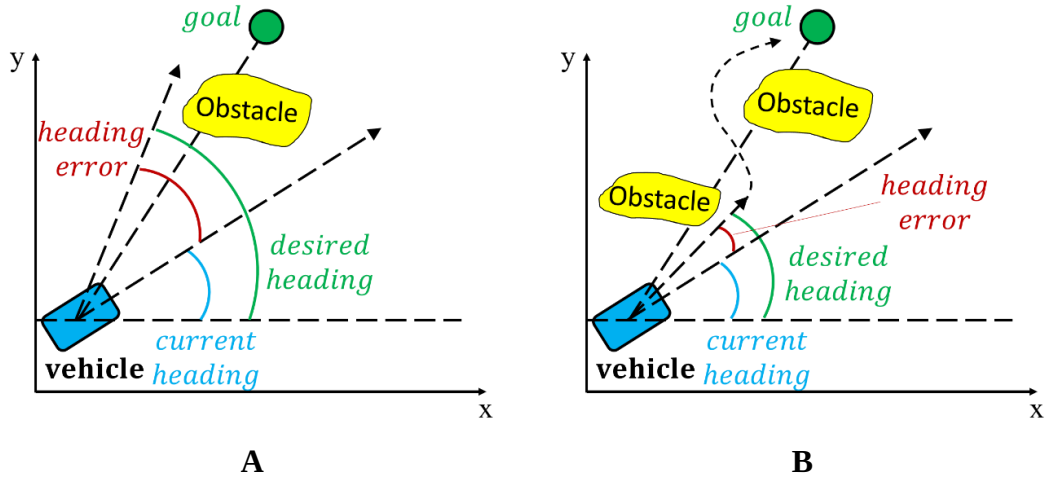


Figure 2.9 Heading error determination. (A) Single obstacle. (B) Multiple obstacles.

Following the FLC process in Figure 2.8, the precise measurements of linear velocity and heading error are first fuzzified into fuzzy quantities. While linear velocity can be described as Slow, Normal and Fast, heading error can be characterized as Far Right, Right, Front, Left, and Far Left, which indicate the direction of the desired heading relative to the current heading (left being positive, right being negative). The membership functions associated with these two measurement quantities are shown in Tables 2.1 and 2.2 and Figures 2.10 and 2.11. Note that different operators with different driving habits and experience have different judgements on the membership values. The following values were determined by the author based on (i) his own driving experience and observation of a real harvest operation, (ii) discussion from online forums on crop harvesting, and (iii) the scaling of the simulation tests.

Table 2.1 Membership values of linear velocity.

	Linear velocity (ft/s)																				
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Slow	1	1	1	1	1	1	1	0.75	0.5	0.25	0	0	0	0	0	0	0	0	0	0	0
Normal	0	0	0	0	0	0	0	0.25	0.5	0.75	1	0.75	0.5	0.25	0	0	0	0	0	0	0
Fast	0	0	0	0	0	0	0	0	0	0	0	0.25	0.5	0.75	1	1	1	1	1	1	1

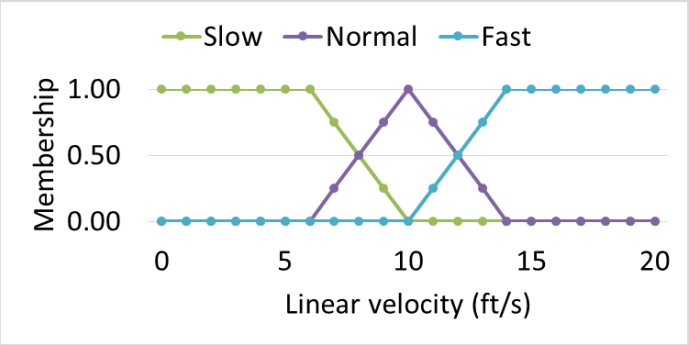


Figure 2.10 Membership functions of linear velocity.

Table 2.2 Membership values of heading error.

	Heading error (rad)																
	$-\frac{8}{32}$	$-\frac{7}{32}$	$-\frac{6}{32}$	$-\frac{5}{32}$	$-\frac{4}{32}$	$-\frac{3}{32}$	$-\frac{2}{32}$	$-\frac{1}{32}$	0	$\frac{1}{32}$	$\frac{2}{32}$	$\frac{3}{32}$	$\frac{4}{32}$	$\frac{5}{32}$	$\frac{6}{32}$	$\frac{7}{32}$	$\frac{8}{32}$
Far Right	1	0.75	0.5	0.25	0	0	0	0	0	0	0	0	0	0	0	0	0
Right	0	0.25	0.5	0.75	1	0.75	0.5	0.25	0	0	0	0	0	0	0	0	0
Front	0	0	0	0	0	0.25	0.5	0.75	1	0.75	0.5	0.25	0	0	0	0	0
Left	0	0	0	0	0	0	0	0	0	0.25	0.5	0.75	1	0.75	0.5	0.25	0
Far Left	0	0	0	0	0	0	0	0	0	0	0	0	0	0.25	0.5	0.75	1

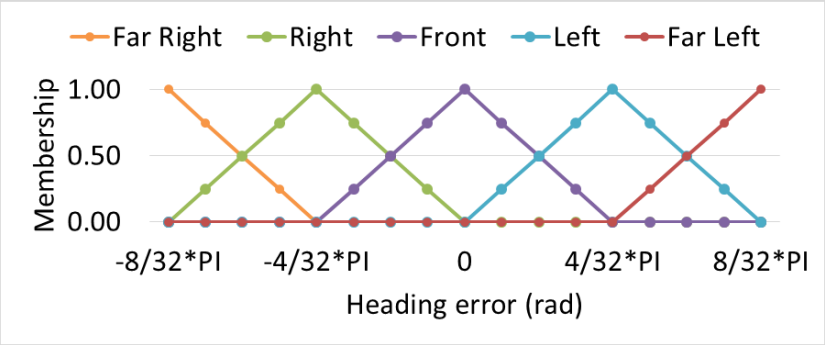


Figure 2.11 Membership functions of heading error.

Before formulating the fuzzy rules, fuzzy sets and membership functions of the fuzzy decision need to be defined. As the output of the FLC part of the planner, the desired steering angle is expressed in terms of Sharp Right, Gentle Right, Straight, Gentle Left, and Sharp Left (left being positive, right being negative), whose membership functions are shown in Table 2.3 and Figure 2.12.

Table 2.3 Membership values of steering angle.

	Steering angle (rad)																
	$-\frac{8}{32}$	$-\frac{7}{32}$	$-\frac{6}{32}$	$-\frac{5}{32}$	$-\frac{4}{32}$	$-\frac{3}{32}$	$-\frac{2}{32}$	$-\frac{1}{32}$	0	$\frac{1}{32}$	$\frac{2}{32}$	$\frac{3}{32}$	$\frac{4}{32}$	$\frac{5}{32}$	$\frac{6}{32}$	$\frac{7}{32}$	$\frac{8}{32}$
Sharp Right	1	0.75	0.5	0.25	0	0	0	0	0	0	0	0	0	0	0	0	0
Gentle Right	0	0.25	0.5	0.75	1	0.75	0.5	0.25	0	0	0	0	0	0	0	0	0
Straight	0	0	0	0	0	0.25	0.5	0.75	1	0.75	0.5	0.25	0	0	0	0	0
Gentle Left	0	0	0	0	0	0	0	0	0	0.25	0.5	0.75	1	0.75	0.5	0.25	0
Sharp Left	0	0	0	0	0	0	0	0	0	0	0	0	0	0.25	0.5	0.75	1

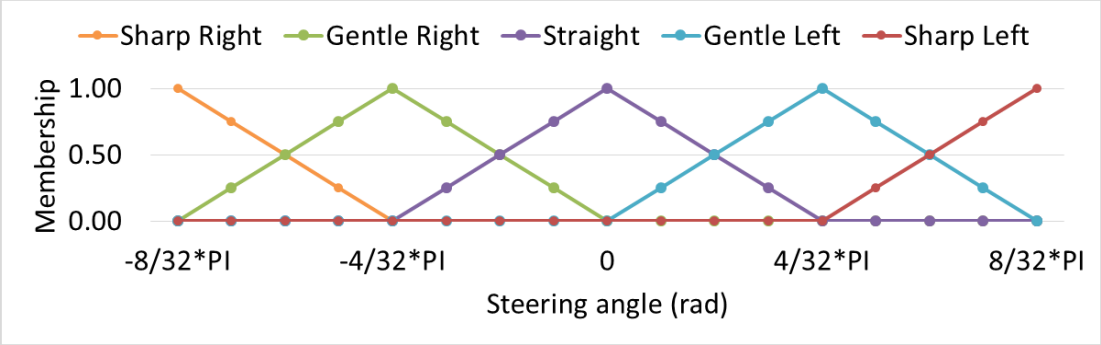


Figure 2.12 Membership functions of steering angle.

Given the fuzzy sets of both the input (heading error and linear velocity) and output (desired steering angle) of the fuzzy planner, the linguistic fuzzy rules are tabulated in Table 2.4.

Table 2.4 Fuzzy rules for steering angle planning.

	Heading error				
Linear velocity	Far Right	Right	Front	Left	Far Left
Slow	Sharp Right	Sharp Right	Straight	Sharp Left	Sharp Left
Normal	Sharp Right	Gentle Right	Straight	Gentle Left	Sharp Left
Fast	Gentle Right	Gentle Right	Straight	Gentle Left	Gentle Left

Finally, the fuzzy steering decision in Table 2.4 can be defuzzified into a precise steering angle in radians and input to the controller. Various defuzzification techniques are available (e.g., max membership principle, centroid method, weighted average method and mean-max membership), among which weighted average method is adopted herein for its computational simplicity [35].

Following the rules in Table 2.4, steering motion plans can be generated given any linear velocity and heading error. For example, if the grain cart travels at 9 ft/s, we know the linear velocity has 0.75 as the degree of membership for being Normal, and 0.25 for Slow; meanwhile, if the heading error is $-\frac{3}{32}pi$ rad, the heading error has 0.25 as the degree of membership for being Front, and 0.75 for Right. According to Fuzzy Set Theory [34], the following inference produces the output steering angle.

IF linear velocity is Normal (0.75) AND heading error is Front (0.25), THEN steering angle is Straight (0.25).

IF linear velocity is Normal (0.75) AND heading error is Right (0.75), THEN steering angle is Gentle Right (0.75).

IF linear velocity is Slow (0.25) AND heading error is Front (0.25), THEN steering angle is Straight (0.25).

IF linear velocity is Slow (0.25) AND heading error is Right (0.75), THEN steering angle is Sharp Right (0.25).

Therefore, the fuzzy decision is Straight (0.25), Gentle Right (0.75) and Sharp Right (0.25), which can be converted to a precise steering angle as output:

$$\frac{0.25 \times 0 + 0.75 \times \left(-\frac{4}{32}\right) + 0.25 \times \left(-\frac{8}{32}\right)}{0.25 + 0.75 + 0.25} = -0.125\ rad$$

2.3.2.4. Integration Mechanism

As described above, APF is based on elegant mathematical concepts and operations that support fast planning in real-time applications, yet simply following the equations could generate irrational or inefficient motion plans. Meanwhile, FLC

leverages human intelligence that can take into account the rationality and efficiency of the plan, complementing APF. That being said, some cases are so intractable that formulating fuzzy rules becomes complex and impractical. For example, dynamic obstacles in a farm field may move around randomly, interfering with the grain cart's motion, in which case APF's attractive and repulsive forces can provide more direct guidance to the grain cart. Integrating APF and FLC is a hybrid approach that is expected to exploit the merits of both techniques while overcoming their limitations. Figure 2.13 illustrates how APF and FLC are integrated. The attractive forces of APF always decide the heading towards the goal. When the grain cart is safely away from obstacles, FLC applies simple and effective human-intelligence-based rules to generate collision-free motion plans that account for the GLOBAL optimality of the motion (i.e., total travel distance). When the grain cart gets dangerously close to the obstacles that FLC has not handled (e.g., closely surrounding static and/or fast approaching dynamic obstacles), the repulsive forces of APF enable prompt avoidance of these LOCAL obstacles. As discussed previously, in parallel with the planning for steering angle, APF is also responsible for determining the linear velocity. One may argue that properly adjusting the influence ranges of the goal and obstacles can upgrade the APF algorithm, but developing explicit rules to adapt the APF to the agricultural environment that is largely unconstructed and unpredictable can be a great challenge.

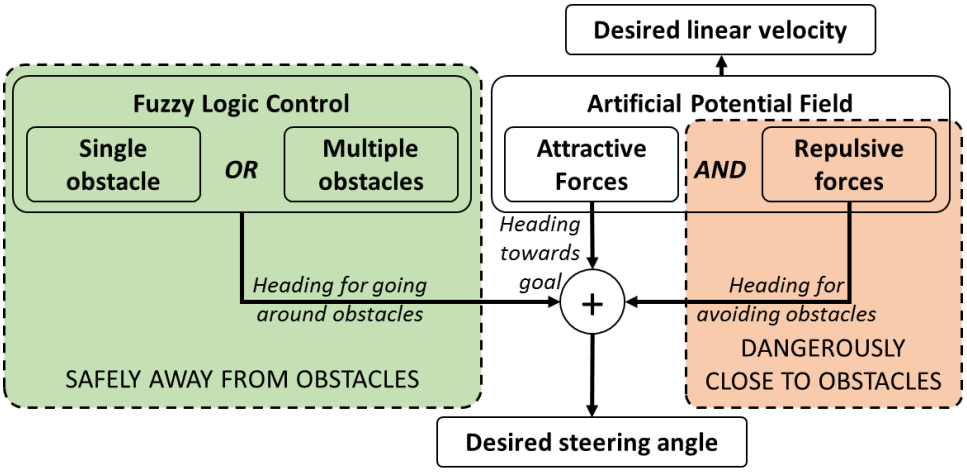


Figure 2.13 Integration mechanism of Artificial Potential Field and Fuzzy Logic Control.

2.3.2.5. Task Scheduling

As previously described, the grain cart should ideally meet the combine exactly when the combine reaches its capacity, a difficult and often impractical challenge. Therefore, to guarantee that harvesting is not interrupted, a grain cart typically arrives early and follows the combine before unloading starts. To reduce this non-productive following time, a special task scheduling strategy was proposed for grain carts: after transferring the grain to the semi-trailer, a grain cart should neither stay still waiting for the combine's unloading request nor go and follow the combine until unloading starts; instead, the grain cart should drive to the boundary of the working area where the combine is harvesting, then stand by in a pose from which a clear path to the combine is available. "Clear" simply implies that between the grain cart and the combine, there are no blocking obstacles (not even unharvested crop rows), allowing the grain cart to approach the combine via a straight route (additional maneuvers might be needed for

heading alignment). Thus, a motion plan that meets the temporal requirement can be easily computed based on the relative poses and velocities of the two vehicles as well as the fill level of the combine transmitted to the cart via wireless communication. Following this strategy, a full work cycle for grain carts in harvest operation is illustrated in Figure 2.14. It can be seen that, in addition to unloading grain from the combine, the logistical tasks in harvest operations for grain carts include go to semi-trailer, transfer grain to semi-trailer, return to standby point, standby and go to combine. To sequence and schedule these tasks, the following constraints and triggers, expressed in pseudo code, are utilized.

(i) Go to combine:

IF standing by AND combine harvesting in field AND estimated meeting point in field AND combine will be full by arrival of grain cart

THEN go to combine

(ii) Unload grain from combine

IF close to combine AND aligned with combine

THEN drive alongside combine in sync

(when to start and finish unloading grain is determined by combine)

(iii) Go to semi-trailer:

IF unloading finished OR grain cart full

THEN go to semi-trailer

(iv) Transfer grain to semi-trailer:

IF going to semi-trailer AND close to semi-trailer AND aligned with semi-trailer

THEN transfer grain to semi-trailer

(v) Go to standby point:

IF transferring grain AND grain cart empty

THEN go to standby point

(vi) Stand by

IF going to standby point AND close to standby point AND facing specified direction

THEN stand by

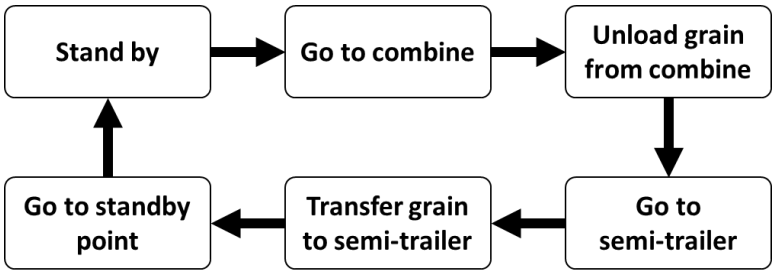


Figure 2.14 Full work cycle of autonomous grain carts in harvest operations.

2.4. Control Technique

2.4.1. Grain Cart Modeling

As stated previously, one of the objectives of this study is to investigate the performance of the proposed motion planning algorithm using simulation tests, in which a mathematical model that represents grain carts' motion dynamics is required. Since the grain cart (i.e., tractor and trailer) model has been widely used in many studies, this project does not spend effort on developing new models. The following introduction and derivation are based on [36] and [37].

Vehicle dynamics refers to a vehicle's responses (position, velocity and/or acceleration) to various control inputs. Theoretically the responses in all possible degrees of freedom should be considered, but the scope of this study has been limited to a plain field (i.e., no significant slopes or curves), constraining the vehicle's motions in a 2D plane.

Two types of vehicle models have been commonly adopted for controller development, namely dynamic models and kinematic models. While a dynamic model uses tire lateral forces to excite the lateral and yaw dynamics of the vehicle and applies Newton's second law of motion to construct the force and moment balance, a kinematic model assumes that the vehicle's tires travel only in the direction they face (i.e., no slip), and the vehicle's responses to steering inputs are determined by only the geometric parameters, which compromises the fidelity of transient behaviors of the vehicle in favor of simplicity. Noting that this study focuses on motion planner development for a low-speed tractor trailer, the relatively simple and mathematically elegant kinematic model was employed for the controller design.

As a typical example of off-road vehicles, a tractor can be simplified and represented by a so-called bicycle model with just one tire each on front and rear axles (Figure 2.15), neglecting the load transfer between right and left wheels.

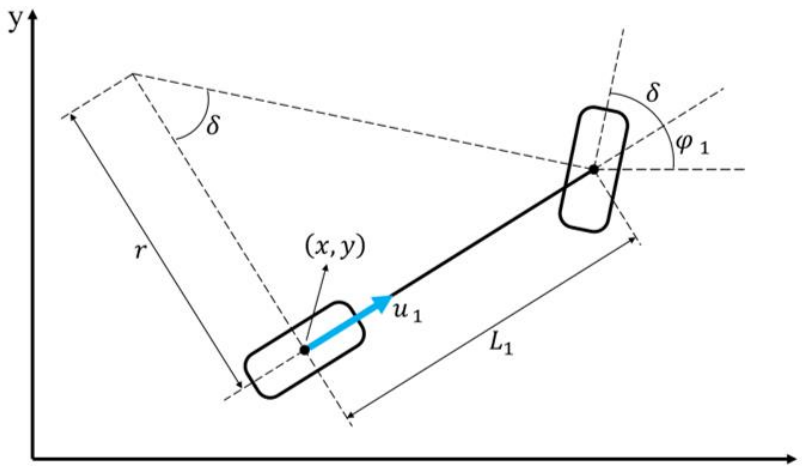


Figure 2.15 Bicycle model.

As shown in Figure 2.15, the geometric relationship between the variables is simple and straightforward, giving the velocity of the center of the tractor's rear axle as

$$\begin{cases} \dot{x} = u_1 \cos \varphi_1 \\ \dot{y} = u_1 \sin \varphi_1 \end{cases},$$

where u_1 is the tractor's linear velocity and φ_1 is the tractor's heading angle (i.e., the angle between the x-axis and the vehicle's longitudinal axis).

Also, from Figure 2.15, with the counterclockwise direction being positive, the yaw rate (i.e., the tractor's angular velocity) is given by

$$\dot{\varphi}_1 = \frac{u_1}{r},$$

where r is the turning radius of the tractor. And with

$$\tan \delta = \frac{L_1}{r},$$

the yaw rate can be expressed as

$$\dot{\varphi}_1 = \frac{u_1}{L_1}\tan\delta,$$

where δ is the steering angle of the front wheel (i.e., the angle between headings of the tractor body and front wheel), and L_1 is the wheelbase (i.e., the distance between the front and rear axles). Thus, the kinematic model of the tractor can be represented by the following set of equations.

$$\begin{cases} \dot{x} = u_1 \cos\varphi_1 \\ \dot{y} = u_1 \sin\varphi_1 \\ \dot{\varphi}_1 = \dfrac{u_1}{L_1}\tan\delta \end{cases}.$$

As shown in Figure 2.16, linking the tractor and a trailer with a revolute joint at the hitch point (circled in Figure 2.17), the velocities of the tractor and trailer are coupled to each other. In Figure 2.16, φ_2 and λ are, respectively, heading angle of the trailer (i.e., the angle between the x-axis and the trailer's longitudinal axis) and the difference between the heading angles of the tractor and trailer.

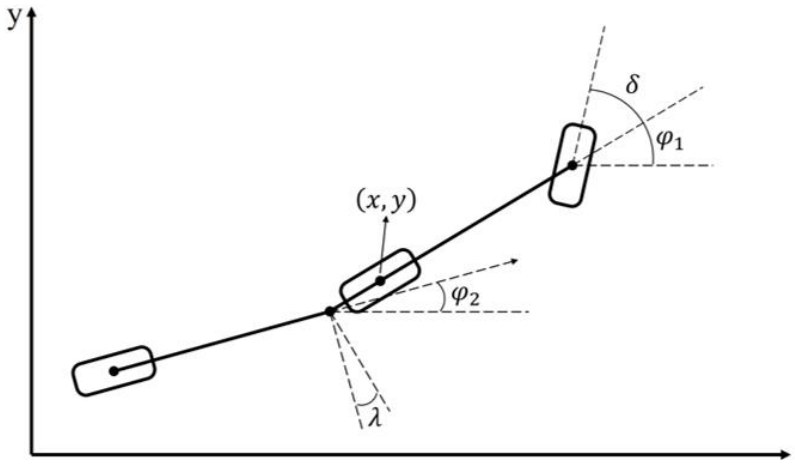


Figure 2.16 Tractor trailer model with key angles.

Figure 2.17 Hitch points on grain carts. Reprinted from [38].

As illustrated in Figure 2.18, yaw motions of the tractor are centered at the rear axle, giving the hitch point a lateral velocity in the direction opposite to that of the tractor's center of gravity (CoG). Denoted by θ, the angle between the resultant velocity and the longitudinal velocity of the hitch point can be used along with $r = \frac{L_1}{\tan\delta}$ to derive the lateral velocity of the hitch point as follows.

$$v_1 = u_1 \tan\theta = u_1 \frac{L_2}{r} = \frac{u_1 L_2}{L_1} \tan\delta,$$

where L_2 is the distance between the center of the rear axle and the hitch point. With a rigid body, the tractor has the same longitudinal velocity u_1 at the hitch point and the center of rear axle.

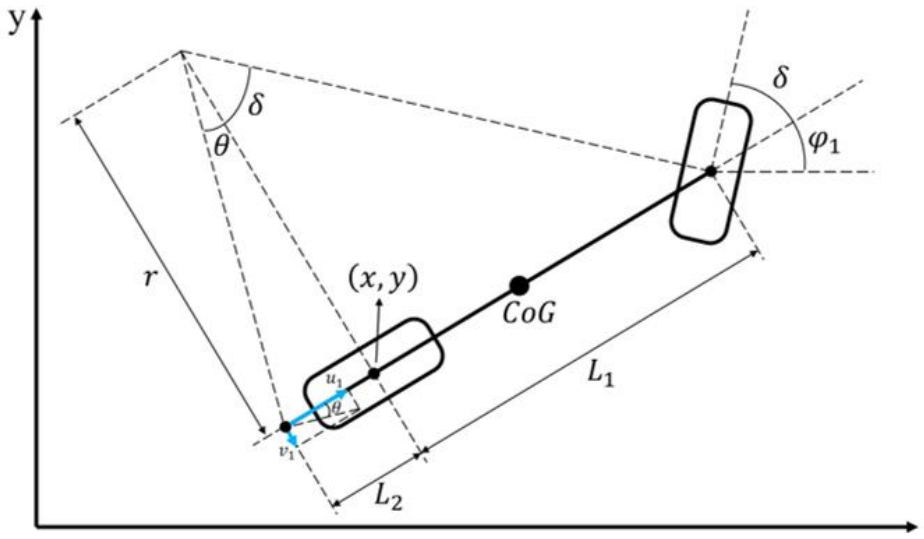


Figure 2.18 Tractor model with hitch point.

As shown in Figure 2.19, taking into account the angle between the tractor heading and trailer heading, denoted by $\lambda = \varphi_1 - \varphi_2$, the longitudinal and lateral velocities of the trailer at the hitch point are given by

$$\begin{cases} u_2 = v_1 \sin\lambda + u_1 \cos\lambda \\ v_2 = v_1 \cos\lambda - u_1 \sin\lambda' \end{cases}$$

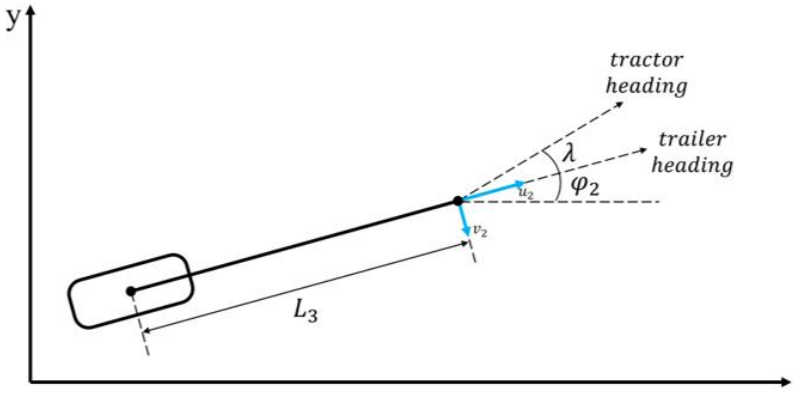


Figure 2.19 Trailer model.

Finally, the above expression for the lateral velocity of the hitch point is used to define yaw rate of the trailer as (again, with the counterclockwise direction being positive)

$$-\dot{\varphi}_2 = \frac{v_2}{L_3} = \frac{1}{L_3}(v_1 \cos\lambda - u_1 \sin\lambda) = \frac{u_1 L_2}{L_1 L_3}\tan\delta\cos\lambda - \frac{u_1}{L_3}\sin\lambda.$$

Adding this equation to the equation set of the tractor's kinematic model, the dominant motion dynamics of the entire tractor trailer system can be represented by the following set of equations.

$$\begin{cases} \dot{x} = u_1 \cos\varphi_1 \\ \dot{y} = u_1 \sin\varphi_1 \\ \dot{\varphi}_1 = \dfrac{u_1}{L_1}\tan\delta \\ \dot{\varphi}_2 = \dfrac{u_1}{L_3}\sin\lambda - \dfrac{u_1 L_2}{L_1 L_3}\tan\delta\cos\lambda \end{cases}.$$

2.4.2. Controller Design

To implement the motion plans generated by the motion planner to regulate the behavior of the grain cart, Proportional, Integral and Derivative (PID) control, a very commonly used feedback control technique, has been adopted in this study. A brief description of PID control technique is presented below [39][40][41].

PID has played a dominant role in process control since the inception of automation because it is simple, efficient, and generally applicable to most control systems. PID has the potential to provide optimal control solutions for vehicle motion control, which is characterized by nonlinear dynamics, unmeasured disturbances, noise, measurement delays and lags. Studies have shown that PID can efficiently solve both

regulator problems (i.e., rejection of disturbances) and servo problems (i.e., setpoint response).

As illustrated in Figure 2.20, all three components of a PID controller work together to produce the control output based on the measured error; i.e., the difference between the desired and actual states of the system. Ideally, when the three PID components function and cooperate properly, any variation in error caused by setpoint change or process disturbance can be quickly eliminated.

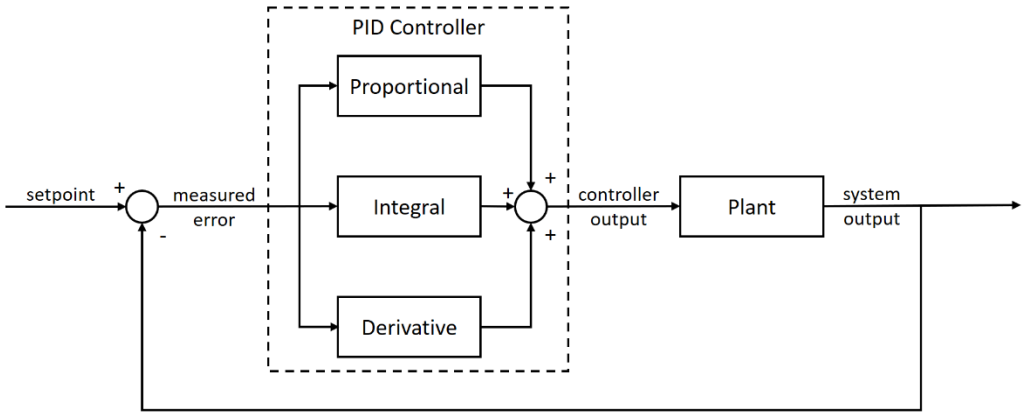


Figure 2.20 PID controller in a control system.

Proportional is the simplest component, whose output is the product of the proportional gain K_P and measured error $e(t)$. Therefore, larger K_P or $e(t)$ leads to larger control actions. While a large K_P normally provides fast and sufficient responses to errors, it also causes the controller to repeatedly overshoot the setpoint, making the system oscillate and become unstable. A Proportional-only control loop has a major drawback: when the error becomes so small that the control action generated by K_P is

negligible, the error is sustained even though the system has reached steady state; in such case, the error becomes an offset.

To address the offset, the Integral component stores all measured errors and takes actions accordingly. Specifically, even though $e(t)$, the error at each instant, is so small that the Proportional component is not effective, the Integral component keeps collecting the errors until they are large enough to be significant. As errors can be positive or negative, sometimes errors fill the Integral storage (i.e., positive added to positive or negative added to negative), and sometimes they empty the storage (i.e., errors with opposite signs cancel out). The storage stays nearly empty when the Integral factor functions properly. Effective for eliminating steady-state errors though, the Integral component makes a considerable contribution to overshoot.

The Derivative component deals with the change rate of the error. The faster the error changes, the larger the control actions the Derivative component produces, counteracting the overshoot that can be caused by P and I. To be more specific, when the error is large, P and I generate large control actions, making the error change quickly, which leads D to counteract the actions more aggressively in favor of overshoot reduction.

In summary, the PID controller deals with the value of the current error, the cumulative errors over a period of time, and the change rate of the error, then determines how much correction to apply. The controller keeps updating the computation and applying the correction until the controlled system has reached the setpoint.

Designing a PID controller essentially consists of the process of tuning the three control parameters (i.e., the P, I and D gains) in pursuit of optimal control performance, which refers to rise time, overshoot, settling time and steady-state error. Before 1942, tuning was performed through trial and error, but in that year, the Ziegler-Nichols method was published, which has been a popular heuristic PID tuning technique in the last few decades.

The Ziegler-Nichols method first sets the I and D gains to zero, then increases K_P from zero until it reaches the ultimate gain K_U, at which time the controlled system has stable and consistent oscillations with the oscillation period T_U. T_U is then adopted to update the P, I, and D gains based on the rules in Table 2.5.

Table 2.5 PID tuning rules by Ziegler-Nichols method.

	K_P	T_I	T_D
Classic PID	$0.6K_U$	$T_U/2$	$T_U/8$
Pessen Integral Rule	$0.7K_U$	$T_U/2.5$	$3T_U/20$
Some overshoot	$0.33K_U$	$T_U/2$	$T_U/3$
No overshoot	$0.2K_U$	$T_U/2$	$T_U/3$

where K_P, T_I and T_D are used to establish the control action $u(t)$ from the measured error $e(t)$ in the form of

$$u(t) = K_P\left(e(t) + \frac{1}{T_I}\int_0^t e(\tau)\,d\tau + T_D\frac{de(t)}{dt}\right)$$

To clarify, the Ziegler-Nichols method is not the optimal solution to all control applications because different criteria are adopted to evaluate the control performance. For example, some may desire fast responses while others may be strict on overshoot. As a more general reference, the rules of thumb for tuning PID controllers are given in Table 2.6.

Table 2.6 General rules for tuning PID controllers.

	Rise time	Overshoot	Settling time	Steady-state error	Stability
Increase K_P	Decrease	Increase	Minor change	Decrease	Degrade
Increase $K_I = \frac{K_P}{T_I}$	Decrease	Increase	Increase	Decrease	Degrade
Increase $K_D = K_P T_D$	Minor change	Decrease	Decrease	No change	Improve if K_D small

In Table 2.6, rise time is the amount of time the controlled system takes to go from 10% to 90% of the steady-state (or final value). Settling time is the time required for the controlled process variable to settle within a certain deviation range (commonly 5%) from the final value.

In this study, the motion planner provides the control setpoint, the desired linear velocity and steering angle. The sensors measure the vehicle's actual linear velocity and steering angle. The differences between the desired and actual values are the errors that

are input to the PID controller, which performs the computation and outputs the steering rate and acceleration or deceleration to regulate the behavior of the grain cart.

2.5. Actuation Plan

Most traditional drive trains are mechanical and provide the operator with direct, physical control over the speed or direction of the vehicle, which is not always practical to deliver actuation functions in autonomous driving. In contrast, drive-by-wire powertrains can support the automation of farming vehicles because they are primarily composed of servomotors or electromechanical actuators that are directly controlled through electric wires instead of mechanical connections. Drive-by-wire functionalities like steer-by-wire, throttle-by-wire, and brake-by-wire are already available from major farm equipment manufacturers like John Deere, CNH, AGCO, CLASS and Kubota [42]. In fact, the increasing trend toward AAVs is making agriculture tractors one of the fastest growing among the off-highway vehicle segment in the drive-by-wire market [43].

2.6. Experimental Procedure

To validate the design of the planning algorithm and the associated navigation solution, three sets of experiments were carried out: first two in a virtual environment with MatLab Simulink simulations, and another one in the real world with mobile robots. The simulation tests were for verification of the effectiveness, robustness, efficiency and computational ease of the proposed motion planning algorithm and the task scheduling strategy. The mobile robot tests were for verification of the effectiveness

and practicality of the navigation solution, and further verification of the effectiveness, robustness and efficiency of the motion planning algorithm.

2.6.1. Simulation Tests: APF+FLC vs. Simple APF

2.6.1.1. Experimental Setup

This set of simulation tests features the comparison between the proposed APF+FLC planner and a simple APF planner. Running on an Intel Core i7-8650U CPU @ 1.90 GHz, MatLab Simulink R2019a was employed as the simulation platform. To imitate a real-world crop harvest, a 2D virtual harvest operation was modeled, executed (at 50 Hz) and plotted (Figure 2.21): a semi-trailer (in pink) waited nearby, a trapezoidal crop field (in orange) was harvested row by row by a combine (in green) following a specific route; a grain cart (in blue) unloaded and transferred the grain, commuting between the semi-trailer and the combine; real-time fill levels of the combine and grain cart were shown on the bottom left of the plot. The simulated harvest was downscaled from real harvest operations in terms of crop row lengths and combine capacity. The goal was to use shorter times and smaller spaces to showcase the effectiveness of the proposed motion planning algorithm in different scenarios that could be encountered by grain carts in harvest operations.

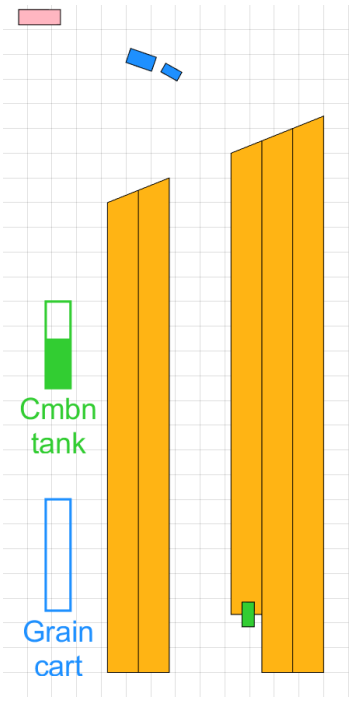


Figure 2.21 2D virtual harvest operation in MatLab Simulink.

2.6.1.2. Field Data

A set of yield monitor data from a corn field in Minnesota were incorporated into the simulation tests. The dataset contained all the information collected by a combine throughout a harvest operation, including the latitude-longitude coordinates, the crop flow in mass and volume, the yield mass and volume (each dry and wet), the moisture, the combine's speed, etc. For the simulation tests, the crop flow and combine speed from the dataset were processed and utilized to validate the specification of the fill rate and speed of the simulated combine. The data processing procedure mainly consisted of filtering out the data collected out of crop rows (e.g., during end-of-row turns) and modeling the selected data. Because the simulation tests were downscaled from real harvest operations, instead of the actual mean and standard deviation, the coefficients of

variation of the crop flow and combine speed were employed, which reflect the extent of variability in relation to the mean. The coefficients of variation of the crop flow and combine speed were 5.61% and 3.24%, respectively.

2.6.1.3. Test Design

The following simulation tests were designed and conducted: (i) simple harvesting with no obstacles other than unharvested crop rows; (ii) harvesting with static obstacles located between the crop rows and the semi-trailer; (iii) harvesting with dynamic obstacles moving around between the crop rows and the semi-trailer; and (iv) the same tests as the above but with a simple APF planner guiding the grain cart for comparison.

2.6.1.3.1. Simple Harvesting

As illustrated in Figure 2.14, a simple harvest operation in which unharvested crop rows are the only obstacles can be broken down into the following tasks for an autonomous grain cart: (i) stand by when the combine is harvesting; (ii) navigate through the field to meet the combine when unloading is needed; (iii) keep the same velocity travelling alongside the combine while the grain is being unloaded; (iv) navigate to the semi-trailer; (v) transfer the grain to the semi-trailer; (vi) return to the standby point. In simple harvesting, the grain cart was expected to follow the motion plans generated by the proposed planner and autonomously accomplish all the tasks above.

2.6.1.3.2. Static Obstacles

As mentioned previously, various static obstacles can be present in the harvesting environment, blocking the straight route between the grain cart and the goal. The grain

cart was expected to navigate around the obstacles and accomplish the logistical activities. No obstacles were located inside the crop rows, but six square static obstacles were placed between the crop rows and the semi-trailer. For a more rigorous test of robustness, the configuration of the obstacles varied every work cycle.

2.6.1.3.3. Dynamic Obstacles

Compared to static obstacles, the presence of dynamic obstacles makes autonomous navigation of the grain cart more challenging. Ideally, the grain cart can anticipate all potential collisions, avoid them and eventually find its way to the goal. This simulation involved the wandering motion of two rectangular obstacles (one circling, one back and forth) between the crop rows and the semi-trailer, intermittently impeding the grain cart.

2.6.1.3.4. Simple APF Tests

Another group of three simulation tests like those above (i.e., simple harvesting, static obstacles and dynamic obstacles) were conducted with a simple APF planner guiding the grain cart. Data from both groups of tests were collected and compared for the evaluation of the efficiency of the proposed planning algorithm.

2.6.1.4. Data Collection and Results Analysis

2.6.1.4.1. Effectiveness and Robustness

The first aspect considered in the results was whether the proposed planner and the associated task scheduling strategy successfully directed the grain cart to autonomously accomplish the logistical tasks in the harvest operations. Accomplishment of all the logistical tasks in simple harvesting and harvesting featuring static and

dynamic obstacles (in addition to unharvested crop rows) would verify the effectiveness as well as the robustness of the proposed motion planning algorithm and the task scheduling strategy. This analysis provided a qualitative result.

As mentioned previously, to guarantee uninterrupted harvesting, a grain cart typically meets the combine early and follows the combine before unloading starts. The focus of the proposed task scheduling strategy is on improving this meeting timing by reducing the non-productive following time, which is measured from the moment the grain cart reaches the left side of the combine until the unloading starts. The less the following time, the more effective the task scheduling strategy. This analysis provided a quantitative result. No comparison between the proposed planner and simple APF planner was conducted herein because meeting timing was exclusively dependent on the design of the task scheduling strategy, which was not involved in the simple APF at all. In addition, when random static or dynamic obstacles were present, they interfered with the grain cart in unpredictable ways, hindering it from performing the tasks on time. Therefore, meeting timing was not considered for these cases.

2.6.1.4.2. Efficiency

In this study, efficiency is the most important indicator for evaluating the performance of the proposed motion planning algorithm. The evaluation was based on the comparison between the proposed planner and the simple APF planner in the length and smoothness of the grain cart's trajectory. In simple harvesting, the comparison ran through the entire harvest operation. In the harvest cases involving static and dynamic obstacles, the comparison was focused on how differently the two planners dealt with

the static or dynamic obstacles between the crop rows and the semi-trailer. Therefore, only when the grain cart was out of the crop rows, either going to the semi-trailer or returning to the standby point, the trajectories were measured and compared; the trajectories of the grain cart when it was going to the combine, unloading and going to the semi-trailer but still between the crop rows, were not included in the comparison. In these comparisons, the shorter and smoother the trajectory, the higher the efficiency, as a shorter trajectory required less consumption of fuel and time, and a smoother trajectory improved traveling stability and comfort. To measure the trajectory length, coordinates of the grain cart in each simulation step were logged. Since the simulation was discrete with very small time steps (20 ms), the curves the grain cart traveled within each simulation step were approximated as straight lines. Therefore, given the coordinates of the grain cart in each step, the lengths of the small segments of the trajectory were easily calculated, and their summation would be the approximate length of the entire trajectory. To measure the smoothness of the trajectory, the following formula was used.

$$\kappa' = \frac{1}{n}\sum_{i=2}^{n} \alpha_i^2,$$

where κ' represents the smoothness, α_i is the angle between two consecutive segments of the trajectory, and n is the total number of segments the trajectory consists of. Assuming the mean of α_i is zero, κ' is the variance of α_i with a unit of rad^2. A smaller κ' indicates smaller angles between consecutive segments of the trajectory in general, meaning the entire trajectory is smoother. Measurements of both length and smoothness of the trajectory provided quantitative results.

2.6.1.4.3. Computational Expense

In each simulation step, the motion planner took states of the grain cart and the combine, performed a series of computations and generated the desired motion plan. The time this process consumed in each simulation step was additional important data to collect. A short time implied computational efficiency, while a long time indicated a heavier burden for the processor. With the MatLab function *cputime*, CPU time consumed by a piece of code in each simulation step was easily measured. These data provided a quantitative result. Computational efficiency has been discussed in many studies [11][13][14] as a key advantage of APF over other techniques, demonstrating that APF is suitable for real-time applications. Since the proposed motion planning algorithm incorporated FLC into APF, it was expected to be more computationally expensive than the simple APF.

2.6.2. Simulation Tests: APF+FLC vs. VFH

2.6.2.1. Experimental Setup

This set of simulation tests compared the proposed APF+FLC planner with a Vector-Field-Histogram (VFH) planner. As one of the most popular real-time motion planning algorithms in mobile robotics, VFH uses a statistical representation of the surrounding obstacles. Specifically, VFH (i) uses data from the range sensors to construct and update in real time a 2D Cartesian histogram grid, (ii) reduces the histogram to a 1D polar histogram with different obstacle densities in the histogram sectors, and (iii) selects consecutive sectors with a polar obstacle density below a predefined threshold and in proximity to the target direction. The center of the selected

sectors will be the desired steering angle. The linear velocity is planned based on the obstacle density in front of the vehicle and the current steering rate of the vehicle. The higher the obstacle density and the steering rate, the lower the linear velocity. The VFH algorithm has proven to be computationally efficient and robust, yet it is more powerful for local collision avoidance and global path optimality is not guaranteed.

On the same simulation platform (i.e., MatLab Simulink R2019a on Intel Core i7-8650U CPU @ 1.90 GHz), five different test cases were modeled, executed (at 50 Hz) and plotted. Figure 2.22 is an example: a grain cart (in blue) started traveling from the bottom left of the figure while multiple obstacles (in red) lay between the grain cart and the goal point (blue circle) at the top right of the figure.

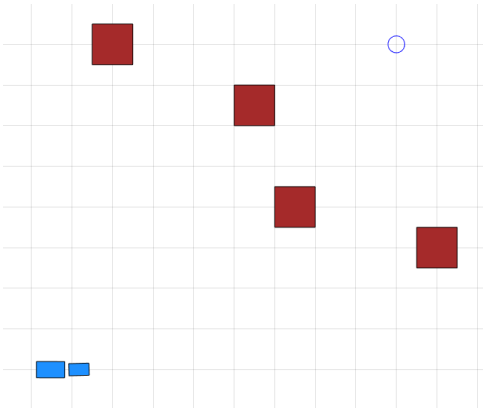


Figure 2.22 Simulation test case: closely spaced static obstacles in the middle.

2.6.2.2. Test Design

The following simulation tests were designed and conducted with the APF+FLC planner: (i) long static obstacle with a narrow gap for the grain cart to go through; (ii) multiple static obstacles close to the goal; (iii) closely spaced static obstacles with a

narrow gap in between for the grain cart to go through; (iv) sparsely spaced static obstacle groups for the grain cart to go around one by one; (v) two dynamic obstacles that moved around potentially blocking the grain cart; and (vi) the same tests as (i) through (v) above but with a VFH planner guiding the grain cart for comparison. Note that key parameters of the VFH planner, namely the histogram thresholds and cost function weights, were tuned to potentially improve the performance, and the best results were collected.

2.6.2.2.1. Long Static Obstacle with Gap

Four static rectangular obstacles formed a long static obstacle (Figure 2.23), resembling a crop row, the most common obstacle a grain cart faces in harvest operations. A narrow gap in the obstacle provided a shortcut towards the goal. The grain cart was expected to take a more efficient route by making use of the gap, rather than going all the way around the entire long obstacle.

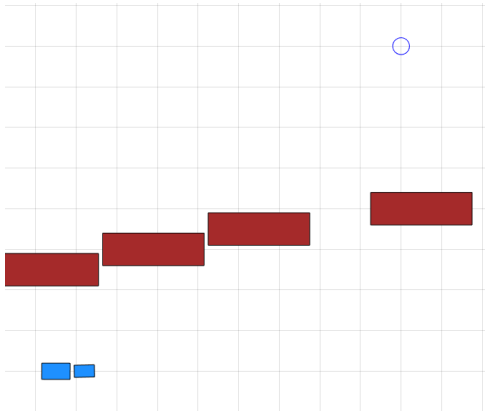


Figure 2.23 Simulation test case: long static obstacle with gap.

2.6.2.2.2. Static Obstacles Close to Goal

In harvest operations, a grain cart is sometimes required to approach a goal that is in close proximity to obstacles. For example, when driving in synchronization with the combine unloading the grain, and when transferring the grain to the semi-trailer that is parked by the roadside. Four static square obstacles were placed around the goal, yet none of them was actually blocking the straight path of the grain cart from the origin (Figure 2.24). The grain cart was expected to go directly towards the goal, ignoring the surrounding obstacles.

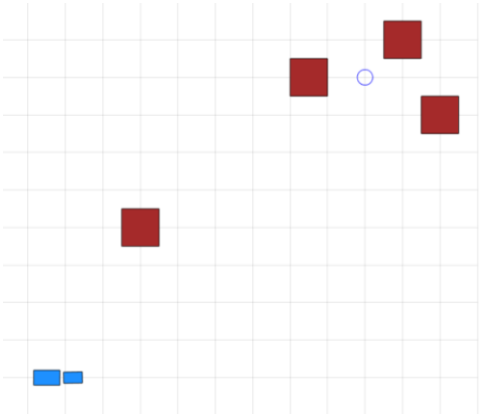


Figure 2.24 Simulation test case: multiple static obstacles close to goal.

2.6.2.2.3. Closely Spaced Static Obstacles

Four static square obstacles were placed between the origin of the grain cart and the goal (Figure 2.22). Overall, the four obstacles were evenly spaced. The most efficient route from the origin to the goal was between the two obstacles in the middle. However, the wider gaps on the two sides could be misleading. In harvest operations, a grain cart

sometimes needs to pass through narrow gaps, for example, when traveling between the unharvested crop rows on the two sides to meet the combine.

2.6.2.2.4. Sparsely Spaced Static Obstacle Groups

Two groups of static square obstacles were located on the grain cart's way towards the goal (Figure 2.25). The two obstacle groups created a wide gap in between, representing sparsely spaced random obstacles in the crop field. The grain cart was expected to identify the large space between the obstacle groups, make use of it and reach the goal. The locations of the two obstacle groups relative to the grain cart's origin were misleading in the sense that the wide gap could not be directly identified at the beginning of the run.

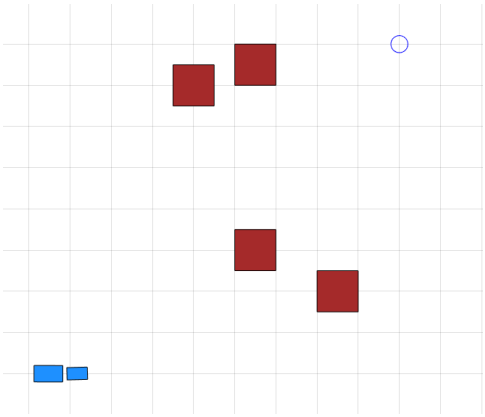


Figure 2.25 Simulation test case: sparsely spaced obstacle groups.

2.6.2.2.5. Dynamic Obstacles

The existence of dynamic obstacles can often make a motion planning task more complex and challenging. Two square obstacles moved around between the origin and the goal, potentially blocking the grain cart (Figure 2.26). While one obstacle moved

back and forth, the other one went in a circle, revolving around the goal counterclockwise. The grain cart was expected to take an efficient route towards the goal while making fast reactions to avoid collisions with the dynamic obstacles.

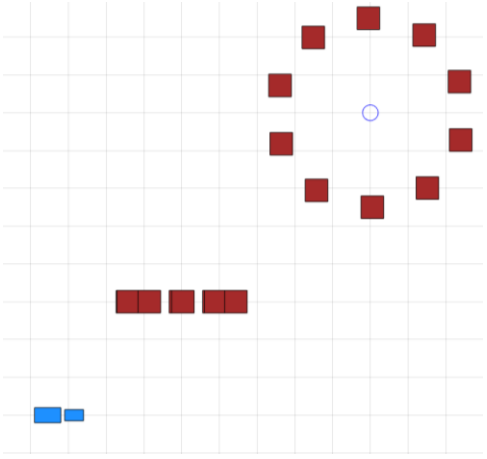


Figure 2.26 Simulation test case: dynamic obstacles.

2.6.2.2.6. VFH Tests

Another group of five simulation tests like those above were conducted with a VFH planner guiding the grain cart. Data from both groups of tests were collected and compared to evaluate the efficiency of the motion plans generated by the proposed planning algorithm.

2.6.2.3. Data Collection and Results Analysis

2.6.2.3.1. Effectiveness and Robustness

The first aspect considered in the results was whether the proposed planner successfully directed the grain cart to accomplish the navigation tasks involving different configurations of static or dynamic obstacles. Accomplishment of the navigation tasks in

all the different test cases would verify the effectiveness as well as the robustness of the proposed motion planning algorithm. This analysis provided a qualitative result.

2.6.2.3.2. Efficiency

Just like the first set of simulation tests (i.e., APF+FLC vs. simple APF), efficiency is the most important indicator of performance. The evaluation was based on the comparison between the proposed planner and the VFH planner in the length and smoothness of the grain cart's trajectory. The comparison was focused on how differently the two planners dealt with the static or dynamic obstacles between the origin and the goal. Again, the shorter and smoother the trajectory, the higher the efficiency. The procedures for measuring the length and smoothness have been discussed above. The efficiency analysis provided a quantitative result.

2.6.2.3.3. Computational Expense

The average CPU time consumed by the proposed planner and the VFH planner in each simulation step was calculated. Given the same navigation tasks, the planner that took longer to process the sensing information and generate collision-free motion plans would be considered more expensive in the computations. Again, the function *cputime* was adopted for measuring the CPU time in each simulation step. These data provided a quantitative result.

2.6.3. Mobile Robot Tests

2.6.3.1. Experimental Setup

The mobile robot tests consisted of harvest operations and general collision avoidance tasks. The harvest operation tests included two autonomous robots, one

representing a grain cart and one representing a combine. As shown in Figures 2.27 and 2.28, different devices were mounted on the robots. A detailed description follows.

Figure 2.27 Robot representing grain cart.

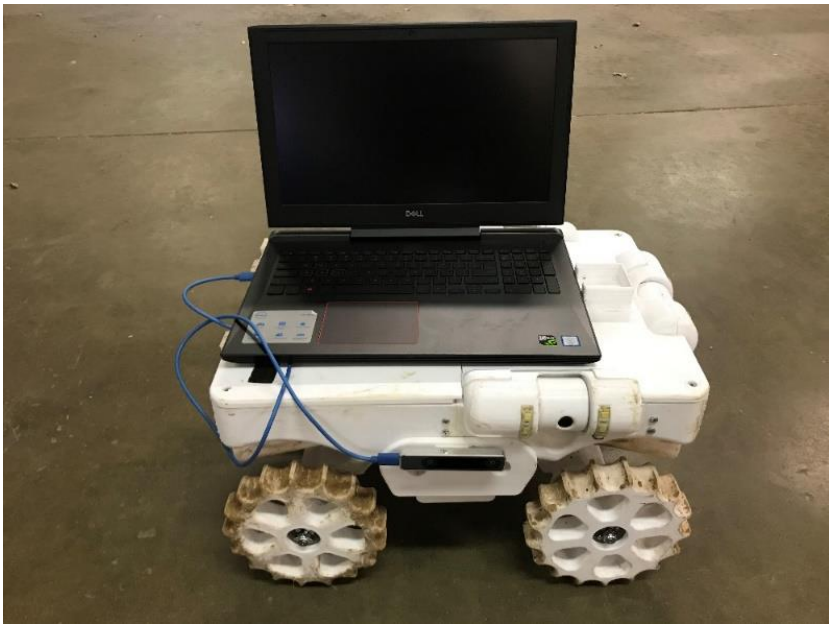


Figure 2.28 Robot representing combine.

2.6.3.1.1. Grain Cart Representation

As shown in Figure 2.29, the mobile robot employed in this study to represent the grain cart was a Jackal, a field robotics platform produced by Clearpath Robotics, Inc. (Kitchener, ON, Canada). The Jackal is small with a length of 508 mm, width of 430 mm and height of 250 mm. It has an onboard computer, which is fully integrated with Robot Operating System (ROS) for out-of-the-box autonomous capability. The Wi-Fi connectivity of Jackal allows for navigation algorithms (e.g., sensing, planning and control) to execute remotely, in this study, on a laptop. The flat top surface of Jackal's waterproof casing provides a simple mounting platform for sensors and other devices. The Jackal features a sturdy aluminum chassis with a high torque 4×4 drive-by-wire drive train.

Figure 2.29 Jackal by Clearpath Robotics, Inc. Reprinted from [44].

2.6.3.1.2. Combine Representation

The mobile robot employed in this study to represent the combine was a TerraSentia (Figure 2.30), produced by EarthSense, Inc. (Champaign, IL, USA). This robot is similar in size to the Jackal, equipped with multiple cameras and an onboard computer, and designed for in-field plant trait data collection, especially for under-canopy traits that cannot be obtained by other technologies. TerraSentia uses deep learning and computer vision software to phenotype and detect variations for crop breeders and researchers. Because the focus of this study was motion planning for autonomous grain carts, the TerraSentia, representing the combine, was remotely controlled by a human with the tablet computer paired with the TerraSentia to move around in the tests.

Figure 2.30 TerraSentia by EarthSense, Inc. Reprinted from [45].

2.6.3.1.3. Semi-Trailer Representation

Unlike the grain cart or the combine, the semi-trailer does not need to move in this study. After unloading the grain from the combine, the grain cart will simply go to the semi-trailer, park alongside and transfer the grain. Therefore, for simplicity, a cardboard box was employed to represent the semi-trailer.

2.6.3.1.4. Local Perception

As discussed previously, a 2D lidar sensor should be able to deliver sufficient local perception for an autonomous grain cart. A scanning laser rangefinder (Model UTM-30LX, Hokuyo, Japan; Figure 2.31) was employed and mounted on top of the Jackal. This 2D lidar is able to detect objects at ranges from 100 mm to 30 m with 1 mm resolution in a 270° arc (with 0.25° angular resolution) at a frequency as high as 40 Hz, specifications that are sufficient for the scaled indoor experiments in this study.

Figure 2.31 Scanning laser rangefinder Hokuyo UTM-30LX. Reprinted from [46].

2.6.3.1.5. Global Localization and Vehicle States Measurement

While RTK-GPS supports high-precision global localization, it is designed for use in outdoor environments. For the localization of the indoor mobile robots, two identical tracking cameras (Model RealSense T265, Intel Corp., Santa Clara, CA, USA; Figure 2.32) were employed, one for the Jackal and one for the TerraSentia. The T265 tracking camera is a stand-alone simultaneous localization and mapping (SLAM) device. SLAM involves constructing or updating a map of an unknown environment based on sensor data while simultaneously keeping track of the sensor location within that environment. The T265 tracking camera consists of two fisheye lens sensors, a visual processing unit (VPU) and an IMU. The built-in vision-based SLAM (V-SLAM) algorithms that runs on the VPU uses visual features in the environment for location tracking. The IMUs were used to measure the linear velocities and headings of the Jackal and TerraSentia.

Figure 2.32 Intel RealSense Tracking Camera T265. Reprinted from [47].

2.6.3.1.6. Computing Devices

A mini PC (Model NUC6i5SYK, Intel Corp., Santa Clara, CA, USA; Figure 2.33) was employed to run the sensors on the Jackal. The mini PC has an Intel Core i5-

6260U processor, a weight of 2.3 pounds and dimensions of 5.5×5×4 inches for length, width and height, which allowed it to be easily mounted on top of the Jackal. Connected with the 2D lidar and tracking camera, the mini PC was operated remotely via wireless connection to run the required sensor programs and send the sensing information to the laptop running the navigation algorithms.

Figure 2.33 Intel NUC6i5SYK mini PC. Reprinted from [48].

A gaming laptop (Model G5, Dell Corp., Austin, TX, USA; Figure 2.34) was mounted on top of the TerraSentia to run the tracking camera as well as interact with the mini PC remotely.

Figure 2.34 Dell G5 gaming laptop. Reprinted from [49].

A laptop (Model ThinkPad T580, Lenovo Corp., Quarry Bay, Hong Kong; Figure 2.35) served as the computation hub, executing the core planning and control algorithms. With an Intel Core i7-8650U processor, the laptop communicated wirelessly with the computer onboard the Jackal, the Intel mini PC on the Jackal and the Dell laptop on the TerraSentia.

Figure 2.35 Lenovo ThinkPad T580 laptop. Reprinted from [50].

2.6.3.1.7. V2V Communication

TAMULink, Texas A&M University's campus wireless network based on WPA Enterprise, was adopted for the wireless communication between the computing devices on the Jackal and TerraSentia.

2.6.3.1.8. Power Sources

Two power banks (XTPower Model MP-16000, Jauch Quartz America Inc., Seattle, WA, USA; Figure 2.36) were employed to power the Intel mini PC and the lidar

on the Jackal. These power banks have a capacity of 16000 mAh/59.2Wh. While the mini PC runs on 19V, the lidar runs on 12V, both of which are available as outputs of the power banks. The power banks are small and lightweight and were easily mounted on top of the Jackal.

Figure 2.36 XTPower power bank MP-16000. Reprinted from [51].

2.6.3.1.9. Crop Row Representation

Cardboard sheets were employed to represent the crop rows. As shown in Figure 2.37, the cardboard sheets were placed vertically on the ground with certain angles relative to each other. Slightly folded on the side and attached to the ground with duct tape, the cardboard sheets stayed upright by themselves, and they were easily run over by the TerraSentia robot, representing the combine. When standing upright, the cardboard sheets resembled unharvested crop rows that could potentially block the Jackal; when run over, the cardboard sheets resembled harvested areas that could be

traversed by the Jackal. Note that when mounted on an actual grain cart for real-world implementations, the lidar would not be on top of the tractor cab. Instead, it would be mounted at a level lower than the height of the crop plants being harvested. For example, on the front hitch point of the tractor.

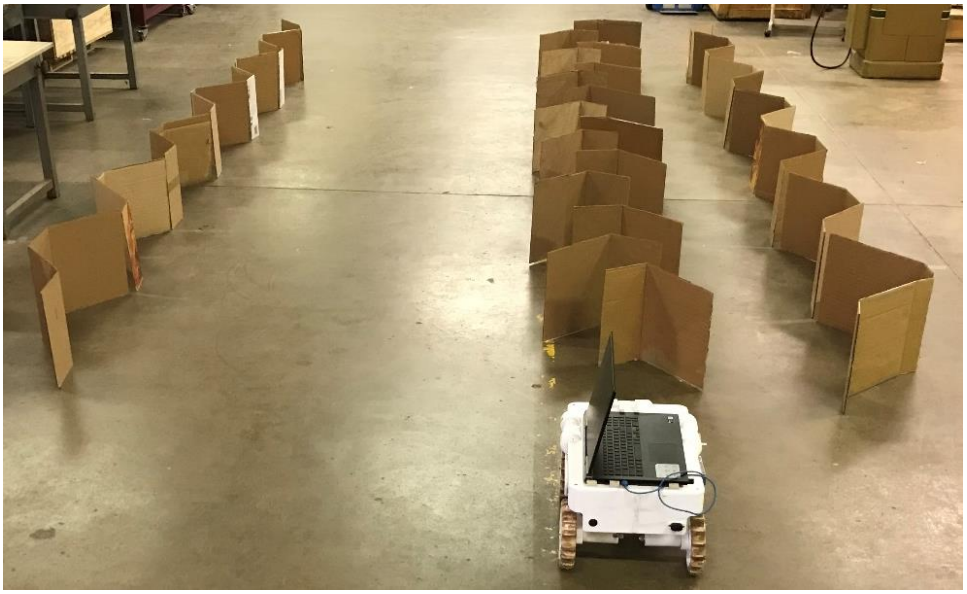


Figure 2.37 Crop rows represented by cardboard sheets.

2.6.3.1.10. ROS Network

As mentioned previously, control of the Jackal was realized via the ROS programs running on the Jackal's onboard PC. As a flexible framework for writing robotics software, ROS is a collection of tools, libraries, and conventions that aim to simplify the creation of complex and robust robot behaviors [52]. ROS's built-in messaging system saves time by managing the communication between distributed nodes via a publication and subscription mechanism, where "node" is the ROS term for

an executable program that is connected to the ROS network. Specifically, when a ROS node publishes a ROS message to a virtual ROS topic, any other node(s) can subscribe to this topic and receive the message. The structure of the ROS network built on the TAMULink was centered at the ROS node of the motion planner running on the ThinkPad laptop (Figure 2.38).

On top of the Jackal, the mini PC running lidar and camera drivers served as two ROS nodes, which published to a topic the messages containing the Jackal's information of the lidar-based local perception, the camera-based global localization and the IMU-based vehicle states measurement. On top of the TerraSentia, the Dell laptop running the camera driver served as another ROS node. The messages containing the pose and linear velocity of the TerraSentia, which were provided by the Intel tracking camera, were also published to a ROS topic. The core planning algorithm, coded and executed in MatLab Simulink, subscribed to the aforementioned topics and received the information about the Jackal and TerraSentia. After a series of computations, the planner published the planned motions to another topic, to which the Jackal's onboard PC subscribed. Because all Jackal robots come with a built-in low-level controller, the planned motions in the form of ROS messages can be directly implemented to control the movement of the Jackal, and no separate controller design is needed.

All the ROS messages were transmitted wirelessly to each ROS node in the ROS network. With the add-on Simulink toolbox, namely the Robotic System Toolbox, the interface between MatLab Simulink and ROS were easily established.

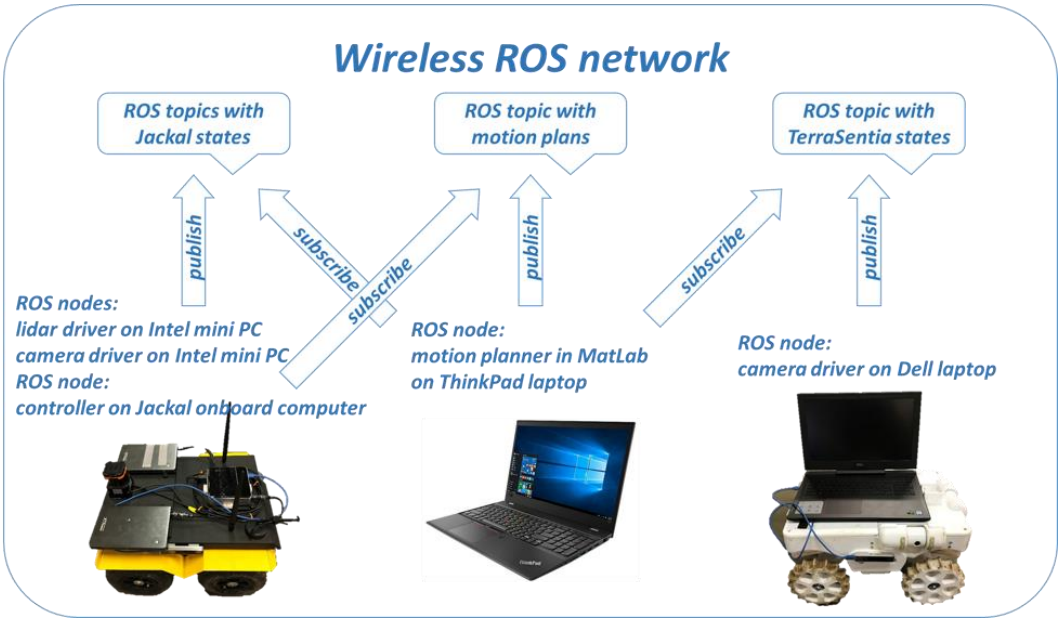


Figure 2.38 ROS network for mobile robot tests.

2.6.3.2. Test Design

The following mobile robot tests were designed and conducted. (i) Simple harvesting where the Jackal and the TerraSentia faced first the same direction and then opposite directions when meeting for unloading; (ii) single long static obstacle for the Jackal to go around; (iii) single static obstacle very close to the goal; (iv) two closely spaced static obstacles for the Jackal to go between; (v) two sparsely spaced obstacles for the Jackal to go around one by one; (vi) multiple static obstacles randomly spaced for the Jackal to navigate through; (vii) dynamic obstacle with no threat; (viii) dynamic obstacle that moved around blocking the way; (ix) malicious dynamic obstacle that chased and blocked the Jackal; (x) same tests as (ii) through (vi) but with a simple APF planner guiding the Jackal for comparison. The default repulsive range of obstacles in the APF was set to 0.5 m.

2.6.3.2.1. Simple Harvesting

Two very common cases of simple harvesting were tested. With the grain cart (Jackal) and the combine (TerraSentia) meeting for unloading, (i) they traveled in the same direction (Figure 2.39), and (ii) they faced opposite directions (Figure 2.40). When facing the same direction, the grain cart could directly approach the combine and drive alongside the combine on the left for unloading the grain. When facing opposite directions, the grain cart needed to make a U-turn to align its heading with that of the combine before unloading started. Beginning at the standby point, the grain cart needed to accomplish all the logistical tasks described previously; i.e., go to combine, unload grain, go to semi-trailer, transfer grain and return to standby point. The combine was manually controlled by a human with a joystick, and the linear velocity fluctuated slightly, resembling the speed variation of a real combine in a real harvest operation. Although simple harvesting was already included in the simulation tests, carrying out the same test with mobile robots was for verification of the effectiveness and practicality of the proposed navigation solution. Note that the comparison between the APF and APF+FLC planners was not conducted in this test, primarily because the obstacle avoidance tasks herein were relatively simple and would not help showcase the advantage of the proposed algorithm.

Figure 2.39 Simple harvesting: grain cart and combine faced same direction.

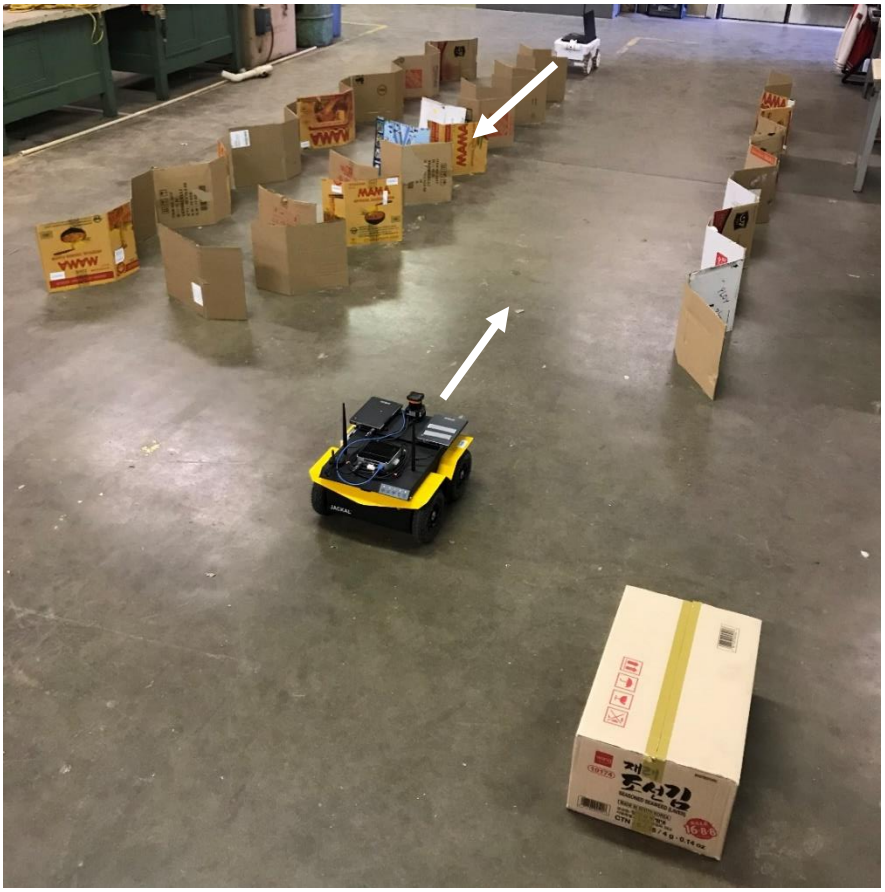


Figure 2.40 Simple harvesting: grain cart and combine faced opposite directions.

2.6.3.2.2. Long Static Obstacle

A single long static obstacle was constructed by a row of cardboard sheets (Figure 2.41). The long static obstacle represented an unharvested crop row, which is the most common obstacle a gain cart faces in harvest operations. The Jackal was expected to go around the long obstacle and reach the goal behind it (green dot). Long static obstacles could cause oscillations with the simple APF planner.

Figure 2.41 Long static obstacle.

2.6.3.2.3. Static Obstacle Close to Goal

In simple harvesting, the task of approaching and parking alongside the semi-trailer for grain transfer can be considered a general test case: to reach the goal with a

specific heading when there is an obstacle in close proximity to the goal. Figure 2.42 shows such an obstacle (cardboard box) close to the goal (green dot). The Jackal was supposed to approach the goal and park alongside it facing right. This may not seem a challenging task, but a simple APF planner may experience difficulty due to force contradiction or cancellation.

Figure 2.42 Obstacle close to goal.

2.6.3.2.4. Closely Spaced Static Obstacles

Two closely spaced static obstacles were constructed by two groups of cardboard sheets. As shown in Figure 2.43, the Jackal needed to go through the gap and reach the goal (green dot) behind the obstacles. This test case is also common in harvest

operations; when a grain cart enters the crop rows, many times it has to make it through the relatively narrow gap between the unharvested crop rows on the two sides. Closely spaced obstacles can potentially cause problems to a simple APF planner by creating local minima.

Figure 2.43 Closely spaced obstacles.

2.6.3.2.5. Sparsely Spaced Static Obstacles

The same two groups of cardboard sheets employed for closely spaced static obstacles were placed farther away from each other, representing random obstacles that are sparsely spaced (Figure 2.44). The Jackal thus needed to go around the obstacles one

by one and reach the goal (green dot) behind them. This test highlighted the human-like intelligence incorporated in the proposed APF+FLC planning algorithm.

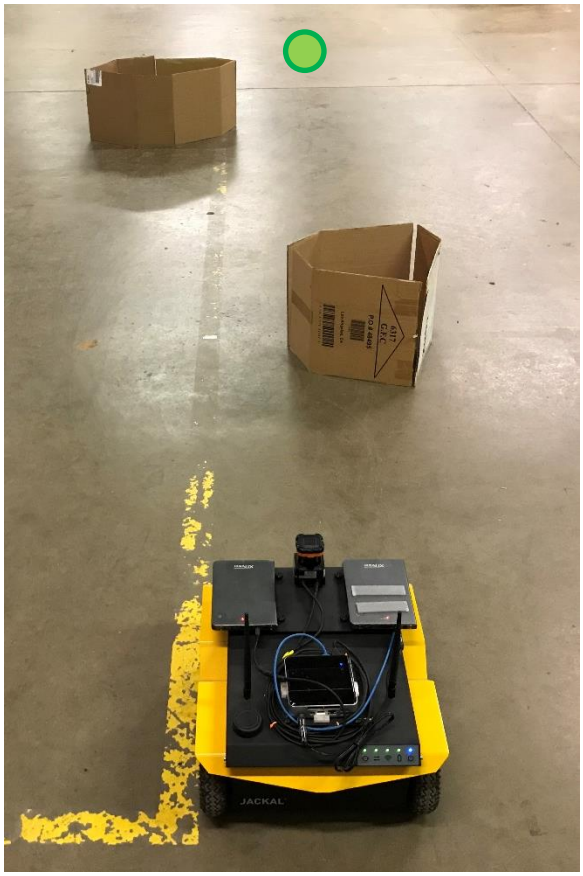


Figure 2.44 Sparsely spaced obstacles.

2.6.3.2.6. Multiple Randomly Spaced Static Obstacles

To verify the robustness of the proposed planning algorithm, the Jackal must be able to navigate between a number of randomly spaced static obstacles and reach the goal behind (green dot). To raise the difficulty of the motion planning task, (i) the obstacles were designed to be concave rather than convex, “embracing” the Jackal, and

(ii) small gaps were intentionally left between some of the obstacles, which could be misleading as possible passable openings (Figure 2.45).

Figure 2.45 Multiple randomly spaced static obstacles.

2.6.3.2.7. Dynamic Obstacle with No Threat

The existence of dynamic obstacles can often make the motion planning task more complex and challenging. Fast response to dynamic obstacles is critical for collision avoidance. Meanwhile, overreacting to dynamic obstacles that are actually of no threat to the host vehicle (always the Jackal in this study) is not desired. In this test, when the Jackal traveled straight towards the goal immediately in front of it, the dynamic obstacle, represented by the TerraSentia, traveled in a direction that crossed the Jackal's path. However, the linear velocity of the TerraSentia was so low that it would

never actually block or hit the Jackal. Note that in this general test case, the TerraSentia was not necessarily representing the combine as in the harvest operation test. Instead, the TerraSentia could be any (farming) vehicle or even a large animal that traversed the field. As explained in the APF+FLC integration mechanism, it was typically the APF part of the algorithm that handles dynamic obstacles, because their behaviors are often unpredictable and can easily get dangerously close to the vehicle. This and the following tests involving dynamic obstacles were designed simply to further verify the effectiveness and robustness of the proposed algorithm. Therefore, the comparison between the APF+FLC and a simple APF planner was omitted.

2.6.3.2.8. Dynamic Obstacle in the Way

The planner's response was also tested when the TerraSentia crossed the Jackal's straight path to the goal in front and actually blocked the way. Two specific cases were considered: when the TerraSentia met the Jackal relatively early, and when the TerraSentia met it relatively late. This test was to investigate whether the proposed algorithm can generate the most efficient motion plans when encountering dynamic obstacles in different relative poses.

2.6.3.2.9. Malicious Dynamic Obstacle

To verify the robustness of the proposed algorithm in handling dynamic obstacles, the TerraSentia in this test maliciously chased the Jackal that was going straight ahead. Specifically, the TerraSentia (i) intentionally blocked the way, forcing the Jackal to make sharp turns; (ii) chased the Jackal and pressed hard on the side, forcing the Jackal to turn around; and (iii) sabotaged the navigation in both the above ways.

2.6.3.2.10. Simple APF Tests

Just like the simulation tests, another set of tests identical to tests (ii) - (vi) were conducted with a simple APF planner guiding the Jackal. Performances of both the simple APF planner and the proposed APF+FLC planner were logged and compared for evaluating the efficiency of the proposed algorithm. To better illustrate the advantage of the proposed algorithm, in some tests, the key parameter of the APF planner, i.e., the repulsive range of obstacles, was adjusted to potentially improve the performance of the APF planner and further compare it with that of the APF+FLC planner.

2.6.3.3. Data Collection and Results Analysis

2.6.3.3.1. Effectiveness and Practicality of Navigation Solution

As stated in the Objectives, the first consideration with the mobile robot tests was whether the proposed navigation solution was effective and practical, which would be primarily reflected by the results of the simple harvesting test. If the grain cart (Jackal) could autonomously accomplish all the logistical tasks, following the motion plans generated by the APF+FLC planner, it would be reasonable to declare that the proposed navigation solution, featuring 2D-lidar-based local perception, IMU-based vehicle states measurement, Wi-Fi-supported V2V communication and drive-by-wire actuation, was effective and practical in implementing the developed motion planning algorithm. This provided a qualitative result.

2.6.3.3.2. Effectiveness and Robustness of Planning Algorithm

Another important subobjective of the mobile robot tests was to further verify the effectiveness and robustness of the proposed motion planning algorithm. While

accomplishment of the logistical tasks in the simple harvesting test would be a positive result, more importance was placed on the results of the other tests involving static or dynamic obstacles. Specifically, in these tests, if the Jackal could always intelligently handle the static obstacles with different configurations and the dynamic obstacles with different levels/types of threats, successfully reaching the goal, the effectiveness and robustness of the proposed motion planning algorithm would be verified. This also provided a qualitative result.

2.6.3.3.3. Efficiency

Again, as the most important aspect of the performance of the proposed motion planning algorithm, efficiency was evaluated based on the comparison between the proposed planner and the simple APF planner. While the efficiency was still mainly dependent on the length and smoothness of the Jackal's trajectory from the origin to the goal, the rationality of the motion plans was also considered an important indicator. The same methods employed in the simulation tests were used to measure the length and smoothness of the trajectory: approximate the tiny curves traveled by the Jackal within each sampling step as straight lines and sum up all the small segments for the length, and calculate the variance of the angles between consecutive segments of the trajectory and take it as the smoothness. Like before, measurement of the trajectory length and smoothness was quantitative, while comparing the rationality of the motion plans was qualitative.

2.7. References

[1] Reina, Giulio, et al. "Ambient awareness for agricultural robotic vehicles." *biosystems engineering* 146 (2016): 114-132.

[2] Noguchi, N., Will, J., Reid, J., & Zhang, Q. (2004). Development of a master–slave robot system for farm operations. *Computers and Electronics in Agriculture*, *44*(1), 1–19. doi: 10.1016/j.compag.2004.01.006

[3] Åstrand, B., & Baerveldt, A.-J. (2005). A vision based row-following system for agricultural field machinery. *Mechatronics*, *15*(2), 251–269. doi: 10.1016/j.mechatronics.2004.05.005

[4] Subramanian, V., Burks, T. F., & Arroyo, A. (2006). Development of machine vision and laser radar based autonomous vehicle guidance systems for citrus grove navigation. *Computers and Electronics in Agriculture*, *53*(2), 130–143. doi: 10.1016/j.compag.2006.06.001

[5] Shalal, N., et al. "A review of autonomous navigation systems in agricultural environments." *2013 Society for Engineering in Agriculture Conference: Innovative Agricultural Technologies for a Sustainable Future*. Engineers Australia, 2013.

[6] Mousazadeh, Hossein. "A technical review on navigation systems of agricultural autonomous off-road vehicles." *Journal of Terramechanics* 50.3 (2013): 211-232.

[7] English, Andrew, et al. "Vision based guidance for robot navigation in agriculture." *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014.

[8] Rovira-Más, Francisco. "Sensor architecture and task classification for agricultural vehicles and environments." *Sensors* 10.12 (2010): 11226-11247.

[9] Shearer, Scott A., Santosh K. Pitla, and Joe D. Luck. "Trends in the automation of agricultural field machinery." *Proceedings of the 21st Annual Meeting of the Club of Bologna. Italy*. 2010.

[10] Wang, Ning, Naiqian Zhang, and Maohua Wang. "Wireless sensors in agriculture and food industry—Recent development and future perspective." *Computers and electronics in agriculture* 50.1 (2006): 1-14.

[11] Tang, Sai Hong, et al. "A review on robot motion planning approaches." *Pertanika Journal of Science and Technology*20.1 (2012): 15-29.

[12] Mohanty, Prases K., and Dayal R. Parhi. "Controlling the motion of an autonomous mobile robot using various techniques: a review." *Journal of Advance Mechanical Engineering* 1.1 (2013): 24-39.

[13] Shiller, Zvi. "Off-line and on-line trajectory planning." *Motion and Operation Planning of Robotic Systems*. Springer, Cham, 2015. 29-62.

[14] Raja, Purushothaman, and Sivagurunathan Pugazhenthi. "Optimal path planning of mobile robots: A review." *International Journal of Physical Sciences* 7.9 (2012): 1314-1320.

[15] González, David, et al. "A review of motion planning techniques for automated vehicles." *IEEE Transactions on Intelligent Transportation Systems* 17.4 (2016): 1135-1145.

[16] Klingensmith, M. (2018, February 16). What Robotics Can Teach Gaming AI. Retrieved from https://sudonull.com/post/62343-What-robotics-can-teach-gaming-AI

[17] Koren, Y., & Borenstein, J. Potential field methods and their inherent limitations for mobile robot navigation. *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. doi: 10.1109/robot.1991.131810

[18] Goodrich, M. A. (2002). Potential fields tutorial. *Class Notes, 157*.

[19] Li, Guanghui, et al. "An efficient improved artificial potential field based regression search method for robot path planning." *Mechatronics and Automation (ICMA), 2012 International Conference on*. IEEE, 2012.

[20] Zohaib, Muhammad, et al. "Control strategies for mobile robot with obstacle avoidance." *arXiv preprint arXiv:1306.1144*(2013).

[21] Anonymous. (2017, March). Voronoi Diagram VI. Retrieved from http://zone.ni.com/reference/en-XX/help/371361P-01/gmath/voronoi_diagram/

[22] LaValle, S. (2012, April 20). Defining the Roadmap. Retrieved from http://planning.cs.uiuc.edu/node265.html

[23] Khaksar, W., Sahari, K. S. M., & Hong, T. S. (2016). Application of sampling-based motion planning algorithms in autonomous vehicle navigation. *Autonomous Vehicle, 735*.

[24] Dreo, J. (2006, May 27). Shortest Path Found by an Ant Colony. Retrieved from https://commons.wikimedia.org/wiki/File:Aco_branches.svg

[25] McFetridge, Lachlan, and M. Yousef Ibrahim. "New technique of mobile robot navigation using a hybrid adaptive fuzzy potential field approach." *Computers & Industrial Engineering*35.3-4 (1998): 471-474.

[26] Park, Jong-Wook, et al. "Advanced fuzzy potential field method for mobile robot obstacle avoidance." *Computational intelligence and neuroscience* 2016 (2016): 10.

[27] Ahmed, Alaa A., Turki Y. Abdalla, and Ali A. Abed. "Path Planning of Mobile Robot Using Fuzzy-Potential Field Method." *Iraqi Journal for Electrical & Electronic Engineering*11.1 (2015).

[28] Tsourveloudis, Nikos C., Kimon P. Valavanis, and Timothy Hebert. "Autonomous vehicle navigation utilizing electrostatic potential fields and fuzzy logic." *IEEE transactions on robotics and automation* 17.4 (2001): 490-497.

[29] Melingui, Achille, et al. "A novel approach to integrate artificial potential field and fuzzy logic into a common framework for robots autonomous navigation." *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 228.10 (2014): 787-801.

[30] Abdalla, Turki Y., Ali A. Abed, and Alaa A. Ahmed. "Mobile robot navigation using PSO-optimized fuzzy artificial potential field with fuzzy control." *Journal of Intelligent & Fuzzy Systems*32.6 (2017): 3893-3908.

[31] Jaradat, Mohammad Abdel Kareem, Mohammad H. Garibeh, and Eyad A. Feilat. "Autonomous mobile robot dynamic motion planning using hybrid fuzzy potential field." *Soft Computing* 16.1 (2012): 153-164.

[32] Siciliano, B. (2010). *Robotics: modelling, planning and control*. London: Springer.

[33] Hunt, Michael. *Autonomous mobile robot navigation using fuzzy logic control*. Diss. Dublin City University, 1998.

[34] Ross, Timothy J. *Fuzzy logic with engineering applications*. John Wiley & Sons, 2005.

[35] Ross, T. J. (2017). *Fuzzy Logic with Engineering Applications*. Wiley.

[36] Salmon, James. *Guidance of an Off-Road Tractor-Trailer System Using Model Predictive Control*. Diss. 2013.

[37] Karkee, Manoj. *Modeling, identification and analysis of tractor and single axle towed implement system*. Iowa State University, 2009.

[38] Richards & Son Agri Supply. (n.d.). Retrieved from http://www.richardsagri.com/equipment-and-parts.html

[39] Payne, L. (2014, July 1). The Modern Industrial Workhorse: PID Controllers. Retrieved from https://www.techbriefs.com/component/content/article/tb/features/articles/20013

[40] McMillan, Gregory K. "Industrial applications of PID control." *PID Control in the Third Millennium*. Springer, London, 2012. 415-461.

[41] McCormack, Anthony S., and Keith R. Godfrey. "Rule-based autotuning based on frequency domain identification." *IEEE transactions on control systems technology* 6.1 (1998): 43-61.

[42] Baillie, C, et al. (2017, July 19). Developments in Autonomous Tractors. Retrieved from https://grdc.com.au/resources-and-publications/grdc-update-papers/tab-content/grdc-update-papers/2017/07/developments-in-autonomous-tractors

[43] MarketsandMarkets. (2019, November 1). Drive By Wire Market by Application (Brake, Park, Shift, Steer, Throttle), Sensor (Brake Pedal, Throttle Position & Pedal, Park, Gearshift, Handwheel, Pinion), Vehicle (On-Highway, OHV, BEV/HEV/PHEV, Autonomous), Component & Region - Global Forecast to 2025. Retrieved from https://www.marketsandmarkets.com/Market-Reports/drive-by-wire-market-64127606.html

[44] Clearpath Robotics, Inc. (n.d.). Retrieved from https://clearpathrobotics.com/jackal-small-unmanned-ground-vehicle/

[45] EarthSense, Inc. (n.d.). Retrieved from https://www.earthsense.co/

[46] RobotShop. (n.d.). Hokuyo UTM-30LX Scanning Laser Rangefinder. Retrieved from https://www.robotshop.com/en/hokuyo-utm-03lx-laser-scanning-rangefinder.html

[47] Intel. (n.d.). Intel RealSense Tracking Camera T265. Retrieved from https://www.intelrealsense.com/tracking-camera-t265/

[48] Intel. (n.d.). Intel NUC Kit NUC6i5SYK. Retrieved from https://www.intel.com/content/www/us/en/products/boards-kits/nuc/kits/nuc6i5syk.html

[49] BestBuy. (n.d.). Dell - Inspiron 15.6" Laptop - Intel Core i7 - 16GB Memory - NVIDIA GeForce GTX 1060 - 1TB HDD + 256GB SSD - Matte Black. Retrieved from https://www.bestbuy.com/site/dell-inspiron-15-6-laptop-intel-core-i7-16gb-memory-nvidia-geforce-gtx-1060-1tb-hdd-256gb-ssd-matte-black/6162034.p?skuId=6162034&intl=nosplash

[50] Newegg. (n.d.). Lenovo Laptop ThinkPad T580 (20L9001KUS) Intel Core i5 8th Gen 8350U (1.70 GHz) 8 GB Memory 256 GB SSD Intel UHD Graphics 620 15.6" Windows 10 Pro 64-Bit. Retrieved from https://www.newegg.com/black-lenovo-thinkpad-t580-mainstream/p/N82E16834850817

[51] Geektech. (n.d.). Mega Laptop Powerbank MP-20000. Retrieved from https://www.geeektech.com/nl/geeek-mega-laptop-powerbank-mp-20000.html

[52] ROS. (n.d.). About ROS. Retrieved from http://www.ros.org/about-ros/

3. RESULTS AND CONCLUSIONS

3.1. Results by Objective

3.1.1. Simulation Tests: APF+FLC vs. Simple APF

3.1.1.1. Simple Harvesting

3.1.1.1.1. Effectiveness

When no obstacles other than unharvested crop rows were present, with the proposed task scheduling strategy, both the APF+FLC and APF planners were able to navigate the autonomous grain cart in the crop field and accomplish all the logistical tasks without any collisions. Figure 3.1 exemplifies a work cycle of the autonomous grain cart, in which it (i) stood by, (ii) went to the combine, (iii) unloaded the grain, (iv) went to the semi-trailer, (v) transferred the grain, and (vi) returned to the standby point. For the six unloading operations that took place during the entire harvest, the grain cart spent an average of only 5.38 s following the combine before unloading started. This non-productive following time was less than 0.6% of the total time of the simulated harvest operation, which was 15 minutes. Therefore, the effectiveness of the proposed motion planning algorithm and the task scheduling strategy was verified.

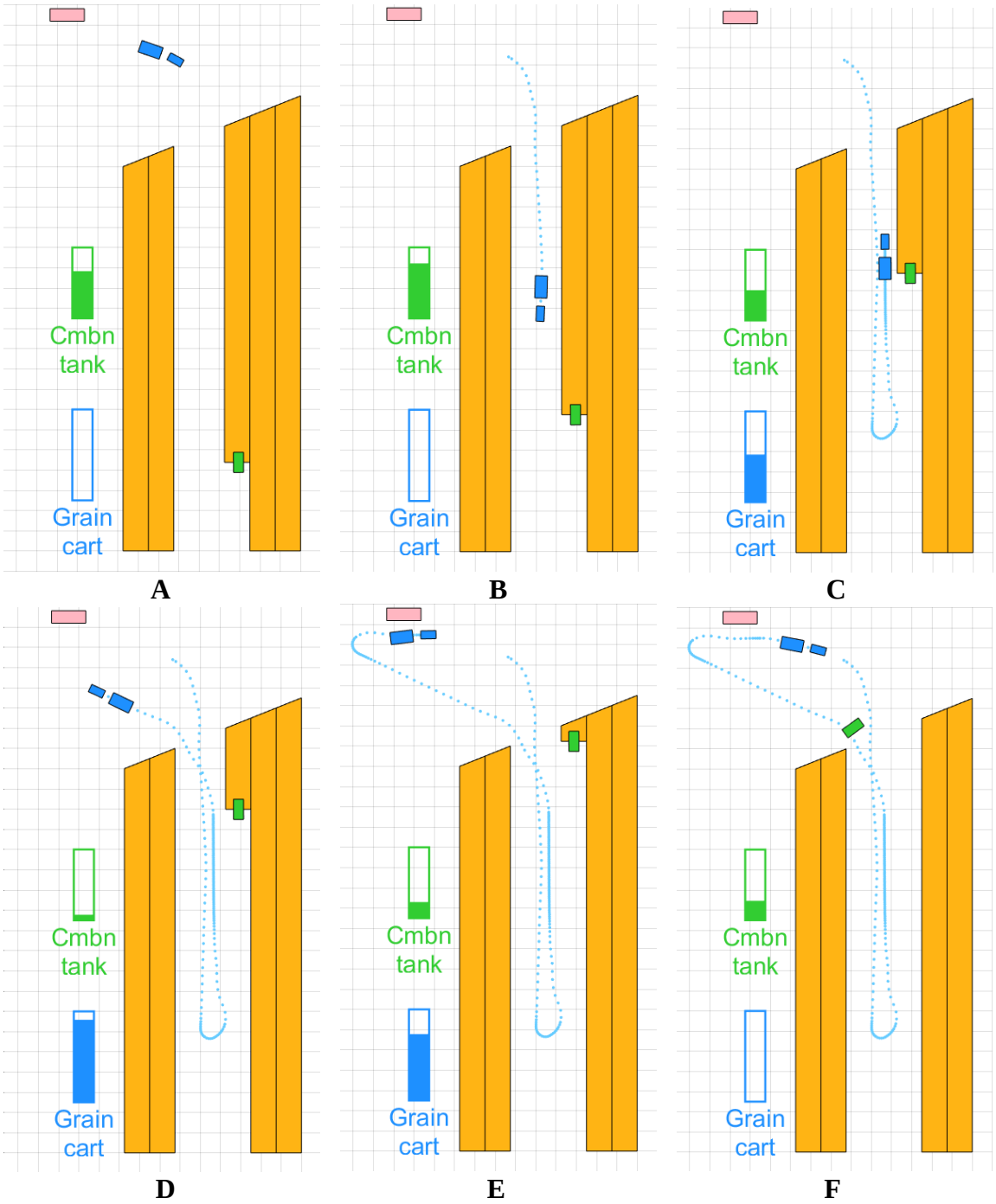


Figure 3.1 An example work cycle of grain cart in simple harvesting. (A) Standby. (B) To combine. (C) Unload. (D) To semi. (E) Transfer. (F) To standby.

3.1.1.1.2. Efficiency

The two planners differed from each other in two scenarios: (i) each time the grain cart left the combine for the semi-trailer, the APF planner tended to drive the grain cart unnecessarily close to the crop rows while the APF+FLC planner, leveraging human-like intelligence, kept the grain cart safely away from the rows (Figure 3.2); (ii) due to its inherent limitations, the APF planner made the grain cart oscillate when it was traveling between the crop rows (Figure 3.3.), while the APF+FLC had no such issue. Therefore, in terms of rationality of the motion plans, the APF+FLC planner outperformed the APF planner. In terms of the trajectory length and smoothness, the APF+FLC trajectory was slightly smoother (0.00062 rad^2) while slightly longer (1590.89 m) than the simple APF trajectory (0.00063 rad^2 for smoothness and 1583.37 m for length). These results imply that, overall, the two planners generated similar motion plans. Two major reasons for the lack of difference in performance between the planners in this test were (i) the obstacle avoidance task was relatively simple when unharvested crop rows were the only obstacles, and (ii) as shown in Figure 3.1 A, the standby point was deliberately selected such that a straight route to the unloading location was always available, further simplifying the obstacle avoidance task.

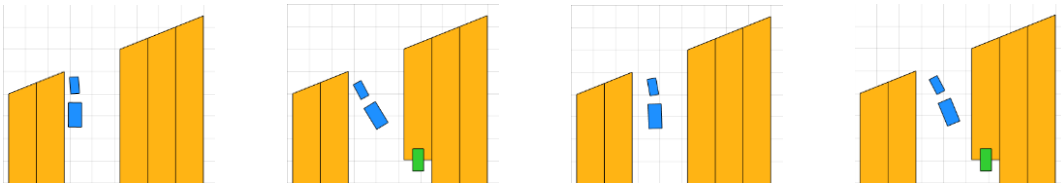


Figure 3.2 Grain cart left combine for semi-trailer in first two work cycles. (A) APF 1st cycle. (B) APF 2nd cycle. (C) APF+FLC 1st cycle. (D) APF+FLC 2nd cycle.

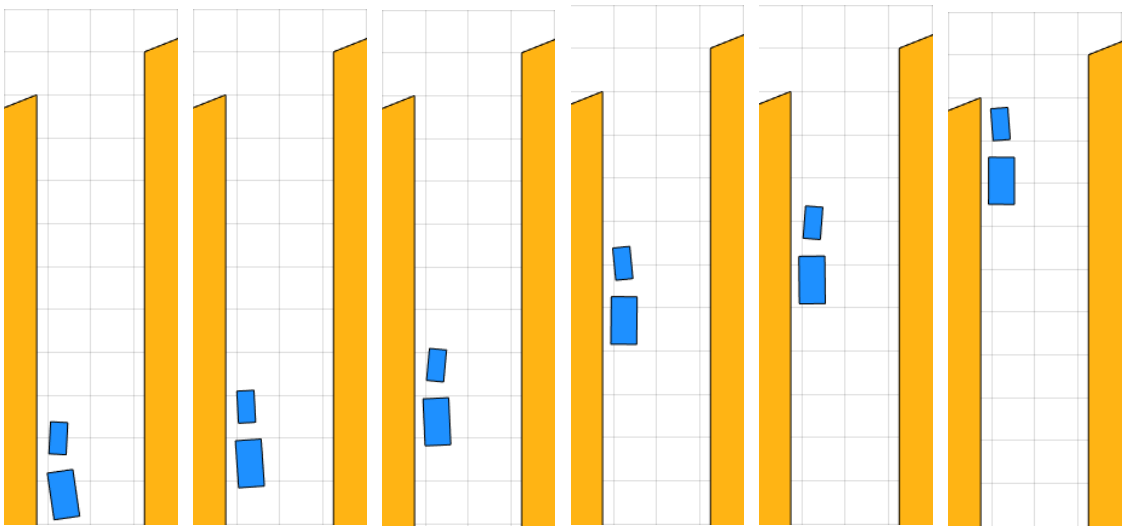


Figure 3.3 Grain cart with APF planner oscillated between crop rows. Sequential motion from (A) to (F).

3.1.1.1.3. Computational Expense

As mentioned previously, computational efficiency has been discussed in many studies as a key advantage of APF over other techniques, demonstrating that APF is suitable for real-time applications. Upgraded from APF, the proposed APF+FLC planner consumed an average CPU time of 0.74 ms in each computation step. Although this computation expense was over two times greater than that of the simple APF planner, which was 0.28 ms, the APF+FLC computations were still very fast. Note that employed in the simulation was an ordinary CPU (Intel Core i7-8650U CPU @ 1.90 GHz), thus it is reasonable to declare that the proposed algorithm is sufficiently expeditious for real-time motion planning.

3.1.1.2. Static Obstacles

3.1.1.2.1. Effectiveness and Robustness

Both the APF+FLC and APF planners have the same task scheduling strategy and thus were similarly able to direct the autonomous grain cart to accomplish the logistical tasks in the harvest operation where static obstacles other than crop rows existed, verifying the effectiveness and robustness of the proposed motion planning algorithm and the associated task scheduling strategy.

3.1.1.2.2. Efficiency

A major difference between the efficiencies of the motions generated by the two planners was observed. Figures 3.4 and 3.5, respectively, show how the APF+FLC and APF planners handled five sets of static obstacles that blocked the straight route when the grain cart was going to the semi-trailer. Taking early actions to smoothly go around the obstacles from the side closer to the goal enabled the APF+FLC planner to deal with the static obstacles more rationally and efficiently, with a trajectory length of 436.74 m and smoothness of 0.0014 rad^2. In contrast, without a global view, the APF planner steered the grain cart only to avoid imminent collisions with local obstacles, resulting in a trajectory with length of 553.13 m and smoothness of 0.0016 rad^2. Thus, the APF+FLC trajectory was 21.04% shorter and 12.50% smoother than that of the simple APF planner.

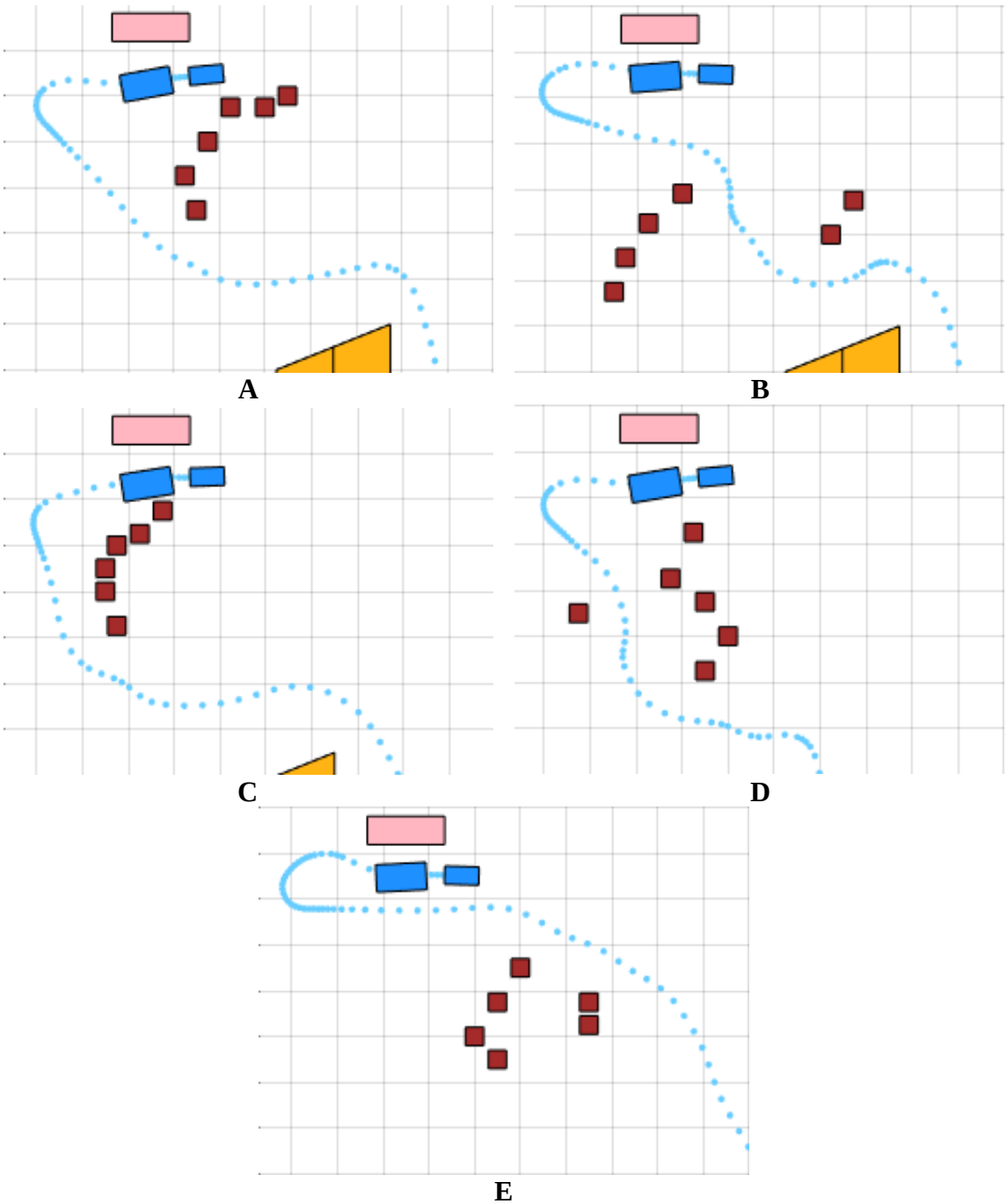


Figure 3.4 Grain cart with APF+FLC planner handled five sets (from A to E) of static obstacles.

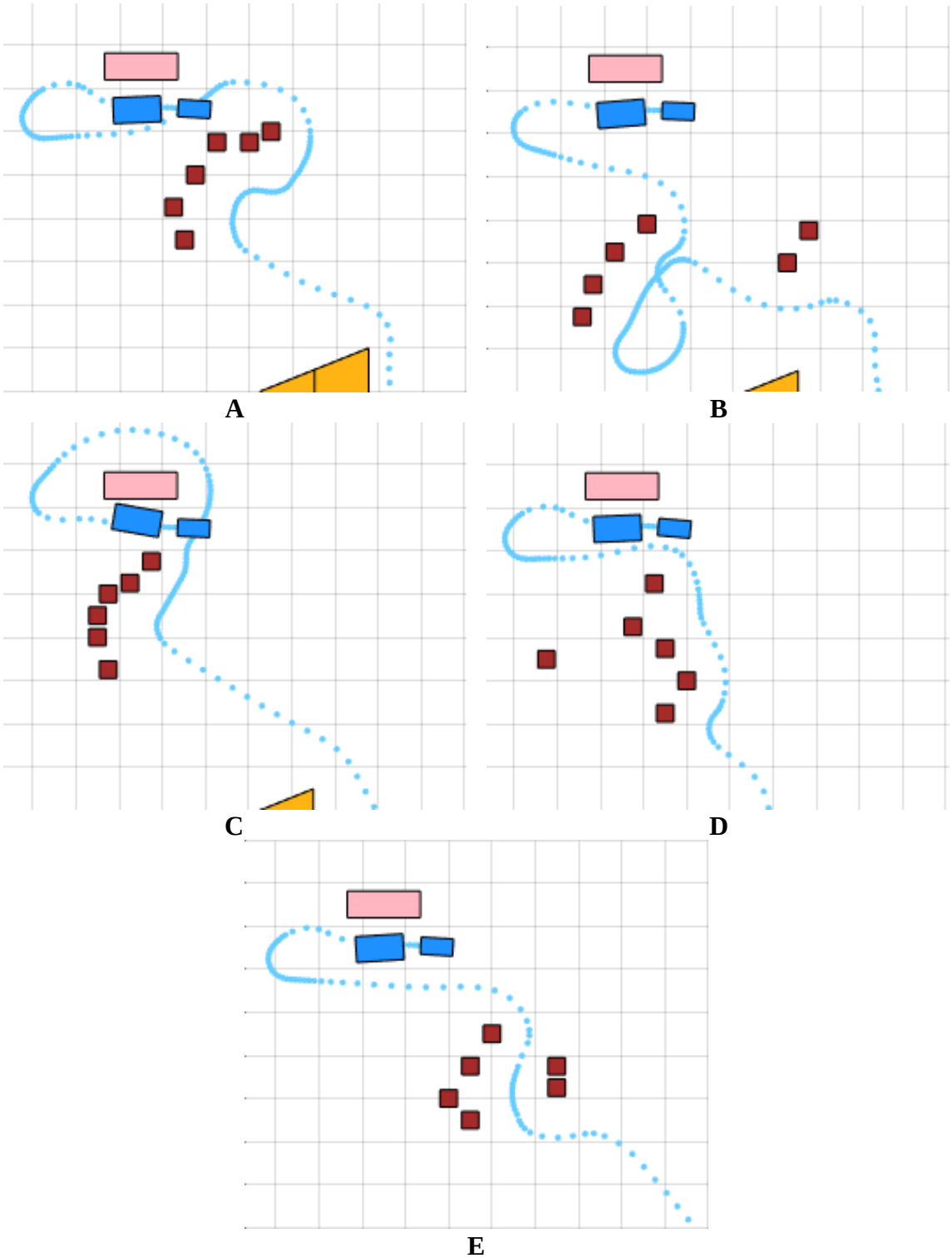


Figure 3.5 Grain cart with APF planner handled five sets (from A to E) of static obstacles.

3.1.1.2.3. Computational Expense

Although the static obstacles made the environmental condition more complex, this fact did not increase the computational expense of the proposed planner as well as the simple APF planner. The average CPU times consumed in each computation step were basically the same as those in the previous test.

3.1.1.3. Dynamic Obstacles

3.1.1.3.1. Effectiveness and Robustness

The dynamic obstacles constantly changed the environment between the crop rows and the semi-trailer (Figure 3.6). As described previously, when the grain cart approached this type of obstacles, the APF+FLC planner first made use of FLC and attempted to drive the grain cart around the obstacles based on their real-time configuration, and it updated the motion plan accordingly in each computation step. When dangerously close to the obstacles, the planner relied on APF to promptly avoid potential collisions. As shown in Figure 3.7, in some cases the grain cart went around the obstacles from the side, while in other cases the grain cart navigated through the obstacle zones when the gap between them was wide enough to provide more efficient routes. Because of the dynamic nature of this test, the "go-around" plans generated by the FLC part of the planner were often overruled as the obstacles typically moved very close to the grain cart, making APF dominate the planning. Both the APF and APF+FLC planners successfully handled the dynamic obstacles, and with the task scheduling strategy they accomplished the logistical tasks in the harvest operation, further verifying their effectiveness and robustness.

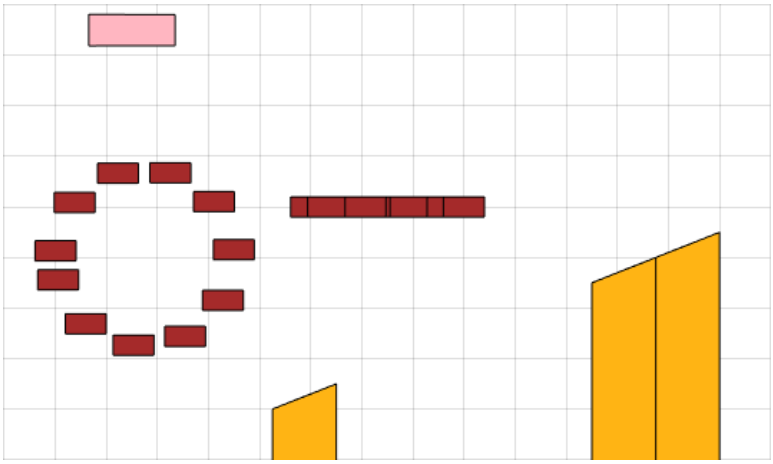


Figure 3.6 Trajectories of two dynamic obstacles: one circled, one moved back and forth.

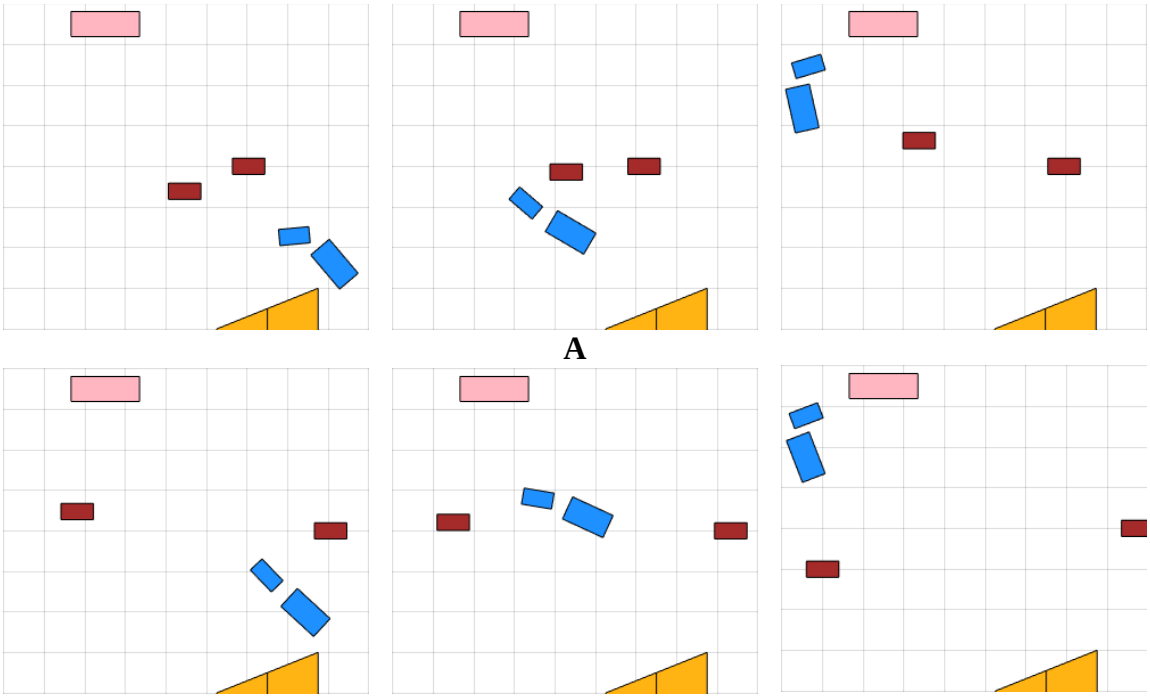


Figure 3.7 Grain cart with APF+FLC planner handled dynamic obstacles in five working cycles (from A to E). Each row for each cycle. Sequential motion from left to right.

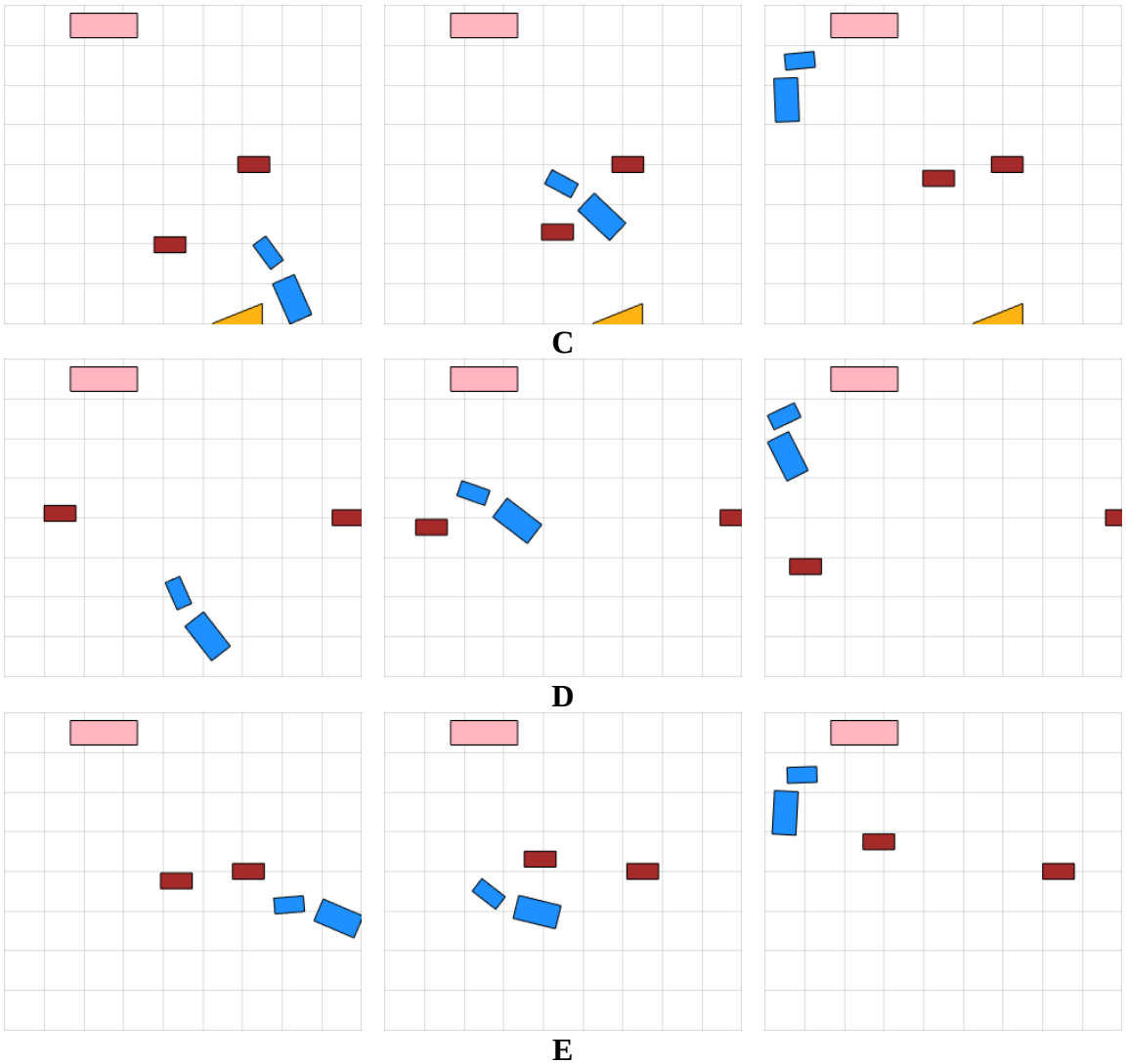


Figure 3.7 Continued.

3.1.1.3.2. Efficiency

The APF planner was effective and prompt in avoiding collisions with dynamic obstacles but ended up with a trajectory that was 831.18 m in length and 0.0012 rad^2 in smoothness, compared with the APF+FLC trajectory that was 18.76% shorter (675.27 m) and 16.67% smoother (0.0010 rad^2). The primary reason for these differences was that, with the APF planner, the grain cart occasionally got trapped in a local minimum.

In these instances, when the dynamic obstacle was wandering close to the standby point although not really blocking the grain cart, its repulsive forces hindered the grain cart (which just finished transferring the grain to the semi-trailer) from approaching the standby point in the specified heading. Therefore, the grain cart had to make U-turns to adjust its pose, only to fail again until the dynamic obstacle moved away, by which time the grain cart was already late for unloading and had to go to the combine immediately (Figure 3.8). In contrast, the APF+FLC planner had no such issue since it has the intelligence to ignore the dynamic obstacle, which actually never blocked the way to the standby point. Compared with the APF planner, the APF+FLC planner exhibited higher efficiency dealing with dynamic obstacles in this study.

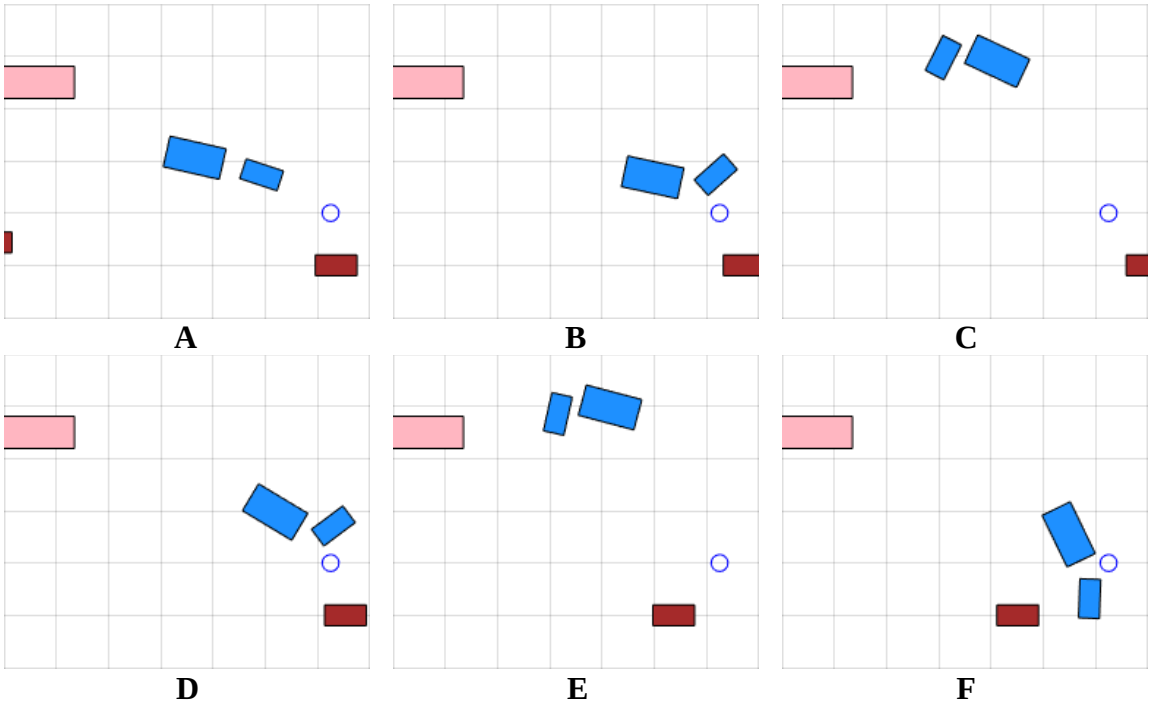


Figure 3.8 Grain cart with APF planner had trouble approaching standby point (blue circle). Sequential motion from (A) to (F).

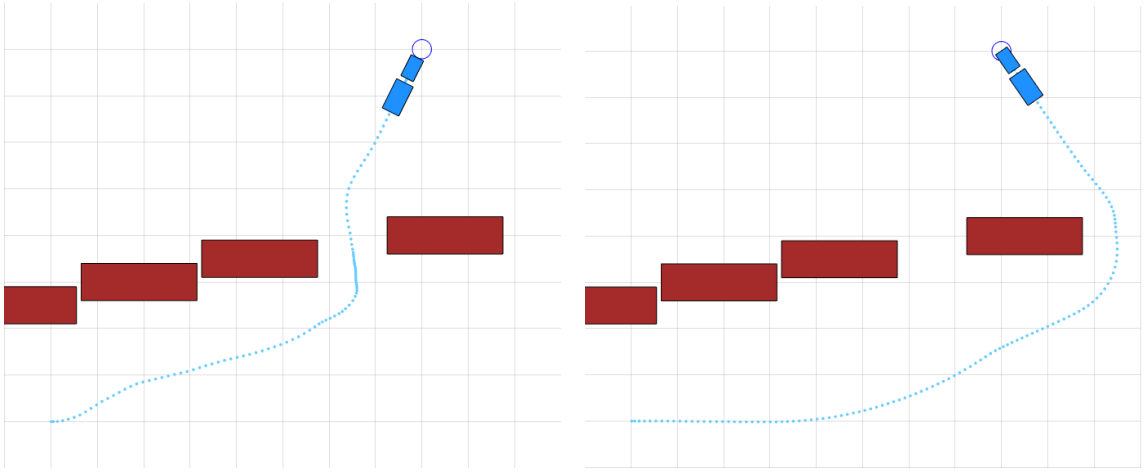
3.1.1.3.3. Computational Expense

Similar to the previous test, dealing with the dynamic obstacles did not add significant extra calculation requirements to the CPU, leaving the computational expenses of both planners similar to those in the previous tests.

3.1.2. Simulation Tests: APF+FLC vs. VFH

3.1.2.1. Long Static Obstacle with Gap

Figure 3.9 shows how the APF+FLC and VFH planners dealt differently with the long static obstacle with a narrow gap. The APF+FLC planner was able to identify the more efficient route through the gap and directed the grain cart to take the shortcut towards the goal. Meanwhile, the VFH planner led the grain cart to travel all the way around the long obstacle without making use of the gap, because the VFH planner decided that the space at the end of the long obstacle had a lower obstacle density than the narrow gap, meaning going around the long obstacle would have lower probability for the grain cart to be blocked by obstacles. Although this decision resulted in a trajectory that was over two times smoother (0.00036 rad^2) than the APF+FLC trajectory (0.00091 rad^2), the length for APF+FLC (122.19 m) was 24.47% shorter than that for VFH (161.78 m). In terms of computational expense in each simulation step, the APF+FLC planner consumed an average of 0.42 ms while the VFH planner consumed an average of 0.62 ms, indicating that the APF+FLC planner was 32.26% faster in the computations. In summary, when handling the long static obstacle with a narrow gap, the APF+FLC planner accomplished the navigation task with a shorter travel distance and less computational consumption.

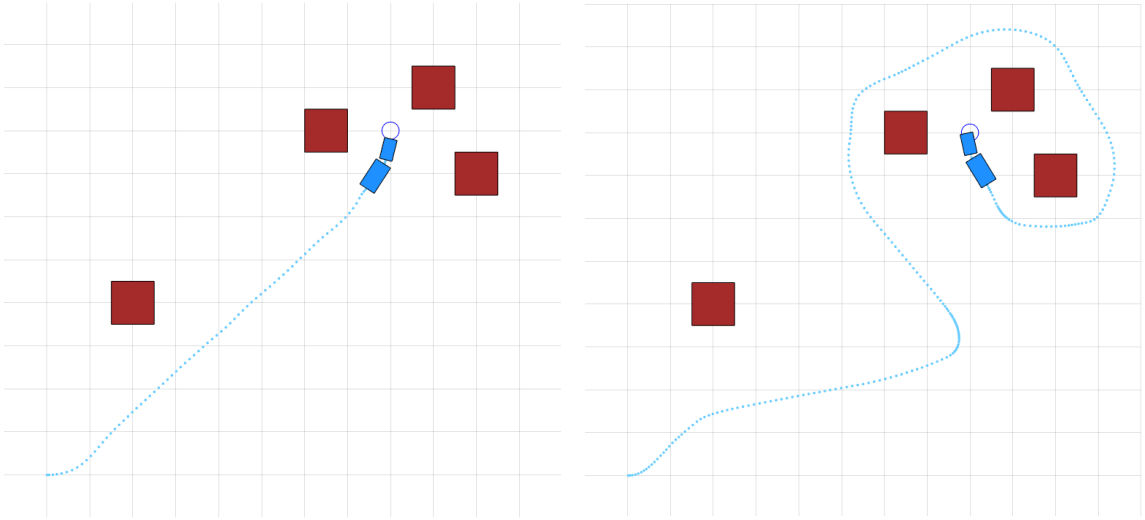


A B

Figure 3.9 Grain cart dealt with long static obstacle with narrow gap. (A) APF+FLC. (B) VFH.

3.1.2.2. Static Obstacles Close to Goal

As shown in Figure 3.10, the APF+FLC planner was effective in directing the grain cart to approach the goal, ignoring the obstacle on the way that hardly blocked the straight path from the origin to the goal as well as the three obstacles closely surrounding the goal. However, these obstacles, especially the three that generated high obstacle density around the goal, caused major problems for the VFH planner. Instead of approaching the goal directly, the grain cart with the VFH planner had to go around all the three surrounding obstacles before identifying the path towards the goal. Thus, the APF+FLC trajectory (109.16 m length and 0.00065 rad^2 smoothness) was 61.34% shorter and 32.29% smoother than the VFH trajectory (282.37 m length and 0.00096 rad^2 smoothness). In terms of average computation time, the APF+FLC planner consumed 0.53 ms while the VFH planner consumed 0.69 ms, meaning the APF+FLC planner was 23.19% more efficient in the computations.

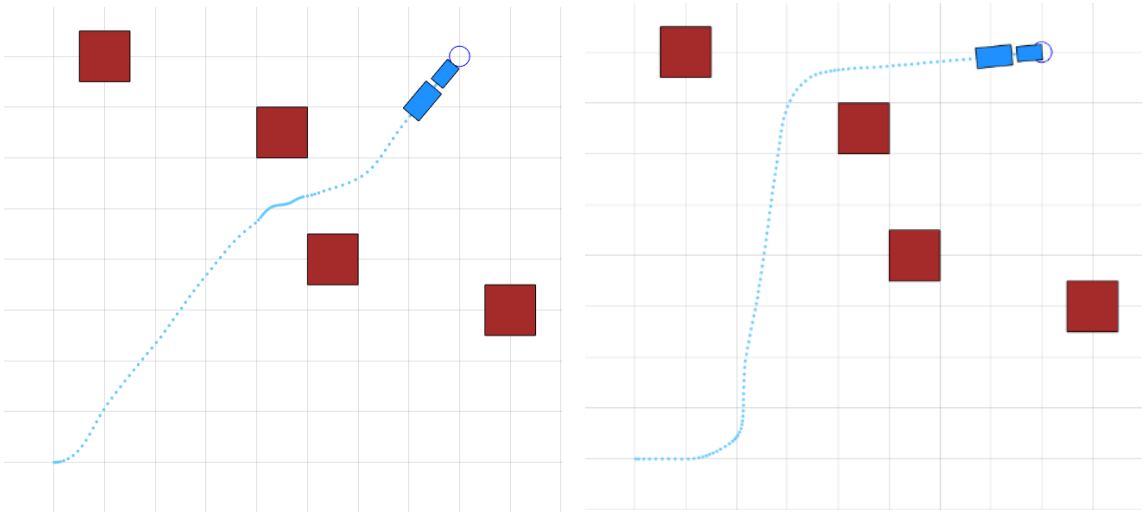


A **B**

Figure 3.10 Grain cart dealt with multiple obstacles close to goal. (A) APF+FLC. (B) VFH.

3.1.2.3. Closely Spaced Static Obstacles

Of the three gaps between the four closely spaced static obstacles, the narrow one in the middle provided the most efficient route towards the goal. The APF+FLC planner was effective in making use of this narrow gap and directed the grain cart to efficiently navigate to the goal (Figure 3.11 A), resulting in a trajectory length of 109.51 m and a smoothness of 0.00091 rad^2. On the other hand, the VFH planner was not able to make use of the narrow gap in the middle, and instead, it directed the grain cart to go through one of the two wider gaps which was considered safer with lower obstacle density (Figure 3.11 B). The VFH trajectory had a length of 133.33 m and a smoothness of 0.00064 rad^2, which was 29.67% smoother but 21.75% longer than the APF+FLC trajectory. Additionally, the computational expense of the APF+FLC planner (0.36 ms) was 58.62% less than that of the VFH planner (0.87 ms).



A B

Figure 3.11 Grain cart dealt with closely spaced obstacles. (A) APF+FLC. (B) VFH.

3.1.2.4. Sparsely Spaced Static Obstacle Groups

The two groups of sparsely spaced static obstacles hardly blocked the straight path between the starting and target points. The APF+FLC was effective in directing the grain cart to smoothly steer towards the wide gap between the two obstacle groups, pass through it and reach the goal (Figure 3.12 A). In contrast, the VFH planner did not enable the grain cart to travel with a global view, so it went around the obstacle group on the right to approach the goal (Figure 3.12 B). The VFH planner did not adopt the obvious shortcut, because the locations of the obstacle groups relative to each other and relative to the grain cart made it challenging for the VFH planner to identify the large space in between at the beginning of the run. Meanwhile, going around the obstacle group from the righthand side where the obstacle density was lower seemed more promising to the VFH planner. Again, the APF+FLC compromised trajectory

smoothness (i.e., VFH 0.00030 rad^2 vs. APF+FLC 0.00066 rad^2; VFH was 54.55% smoother) for a reduction in trajectory length (APF+FLC 107.62 m vs. VFH 141.70 m; APF+FLC was 24.05% shorter). In addition, the computational expense of the APF+FLC planner (0.35 ms) was only about half of that of the VFH planner (0.71 ms).

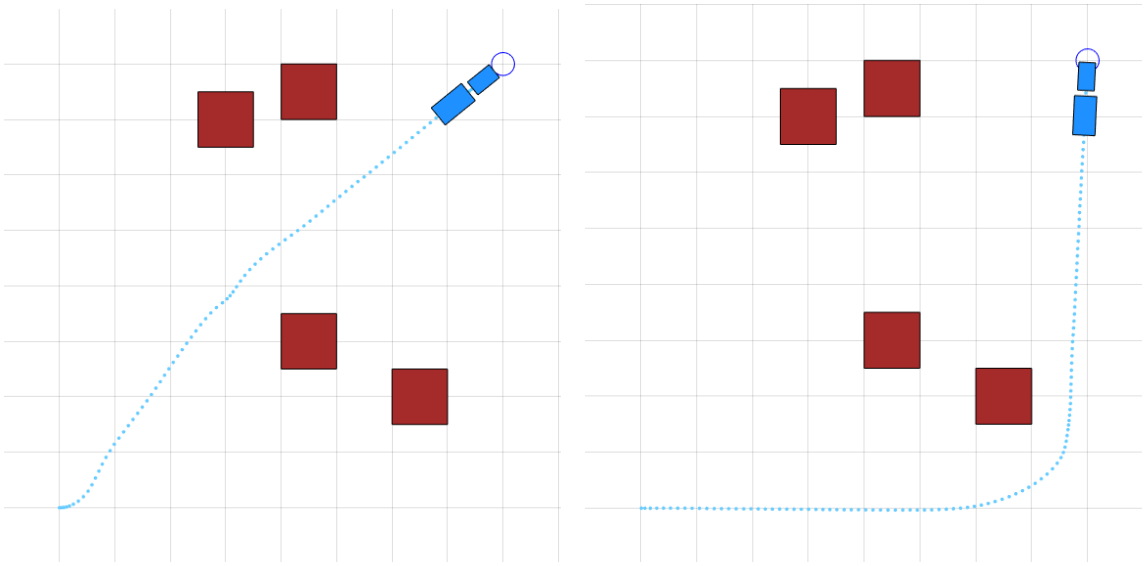


Figure 3.12 Grain cart dealt with sparsely spaced obstacle groups. (A) APF+FLC. (B) VFH.

3.1.2.5. Dynamic Obstacles

As shown in Figure 3.13, when directed by the APF+FLC planner, the grain cart was able to efficiently approach the goal. While the grain cart attempted to follow the most efficient route (i.e., straight line from origin to goal), it adjusted the motions to avoid collisions with the dynamic obstacles. Specifically, the grain cart went around the first obstacle from the left, then tried to circumvent the second obstacle from the right. However, as the second obstacle circled counterclockwise, the grain cart steered left to

let the obstacle pass by, and approached the goal more efficiently. As shown in Figure 3.14, when guided by the VFH planner, the grain cart encountered major problems. Just like the previous test (i.e., sparsely spaced obstacle groups), the grain cart did not attempt to make use of the momentary large space between the two dynamic obstacles, and went around the first obstacle from the right. After that the grain cart encountered the second obstacle, which traveled between the grain cart and the goal for almost half of the circle around the goal. Eventually, when the relatively faster obstacle passed the grain cart, leaving a clear space between the grain cart and the goal, the grain cart was able to steer towards the goal. The VFH planner failed to prevent the grain cart from following alongside the dynamic obstacle, which led to a trajectory (182.47 m) much longer than that of the APF+FLC planner (114.50 m, 37.25% shorter). On the other hand, the smoothness of the VFH trajectory (0.00092 rad^2) was 23.33% higher than that of the APF+FLC trajectory (0.0012 rad^2). Just like the previous tests, the APF+FLC proved to be much faster (25.93%) computationally than the VFH planner (APF+FLC 0.60 ms vs. VFH 0.81 ms).

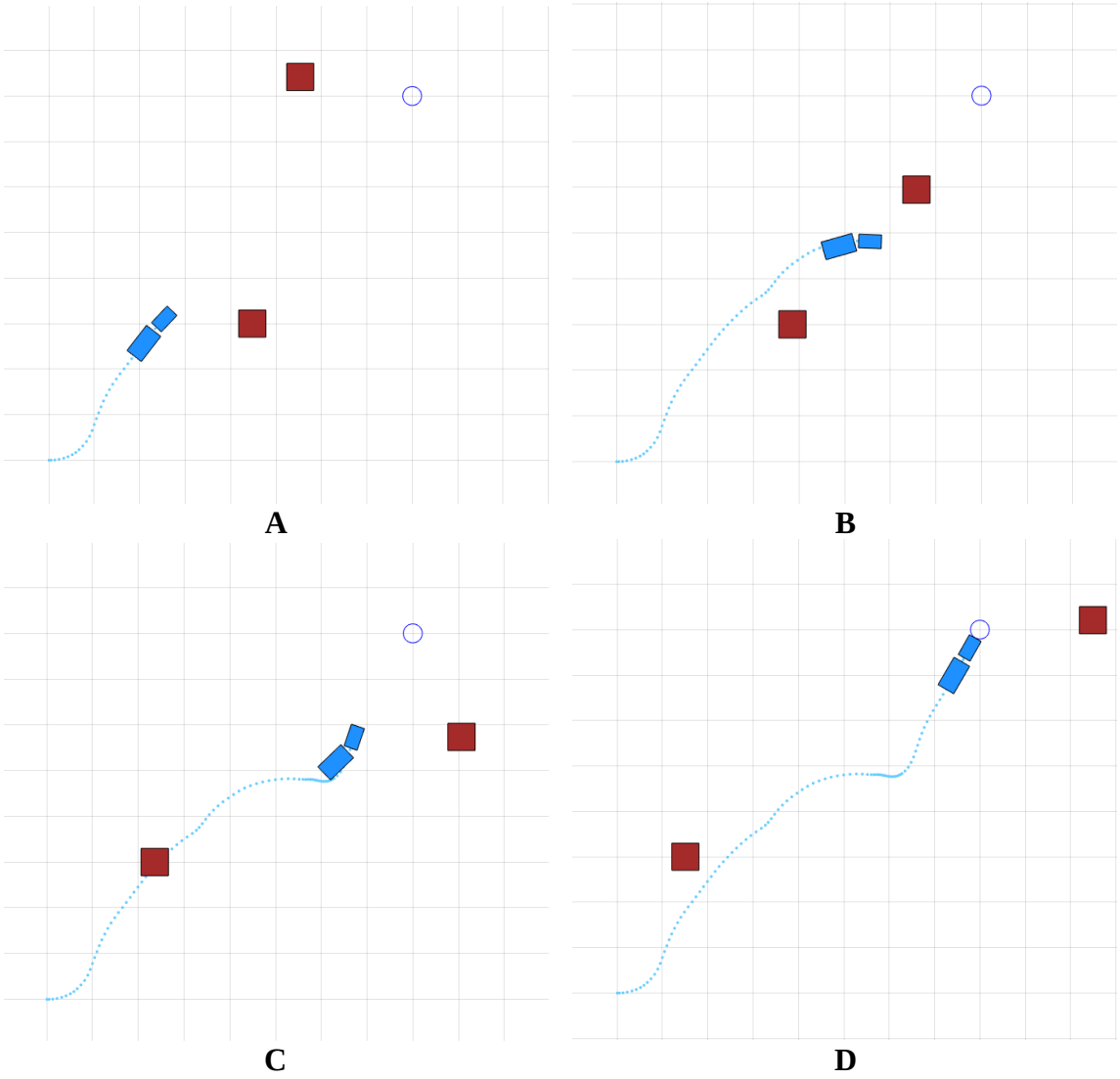


Figure 3.13 Grain cart with APF+FLC planner dealt with dynamic obstacles. Sequential motion from (A) to (D).

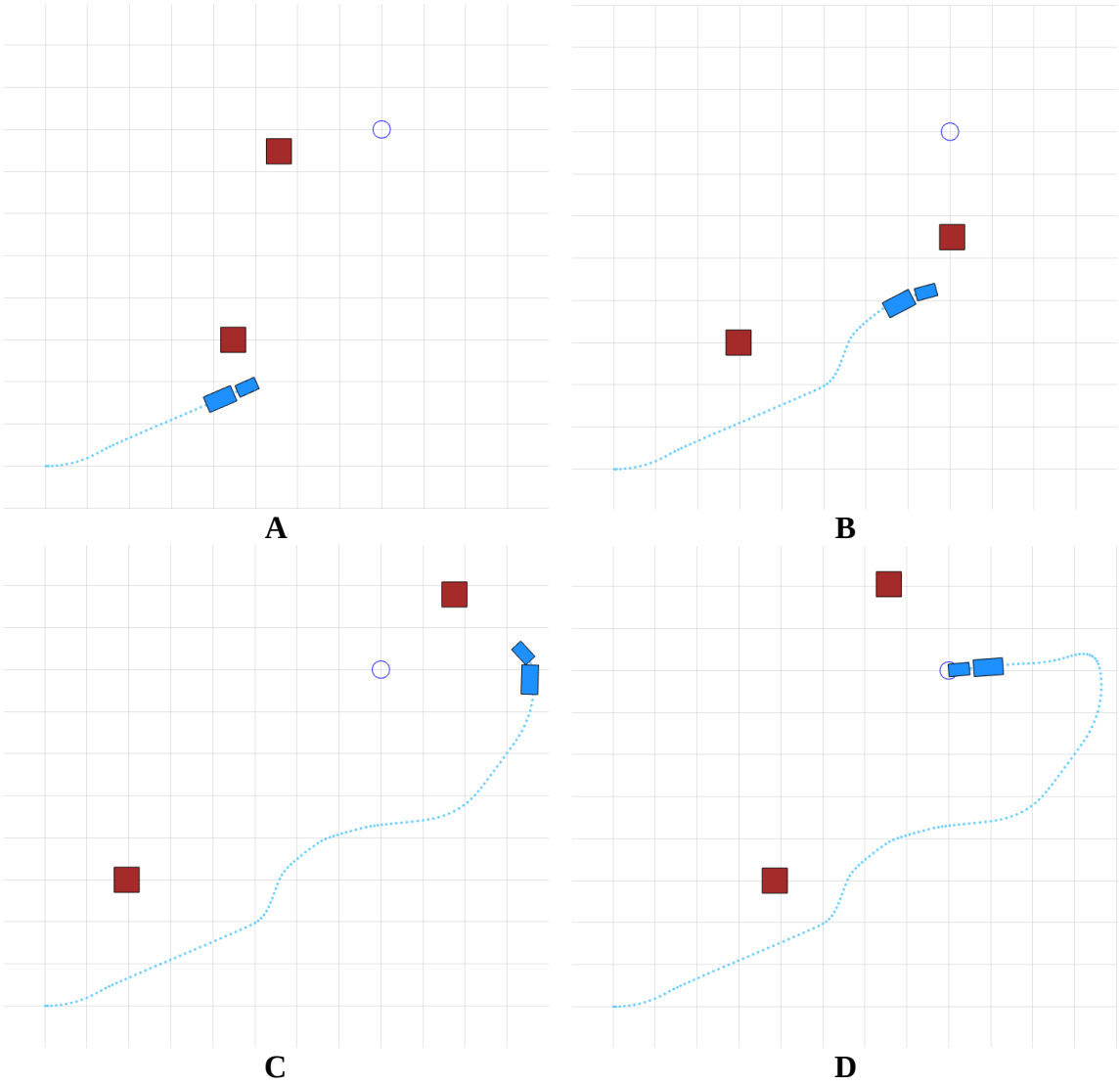


Figure 3.14 Grain cart with VFH planner dealt with dynamic obstacles. Sequential motion from (A) to (D).

3.1.3. Mobile Robot Tests

3.1.3.1. Simple Harvesting

In both test cases -- when the grain cart (Jackal) and combine (TerraSentia) faced the same direction or opposite directions when meeting for unloading -- the proposed navigation solution, implementing the APF+FLC planner along with the task scheduling

strategy, was able to direct the autonomous grain cart to accomplish all the logistical tasks. Illustrated in Figures 3.15 and 3.16, where the grain cart and combine faced the same direction or opposite directions, respectively, the autonomous grain cart stood by, went to the combine, unloaded the grain, went to the semi-trailer, transferred the grain, and returned to the standby point. Therefore, the effectiveness and practicality of the proposed navigation solution were verified.

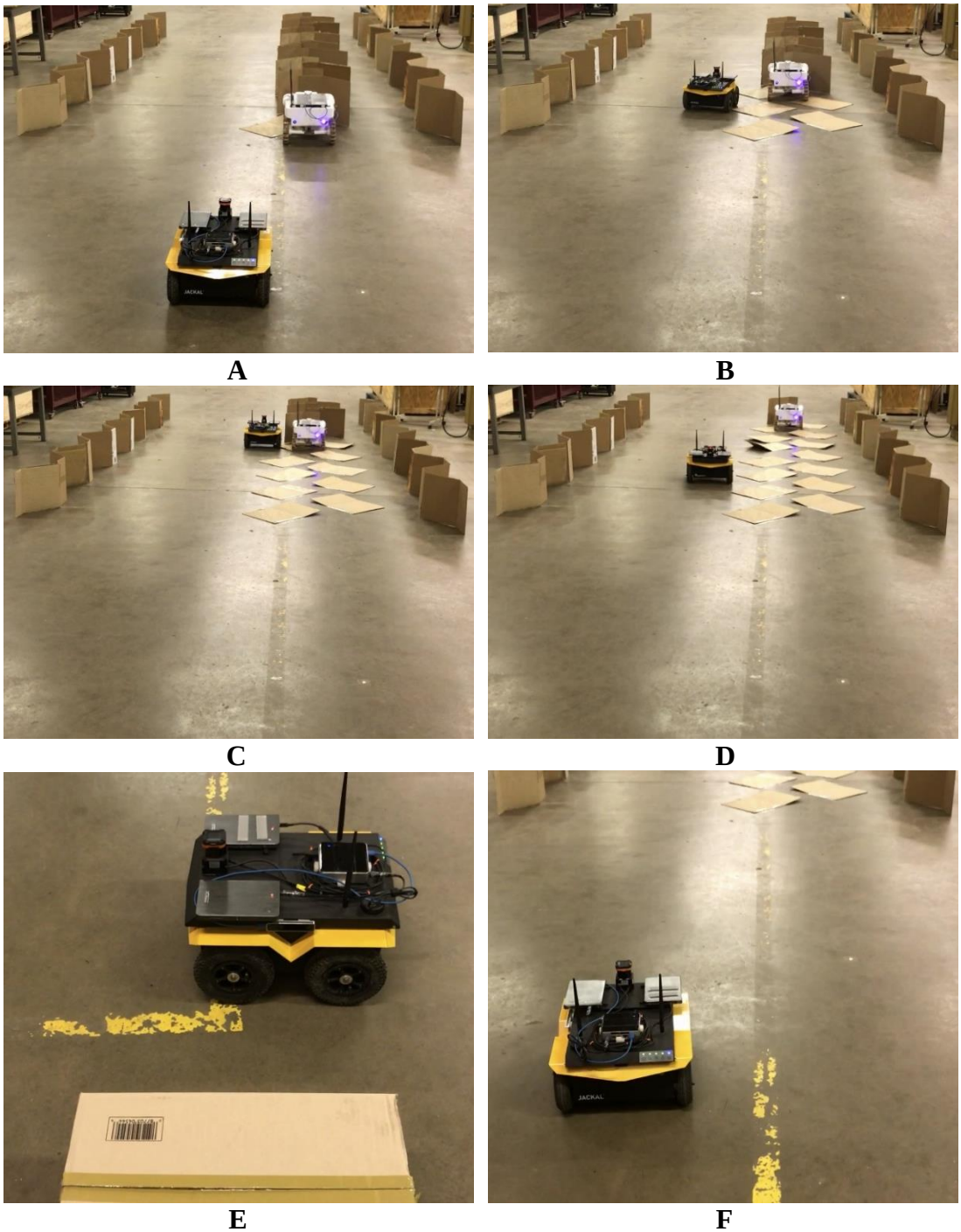
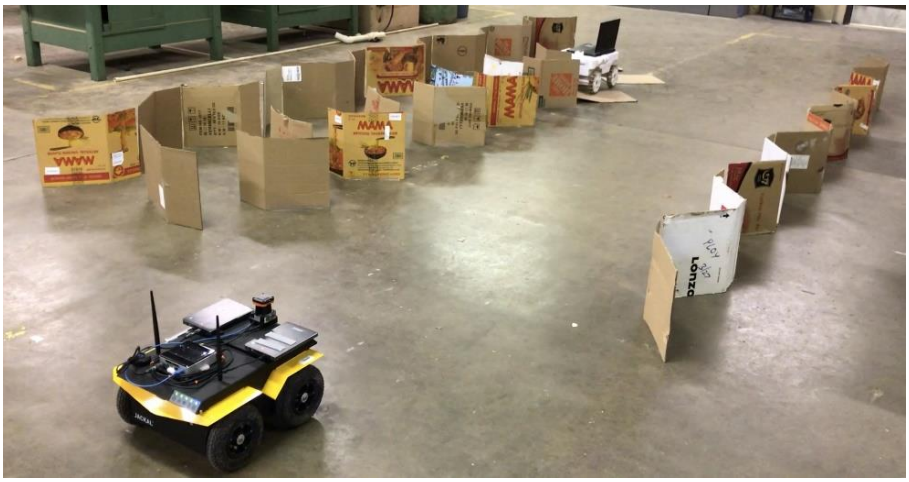
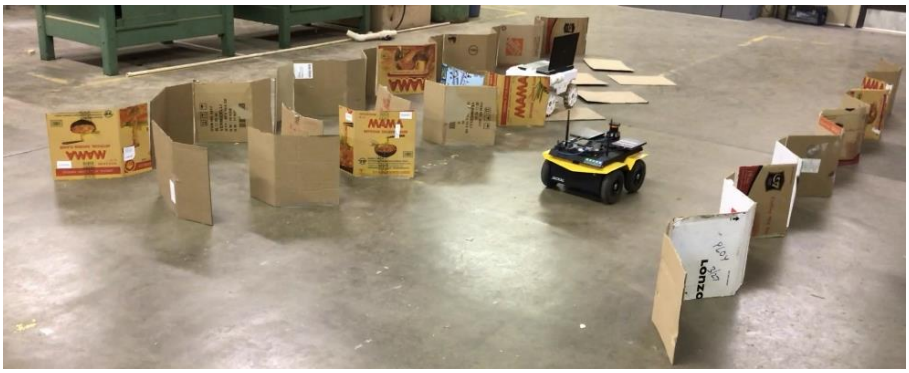


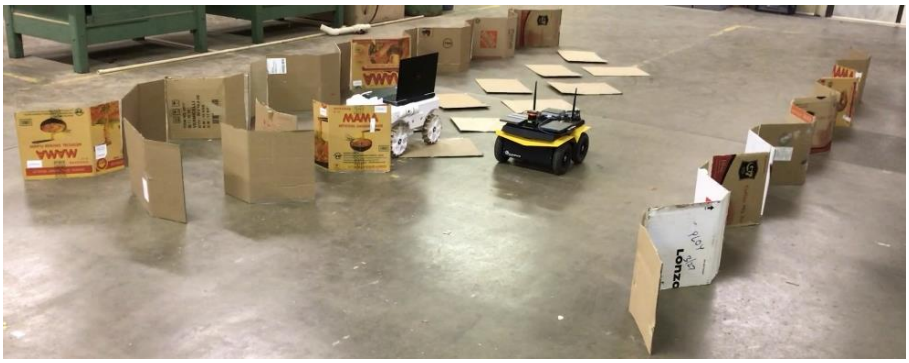
Figure 3.15 Grain cart accomplished logistical tasks in simple harvesting (combine had same facing). (A) Standby. (B) To combine. (C) Unload. (D) To semi. (E) Transfer. (F) To standby.



A



B

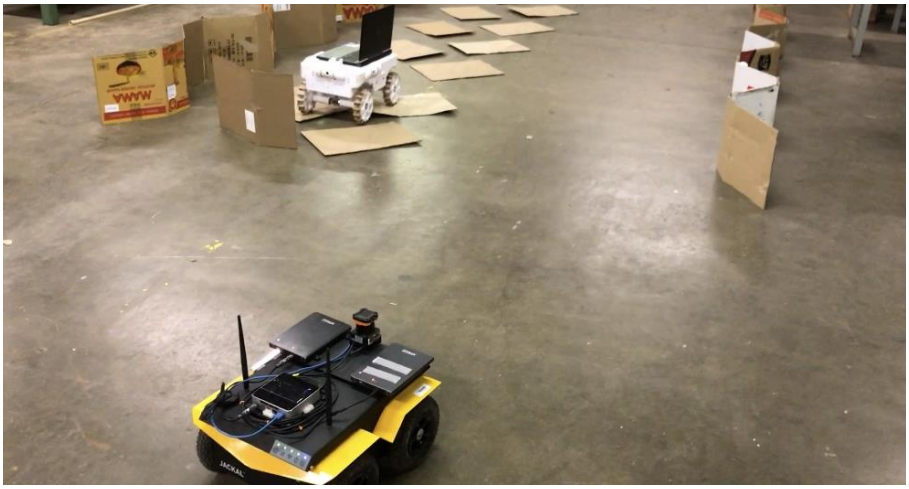


C

Figure 3.16 Grain cart accomplished logistical tasks in simple harvesting (combine had opposite facing). (A) Standby. (B) To combine. (C) Unload. (D) To semi. (E) Transfer. (F) To standby.

D

E

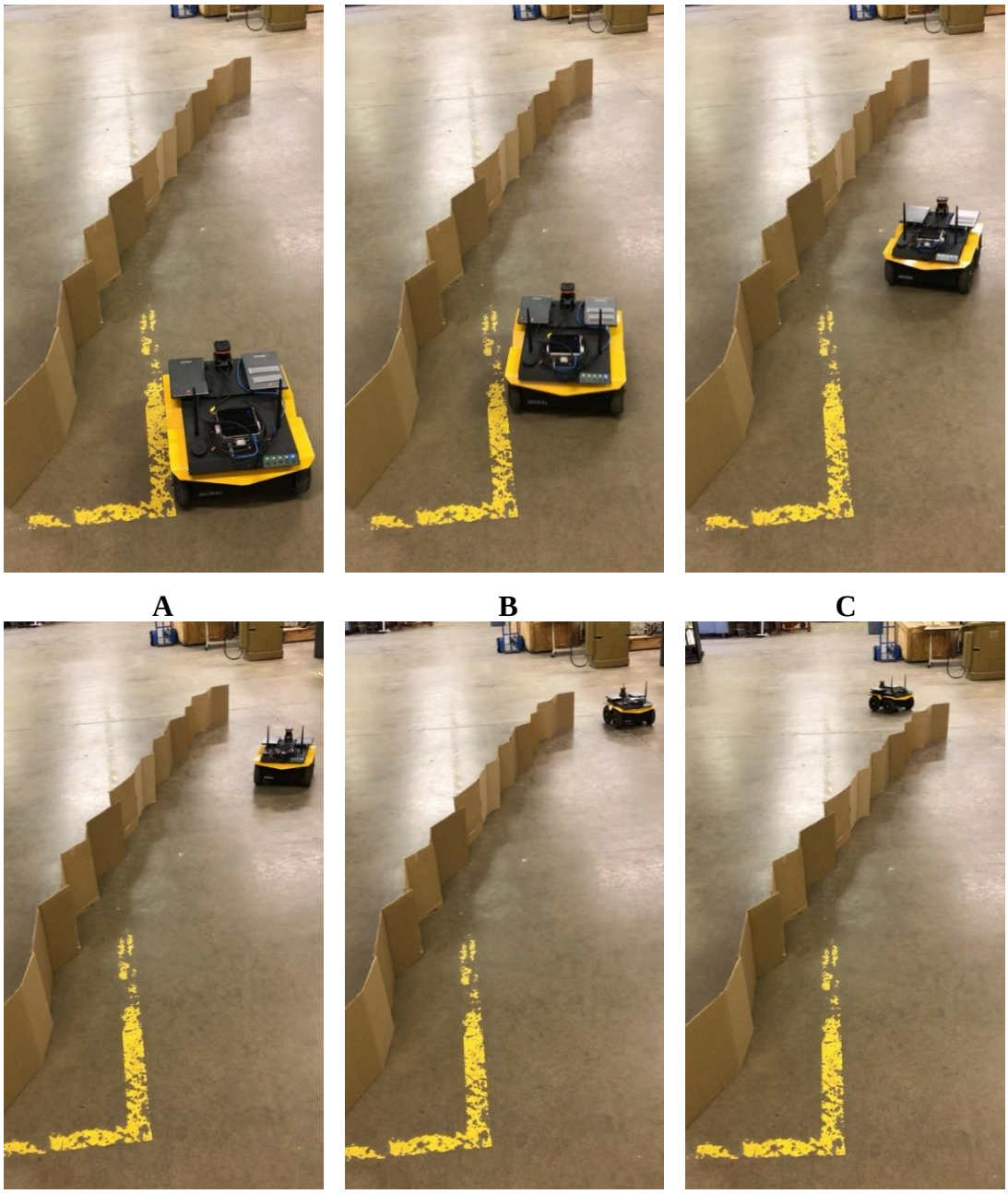


F

Figure 3.16 Continued.

3.1.3.2. Long Static Obstacle

As shown in Figure 3.17, when the Jackal faced the long static obstacle, the APF+FLC planner was effective in directing it to go around the obstacle smoothly and reach the goal behind it. The path the Jackal took was simple and efficient with a safe distance away from the long obstacle. Meanwhile, as shown in Figure 3.18, when the APF planner was in charge, the Jackal tended to go straight to the goal until it was dangerously close to the long obstacle and had to steer away. Every time an imminent collision was avoided, the Jackal tended to steer back to the original heading and proceeded, only to run into another imminent collision and had to steer away again. This tendency caused the Jackal to oscillate all the way until it eventually passed the long obstacle. The trajectory length in the two cases was, respectively, 8.77 m and 9.00 m. Although the lengths of the APF and APF+FLC trajectories were almost the same, the latter was considered more efficient, as it was more rational and over three times smoother (APF+FLC 0.015 rad^2 vs. simple APF 0.047 rad^2).



A **B** **C**

D **E** **F**

Figure 3.17 Jackal with APF+FLC planner went around long static obstacle. Sequential motion from (A) to (F).

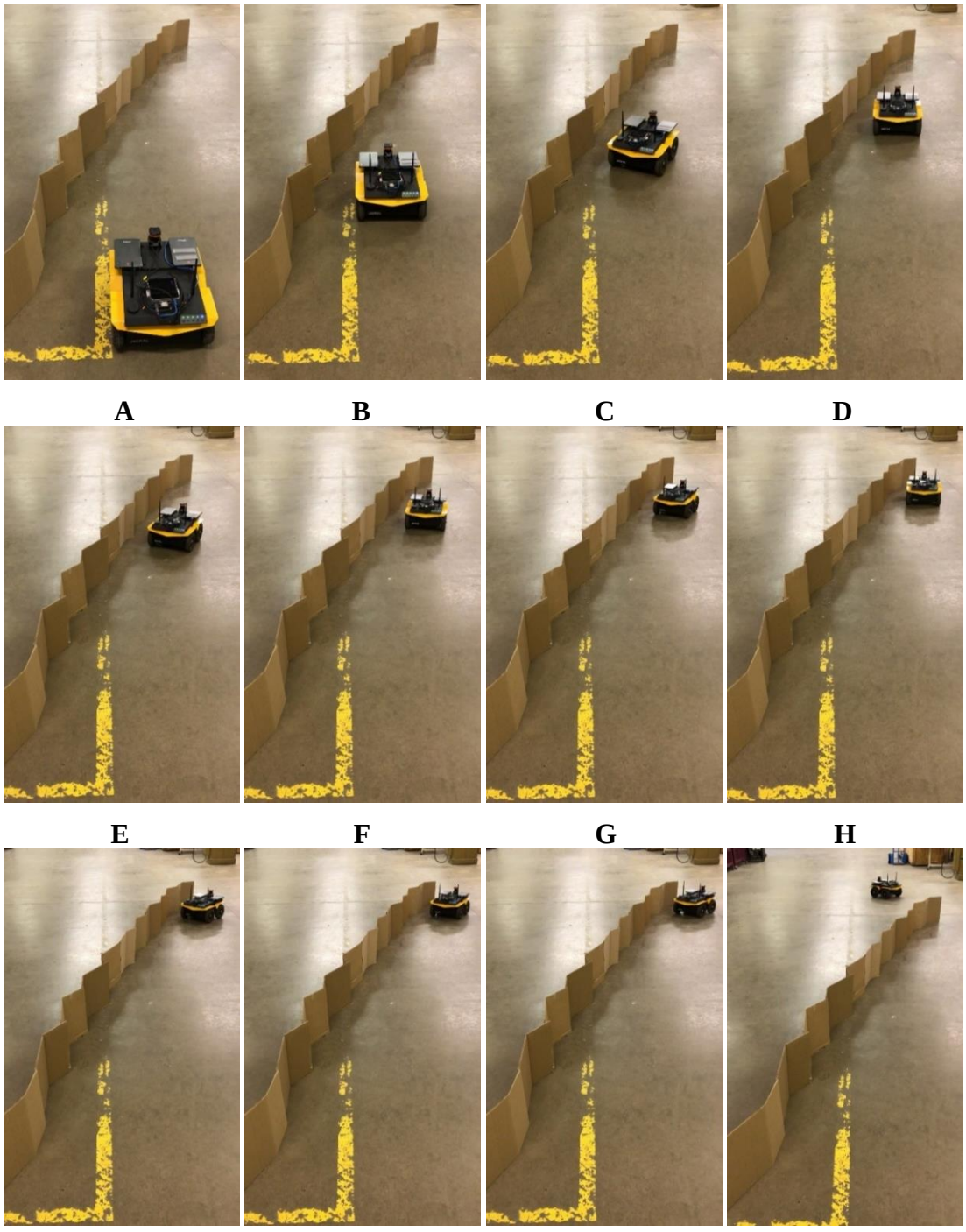
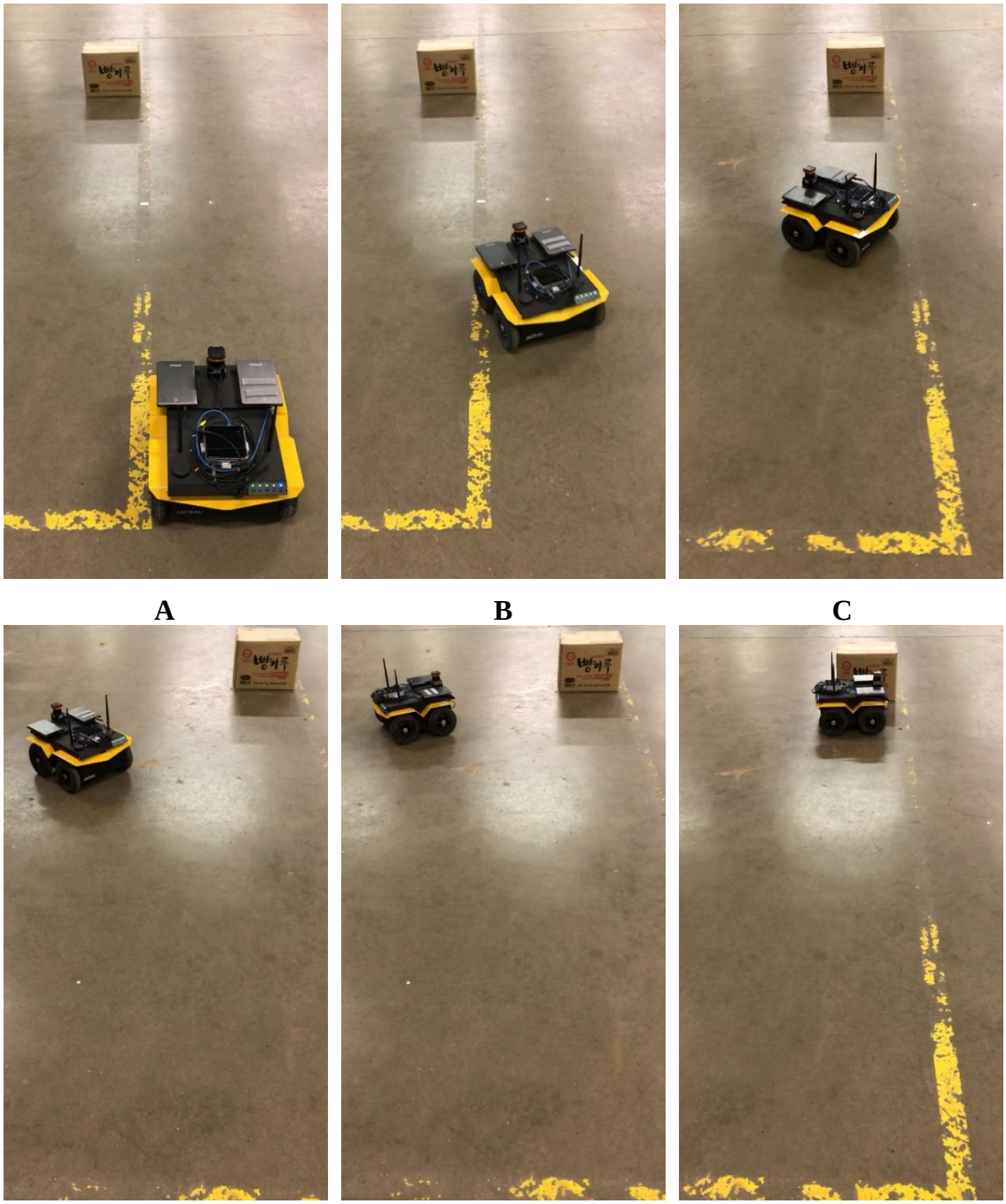


Figure 3.18 Jackal with APF planner oscillated going around long static obstacle. Sequential motion from (A) to (L).

3.1.3.3. Static Obstacle Close to Goal

In simple harvesting, the grain cart (Jackal) accomplished all the logistical tasks, including approaching and parking alongside the semi-trailer to transfer the grain. This task was separated to be a general test in which the goal was very close to a static obstacle. As shown in Figure 3.19, the APF+FLC planner was effective in directing the Jackal to approach the obstacle and stop right in front of it, just like the grain cart went to the semi-trailer and parked alongside in the simple harvesting test. This seemingly simple task turned out to be a great challenge for the APF planner. The goal was so close to the static obstacle that the attractive and repulsive forces created local minima around the area in front of the obstacle. Whenever the Jackal with the APF planner approached the goal, the obstacle behind it would "push" the Jackal away, preventing it from reaching the goal. As shown in Figure 3.20, after failing two times in approaching the goal by the obstacle, the Jackal was turning around for more attempts, which would always end up with the same failure.



A **B** **C**

D **E** **F**

Figure 3.19 Jackal with APF+FLC planner approached goal close to obstacle. Sequential motion from (A) to (F).

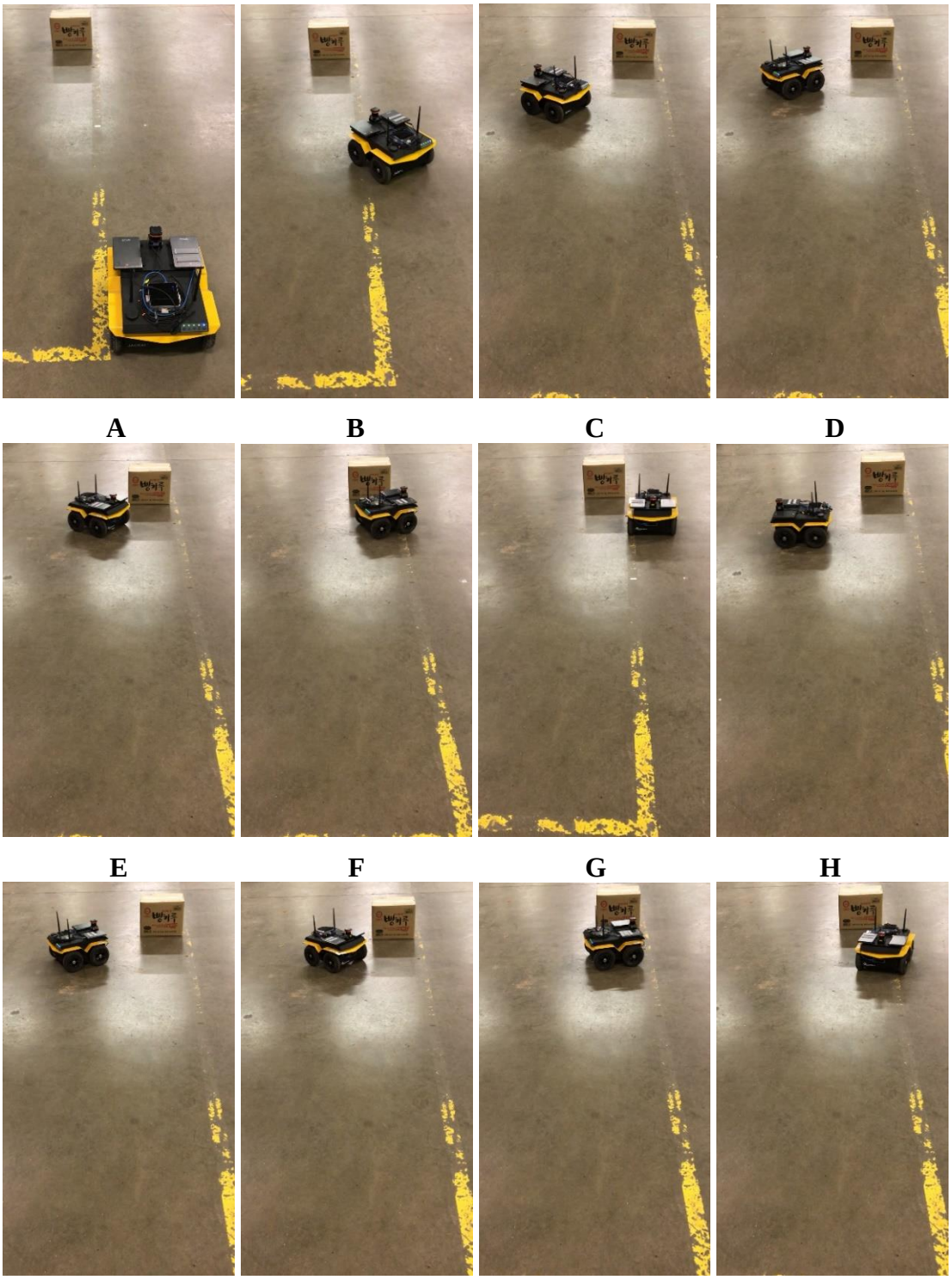
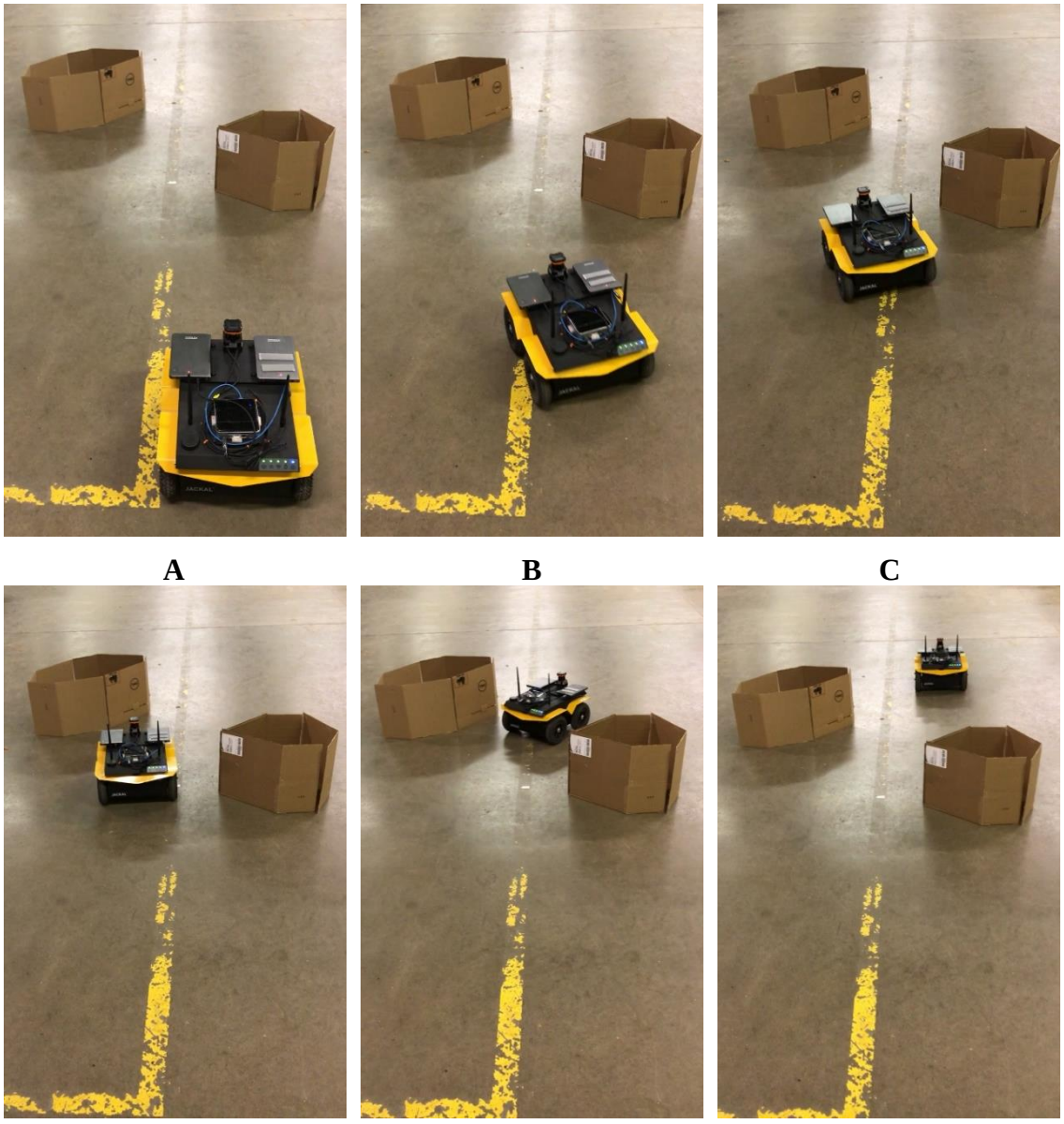


Figure 3.20 Jackal with APF planner failed approaching goal close to obstacle. Sequential motion from (A) to (L).

3.1.3.4. Closely Spaced Static Obstacles

The APF+FLC planner was effective in making use of the narrow gap between the two closely spaced static obstacles and directed the Jackal to efficiently pass through the gap to reach the goal (Figure 3.21), resulting in a trajectory length of 5.59 m and smoothness of 0.043 rad^2. To first go around the closer obstacle on the right with a safe distance, the Jackal inevitably got a little too close to the relatively farther obstacle on the left, where the repulsive force immediately steered the Jackal away from potential collisions, forcing it to make a relatively sharp turn to go through the gap. Illustrated in Figures 3.22, 3.23 and 3.24, the Jackal faced the same two obstacles but the directing APF planner had different repulsive ranges for the obstacles. In Figure 3.22, the obstacles were defined such that they had a repulsive range of 0.5 m, which did not repulse the Jackal until it traveled extremely close to the obstacles. Making sharp turns at the last minute, the Jackal eventually made it through the gap and reached the goal with a trajectory of 6.12 m in length and 0.088 rad^2 in smoothness, which made the APF+FLC trajectory 8.66% shorter and 51.14% smoother in comparison. When the repulsive range was increased to 1.0 m, the Jackal had similar behavior (Figure 3.23); it attempted to go straight towards the goal until it had to circumvent the obstacle to avoid imminent collisions. In addition, with the longer repulsive range of the obstacles, the Jackal experienced oscillation caused by the force contradiction passing through the gap. The APF trajectory with longer repulsive range was 7.00 m in length and 0.061 rad^2 in smoothness, while the APF+FLC trajectory was 20.14% shorter and 29.51% smoother. Illustrated in Figure 3.24, when the repulsive range was set to 1.5 m, the Jackal did not

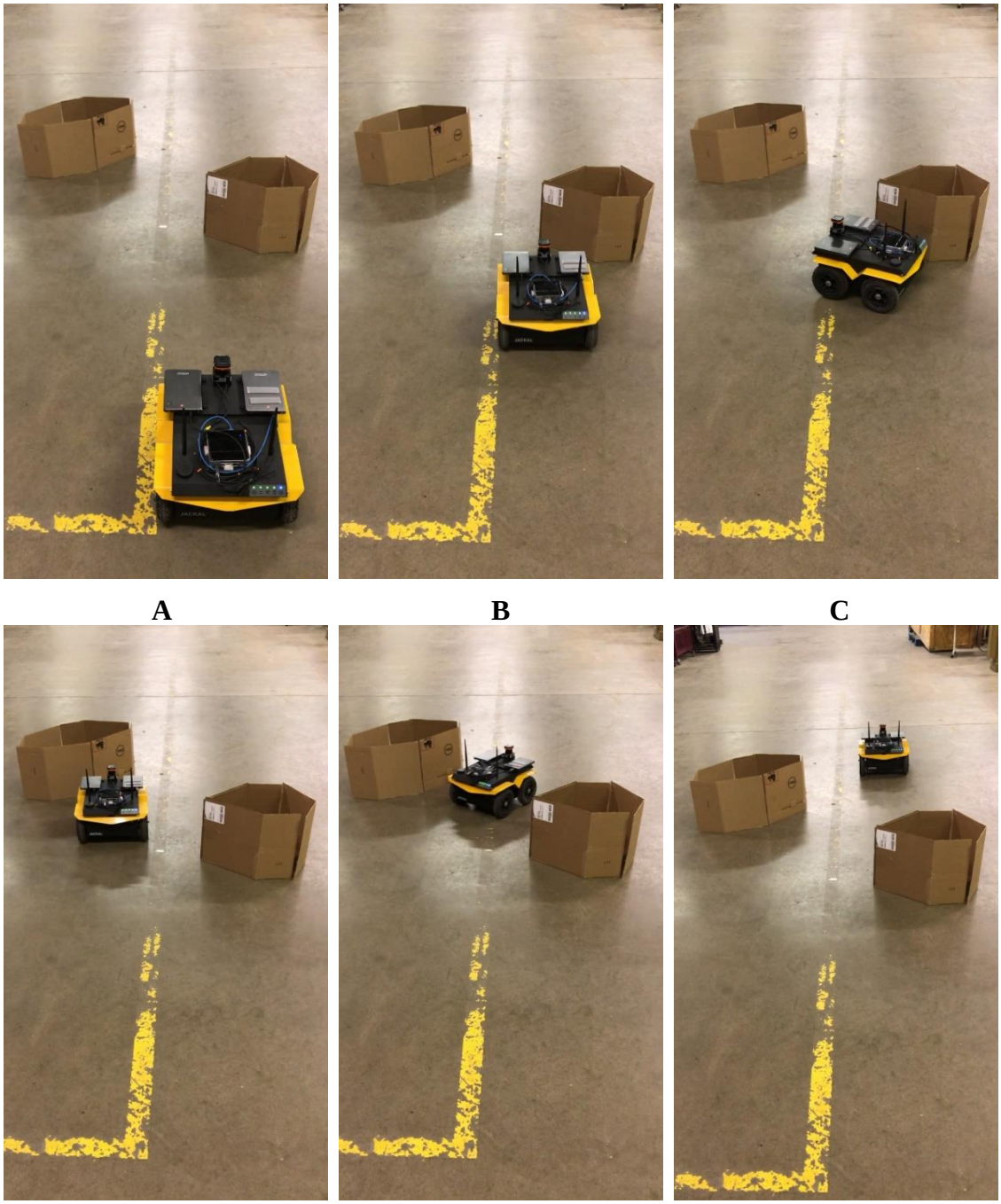
wait until the last minute to make sharp turns to avoid collisions, which seemed at first to be an improvement. However, the long repulsive range of the two obstacles prevented the Jackal from passing through the narrow gap, forcing the Jackal to go around the farther obstacle on the left to approach the goal. Compared with the resulting trajectory (APF, 1.5-m repulsive range) of 7.48 m in length and 0.085 rad^2 in smoothness, the APF+FLC trajectory was 25.27% shorter and 49.41% smoother.



A **B** **C**

D **E** **F**

Figure 3.21 Jackal with APF+FLC planner went between closely spaced obstacles. Sequential motion from (A) to (F).



A **B** **C**

D **E** **F**

Figure 3.22 Jackal with APF planner went between closely spaced obstacles (repulsive range = 0.5 m). Sequential motion from (A) to (F).

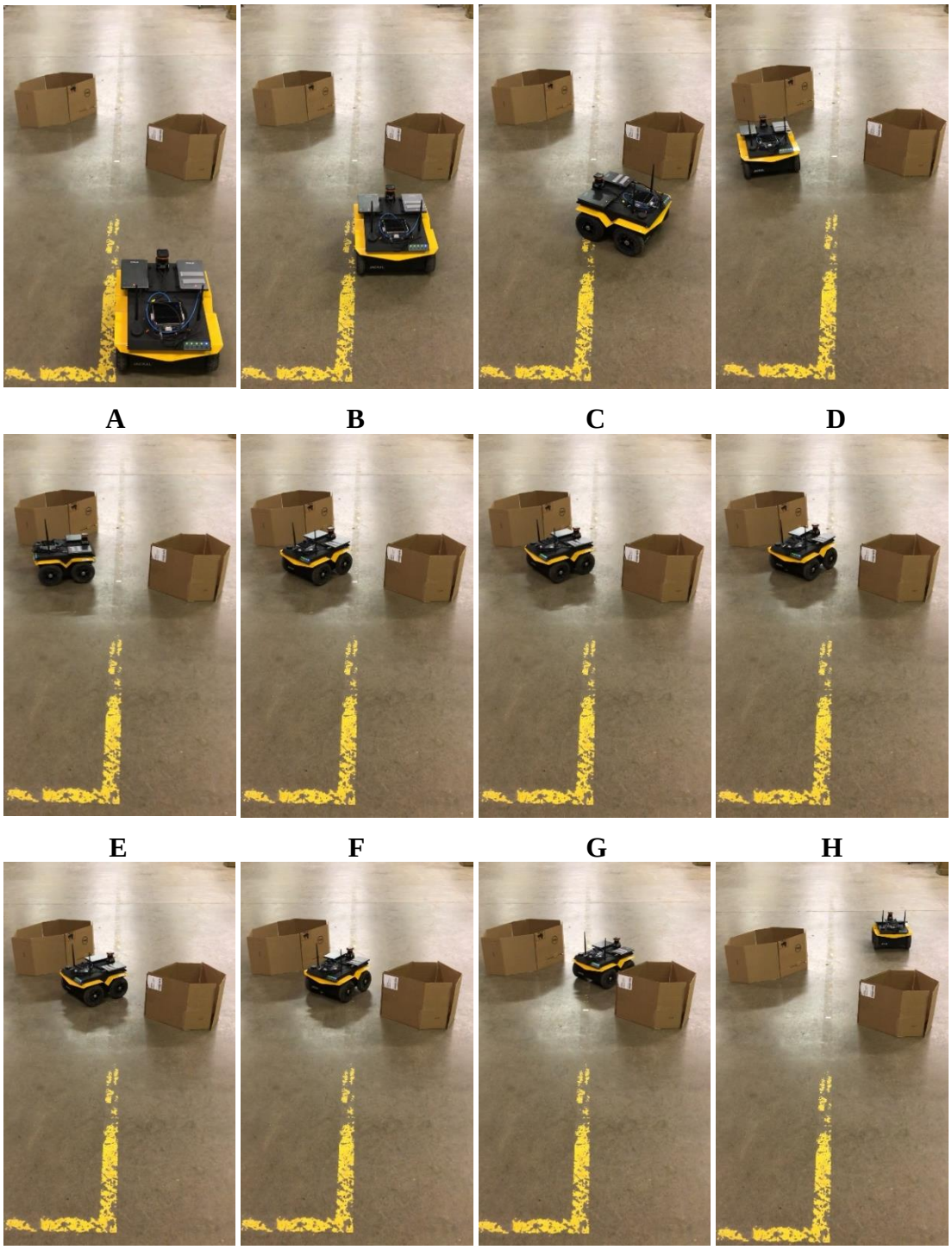
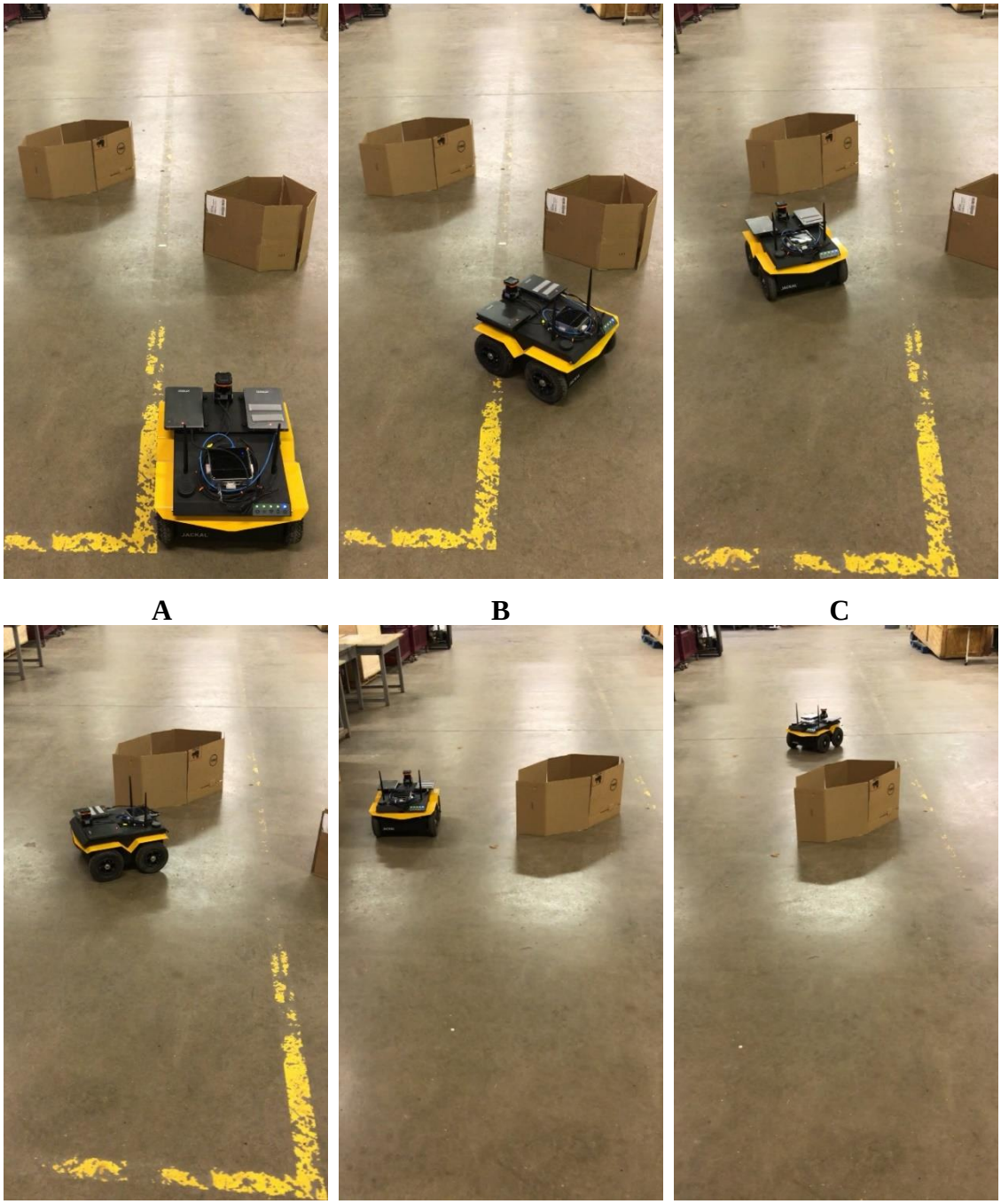


Figure 3.23 Jackal with APF planner went between closely spaced obstacles (repulsive range = 1.0 m). Sequential motion from (A) to (L).



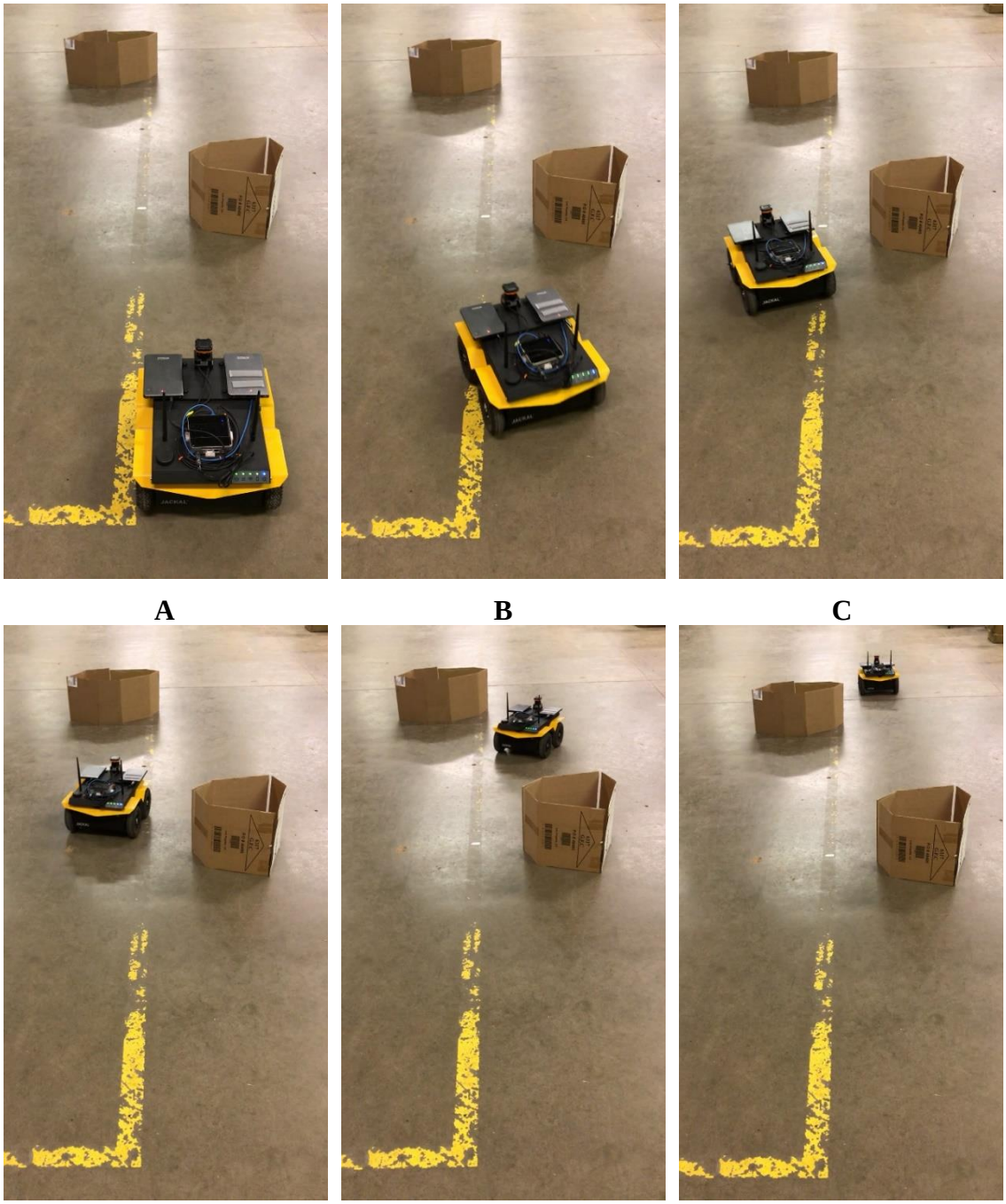
A **B** **C**

D **E** **F**

Figure 3.24 Jackal with APF planner went between closely spaced obstacles (repulsive range = 1.5 m). Sequential motion from (A) to (F).

3.1.3.5. Sparsely Spaced Static Obstacles

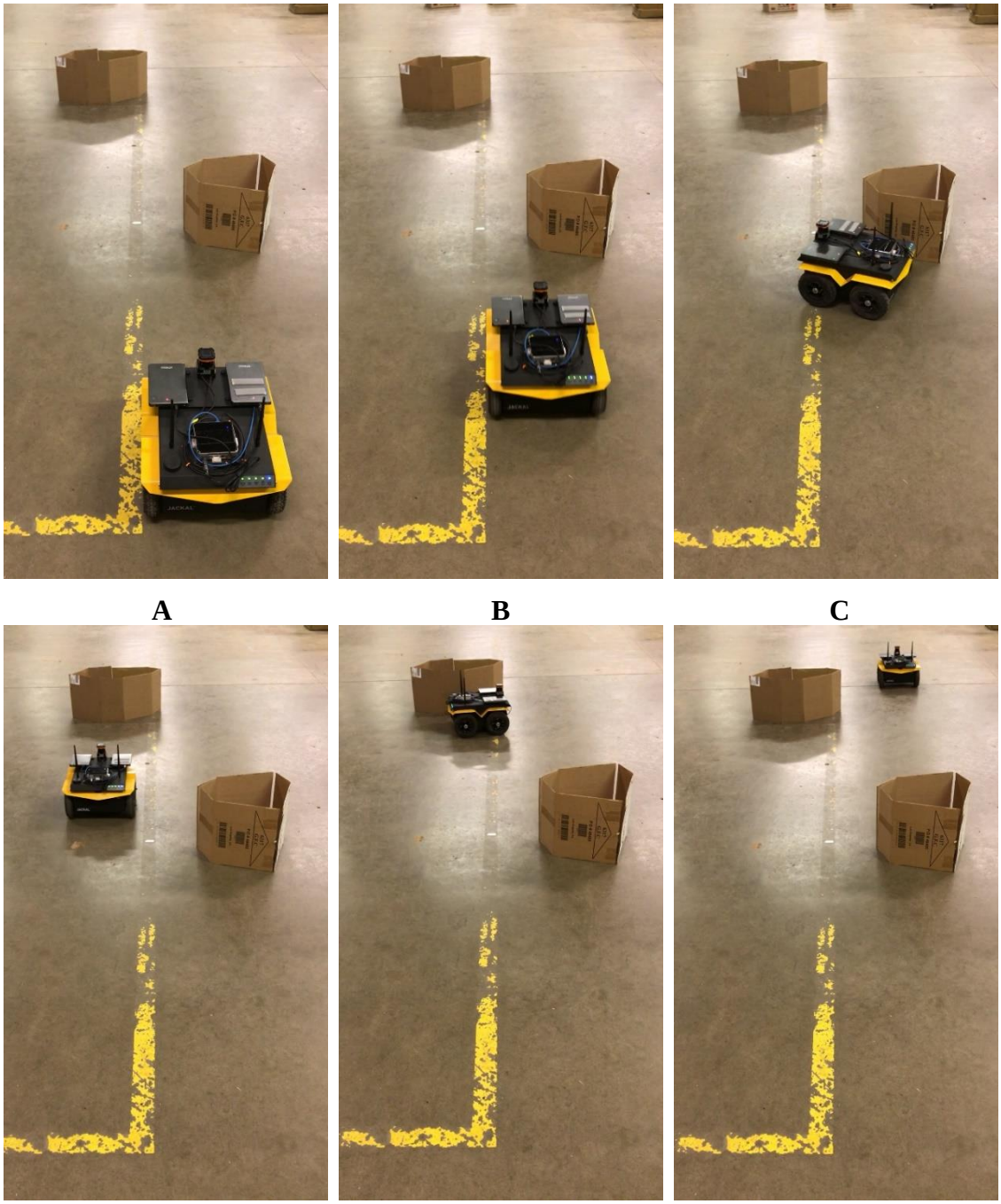
As shown in Figure 3.25, when the two obstacles were farther away from each other, the APF+FLC planner was effective in directing the Jackal to handle the obstacles one by one, taking a smoother path to reach the goal. In this test, because of more available space, the Jackal was always able to take early actions to efficiently go around the obstacles without making sharp turns. On the contrary, when the APF planner was in charge, the Jackal traveled without a global view, always going straight towards the goal until a collision was imminent. For the APF planner, the repulsive range of the obstacles was first set to 0.5 m (Figure 3.26), and then 1.5 m (Figure 3.27). In either case, the Jackal only changed directions when it was "sufficiently" close to the obstacles. In the 1.5 m case, the Jackal took steering actions a little earlier than in the 0.5 m case, which seemed at first to be an improvement. However, in the 1.5 m case, the Jackal circumvented the obstacles with an unnecessarily large gap in between, resulting in a trajectory of 7.52 m in length and 0.085 rad^2 in smoothness, while the 0.5 m case had a length of 6.80 m and a smoothness of 0.159 rad^2. Compared with these results, the APF+FLC trajectory of 6.32 m in length and 0.045 rad^2 in smoothness, was 15.96% and 7.06% shorter, and 47.06% and 71.70% smoother, in the 1.5 m and 0.5 m cases, respectively. Hence, the APF+FLC planner was more efficient than the APF planner in this test.



A **B** **C**

D **E** **F**

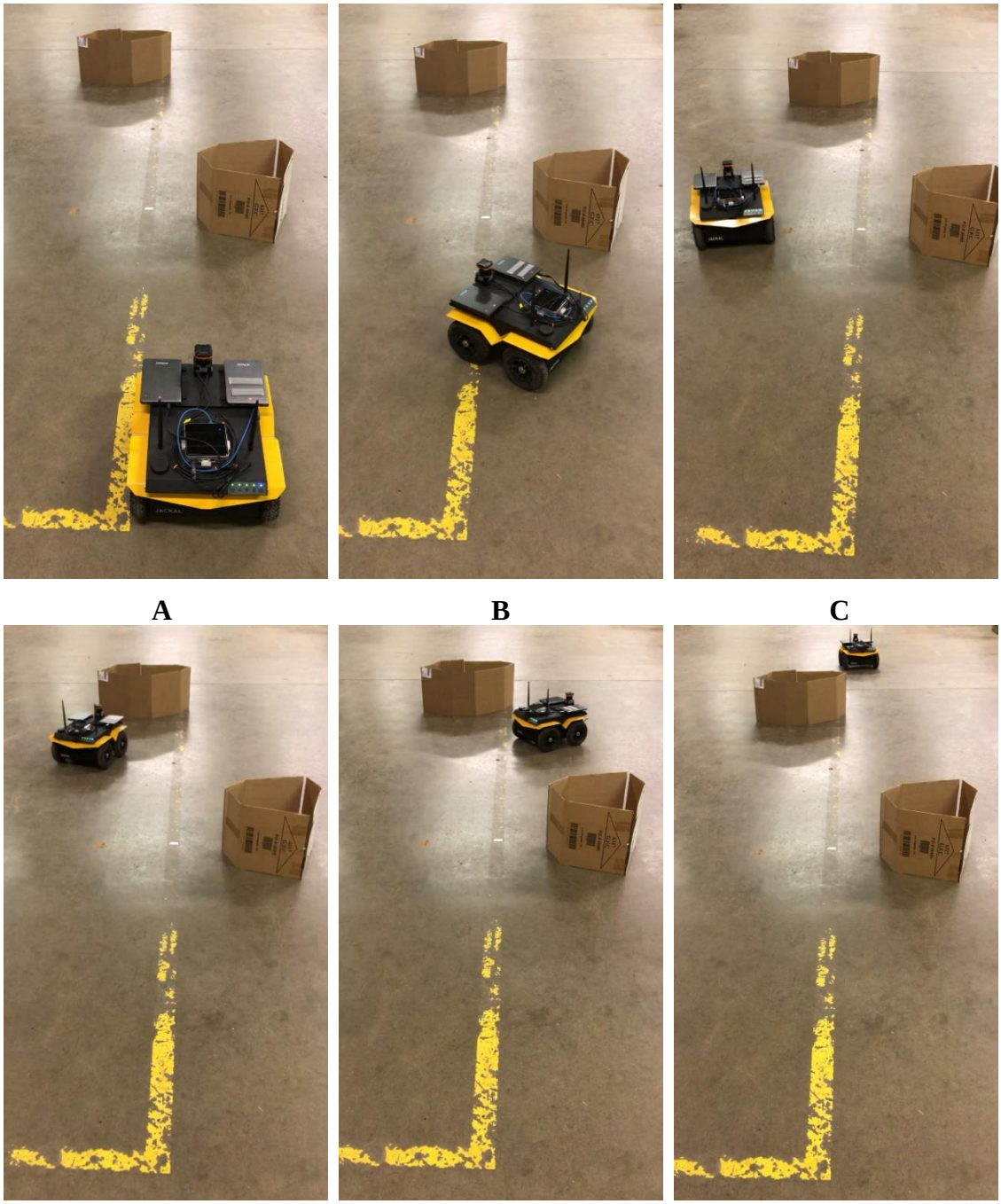
Figure 3.25 Jackal with APF+FLC planner went between sparsely spaced obstacles. Sequential motion from (A) to (F).



A **B** **C**

D **E** **F**

Figure 3.26 Jackal with APF planner went between sparsely spaced obstacles (repulsive range = 0.5 m). Sequential motion from (A) to (F).



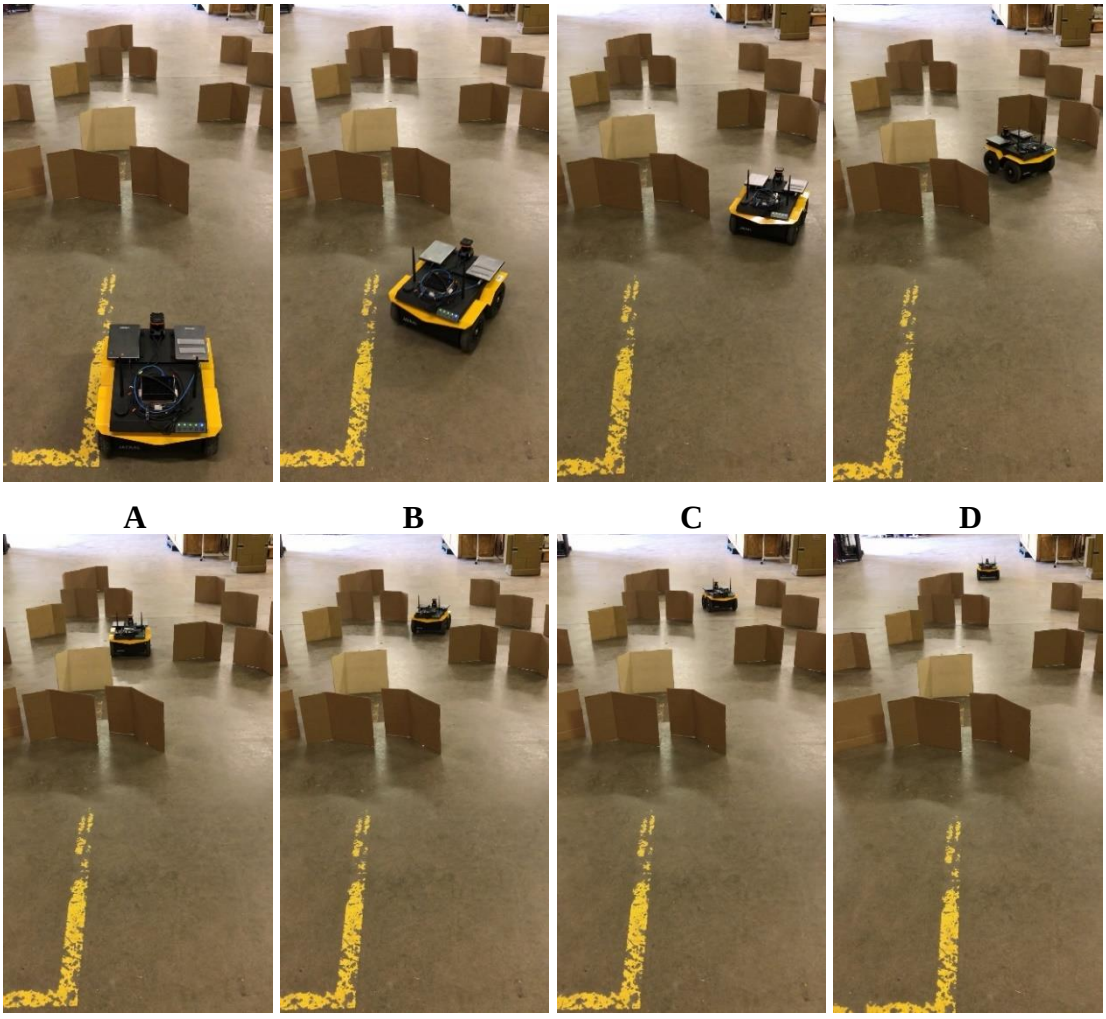
A **B** **C**

D **E** **F**

Figure 3.27 Jackal with APF planner went between sparsely spaced obstacles (repulsive range = 1.5 m). Sequential motion from (A) to (F).

3.1.3.6. Multiple Randomly Spaced Static Obstacles

Even facing a number of concave obstacles with misleading small gaps in between, the APF+FLC planner was effective in leading the Jackal with the most efficient motions to tackle the navigation task and eventually reach the goal (Figure 3.28). The length and smoothness of the trajectory were 8.96 m and 0.052 rad^2, respectively. Illustrated in Figures 3.29 and 3.30, the APF planner led the Jackal into the "embraces" of the obstacles. With either 0.5 m or 1.0 m repulsive range, the three cardboard sheets in the very front, which the Jackal ran into first, caused the Jackal to deviate from the most efficient path and to oscillate. Directed by the APF planner, the Jackal kept running towards the obstacles and made sharp turns at the last minute. The trajectory length of the APF+FLC planer was 21.68% shorter than that of APF in the 0.5 m case (11.44 m) and 25.95% shorter than that in the 1.0 m case (12.10 m). The trajectory smoothness of the APF+FLC planner was 70.79% higher than that of APF in the 0.5 m case (0.178 rad^2) and 51.85% higher than that in the 1.0 m case (0.108 rad^2). Hence, both the superior robustness and efficiency of the APF+FLC planner were verified.



A **B** **C** **D**

E **F** **G** **H**

Figure 3.28 Jackal with APF+FLC planner navigated between multiple static obstacles. Sequential motion from (A) to (H).

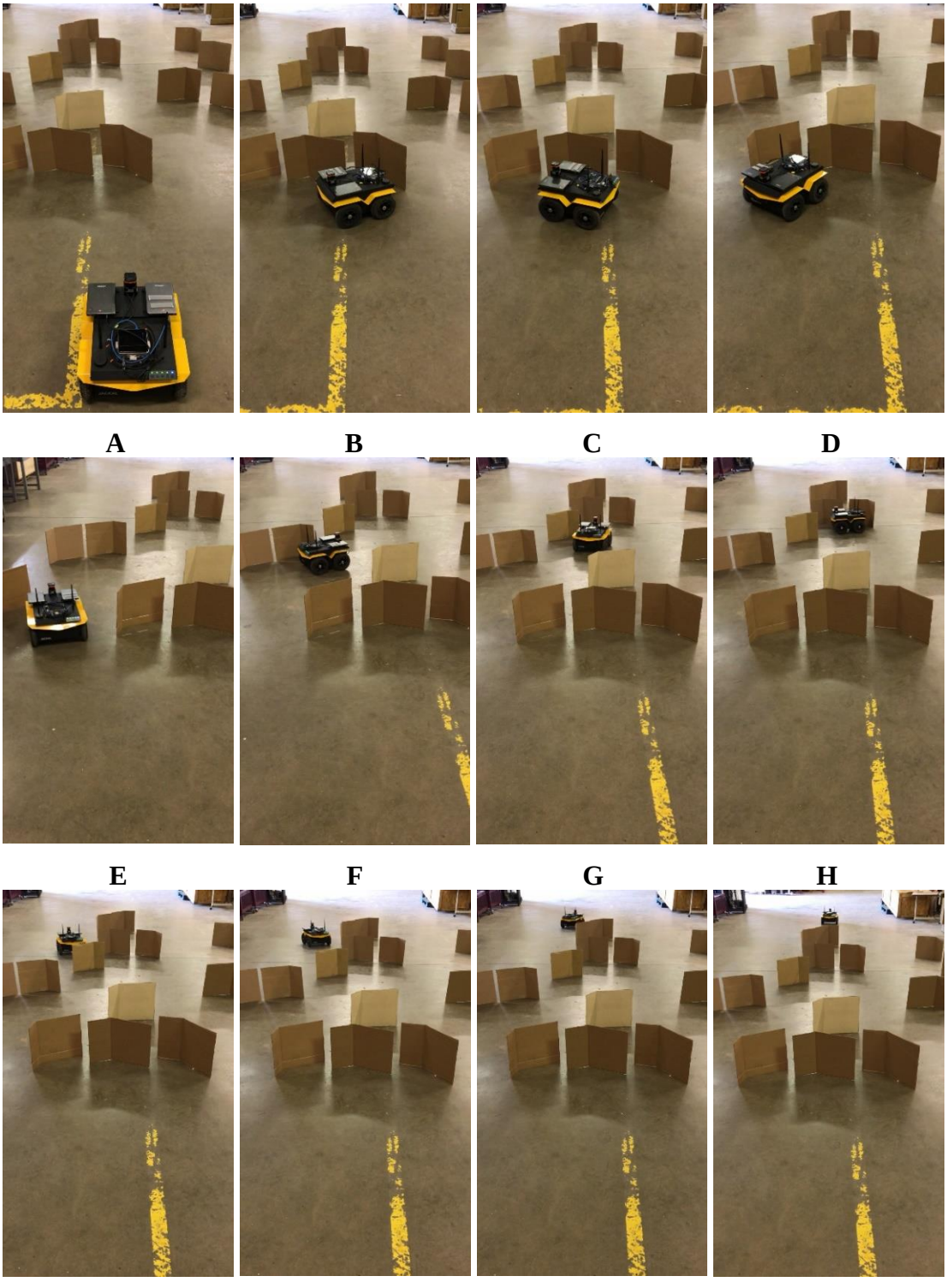


Figure 3.29 Jackal with APF planner navigated between multiple static obstacles (repulsive range = 0.5 m). Sequential motion from (A) to (L).

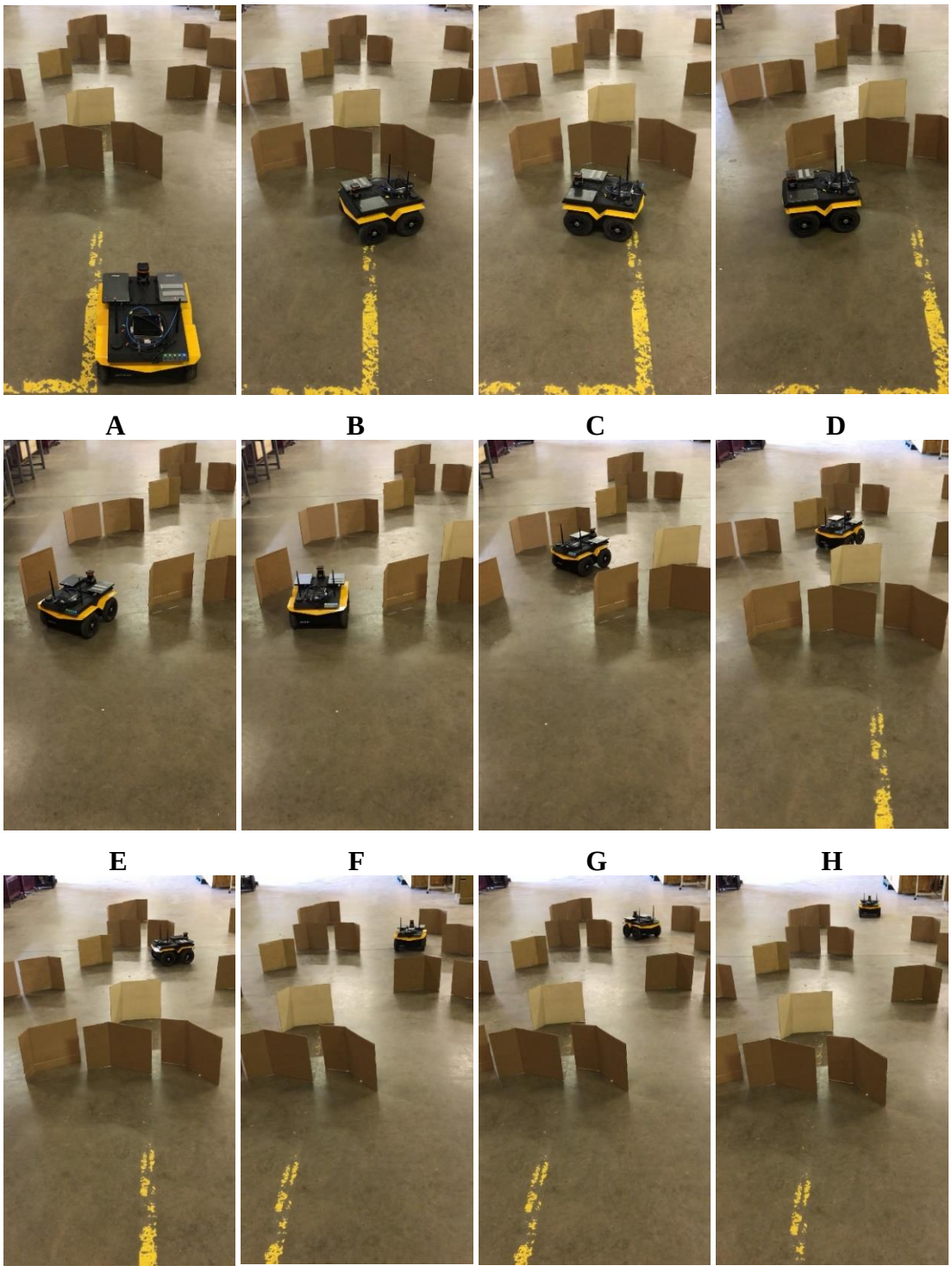


Figure 3.30 Jackal with APF planner navigated between multiple static obstacles (repulsive range = 1.0 m). Sequential motion from (A) to (L).

3.1.3.7. Dynamic Obstacle with No Threat

As shown in Figure 3.31, the APF+FLC planner was effective in leading the Jackal directly towards the goal in front, neglecting the TerraSentia that was traveling across the space. Although the dynamic obstacle could potentially run into the Jackal, based on its relative position the APF+FLC planner determined that the obstacle was neither blocking the straight path to the goal, nor actually threatening the Jackal. Successfully dealing with the approaching dynamic obstacle without overreacting to it verified the robustness and efficiency of the APF+FLC planner.

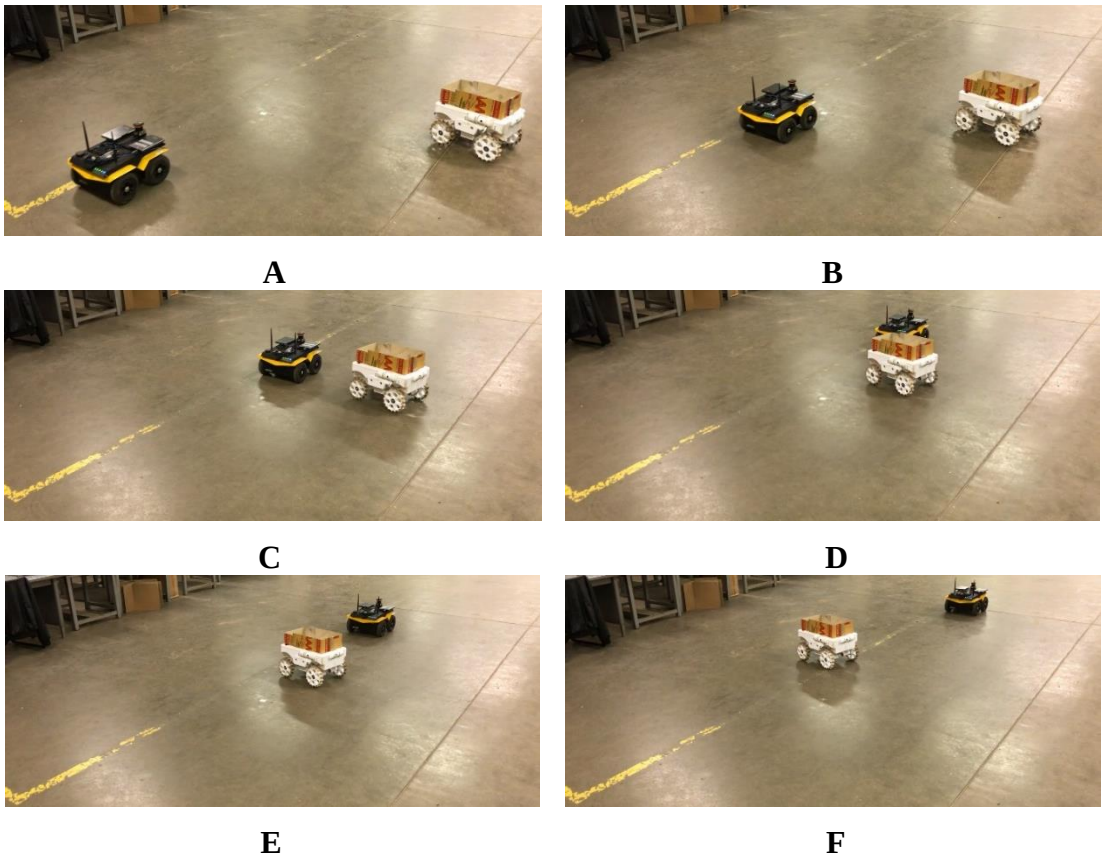
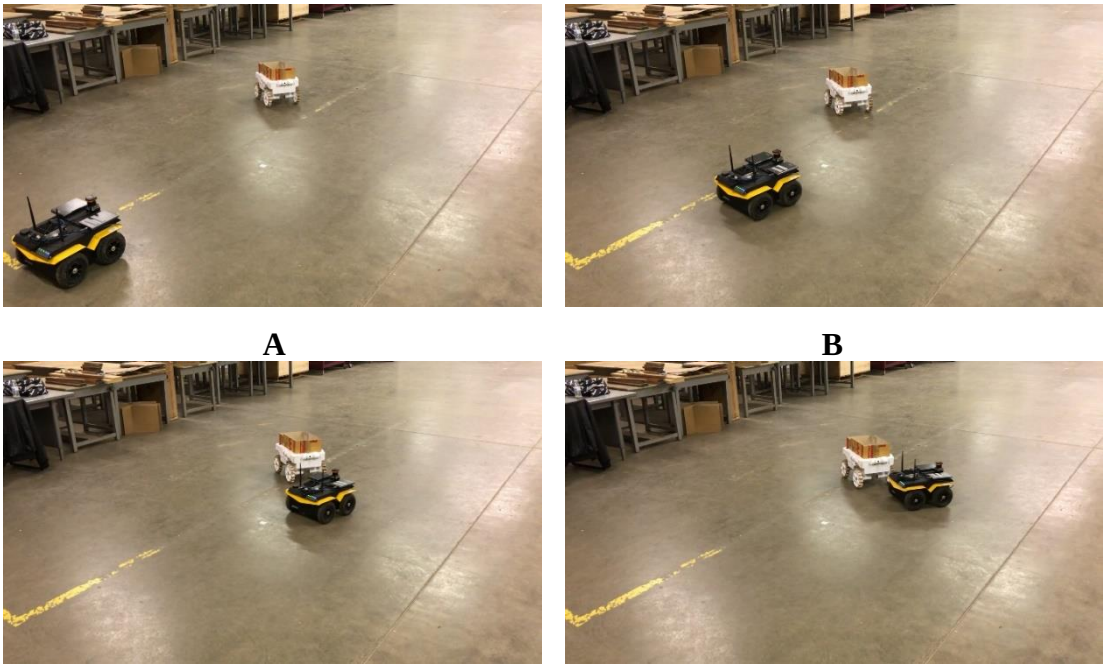


Figure 3.31 Jackal with APF+FLC planner dealt with dynamic obstacle with no threat. Sequential motion from (A) to (F).

3.1.3.8. Dynamic Obstacle in the Way

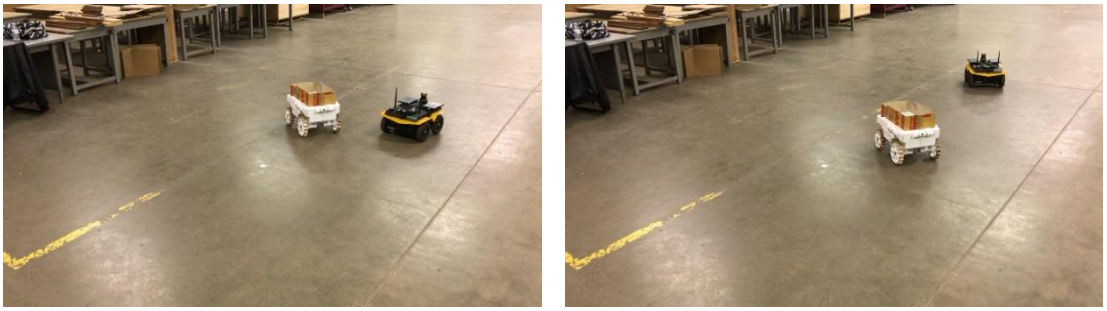
In this test, the APF+FLC planner was effective in directing the Jackal to go around the dynamic obstacle (TerraSentia) and reach the goal in front. In Figure 3.32, although the obstacle was moving across the path and could potentially crash into the Jackal, the APF+FLC planner determined that circumventing the obstacle from the front was still safe and more efficient. In contrast, in the case of Figure 3.33, the Jackal encountered the dynamic obstacle slightly earlier, which made the APF+FLC planner first try the path in front of the obstacle, but eventually direct the Jackal to go around the obstacle from the behind, a safer and more efficient path. This test also verified the robustness and efficiency of the APF+FLC planer.



A **B**

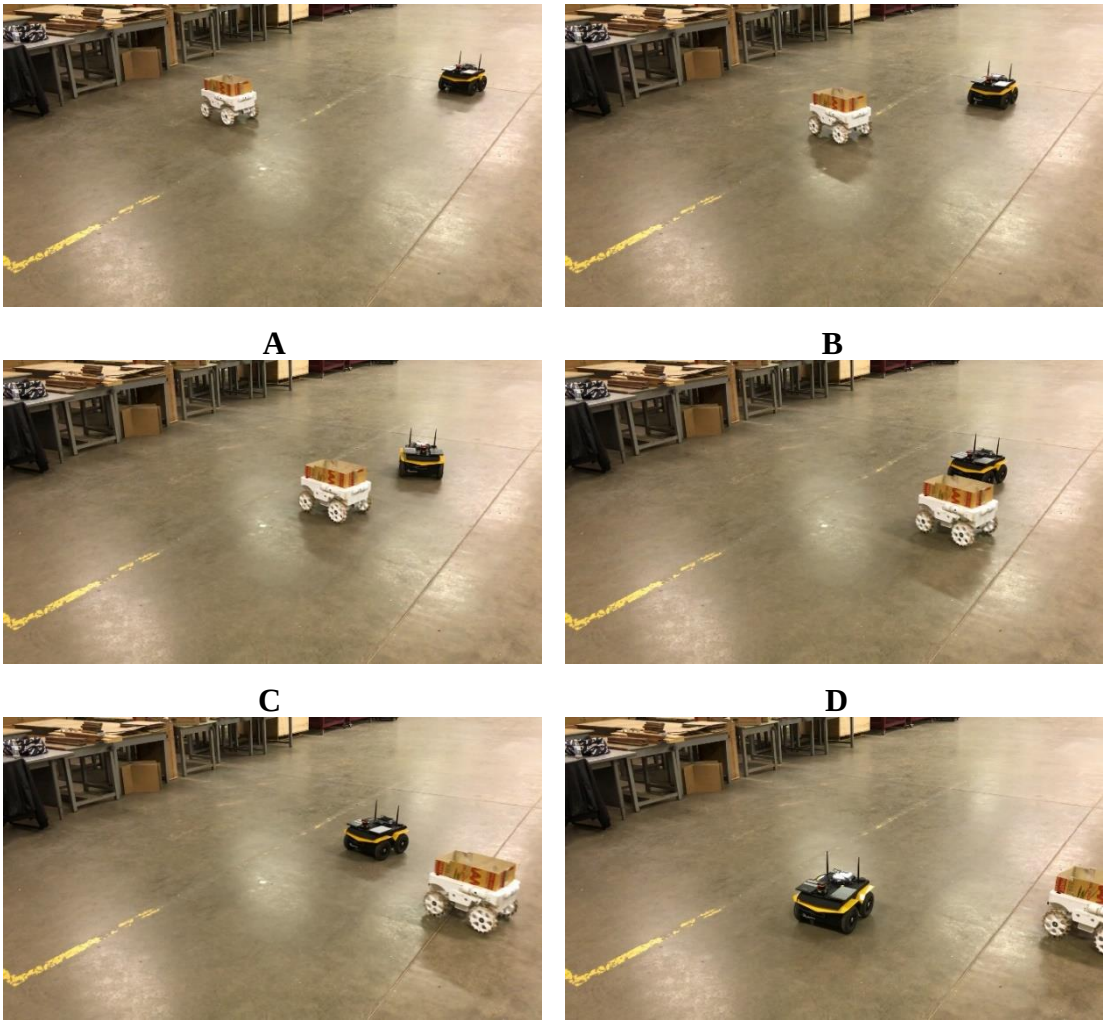
C **D**

Figure 3.32 Jackal with APF+FLC planner went around dynamic obstacle from front. Sequential motion from (A) to (F).



E **F**

Figure 3.32 Continued.



A **B**

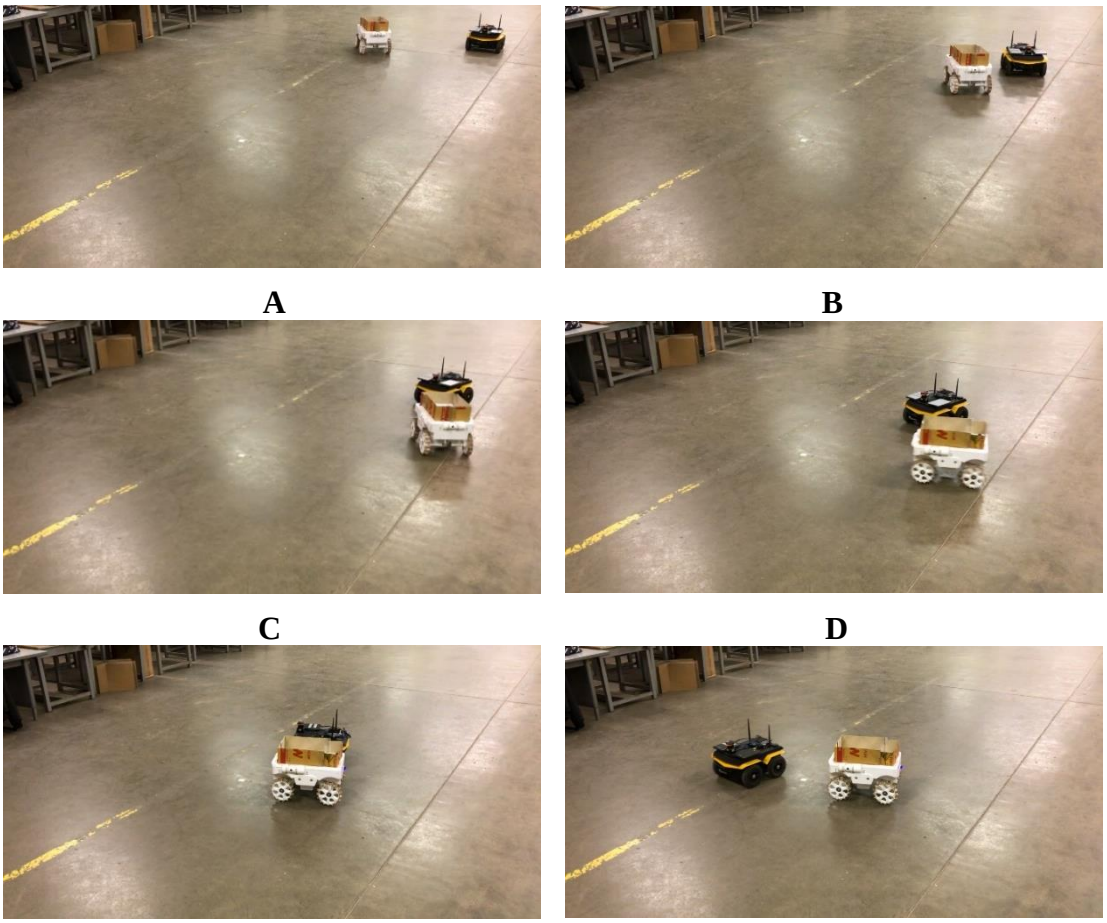
C **D**

E **F**

Figure 3.33 Jackal with APF+FLC planner went around dynamic obstacle from behind. Sequential motion from (A) to (F).

3.1.3.9. Malicious Dynamic Obstacle

In this test involving a dynamic obstacle that intentionally impeded the Jackal, the APF+FLC planner was effective in handling the dynamic obstacle (TerraSentia). The planner directed the Jackal to reach the goal in all three cases, in which the malicious dynamic obstacle (i) intentionally blocked the way, forcing the Jackal to make sharp turns (Figure 3.34); (ii) chased the Jackal and pressed hard on the side until the Jackal turned around (Figure 3.35); and (iii) sabotaged the navigation in both of the above ways (Figure 3.36). Being able to handle dynamic obstacles with unpredictable malicious behaviors, the APF+FLC planner was verified to be effective and robustness.

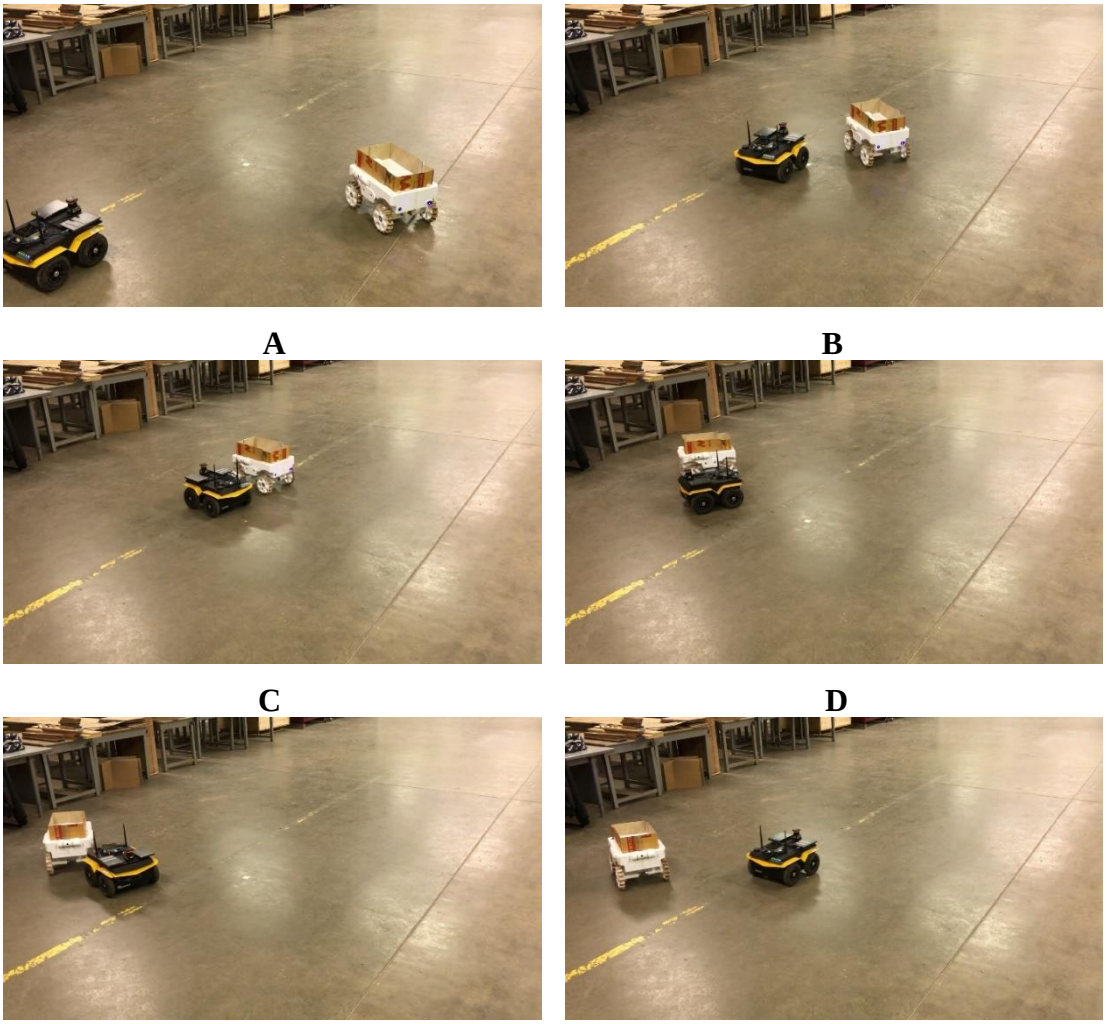


A **B**

C **D**

E **F**

Figure 3.34 Jackal with APF+FLC planner dealt with malicious dynamic obstacle: forced sharp turns. Sequential motion from (A) to (F).



A **B**

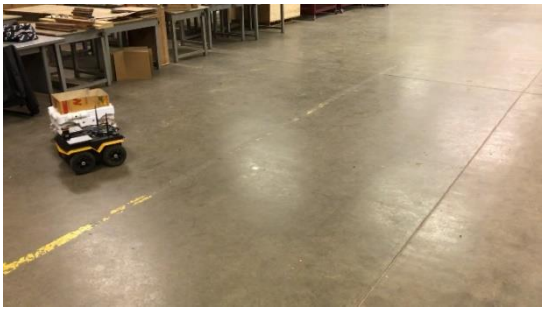
C **D**

E **F**

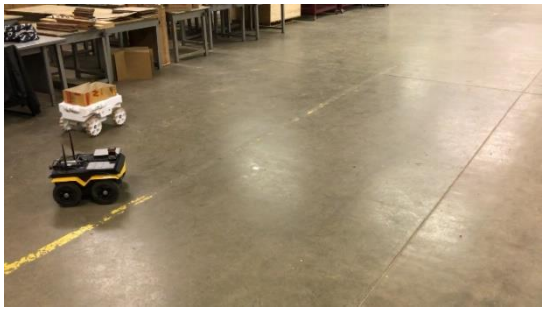
Figure 3.35 Jackal with APF+FLC planner dealt with malicious dynamic obstacle: forced turnaround. Sequential motion from (A) to (F).

A **B**

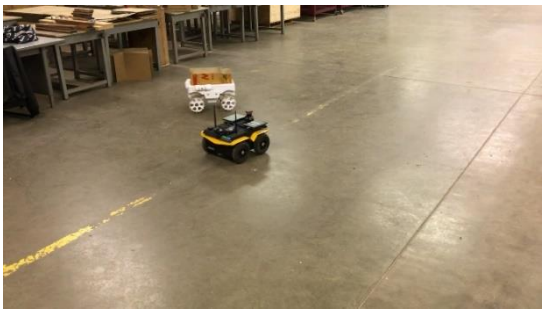
Figure 3.36 Jackal with APF+FLC planner dealt with malicious dynamic obstacle: constant chasing. Sequential motion from (A) to (J).



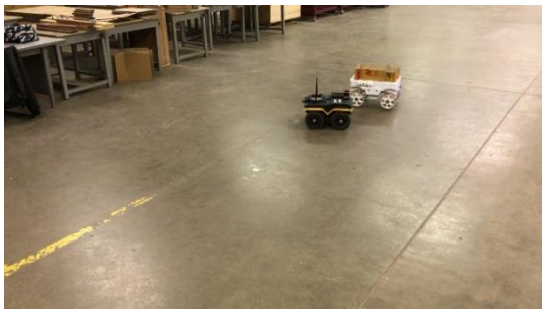
C



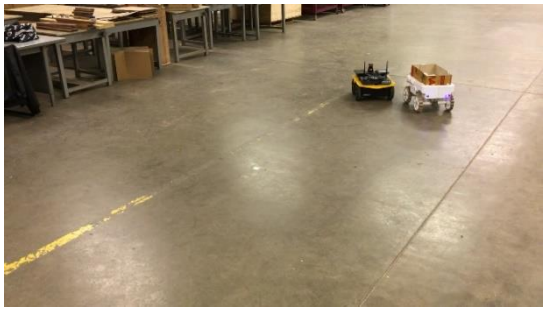
D



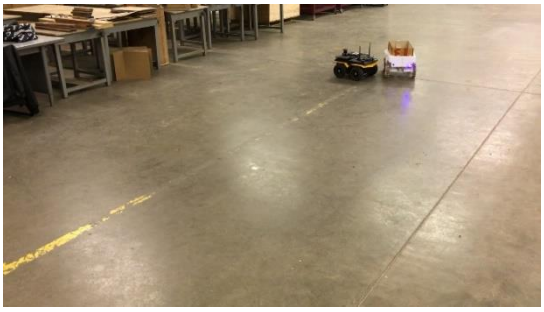
E



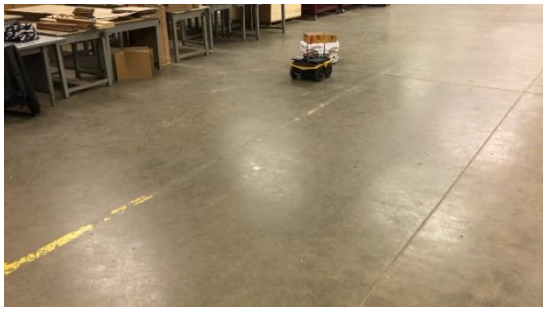
F



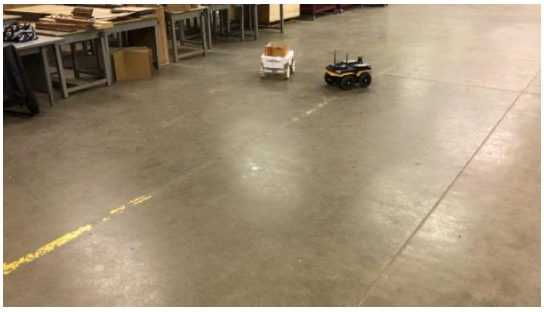
G



H



I



J

Figure 3.36 Continued.

3.2. Conclusions

This study focused on the development of a motion planning algorithm for autonomous grain carts to intelligently navigate in crop fields and efficiently service combines in harvest operations. A high-level navigation solution was also proposed for implementing the motion planning algorithm.

Based on the results of the simulation tests, it can be concluded that the developed motion planning algorithm and the associated task scheduling strategy for autonomous navigation of grain carts were effective, robust, efficient and computationally expeditious.

(i) Effectiveness. The autonomous grain cart was able to navigate in the field intelligently and accomplish all the logistical tasks in the harvest operations, including standby, go to combine, unload grain, go to semi-trailer, transfer grain, and return to standby point. Navigating in the field, the grain cart was able to circumvent the crop rows when they were unharvested and leverage the harvested areas for efficient traverse. When meeting the combine for unloading, the autonomous grain cart spent negligible time following the combine before unloading started, preventing excessive fuel consumption or interruption of the harvest operation. In summary, the motion planning algorithm and the task scheduling strategy have addressed both the spatial and temporal constraints. In addition, the proposed motion planning algorithm was effective in guiding the grain cart to accomplish more general navigation tasks like going around long obstacles, passing through narrow gaps, and avoiding collisions with dynamic obstacles.

(ii) Robustness. In addition to simple harvesting where unharvested crop rows were the only obstacles, harvest operations involving random static and dynamic obstacles as well as more general test cases were also simulated. The autonomous grain cart, directed by the proposed motion planning algorithm, successfully handled the different configurations of static and dynamic obstacles and managed to accomplish the navigation tasks.

(iii) Efficiency. In the tests comparing the proposed APF+FLC planner with the simple APF planner, either simple harvesting or more complex harvesting involving static or dynamic obstacles, the motion plans generated by the APF+FLC planner were always smoother and more rational, leading the autonomous grain cart to take more efficient routes with shorter travel distance. In contrast, the simple APF planner experienced problems such as oscillations and local minima traps, which significantly degraded the navigation performance of the autonomous grain cart. In the more general tests comparing the proposed APF+FLC planner with the VFH planner, the motion plans generated by the APF+FLC planner tended to be less smooth than those in the VFH cases, but they were more rational and efficient as their trajectory lengths were much shorter. In contrast, the VFH planner was often unable to take advantage of shortcuts and tended to be misled by tricky obstacle configurations.

(iv) Computational ease. Running on an ordinary CPU that can be easily accessed with a reasonable cost, the motion planning algorithm consumed little time in each computation step to process the sensing information and generate updated motion plans. This computational expense was greater than that of the simple APF planner (as

expected), but much less than that of the VFH planner, and was sufficiently low for real-time applications.

Based on the results of the mobile robot tests, it can be concluded that the proposed high-level navigation solution for implementing the motion planning algorithm was effective and practical. Additionally, the effectiveness, robustness and efficiency of the motion planning algorithm were further verified.

(i) Effectiveness and practicality of the navigation solution. The autonomous grain cart (Jackal) was able to navigate between the crop rows (cardboard sheets) and accomplish the logistical tasks in the harvest operations. When the combine (TerraSentia) was traveling in the same direction, the grain cart approached directly to meet for unloading; when the combine was traveling in the opposite direction, the grain cart made a U-turn to approach and align heading for unloading. Therefore, the proposed navigation solution featuring 2D-lidar-based local perception, IMU-based vehicle states measurement, Wi-Fi-supported V2V communication and drive-by-wire actuation has been verified to be effective and practical.

(ii) Effectiveness and robustness of the motion planning algorithm. Besides the simple harvesting test, a number of more general tests with various obstacle avoidance tasks were carried out. These tasks included single long static obstacle, single static obstacle close to the goal, two closely spaced and two sparsely spaced static obstacles, multiple randomly spaced static obstacles, dynamic obstacle with no threat, dynamic obstacle crossing the space, and malicious dynamic obstacle intentionally sabotaging the navigation. Dealing with this variety of navigation tasks with different obstacle

configurations, the proposed motion planning algorithm experienced no difficulty and was always able to generate intelligent motion plans for the Jackal to accomplish the tasks, verifying the effectiveness and robustness of the proposed motion planning algorithm.

(iii) Efficiency. In some of the aforementioned general test cases, the APF+FLC planner was compared with a simple APF planner. The simple APF planner sometimes ran into problems such as oscillations and local minima traps. Adjusting the repulsive range of obstacles did not solve the problems, but instead raised other issues. Meanwhile, the APF+FLC planner was always able to generate efficient motion plans for the Jackal to accomplish the navigation tasks. Being smoother and more rational than the APF paths, the APF+FLC paths were always shorter, verifying the efficiency of the proposed motion planning algorithm.

3.3. Practical Implications

The simulation tests verified the effectiveness, robustness, efficiency and computational ease of the proposed motion planning algorithm and the associated task scheduling strategy. The mobile robot tests first verified the effectiveness and practicality of the proposed navigation solution, and then further verified the effectiveness, robustness and efficiency of the proposed motion planning algorithm. These test results imply that the motion planning algorithm, the task scheduling strategy and the navigation solution can be potentially adopted to facilitate full autonomy of real grain carts in actual harvest operations. When a combine harvests crops like wheat, soy and corn, an autonomous grain cart could perform the supporting logistical tasks by

itself, including standby, go to combine, unload grain, go to semi-trailer, transfer grain and return to standby point. Especially when the combine's tank fills up, the autonomous grain cart would approach in time and drive alongside the combine to unload the grain without interrupting the harvesting. With autonomous grain carts, harvest operations would require less or even no human labor, overcoming the issues of labor shortages and increasing labor cost. Implementing the proposed motion planning algorithm, autonomous grain carts could navigate in crop fields with high efficiency and service combines with high precision, saving time and fuel while improving productivity. The proposed motion planning algorithm would also enhance the operational safety of autonomous grain carts by providing smooth and safe motion plans, preventing grain carts from traveling too fast or steering too sharply.

3.4. Further Research

With the long-term goal of implementing the proposed motion planning algorithm for developing real autonomous grain carts, further research should focus on testing the algorithm with real grain carts in actual harvest operations. Although the simulation tests and mobile robot tests have produced satisfactory results, there are differences between the tests and actual harvest operations, which could be more complex and challenging. For example, more sensing noise, uneven and slippery terrain, animals with fast and sudden movements, mechanical malfunctions, software failures, etc. That being said, the major challenge of real harvest tests would lie in the process of setting up and actually conducting the experiments. Specifically, it could take great effort to install and configure the sensors, computing devices and communication

networks. If the grain cart was a relatively old model, a retrofit might be necessary to convert the traditional drive train to drive-by-wire. In actual tests, a human operator would be required to drive the combine in the field, harvesting real crop plants during harvest seasons, an expensive and time-consuming scenario. That being said, real harvest tests would be significant in verifying the actual effectiveness, robustness and efficiency of the proposed motion planning algorithm as well as the effectiveness and practicality of the associated navigation solution. Satisfactory results of real harvest tests would facilitate the actual implementation of the outcome of this research for the development of autonomous grain carts.