HAL
open science

Development towards autonomous navigation and automatic handling of palletized freight units by an autonomous vehicle with forks

Khayyam Masood

To cite this version:

Khayyam Masood. Development towards autonomous navigation and automatic handling of palletized freight units by an autonomous vehicle with forks. Automatic. Ecole Centale de Nantes; Universita degli Studi di Genova, 2021. English. NNT : . tel-03530217v1

HAL Id: tel-03530217

https://hal.science/tel-03530217v1

Submitted on 17 Jan 2022 (v1), last revised 9 Jul 2024 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DEPARTMENT OF MECHANICAL, ENERGY, MANAGEMENT AND TRANSPORT ENGINEERING (DIME)

LABORATOIRE DES SCIENCES DU NUMERIQUE DE NANTES (LS2N)

PhD in Mechanical, Energy and Management Engineering (UNIGE)

Doctorat Automatique, Signal, Productique, Robotique (ECN)

Curriculum: Robotics and Mechatronics

Development towards autonomous navigation and automatic handling of palletized freight units by an autonomous vehicle with forks

Khayyam Masood

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy

Supervisor: Prof. Matteo Zoppi
Supervisor: Prof. Vincent Fremont
Co-Supervisor: Prof. Rezia Molfino

Coordinator of the PhD Course: Prof. Giovanni Berselli

OCTOBER 2021 - XXXIV CYCLE

Contents

Contents i

List of Figures vii

List of Tables vii

Acknowledgements ix

Abstract xi

Resume xii

1 Introduction and Synthesis 1
- 1.1 Background and brief history . . . 1
- 1.2 Motivation . . . 2
- 1.3 Problem definition and scope . . . 3
  - 1.3.1 Assumptions for way forward in research . . . 4
- 1.4 Research approach and methodology . . . 4
- 1.5 Contributions and publications . . . 5
- 1.6 Synthesis and Thesis Structure . . . 6
  - 1.6.1 Synthesis of thesis . . . 6
  - 1.6.2 Thesis structure . . . 8
- 1.7 References . . . 9

2 Hydraulic Pressure-Flow Rate Control of a Pallet Handling Robot for an Autonomous Freight Delivery Vehicle 12
- 2.1 Abstract . . . 12
- 2.2 Introduction . . . 12
  - 2.2.1 Hydraulics on FURBOT . . . 14
  - 2.2.2 Synoptic View of Control Architecture . . . 16
- 2.3 Research Objectives . . . 17
- 2.4 Hydraulic Equations and Control Strategy . . . 17
- 2.5 Implementation of the Control Algorithm . . . 18
  - 2.5.1 The Control Strategy . . . 18
  - 2.5.2 Algorithm . . . 19
- 2.6 Preliminary Work . . . 20
  - 2.6.1 The Open Motion Controller and CAN . . . 20
  - 2.6.2 CANOpen . . . 21
  - 2.6.3 Preliminary Work on the Pressure Sensor . . . 21
- 2.7 Modeling and Results . . . 22

2.7.1 Simulation Modeling 22
2.7.2 Simulation Results 23
2.8 Prototype Testing and Results 24
2.8.1 Pressure Sensor Test and Results for the Pressure Loop 24
2.8.2 Final Algorithm Test 26
2.9 Conclusion 28
2.10 References 29

3 Mathematical Modelling for Performance Evaluation Using Velocity Control for Semi-autonomous Vehicle 32
3.1 Abstract 32
3.2 Introduction 32
3.3 Vehicle Dynamics 33
3.3.1 Constraints 33
3.3.2 Mathematical Modeling 33
3.3.3 Velocity Control 35
3.4 Simulation Setup and Results 36
3.4.1 Simulation Setup 36
3.4.2 Results 38
3.5 Conclusion 40
3.6 References 40

4 Simulated Sensor Based Strategies for Obstacle Avoidance Using Velocity Profiling for Autonomous Vehicle FURBOT 43
4.1 Abstract 43
4.2 Introduction 43
4.3 Lateral and Longitudinal Equation of Motion 45
4.4 Obstacles Characteristics and Modelling 46
4.4.1 Traffic Signal 46
4.4.2 Human/Stationary Object 48
4.4.3 Speed Breaker 49
4.4.4 Slow Vehicle Interaction 50
4.5 Simulation Platform and Results 51
4.6 Conclusions 54
4.7 References 55

5 From Drive-By-Wire to Autonomous Vehicle: Urban Freight Vehicle Perspectives 58
5.1 Abstract 58
5.2 Introduction 58
5.3 State of the Vehicle 61
5.3.1 Vehicle Components Breakdown 61
5.3.2 Current Features of FURBOT 64
5.4 Work Progress for Automation 65
5.4.1 Current Achievements 65
5.4.2 Future Goals 66

5.5 Technical Classification for Vehicle . . . 66
5.5.1 L7E-CU . . . 67
5.5.2 N1-1 . . . 67
5.5.3 Possible Licensing Issues . . . 68
5.6 Foreseeable Issues . . . 68
5.7 Upgrading of Vehicle (Requirements and Expectations) . . . 68
5.7.1 Resolution of 3D LiDAR Purchase/Placement/Quantity Issues . . . 69
5.7.2 Requirements . . . 70
5.7.3 Operative Modes of the Vehicle . . . 73
5.7.4 Expectations . . . 73
5.7.5 Recommendations . . . 74
5.8 Conclusions . . . 74
5.9 References . . . 76

6 A Qualitative Analysis to Develop Handover Strategies for Freight Handling Autonomous Vehicle 80
6.1 Abstract . . . 80
6.2 Introduction . . . 80
6.3 Automation Challenges and Complexity . . . 81
6.3.1 Objective Complexity . . . 81
6.3.2 Subjective Complexity . . . 81
6.4 Haptic Take-Over System . . . 82
6.4.1 Haptic Takeover Control Framework . . . 82
6.4.2 Desired Input Sequence Of System . . . 83
6.4.3 Driver's Cognitive Workload Evaluation . . . 83
6.4.4 Driver's Muscle Readiness Evaluation . . . 84
6.4.5 Torque controller for the system . . . 84
6.5 Bringing the Vehicle to a Safe Stop Within the Allotted TOR Time . . . 84
6.5.1 TOR Time Interval . . . 84
6.5.2 Static Object Constraint . . . 85
6.5.3 Lead Vehicle Constraint . . . 85
6.5.4 Environment Observability Constraint . . . 85
6.5.5 Applying these constraint . . . 85
6.6 Conclusion . . . 85
6.7 References . . . 86

7 Parking Pose Generation for Autonomous Freight Collection by Pallet Handling Car-like Robot 88
7.1 Abstract . . . 88
7.2 Introduction . . . 88
7.3 Problem and Approach . . . 90
7.3.1 Freight Detection . . . 90
7.3.2 Orientation of Vehicle . . . 91
7.3.3 Defining Parking Spot Reference to Freight . . . 92
7.4 Software and Control Architecture . . . 92
7.5 Mathematical Modeling and Notation . . . 94

7.5.1 Vehicle Kinematic Model 94
7.5.2 Points Acquisition for Parking Spot 95
7.5.3 Parking Area with Respect to Loading Bays 95
7.6 Results 97
7.6.1 Parking Spot Definitions w.r.t Loading Bays 97
7.6.2 Results for Different Freight Placement 97
7.6.3 Vehicle Parking in ROS Environment 98
7.7 Conclusions and Perspectives 99
7.8 References 100

8 Path Planning for Autonomous Approach to Freight 104
8.1 Introduction 104
8.2 Rapidly-exploring random tree (RRT and RRT*) 104
8.2.1 Metric function for RRT and RRT* 105
8.2.2 Minkowski sum for obstacle avoidance 105
8.3 Results from RRT and RRT* path planning approach 106
8.4 Bézier curves for mobile robots 108
8.5 Application of Bézier curves on RRT and RRT* 108
8.6 Conclusion 110
8.7 References 110

9 Conclusion and Perspectives 112
9.1 Summary of thesis 112
9.2 Perspectives 113
9.3 Future research 114

List of Figures

2.1 Overview of the hydraulic systems installed in FURBOT. 15
2.2 Hydraulic supply of FURBOT. 15
2.3 Hydraulic forks. 16
2.4 Hydraulic suspensions. 16
2.5 Typical servo controller. 16
2.6 Regulation algorithm 19
2.7 The OpenMCP. 20
2.8 (left) IXXAT USB to CAN, (right) open-source Serial to CAN. 21
2.9 Back of FURBOT. 22
2.10 Mathematical model schematics. 22
2.11 Simulation results. 23
2.12 Zoomed view of pressure rise. 23
2.13 Pump running at 1400 rpm. 24
2.14 Pump running at 1000 rpm. 24
2.15 Pump running at constant pressure (130 bar). 25
2.16 Pressure loop at 100 bar. 25
2.17 Pressure loop at 120 bar. 26
2.18 Speed step test. 26
2.19 Pressure loop at 70 bar with an inefficient pump 27
2.20 Bi-state regulator results. 27
2.21 Second experiment of the bi-state regulator. 28

3.1 Selected elevation route 37
3.2 Extracted elevation data 37
3.3 Velocity profile over time 38
3.4 Velocity error profile 39
3.5 Traction force profile 39
3.6 Braking force and velocity error profile 40

4.1 Demonstration area for Freight Urban Robotic Vehicle (FURBOT) in Trikala/-Greece 44
4.2 Traffic signal output. 46
4.3 Traffic light sensor and derivative feedback. 47
4.4 Traffic signal output. 48
4.5 Velocity profiling for stationary object. 48
4.6 Vehicle distance to obstacle 49
4.7 Stationary obstacle collision avoidance. 49
4.8 Speed breaker hurdle avoidance. 50
4.9 Slow vehicle hurdle avoidance. 51
4.10 Flowchart of obstacle avoidance algorithm 51

4.11 Reference velocity of the vehicle 52
4.12 Reference velocity of the vehicle 53
4.13 Braking force wrt velocity difference 53
4.14 (a) Ground velocity of FURBOT (b) Distance travelled 54

5.1 FURBOT while loading freight 61
5.2 FURBOT CAD model. 62
5.3 Chassis of FURBOT. 62
5.4 CAD model of forklifts. 63
5.5 FURBOT cover 63
5.6 Electronic batteries. 64
5.7 Vehicle velocity adjustment. 66
5.8 PANDARXT LiDAR placed on top middle with 0.5 m elevation from vehicle top-base. 69
5.9 (a) Two LiDARs placed front/back (side view); and (b) two LiDARs placed front/back (isometric view). 70
5.10 Street view of demonstration area with few identifiable hurdles (source street view, Google Maps [46]). 70
5.11 Street view of clear setting of demonstration area in Trikala/Greece (source street view, Google Maps [46]). 71
5.12 Autonomous freight handling flow 73

6.1 Handover strategy module in ADS model 81
6.2 Haptic Take over Request control framework 83
6.3 Kinematic model and control diagram 83
6.4 a) Driver performance and need for automation b) Control authority and driver workload c) Modelled control authority and driver muscle readiness . 83
6.5 Take over transition Time line 85
6.6 Lead vehicle constraint Time line 85

7.1 The design of FURBOT freight box 90
7.2 Loading of freight. 91
7.3 Parking w.r.t freight. 91
7.4 Vehicle dimensions w.r.t loading bays. 92
7.5 FURBOT software architecture 93
7.6 Control structure 93
7.7 (a) Vehicle kinematic model. (b) FURBOT vehicle 94
7.8 Point definition for parking spot 95
7.9 Flowchart for parking pose generation. 96
7.10 Parking spot with respect to loading bays. 97
7.11 Parking spot relative to oriented loading bays. 98
7.12 Vehicle parking w.r.t freight at different time intervals. 99

8.1 Result from RRT path planning approach 107
8.2 Result from RRT* path planning approach 107
8.3 Bézier curve fitted on RRT path 109
8.4 Bézier curve fitted on RRT* path 109

List of Tables

3.1 Velocity controllers 36

4.1 Type of obstacles present in simulated drive 52
4.2 Velocity controllers 52

5.1 Vehicle components mass breakdown. 61
5.2 Vehicle dimensions in meters 62
5.3 Velocity and steering controllers 66
5.4 L7E-CU Classification 67
5.5 Technical aspects of L7E-CU. 67
5.6 N1 sub-classes definitions 68
5.7 Comparison of 3D LiDARs 69
5.8 Vehicle infrastructure for demonstration 71

6.1 Automation level defined by SAE 80
6.2 Functional Failures and Automated Driving System (ADS) Handover scenarios 81

7.1 Vehicle dimensions in meters 92

Acknowledgements

I am incredibly grateful to my supervisors Prof. Matteo Zoppi and Prof. Rezia Molfino from Universita degli Studi di Genova and Prof. Vincent Frémont from École Centrale de Nantes, for mentoring me throughout my Ph.D. and especially for their patience at times where I struggled during this era. Their constructive feedback and precious directions really made my journey easier.

I am also grateful to my colleagues in the PMAR lab in Universita degli Studi di Genova and LS2N in École Centrale de Nantes for their optimistic and go-getter attitude. Their attitude towards life and, more precisely, towards research sparked a continuous passion for research inside me which is extremely important. I spent many good times with them over lunches and coffees, which helped me focus on research throughout these times.

Also, I would like to thank Universita degli Studi di Genova for providing me financial support and the necessary equipment for conducting this research. Without financial independence, this journey would indeed turn a lot tougher.

I also show extreme gratitude towards my friends in Genova, Nantes, and back home for their support. Their moral, physical and verbal support was vital for me to go through the hard parts of my Ph.D. No doubt, the most challenging journey also becomes easy if you have made friends along the way.

I am thankful to my parents, who had to suffer hard times without me. No doubt, their prayers helped me in achieving my goal.

I am highly thankful to my wife and two daughters, who sacrificed the most for my research journey. The time we spent apart from each other was excruciating for all of us, and yet they stood by my side throughout this journey and indeed proved to be my strength in hard times. This research and thesis is dedicated to them for their support for me.

”Evolution is a continuous process....it is what made me into what I am today”

Abstract

In the mobile robots community, particular interest in research in autonomous vehicles has been carried out extensively since the last decade. However, the role of mobile robots capable of delivering freight for the last-mile delivery solution is still under study. Freight Urban Robotic vehicle; FURBOT is one such vehicle that is being developed continuously for the last-mile delivery solution. The difference in this vehicle is the capability to load the freight into itself without physical human intervention. Currently, the vehicle is a complete drive-by-wire vehicle, and studies are being conducted to introduce autonomous capabilities in the vehicle. This thesis is an attempt forward in that direction. Three different aspects of the vehicle, which include; autonomous navigation, autonomous freight handling capabilities, and technical management associated with this upgrading, are studied in this thesis. A mathematical model for the vehicle was built for autonomous navigation, which did not exist beforehand for the vehicle. After developing the model, obstacle avoidance techniques were created, and the controller was verified for the proposed methods. The highlight of the thesis involves enhancing the capabilities of an already built freight handling structure. This involves upgrading the pre-existing pallet handling robot, parking pose generation for autonomous freight handling and the design of the approach to the specific parking pose. To complete the vehicle's journey from a drive-by-wire to an autonomous vehicle, in-depth analysis is performed, incorporating studies for the vehicle's amalgamation in the urban environment. Additionally, handover strategies are also studied in case of the vehicle fails during its autonomous drive. The outcome of this research produced multiple enhancements for the vehicle, including independent parking pose generation, correct categorization of the vehicle, software and hardware requirements, conservation of energy in pallet handling robot, and designing digital twin for the vehicle. Extensive simulation-based validations are carried out for the proposed research, which produced desirable results for the vehicle and thus has created a baseline for the way forward for converting the vehicle into a complete autonomous one.

Keywords: autonomous vehicles; freight handling; autonomous navigation; logistics; policies management; control; obstacle avoidance; handover strategies

Résumé

Dans la communauté des robots mobiles, la recherche sur les véhicules autonomes fait l'objet d'un intérêt particulier depuis la dernière décennie. Cependant, le rôle des robots mobiles capables de livrer des marchandises comme solution de livraison sur le dernier kilomètre est toujours à l'étude. Le véhicule robotique urbain de transport de marchandises (FURBOT) est l'un de ces véhicules qui est développé en permanence pour résoudre le problème de livraison du dernier kilomètre. La spécificité de ce véhicule est la capacité de charger des marchandises dans le véhicule sans intervention humaine physique. Actuellement, ce véhicule est un véhicule piloté électroniquement et des études sont menées pour introduire des capacités autonomes dans le véhicule. Cette thèse est une tentative dans cette direction. Trois aspects différents du véhicule, à savoir la navigation autonome, les capacités de manutention autonome des marchandises et la gestion technique associée à cette mise à niveau, sont étudiés dans cette thèse. Pour la navigation autonome, un modèle mathématique du véhicule a été construit, ce qui n'existait pas auparavant pour le véhicule. Après le développement du modèle, des techniques d'évitement d'obstacles ont été développées et le contrôleur a été vérifié pour les techniques proposées. Le point fort de la thèse concerne l'amélioration des capacités de la structure de manutention de marchandises déjà construite. Cela implique l'amélioration du robot de manutention de palettes préexistant, la génération d'une position de stationnement pour la manutention autonome de marchandises et la conception de la trajectoire d'approche de la position de stationnement spécifique. Pour achever le passage d'un véhicule à conduite par câble à un véhicule autonome, une analyse approfondie est effectuée qui comprend des études pour l'intégration du véhicule dans l'environnement urbain. En outre, des stratégies de transfert sont également étudiées en cas de défaillance du véhicule pendant sa conduite autonome. Le résultat de cette recherche a permis d'apporter de nombreuses améliorations au véhicule, notamment la génération d'une position de stationnement autonome, la catégorisation correcte du véhicule, les exigences logicielles et matérielles du véhicule, la conservation de l'énergie dans le robot de manutention de palettes et la conception d'un jumeau numérique pour le véhicule. Des validations basées sur des simulations extensives ont été effectuées pour la recherche proposée ont produit des résultats souhaitables pour le véhicule et ont ainsi créé une base pour la voie à suivre afin de convertir le véhicule en un véhicule complètement autonome.

Mots-Clés: véhicules autonomes; manutention de marchandises; navigation autonome; logistique; gestion des politiques; contrôle; évitement d'obstacles; stratégies de transfert

Chapter 1

Introduction and Synthesis

1.1 Background and brief history

Invent of delivering mail and couriers services date back to Persian Empire back in 6th Century [1]. Afterward, many civilized nations further developed postal services where the distance of the countries exceeded one day's travel time. Later, in the eightieth Century, railroads put wheels on personalized parcel delivery with the help of locomotives. Afterward, in the early nineteenth Century, parcel post services began, which enabled postal consumers with package delivery services and regular mail. With the advent of petroleum and diesel-based auto motives, and the improvement of the highway system after World War II, the use of vehicles in delivering freight packages increased considerably. In the current era, with the invention of online stores and retailers in early 2010, the burden on freight delivery is at its peak. Recently, according to studies carried out in 2019 (Pre-Corona Virus outburst), on average, 3,248 parcels were shipped every second in the world's thirteen major markets, equating to an average of 27 packages shipped per person per annum [2]. Additionally, the impact of fuel-based vehicles on the environment is causing users and researchers to search for alternative parcel delivery methods.

With such a high number of personalized freight shipments, the need to create more robust, automated, and environmentally friendly solutions is the need of the day. This is additionally important, especially for last-mile freight delivery to the users. Because optimizing solutions in the last mile freight delivery can yield significant economic incentives as last-mile delivery is considered the most expensive leg in the logistics/freight delivery supply chain [3]. For personalized last-mile delivery of packages, researchers are working tirelessly to develop autonomous delivery solutions. Currently, Amazon Lockers is placed all across Europe, and the globe provides some answers for last-mile freight delivery. US-based Amazon and China-based Alibaba have also used drones to minimize costs for this leg of parcel delivery to complement this service. However, this billion-dollar freight delivery industry still needs many additional solutions for its expansion and reducing its costs so that an average user can utilize the benefits of efficient last-mile delivery solutions in a more automated, user-friendly, and environmentally friendly manner.

Currently, several freight delivery options are available within an urban setting for last-mile delivery, which includes advance delivery locations (reception boxes (parcel boxes), delivery boxes, controlled axes points, collection points, and locker banks) and advanced delivery vehicles (electric delivery vehicles, electric cargo bicycles, and autonomous driving vehicles) [4]. Still, any single solution is not enough for intelligent last-mile delivery because of its advantages and disadvantages. Bringing autonomous vehicles on the road is not a new con-

cept for delivering freight to the public. Recently, JD.com used robot delivery services in universities of China for personalized freight delivery [5]. Soon, it is envisioned that we will see automated vehicles delivering our freight/parcels autonomously at our doorstep/point of interest.

Freight Urban RoBOTic vehicle (FURBOT) is an urbanized solution for autonomous last-mile freight delivery in an urban setting. Its unique design of capturing the advantages of autonomous vehicles (AVs) with delivery boxes adds personal delivery. Furthermore, its design is also capable of delivering large parcels suitable for storefronts. Its multiple loading bays helps in integrating different solutions of last-mile delivery simultaneously [6]. Currently, it is a complete drive-by-wire vehicle that is being upgraded to an autonomous vehicle. This research revolves around various aspects of FURBOT and its transition/upgrade from a drive-by-wire to an autonomous vehicle capable of delivering freight autonomously within an urban setting.

1.2 Motivation

After the invention of Lidar technology, the development of autonomous vehicles is on the rise, and autonomous vehicles are now sweeping the market [7]. However, autonomous vehicles are still focused on passenger vehicles, and there is not much evidence of using autonomous vehicles for freight carriage. Currently, constant progress is being made to integrate autonomous vehicles or personalized robots for delivering freight within urban settings. However, literature shows that there is still a wide gap in integrating autonomous vehicles for delivering freight.

FURBOT is a unique robot/vehicle that can enable the community to use autonomous vehicles to deliver freight, especially for last-mile delivery. Its unique design of handling two specially designed palletized freight units using forklifts is a design of its own. The design of FURBOT helps in not only delivering pallets as an autonomous vehicle but also due to the presence of multiple forklifts. It has the capability of loading and unloading its cargo autonomously without human intervention. The unique features of FURBOT, its capacity to drive on urban roads, load/unload cargo without human intervention, and handling large pallet size cargo for mass delivery, create commercial interest for a wide range of customers.

FURBOT is currently designed as a complete drive-by-wire vehicle, and now it is time to put the vehicle to the test in an urban environment delivering freight autonomously. The vehicle was part of the European Green Vehicles Initiative (EGVI) funded under the umbrella of the FP7-Transport European project ending in December 2015 [6]. It was completed as a complete drive-by-wire vehicle and conceptually designed to be converted to an autonomous vehicle. Recently, FURBOT has taken part in SHOW (SHared automation Operating models for Worldwide adoption) project where 69 partners from all across the EU will take part in the adaptation of autonomous vehicles in 20 cities across Europe where autonomous vehicles will demonstrate their autonomy for 24 months (https://show-project.eu/). For this purpose, the vehicle has to be converted from a conventional drive-by-wire vehicle to an autonomous vehicle. This change in design and requirements pose new challenges for the vehicle, and thus many unique aspects of the vehicle have to be looked into for its conversion.

Since FURBOT is expected to perform autonomously in an urban setting, this upgrading has raised issues that need to be resolved/addressed for the vehicle to achieve a higher level of autonomy. There are two main aspects which are of primary concern. One is the technical aspect of the vehicle that includes the simulation development, including the modeling of the vehicle. It's an approach to the freight and handling of obstacles during its maneuvers in urban settings. The second is the managerial aspect, which includes resolution of its legislative issues, the vehicle requirements identification, and handover strategies to the driver/human in case of vehicle components failure.

Thus the motivation behind this research is the answer to the following question: How to convert a drive-by-wire freight delivery capable vehicle into an autonomous vehicle while resolving it is a multidimensional managerial, legislative, simulation, and technology-based issues.

1.3 Problem definition and scope

As discussed in the motivation of our work, this thesis attempts to resolve several problems/issues related to the automation of our vehicle. These issues range from simulation-based issues involving the development of the digital twin of the vehicle and managerial/technical issues, including upgrading requirement assessments/partial upgrading of the vehicle plus development of handover strategies to the operator in case of failure of an autonomous system. The topics investigated in this thesis are discussed in the form of bullets below:

- For simulation related issues:
  - Development of a mathematical model for the vehicle for creating the base for the digital twin of the vehicle.
  - Creating control-based strategies for obstacle avoidance for the digital twin to navigate the urban area autonomously.
  - Development of techniques for approaching the freight autonomously for loading the freight into the respective loading bays.
  - Creating correct parking pose for autonomous freight collection when correct pose of freight is available through sensors.
- For managerial/technical issues:
  - Upgrading the pre-installed hydraulic system for operating a pallet handling robot installed inside the vehicle for loading freight into respective loading bays.
  - Categorization of the vehicle into existing European vehicle class for issuance of its license/number plates.
  - Technical evaluation and examination of the requirements and sensors for the vehicle for upgrading it from a drive-by-wire vehicle into an autonomous vehicle capable of collecting and delivering freight autonomously.

- Development of handover strategies in case of failure of vehicle automation, which can lead to handing over the control of the vehicle back to the operator/-driver.

In the context of freight handling robotic vehicles, the issues mentioned above/problems are looked into to develop the vehicle into an autonomous one. As this vehicle is expected to autonomously navigate in a couple of years in an urban environment, the automation of the vehicle is still going. However, in this thesis, the aforementioned issues for amalgamating the vehicle into the SHOW project are discussed, and solutions are proposed.

1.3.1 Assumptions for way forward in research

The flow of this thesis relies on the following assumptions for carrying out the proposed research.

- Apart from the proposed upgrading of the vehicle that is already carried out in this research (upgrading the hydraulic systems for pallet handling robot), all the developed schemes are implementable on the vehicle for amalgamation of the vehicle for automation in an urban environment.
- The installed software of the vehicle can be customized with the help of proposed solutions in this research.
- It is possible to customize the complete vehicle with new hardware as it is an in-built vehicle; moreover, it is possible to install new sensors on the vehicle to develop new strategies for autonomous navigation and freight collection.
- The vehicle is designed for an operator on board; however, the installed hardware can be operated virtually (through autopilot) or through a virtual operator driving the vehicle placed away from the vehicle in a control tower (if necessary).
- The vehicle can handle freight in both loading bays/compartments of the vehicle; thus, all necessary strategies need to be developed to utilize these features in an optimized way.
- The vehicle will operate like a slow-moving vehicle (max 20 km/hr speed) in an urban environment. Thus it will always operate in its designated lane. Thus, all obstacle avoidance schemes designed for urban navigation should be based on velocity control.

1.4 Research approach and methodology

The research approach for this thesis involves systematic targeting of upgrade-able components, software, and theories that can enable the shift of the FURBOT van from a drive-by-wire to an autonomous vehicle. The approach taken targets different research areas where there is a scope of improvement within the vehicle. The methodology taken targets multiple aspects of the vehicle simultaneously, and periodic growth in each segment is achieved.

The vehicle is studied in-depth, especially about its automation and upgrade requirements

for its amalgamation in the SHOW project. The vehicle is set to perform autonomous navigation plus autonomous loading and unloading of the freight in an urban environment. Data collected from previous work conducted on the vehicle show that at the start of the research, the vehicle was completed as a complete drive-by-wire vehicle with customized software. Yet, no previous work was completed on the vehicle for its automation in terms of navigation and future aspects of loading/unloading of freight autonomously. However, multiple sections on semi-automation on the vehicle were carried out. Thus the niche point for the vehicle for further research produced research direction in autonomous freight-based vehicles with the vehicle's challenges that can differentiate it from already available autonomous vehicles in the market. This is mainly achieved by focusing the direction in different aspects of research in freight-based autonomous vehicles.

The methods of analysis for the research carried out varies within each chapter of the thesis. The primary technique used within all the topics addressed in this research is background study on the subject and finding the niche issue which needs to be addressed for the vehicle under discussion, primarily focusing on its utility for using the proposed techniques/methods for the automation of the vehicle. The methods used to follow the approach to resolve futuristic issues and the foreseeable vehicle applications within its operations in an urban environment.

1.5 Contributions and publications

The contributions of the research carried out in this thesis are summarized below.

- Upgrading of the pre-installed hydraulic system of pallet handling robot designed for loading and unloading of the freight.
- Simulation setup for autonomous navigation for the vehicle includes developing a mathematical model of the vehicle, control algorithm for navigation, and velocity controlled based obstacle avoidance schemes.
- Development of autonomous schemes for parking pose generation for freight collection and path generation for the approach to the pose.
- Analysis for the development of handover strategies for the vehicle in case shift of vehicle control is required from autopilot to operator.
- In-depth analysis of the assessment of requirements for bringing a freight-based drive by wire vehicle on the road as an autonomous vehicle.

Most of the chapters presented in this thesis are also published in the following list of publications.

1. Masood K., Zoppi M., Molfino R. (2021) Mathematical Modelling for Performance Evaluation Using Velocity Control for Semi-autonomous Vehicle. In: Herrero Á., Cambra C., Urda D., Sedano J., Quintián H., Corchado E. (eds) 15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020). SOCO 2020. Advances in Intelligent Systems and Computing, vol

1268. Springer, Cham. https://doi.org/10.1007/978-3-030-57802-2_59

2. Masood, K.; Molfino, R.; Zoppi, M. Simulated Sensor Based Strategies for Obstacle Avoidance Using Velocity Profiling for Autonomous Vehicle FURBOT. Electronics 2020, 9, 883. https://doi.org/10.3390/electronics9060883

3. Masood, K.; Dauptain, X.; Zoppi, M.; Molfino, R. Hydraulic Pressure-Flow Rate Control of a Pallet Handling Robot for an Autonomous Freight Delivery Vehicle. Electronics 2020, 9, 1370. https://doi.org/10.3390/electronics9091370

4. Masood, K.; Zoppi, M.; Fremont, V.; Molfino, R.M. From Drive-By-Wire to Autonomous Vehicle: Urban Freight Vehicle Perspectives. Sustainability 2021, 13, 1169. https://doi.org/10.3390/su13031169

5. Masood, K.; Morales, D.P.; Fremont, V.; Zoppi, M.; Molfino, R. Parking Pose Generation for Autonomous Freight Collection by Pallet Handling Car-like Robot. Energies 2021, 14, 4677. https://doi.org/10.3390/en14154677

6. Anbarasan, S.; Masood, K.; Zoppi, M.; Fremont, V.; Molfino, R. A Qualitative Analysis to Develop Handover Strategies for Freight Handling Autonomous Vehicle. 2021 IEEE International Intelligent Transportation Systems Conference (ITSC). 19-22 Sep. 2021. pp. 2582-2588, https://doi.org/10.1109/ITSC48978.2021.9565043

1.6 Synthesis and Thesis Structure

1.6.1 Synthesis of thesis

The synthesis of this thesis involves six chapters comprising of six published papers on how the work on the freight vehicle has progressed; furthermore, it also affects the seventh chapter of unpublished work, which is being done in collaboration with a Master student in Robotics department in the University of Genova. Apart from these chapters, the introduction and conclusion chapters are also presented. In total, the thesis comprises nine individual chapters.

Since the thesis comprises separately published papers, some of the text in each chapter is overlapping, especially in the introductory section of each chapter. All the papers' introductions have a common background of the vehicle and its amalgamation into the SHOW project. As all published articles are published independently, a few technical pieces of information overlap in these chapters. For example, chapter three, which focuses on mathematical modeling of the vehicle, demonstrates in-depth analysis on the building of the model; however, still, some of that information is repeated as subsections in chapters four, five, and seven because of the necessity to show part of that work within the independent paper. Also, the CAD model of the vehicle presented in chapter four is partially repeated in chapter six due to the necessity of the given work. Additionally, there is a slight discussion in each progressive paper on the previous work accomplished. Apart from that, the ideas and solutions presented in each chapter are unique and represent their unique results.

The key ideas presented in each chapter are

- Hydraulics of pallet handling robots.
- Mathematical modeling of vehicle.
- Velocity-based obstacle avoidance schemes.
- Conversion from drive-by-wire to an autonomous vehicle.
- Formulation of handover strategies for an autonomous vehicle.
- Autonomous parking pose generation for freight collection.
- Autonomous approach to freight.
- Conclusion.

This thesis and upgrading of the vehicle resulted in several collaborations. Part of those collaborations is also part of this thesis. Chapter two on hydraulics is converted into a paper from work done by a Master's student, Xavier Dauptain, during his internship at the University of Genova. Chapter six and chapter eight are done in collaboration with a Robotics Masters student at the University of Genova Srinivasan Anbarasan, which is being supervised. Partial collaboration is also done in chapter seven with Postdoc researcher David Pérez Morales working in Ecole Centrale de Nantes, especially on visualization of the approach to the proposed freight and control algorithm.

The thesis focuses on three main characteristics of vehicle; Hardware upgrading with experimental results, software developments with simulation-based results, and addressing technical, managerial issues for the vehicle. Chapter two focuses on hardware upgrading part of the vehicle (hydraulic pressure flow rate control for pallet handling robot) with experimental results. Chapters three, four, seven, and eight follow a progressive approach in the same direction of autonomous navigation and autonomous freight collection and its course by the vehicle. All four of these chapters focus on CAD modeling and development of the digital twin of the vehicle, i.e., they are based on simulation results of the schemes developed. Chapter five focuses on upgrading the vehicle comprehensively, and chapter six focuses on the analysis of handover strategies. These two chapters mainly focus on the technical management of the vehicle.

Overall, the thesis looks at the development of the freight urban robotic vehicle comprehensively. Having prime focus on the algorithm development for autonomous navigation and freight collection and addressing technical, managerial issues that must be resolved for the vehicle to upgrade from drive-by-wire successfully to an autonomous vehicle. However, partial work is also done in upgrading hardware for the vehicle (chapter two). The theme of the thesis is thus; upgrading the drive by wire freight vehicle FURBOT to an autonomous freight handling vehicle capable of approaching and handling the freight autonomously while addressing the upgrading and autonomous challenges associated with the vehicle.

1.6.2 Thesis structure

Chapter one is self-explanatory and involves the introduction of the thesis in detail. Whereas chapter two focuses on the upgrading of the pre-installed hydraulic-based pallet handling system installed on the vehicle. The forklifts present on the vehicle are operated using the hydraulic system on the vehicle. The previous system operated using a fixed displacement pump and a bypass valve which was upgraded using a newly installed pressure sensor that changed its rotational speed with the help of gear pumps. This helped the vehicle conserve more energy in its battery, enabling the use of the same power in other tasks performed by the vehicle.

In chapter three, a detailed mathematical model of the vehicle is presented and simulated. The mathematical model proposed also maintains the vehicle velocity through a PID controller. The vehicle model is simulated for a simulated hilly terrain in Genova, Italy, where the vehicle's velocity might increase or decrease due to the slope of the road it is running on. Thus the need for brakes or accelerators depending on the situation is used to maintain the overall velocity of the vehicle. Thus autonomous traction and braking for the vehicle are introduced in this chapter.

Chapter four focuses on the obstacle avoidance techniques for the vehicle using velocity control as the preferred mode of obstacle avoidance for the vehicle is by adjusting its velocity. Four different types of obstacles are presented and simulated for the vehicle to check whether the vehicle can alter its velocity if it is presented with an obstacle. These obstacles include stationary objects/pedestrians, slow-moving vehicles, traffic signals, and speed bumps. The process of simulating these obstacles and vehicle velocity control is highlighted in this chapter.

Chapter five is more on the technical management of the vehicle and highlights the points which need to be addressed for the vehicle to convert itself from a typical drive-by-wire vehicle into an autonomous vehicle. The chapter focuses on the legislative side of the vehicle necessary for it to be amalgamated into the European urban environment and technical side, which involves the requirements of the sensors, highlighting their necessities and placement for optimum use. Furthermore, it highlights the use cases of the vehicle for it to perform in an urban environment and its complete operational cycle. Moreover, the chapter focuses on the foreseeable issues, expectations from the vehicle, and its requirement for its autonomous navigation and freight handling into an urban environment.

Chapter six also focuses on the technical management of the vehicle, but in a different aspect. In this chapter, a qualitative analysis of handover strategies in case the control of the vehicle is handed back to the operator is performed. Discussions on different techniques on handover strategies are evaluated, and the analysis performed highlights those strategies so that an optimum design can be adapted for our vehicle.

Chapter seven focuses on the autonomous calculation of parking pose for the vehicle for the autonomous freight collection. As the vehicle is equipped with two separate loading bays, estimates for separate parking poses for each loading bay for freight calculation are performed. This chapter concludes that if the freight location and orientation are calculated previously through the onboard equipped sensors, a correct parking pose can be estimated,

which can result in the autonomous collection of freight. The formulation proposed is verified against different orientations and cargo locations to check whether the parking pose created is correct. The mathematical simulation results highlight that irrespective of the orientation/location of the freight, if the required information from the freight is calculated correctly, then the parking pose generated through the mathematical formulation will be correct.

Chapter eight presents the work completed on the vehicle in collaboration with a Master's students from the Robotics department in UNIGE. The work comprises of the approach to the parking pose to the freight, which can be the freight loaded inside the vehicle. In the approach to the freight, the concept of using RRT and RRT* techniques for path planning for an approach to the freight is discussed and highlighted. Furthermore, once attaining the path through the RRT and RRT* techniques, bezier curves are fitted on the course. The approach presented is also viable by a vehicle-type robot that cannot rotate on its axis. This topic discussed conclude chapter eight and technical work performed on the vehicle in this thesis.

Finally, chapter nine concludes the thesis and the work done for attaining the level of research conducted for this thesis. This chapter highlights the analysis shown, elaborates on the findings, and finishes the thesis with the conclusion.

1.7 References

Bibliography

[1] "History," Encyclopædia Britannica. [Online]. Available: https://www.britannica.com/topic/postal-system/History. [Accessed: 16-Sep-2021].

[2] E. Mazareanu, "Parcel shipping volume worldwide 2013-2026," Statista, 13-Oct-2020. [Online]. Available: https://www.statista.com/statistics/1139910/parcel-shipping-volume-worldwide/. [Accessed: 16-Sep-2021].

[3] S. Dolan, "The challenges of last mile delivery logistics and the tech solutions cutting costs in the final mile," Business Insider, 21-Jan-2021. [Online]. Available: https://www.businessinsider.com/last-mile-delivery-shipping-explained?IR=T. [Accessed: 16-Sep-2021].

[4] Behnke M. Recent Trends in Last Mile Delivery: Impacts of Fast Fulfillment, Parcel Lockers, Electric or Autonomous Vehicles, and More. In Logistics Management; Lecture Notes in Logistics; Bierwirth, C., Kirschstein, T., Sackmann, D., Eds.; Springer: Cham, Switzerland, 2019.

[5] Li Dongmei, "JD.Com launches robot delivery services In Chinese universities," China Money Network, 07-Feb-2018. [Online]. Available:

https://www.chinamoneynetwork.com/2017/06/19/jd-com-launches-robot-delivery-services-in-chinese-universities. [Accessed: 16-Sep-2021].

[6] Freight Urban RoBOTic Vehicle. 8 August 2016. Available online: https://cordis.europa.eu/project/id/285055 (accessed on 25 November 2020).

[7] "The rise of autonomous vehicles," The rise of autonomous vehicles | Digital Watch, 10-Dec-2020. [Online]. Available: https://dig.watch/trends/rise-autonomous-vehicles. [Accessed: 16-Sep-2021].

electronics

MDPI

Article

Hydraulic Pressure-Flow Rate Control of a Pallet Handling Robot for an Autonomous Freight Delivery Vehicle

Khayyam Masood [1,*], Xavier Dauptain [2], Matteo Zoppi [1] and Rezia Molfino [1]

1 PMAR Robotics, DIME, University of Genova, 16145 Genova, Italy; zoppi@dimec.unige.it (M.Z.); molfino@dimec.unige.it (R.M.)

2 Ecole Centrale de Nantes, 1 Rue de la Noë, 44300 Nantes, France; xav.dauptain@orange.fr

* Correspondence: khayyam.masood@edu.unige.it; Tel.: +39-320-530-4054 or +39-010-353-2837

Received: 24 July 2020; Accepted: 20 August 2020; Published: 24 August 2020

Abstract: The current paper presents an upgrade of a pre-installed hydraulic system for the operation of a pallet handling robot for a freight delivery vehicle known as FURBOT (freight urban robotic vehicle) . The automated forklift installed on FURBOT for loading/unloading of cargo is powered with the help of hydraulics. The previous hydraulic system worked via a classical approach with a fixed displacement pump and a bypass valve, making it work on full power when in use. An alternative design was proposed, simulated and installed on FURBOT; it uses a fixed displacement pump and changes the rotation speed in real time using a pressure sensor. Novelty was attained with the use of gear pumps for said scenario. A control algorithm is implemented in the processing unit for controlling the speed of the motor driving the pump. The main advantage of this approach is better use of energy for the vehicle's battery. The aim of this research is to control both the speed and maximum force exerted by the actuators with the help of a single sensor and an inexpensive pump.

Keywords: service robotics; hydraulics; control; forklift; FURBOT

1. Introduction

The idea behind the FURBOT (freight urban robotic vehicle) project was to propose a light-duty, efficient, sustainable and fully electrical freight vehicle for urban environments. The truck features a pallet handling robot and active suspension for loading/unloading operations. The main purpose of this vehicle is to deliver cargo, for which loading/unloading of cargo is of prime importance. The pallet handling robot installed in the truck not only requires automation, but also it is desired that it should be energy efficient. Furthermore, it should be optimized for performance and be safe for use.

The upgradation of FURBOT is being carried out presently, as the vehicle is required to perform at the H2020 Research and Innovation Programme, SHOW (SHared automation Operating models for Worldwide adoption). The vehicle is set to be fully functional and optimal in time to qualify for its pre-demo in October 2021 for SHOW. Apart from the work which is being carried out for its automation [1,2], the hardware upgradation of the vehicle is also very critical for its performance, especially the safe operation of its robotic forklift. Thus the need for this work was generated.

The vehicle has a control mode for loading/unloading of pallets defined as "Forkmode". The vehicle only handles the pallets when the vehicle is in this particular mode. The automated forklift of FURBOT operates with the help of hydraulics rather than electric drives. The selection of hydraulics over electric drives is due to two main reasons—one being that force density is required under limited space/mass, and secondly because other subsystems have similar requirements that hydraulics can serve. Additionally, hydraulics are economically more viable. Costs play a vital role in the design of FURBOT, as it is desired that the vehicle remains cost effective to keep it commercially attractive.

Another novelty in the work is the use of gear pumps instead of axial piston pumps. Gear pumps are always driven at constant nominal speed (at the speed of highest efficiency for the component; efficiency degrades at speed lower than nominal non-linearly, although not in a way compromising the use of the component), and it is a common practice to use axial piston pumps to generate variable flow rates. This issue is industry-driven, since diesel engines and electric motors usually operate at constant speed. However, an axial piston pump costs typically six to eight times what a gear pump does; it is also subject to more wear with standard filtered oil circuits, as one would expect to use in a small van like FURBOT (not sophisticated overall hydraulic components in the circuits); finally, gear pumps are on the market for use with water–glycol, but for the same fluid axial piston pumps are extremely expensive due to rare use. Hence the novelty and idea investigated was to mount a servomotor on the gear pump: the motor was designed to operate at variable speed and in dynamic conditions; and not be regulated the way it is done with an inverter on asynchronous motor, but controlled in time to promptly serve instantaneous fluid rate.

It was observed that very few publications are available on the optimization problem of the hydraulics of a forklift. Some researchers have focused their research on either vision of the forklift operator [3] or controller design for automated forklift operations [4–6]. The focus of optimizing the hydraulics of the system for a forklift is still unexplored. Most of the optimizations are either theoretical [7] or for dynamic modeling [8].

Displacement-controlled actuators and hydraulics are playing vital roles in the development of new and clean technologies for mobile robots [9]. Major challenges being faced in the development of hydraulic systems for such robots are efficiency improvements, noise reduction and advancements in pump technology [9]. The innovative techniques used in this article mainly comprise on/off valves and fixed displacement pumps. The advantages of using fixed displacement pumps over variable displacement pumps are that they are typically smaller, lighter and cheaper than their counterparts [10]. The efficacy of on/off valves is well established [11], and for this reason, such valves are used in our work for development of an energy efficient hydraulic system with a lower cost at the same performance level.

New approaches to solving the intricacies of valve operation and fixed displacement pumps are being studied continuously, and mathematically modeled in order to better simulate their results [12]. The biggest drawback of these displacement pumps is the energy loss which is associated with them. However, these losses can be significantly reduced in the presence of load/pressure sensors [13]. The application of hydraulic systems in itself is a very demanding task. Many researchers have published their results just on the application [14], as a simple condition such as cold weather has the potential to decrease the overall efficiency of the hydraulics system [15]. The discussed articles bring good validity in the proposed work.

Another recent work used the same approach for flow control [16] but they used it specifically for electric drives. Within [16], the process for simulating the controller was similar and MATLAB/Simulink was used for simulation purposes. Additionally, the technique for hardware validation was also similar to our work. Furthermore, in another study, direct driven hydraulics were used to reduce the power consumption of the system [17], which is especially useful for heavy vehicles. This is very helpful in our vehicle as reducing power consumption is one of the goals of our research. Upgrading the energy management for forklifts has also been studied and verified in experimental settings [18]. Our research also uses the same experimental validation technique for our forklift operations to verify our simulation results.

Pressure flow rate control is an essential and emerging necessity. New devices are constantly being patented to safeguard the intellectual property [19,20]. Further work is also being done to save energy by using mutual pressure and flow control [21] and optimization pressure control valve for automatic transmission [22]. In reference to our work, we also tried to optimize our hydraulic system by using an inexpensive displacement pump and pressure sensor. Using a variable displacement

pump was never viable for our research as cost effectiveness plays a vital role in the overall design of our robotic forklift.

This study was unique because an already installed hydraulic system was upgraded for better performance through control modifications and new hardware installation. The work completed not only upgraded the hydraulic system of FURBOT, but also produced a safer environment for working. The whole problem was simulated in LMS and SimuLink before implementation. Additionally, the work produced has practical applications in forklift design and is considered vital in the performance of FURBOT. To summarize, the contributions of this research are given below.

1. It is not common practice to use hydraulics in robotic systems. We have taken a new approach by using hydraulics instead of electric drives for our robotic forklift.
2. Furthermore, the axes need force density and very modest precision: An electric motor and gears also give additional precision not needed for our tasks. Additionally, the electric motor and gear pump can have a rigid transmission, and this too is a feature not required for FURBOT application: for this reason, hydraulics are not used much in robots because robots are designed for tasks needing accuracy and made stiff to simplify the control (controlling flexible axes is more complex in trajectory and under variable loads).
3. Validation of our methodology was achieved by conducting separate tests for speed control, the pressure loop, the speed test and the bi-state regulator. Moreover, repetitive experiments were performed to increase the faith in the robustness of our system.
4. The classical approach to regulate speed/force is through pressure reducing valves while the pump turns at full speed; this means it is not possible to set both the maximum force that will be exerted and the speed during the motion. With the help of bi-state regulator, we are able to set both the maximum force exerted and speed during motion.
5. A lot of energy is wasted using pressure reducing valves. The technique proposed is not only safer, but also reduces the energy loss in the system.

1.1. Hydraulics on FURBOT

FURBOT is equipped with a complete hydraulic circuit for handling of freight through the forklift, for suspensions and for the hydraulic braking. The complete schematics of the installed hydraulics on FURBOT are shown in Figure 1. The architecture of the hydraulic supply of FURBOT is also given in Figure 2. For each forklift axis, a double acting cylinder is used and controlled by a 4/3 mono-stable distributor, as shown in Figure 3. The suspension uses the same structure, but also includes nitrogen tanks, which can be linked to the cylinder chamber when driving, as shown in Figure 4.

The tank supplies the circuit with fluid and allows heat dissipation. The hydraulic pump is a fixed displacement pump, making the flow rate responsible for regulating the speed of the motor driving it. The bypass valve enables the operator to temporary disable the feed circuit without stopping the pump. With the current design, the pump is running at full speed when the operator enters the "FORK mode" needed to enable the hydraulics to operate the forklifts. Without actuation and negligible oil compressibility, a sharp increase in pressure is observed. Without any kind of regulation, this phenomenon has the capacity to waste energy because the excess of fluid goes back to tank through the pressure relief valve. The pressure valve is required to bypass the circuit to the tank when the pressure exceeds 140 bars. Another pressure regulator generates a lower pressure stream of oil for the suspensions (30 bars).

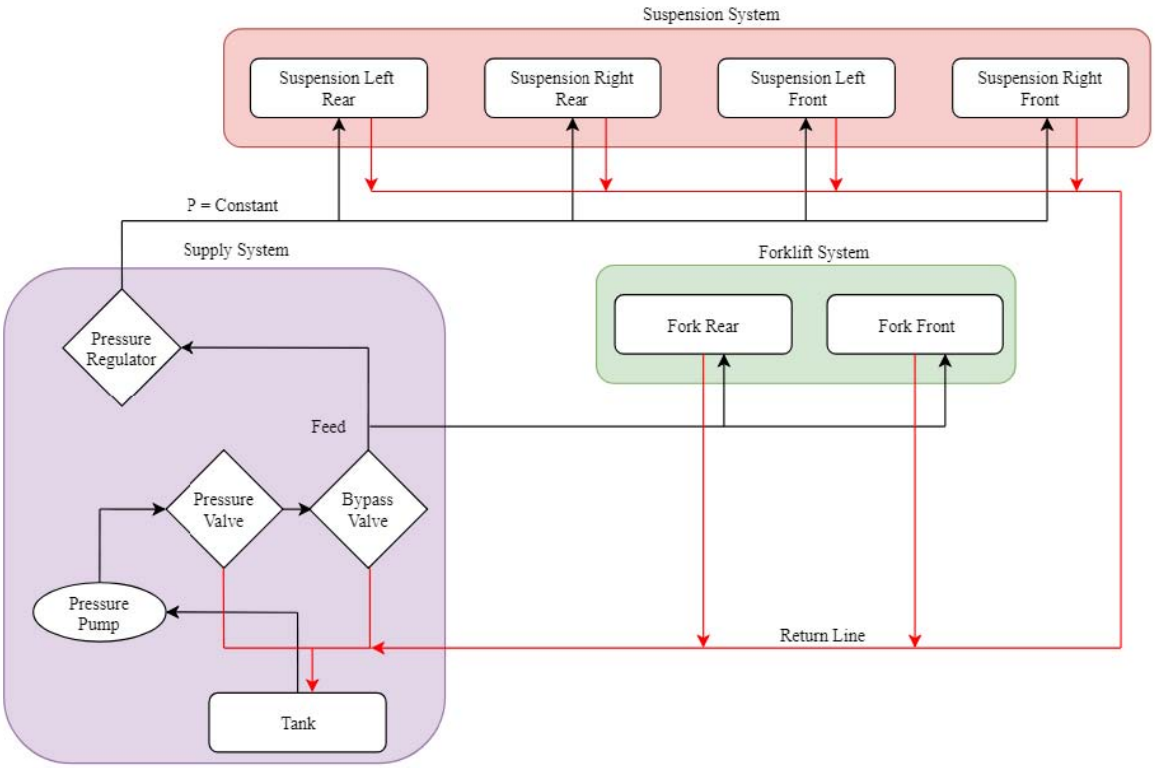


Figure 1. Overview of the hydraulic systems installed in FURBOT.

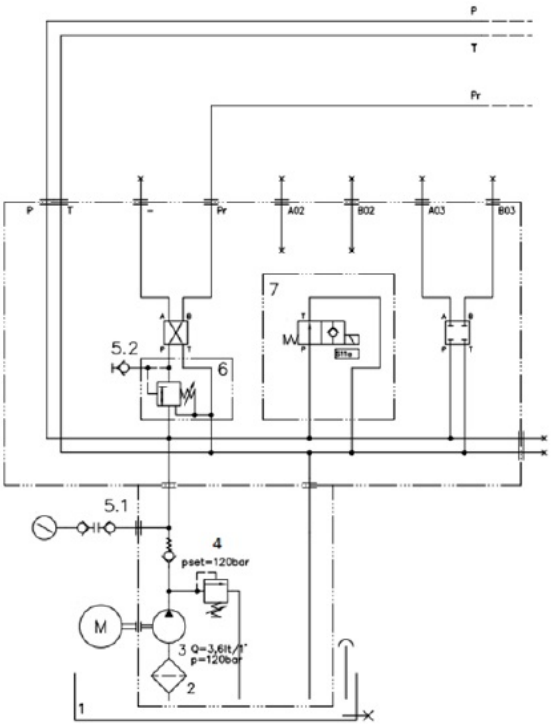


Figure 2. Hydraulic supply of FURBOT.

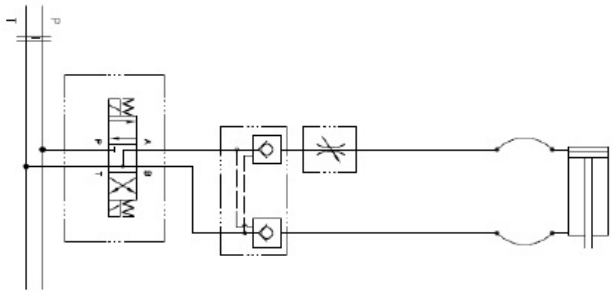


Figure 3. Hydraulic forks.

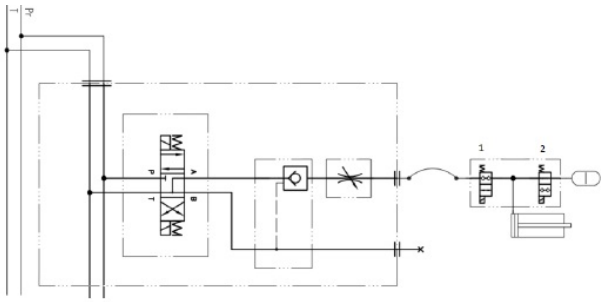


Figure 4. Hydraulic suspensions.

1.2. Synoptic View of Control Architecture

A synchronous brush-less motor drives the pump. The motor is controlled by a servo-controller and makes the set controller + motor a servomotor, as shown in Figure 5. The controller generates the three-phase currents for the motor, and regulates the current (thus the torque) and the speed. This is a current servo-drive where the minor current loop is embedded into the speed loop.

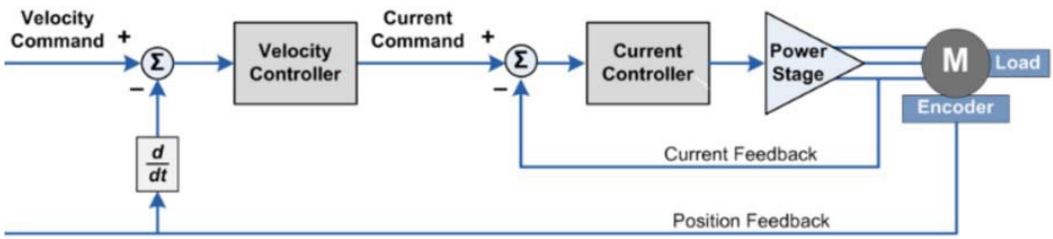


Figure 5. Typical servo controller.

The servo-controller is connected via a digital communication link to a custom controller board and implemented on a Linux computer featuring a PowerPC processor, RAM, an FPGA for GPIOS control, an Ethernet layer (MAC and PHY) and onboard CAN transceivers. The controller board, called the open motion controller, elaborates the control of all the low-level devices on the FURBOT via CAN buses and a CANOpen layer: these include hydraulic motor via a drive manufactured by SEVCON, the traction drives, the steering drive, the signaling of the vehicle and other devices via a CAN-to inputs outputs board (DIO).

The current design is sub-optimal from an energetic point of view and has safety issues, especially regarding the fork's actuation, yet this is a classical architecture. Reasons for this design being sub-optimal are:

1. The pump is always running at full speed and the pressure regulators are used uneconomically. Given the oil compressibility, the rise in pressure is quasi-instantaneous; in other words the pump can start when actuation is needed without wasting energy for the majority of the operation time when in fork mode and waiting for actuation.

2. The forklift's cylinders do not always need 120 bars of pressure to work properly; so the maximum flow rate is not always required.
3. There is no safety regarding the speed of the unloaded cylinders as it is not controlled while working at maximum capacity. However, when the pressure is regulated, lamination valves allow regulating the speed of actuation by limiting the flow rate. Additionally, if the cylinders are stuck, the hydraulic pressure rises to 120 bars of pressure. Thus, it is better to set a maximum pressure set point in order to set the maximum force exercised by the cylinders. The pressure set point will give the maximum force and the lamination valve the maximum speed in steady state.

2. Research Objectives

The objective was to regulate the pressure of the oil before the fork actuators. The pressure regulating valves are able to regulate the pressure of the oil; however, the problems are:

- ◇ They are designed for a simple service and dissipate energy by bypassing a certain amount of oil into the main tank.
- ◇ One can set the pressure manually but cannot control or regulate it over time.

Hence the idea of regulating the pressure by controlling the flow rate of the pump was formulated (the amount of oil fed into the system in a time period, while compensating for the leaks and changing the pressure according to compressibility formulas). Usually, a variable displacement pump is used for such an application. When it comes to "load sensing" in hydraulics, the flow rate of a variable displacement pump is modified (without changing the rotation speed). In load sensing with variable displacement axial piston pumps, the flow rate is adapted in the time to have no flow through the pressure relief valve, which is there just for safety. Additionally, the cost target of the vehicle does not let us use the normal variable displacement architecture because it is too expensive. Additionally their proneness to failure with fluids different from oil is also an issue, since we desire to shift to water additives with ethylene or propylene glycol in the future.

The idea was to use a constant displacement pump and to change the rotation speed in real time. The flow rate is actually proportional to the rotational velocity of the pump. This required measuring the pressure with a new dedicated sensor, implementing an algorithm in the processing unit (the OpenMCP) and controlling the speed of the motor driving the pump.

3. Hydraulic Equations and Control Strategy

We define the bulk modulus in hydraulics β [23] as

$$\beta = \rho \frac{dp}{d\rho} \tag{1}$$

where p is the pressure in the line and ρ is the volumetric mass density of the fluid. Considering that rate of change of density is directly proportional to the rate of change of mass flow rate of the fluid (defined as q) we get (if Volume (V) = constant)

$$\frac{dp}{dt} = \frac{\beta}{\rho V} q \tag{2}$$

Since rate of change of volume is not constant, especially when the actuators are in motion, our equation becomes

$$\frac{1}{\beta} = \frac{1}{\rho}\left(\frac{1}{V}\frac{dM}{dp} - \frac{M}{V^2}\frac{dV}{dp}\right) \tag{3}$$

Since

$$q = \frac{dM}{dt} \quad and \quad \rho = \frac{M}{V} \tag{4}$$

where "M" and "t" define "mass" and "time" respectively, we get

$$dp = \frac{\beta}{\rho V}\Big(qdt - \rho dV\Big) \tag{5}$$

$$\frac{dp}{dt} = \frac{\beta}{\rho V}q - \frac{\beta}{V}\frac{dV}{dt} \tag{6}$$

It is desired that we control the output pressure with the input q despite the unknown variations in V. Thus, the "expansion of the walls" then appears as a perturbation term. Curve fitting on volume around a value is non-realistic as V changes a lot in a non-deterministic way.

Let us assume that $\sigma = p_{cons} - p$. With the Lyapunov candidate function $V = 1/2\sigma^2$ we want: $V = \dot{\sigma}\sigma < 0$ to achieve control whatever the perturbation may be (change of V).

$$(p_{cons} - p)\dot{p} < 0 \tag{7}$$

$$(p_{cons} - p)\frac{\beta}{V(t)}\Big[\frac{q(t)}{\rho} - \dot{V}(t)\Big] > 0 \tag{8}$$

Then we have

$$\begin{cases} q > p\frac{dV}{dt} & if\ p_{cons} > p \\ q < p\frac{dV}{dt} & if\ p_{cons} < p \end{cases} \tag{9}$$

It is possible to take the following first order sliding-mode control for $q : q = \rho K sign(p_{cons} - p) \approx \rho K \frac{\sigma}{|\sigma|+\epsilon}$, where $\sigma = p_{cons} - p$ and K is the upper bound for $|\frac{dV}{dt}|$ and achieve regulation. This is interesting conceptually but of little use for the real life system because such a command law is basically a high-gain, high-frequency control. Since flow rate is controlled by changing the speed of the volumetric pump, we have the following nested loops: current in the $motor \rightarrow speed\ of\ the\ shaft \rightarrow\ pressure$. Sliding mode is well suited for systems that have a low-pass behavior and can be driven at high frequency. Here, the bandwidth of the inner loops, the logic limitations in the electric controller and the delays with the communication link prevent us from having high frequency switching which will result in instabilities: the electric drive cannot cope with the speed changing too quickly. The current loops operating in FURBOT run at 16 kHz, the speed loop runs at 200 Hz and the minimum delay between two consecutive set-points for the speed is 20 ms (50 Hz). Thus the pressure loop can run at a frequency between 200/10 = 20 Hz and 200/5 = 40 Hz < 50. It is required that the outer loop control be as continuous and smooth as possible.

Note that the pressure control mode will be enabled when the actuator is not moving or there is a high load on it, which will result in slow motion under pressure limitation. For a non-time-varying volume, a proportional controller is enough. If an integral term is introduced, it can easily match a small enough $\dot{V}$ with a short amount of time, especially if a high integrator gain is used in conjunction with a good anti-windup (to avoid instabilities with steps in pressure).

4. Implementation of the Control Algorithm

4.1. The Control Strategy

The objective was to be able to set a maximum working force and a nominal speed for each hydraulic axis on the FURBOT. Since it is not possible to control the speed while limiting the force with a single volumetric pump at the same instance, a control mode strategy was devised which switches the control mode according to a defined logic. It works according to the following algorithm (Figure 6) where the controller is aware of

- ◇ The speed of the pump: through intelligent servo-controller.
- ◇ The hydraulic pressure: through the installed pressure sensor.

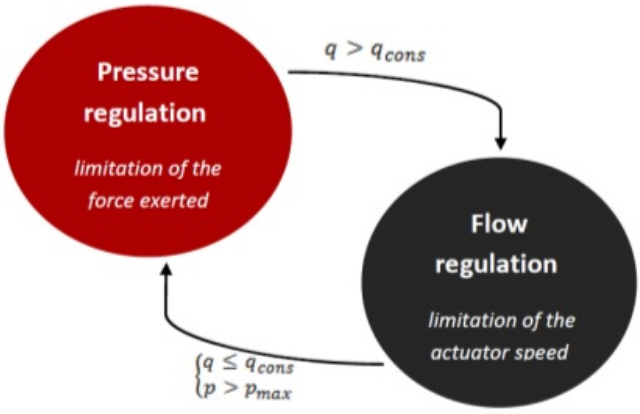


Figure 6. Regulation algorithm.

This is basically a bi-state machine ensuring smooth actuator operation while limiting the power of the cylinders and thus limiting the speed of the fork operation. Pmax is directly proportional to the maximum operating force and Qcons is in keeping with the nominal speed of the actuator (the factor is the cylinder area S).

The operation is the following: most of the time, the actuator is driven with constant flow. If for a reason something is preventing it from moving or the cylinder reaches an end-position, the pressure will raise. For safe operation and to avoid using the pressure limiting valve, the pressure is regulated to Pmax once it goes over Pmax. Once the load is removed or the actuation is done in reverse (pulling instead of pushing), the flow rises until it eventually goes over Qcons, which is the nominal flow rate. Then the controller switches to flow regulation again.

4.2. Algorithm

The whole build system of the OpenMCP, which involves the kernel and the filesystem, the cross-compiler, configuration bash scripts and makefiles was cloned using LINEAR mercurial repository. Within the folders and files, there is the furbot application folder with all the source files (.hpp and .cpp), object dictionary description files (.ods) and the latest binary executable. The architecture of the code is the following:

- The libcanopen++ API is used to create new can buses handlers and CANOpen "master controllers" which can be linked to a bus. A "master controller" is a CANOpen master node with a specific master object dictionary and which can run several "application controllers" with the use of threads.
- The software's "application controllers" are linked to a specific hardware-device on the bus. They work with the use of threads and callbacks from the libcanopen++ stack and have an internal state machine to operate the vehicle safely.
- A FURBOT class wraps all the buses and controllers. The latter also includes a network handler to broadcast information via UDP (User Datagram Protocol) to all hosts on the local network. All the objects are instantiated and the threads launch with the call to the Furbot() constructor inside the main function.

Changes to the existing architecture included:

- Changing the object dictionary for one of the master nodes on the bus. New user entries were added for the pressure process-values coming from the pressure sensor and we did the right PDO (Process Data Object) mapping.
- Writing a controller for the pressure sensor (two new files), to get the data and to send it to the local network inside UDP packets. A network subsystem for the pressure sensor was added using pre-existing functions.

- ◇ Creating a class for the algorithm (two new files) to be able to get the behavior explained above. The pressure control mode is a PID control while the flow regulation just consists of setting the speed of the pump.
- ◇ Changing some functions into the furbot class, especially the function starting the pump and running it at full speed by a new function according to the current requirements.

5. Preliminary Work

5.1. The Open Motion Controller and CAN

The vehicle uses several CAN buses to control the different modules and drives. The algorithm is implemented on the motion controller; the latter runs under a Linux operating system and the "furbot" application runs in the background.

5.1.1. The OpenMCP

It is the main motion controller of the vehicle. The hardware of the OpenMCP is shown in Figure 7. Ethernet links it to all the devices by CAN buses and to the human–machine interface on a local network. Physically, it is composed of two boards: a digital processing board stacked upon a digital interface and a signal handling board that includes network transceivers (CAN and Ethernet), protection components (opto-isolators, fuses and transient voltage suppressors) and an IO interface. On the computing board, there is a PowerPC processor, flash memory, RAM and an Altera FPGA for network layer implementations and some IO logic. Two automotive AMPSEAL connectors are used to physically wire the board inside the vehicle.

The "furbot" application is cross compiled and loaded as part of a Unix file system. A Linux kernel and the file system are loaded in the flash by a separate tool. When the device boots, it decompresses the kernel and the file system (SquashFS) on the RAM. The board allows for two kernels and two file systems being available on the flash memory. It automatically loads the latest ones but allows using the old revision if a problem is detected. Particularly if the boot fails with one kernel, it automatically tries the older one after a few attempts. An executable, added to the path, enables one to load a newer revision of the kernel or the file system via the network and the protocols SSH/SCP.

Figure 7. The OpenMCP.

5.1.2. CAN

The CAN (controller area network) is an automotive fieldbus designed to be very reliable in harsh environments and is an asynchronous multi-master serial bus. The physical layer uses a differential signaling scheme on copper with a 120 Ohms termination resistor to avoid signal reflections at the end of the lines. In the most common implementation, all the nodes share the same link (bus). A complex

arbitration process based on a message priority system and several kinds of frames allow only one node to talk at time. Acknowledgment of the messages enables it to detect errors and faults. There are four frame types:

- ◇ The data frame transports the data.
- ◇ The remote frame asks for the transmission of a data frame.
- ◇ The error frame is broadcast by one node which detects an error.
- ◇ The overload frame is broadcast by a node which detects an overload.

All the transceivers have their own clock, but these are re-synchronized on a frame reception using the signal logic transitions: for that purpose, the bit stuffing ensures that there are always enough transitions to synchronize the nodes.

5.2. CANOpen

Getting the CAN messages from the hardware to a computer requires an adapter (made in silicon with the OpenMCP) and a driver on the computer. An IXXAT "USB to CAN" adapter was used for the early experiments with Linux machine and the drivers from IXXAT. Hardware is shown in Figure 8.

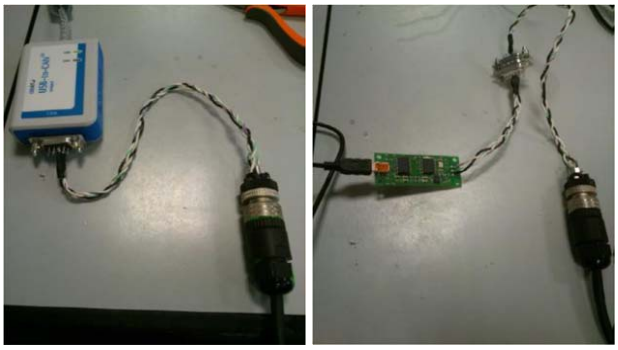


Figure 8. (**left**) IXXAT USB to CAN, (**right**) open-source Serial to CAN.

On UNIX, several tools are useful for talking to the bus, such as the can-utils tools. Other than that, the SocketCAN stack enable the user to use a network interface to talk to the CAN bus. The tools involved are candump and cansend to filter/see the received messages or send a message on the bus. The application talks with the SocketCan stack via a library. CANFestival CANOpen library is used to interface an application code with the bus working with CANOpen. CANOpen master node was written, with its own object dictionary generated from a python utility given with the CANFestival library, and was able to control the motor in speed regulation and position regulation.

5.3. Preliminary Work on the Pressure Sensor

The pressure sensor operates from 1 to 160 bars in absolute pressure and operates according to the CANOpen ds404 profile. The pressure sensor was mounted on the hydraulic circuit of the FURBOT and was first directly wired to the personal computer using IXXAT. It was then configured using the cansend tool for Linux and the LSS protocol. CANOpenShell utility provided with the CANFestival libraries was used to take feedback from the sensor, set the state with NMT and get the data via SDO/PDO messages. All the electronic hardware was mounted on the back of the vehicle as shown in Figure 9. A python script was written to get the pressure plot in real time. This ensured that the pressure sensor was working properly.

Figure 9. Back of FURBOT.

6. Modeling and Results

6.1. Simulation Modeling

The model falls into three parts: the corrector, the electric drive and the hydraulic circuit. The whole SimuLink model is shown in Figure 10. One of the objectives was to check whether the PI controller proposed above could regulate the pressure with a certain class of system composed of an electric drive and a constant displacement hydraulic pump, without the need for a full mathematical model of the Furbot subsystems. The main output is the pressure of the pump, and the inputs are the set point for the pressure and a pulse generator to command the cylinder in the hydraulic circuit. From the pressure set point, the controller generates a speed set point for the pump that is reached by the drive. The final controller is a PID with anti-windup integrator clamping.

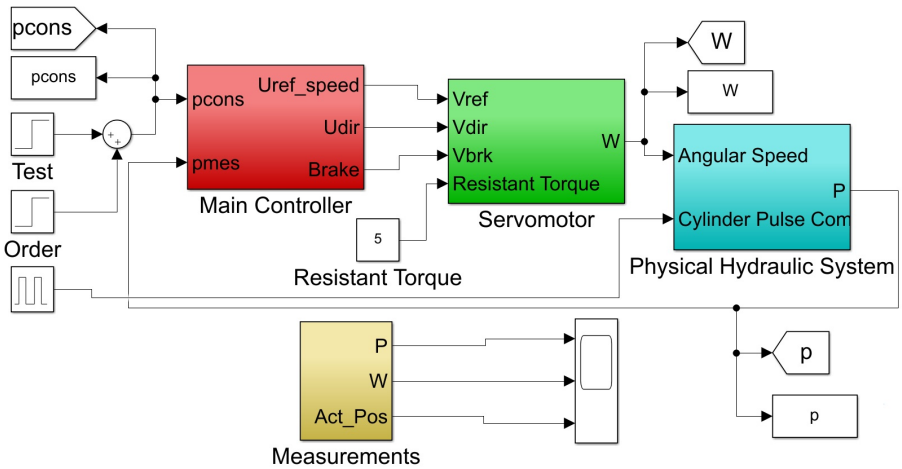


Figure 10. Mathematical model schematics.

6.2. Simulation Results

The control scheme proved successful as long as the speed of the cylinder was limited with a flow limiter and there was enough oil behind the pump. Indeed, if there was no flow limiter, the piston moved too fast, and the circuit could not cope with the corresponding dV/dt. Theoretically, if there is not enough oil, instabilities (oscillations) may happen. A tank with a fixed volume of oil behind the pump can have a damping action. In the real system, we also want to get rid of the flow limiter, but most of the time it will be controlled in constant flow (constant actuator speed) mode, and if it switches to pressure limiting mode it means that a counter action is preventing the piston from moving too fast.

It was observed that there was drop in pressure when the motion of the actuator changed. Nevertheless, the integral term helped the controller to recover regulation around the set point, i.e., 30 bars. The following figure shows the details of the pressure rise at the beginning. A delay was observed between commanded and current speed for the pump. Modeling and the simulation proved that the controller could possibly work in theory. Figure 11 shows the simulation results, whereas a zoomed view of pressure rise is shown in Figure 12.

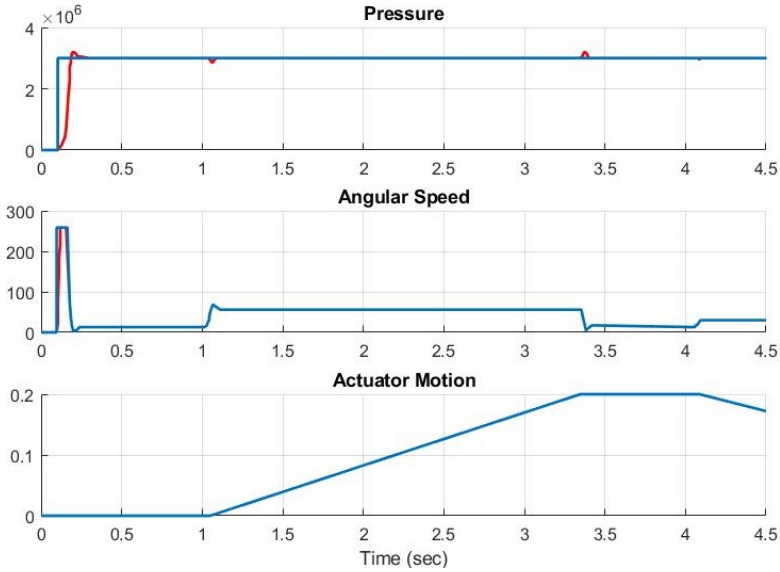


Figure 11. Simulation results.

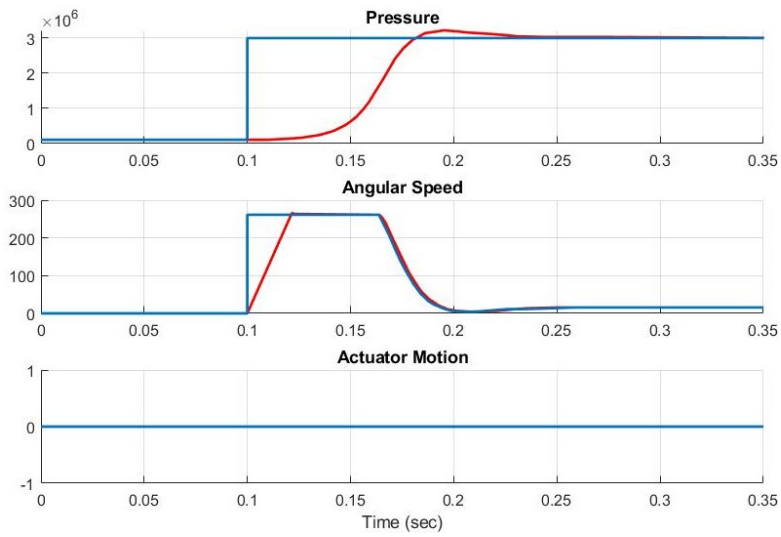


Figure 12. Zoomed view of pressure rise.

7. Prototype Testing and Results

7.1. Pressure Sensor Test and Results for the Pressure Loop

The pump was driven at constant speed to verify the results. The plots below are from the python script with which the data were superposed to see "moving curves" in real time. The time basis was 20 ms between samples, and there was a 255 sample-buffer. Figures 13 and 14 are results of the pump driven at constant speed (rpm). The results obtained reveal the hardware's performance at a constant speed, enabling one to judge the robustness of the controller.

- ◇ Blue represents pressure.
- ◇ Red represents set speed from the control loop.
- ◇ Green represents actual speed.
- ◇ Yellow represents torque from the motor, measured from the servomotor.

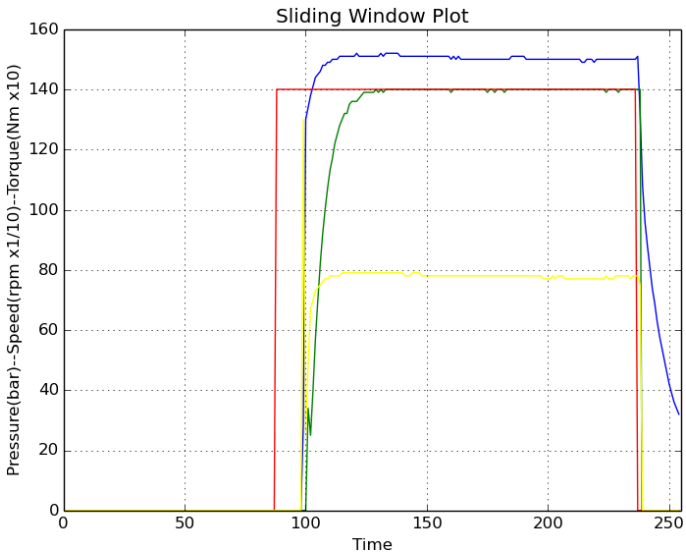


Figure 13. Pump running at 1400 rpm.

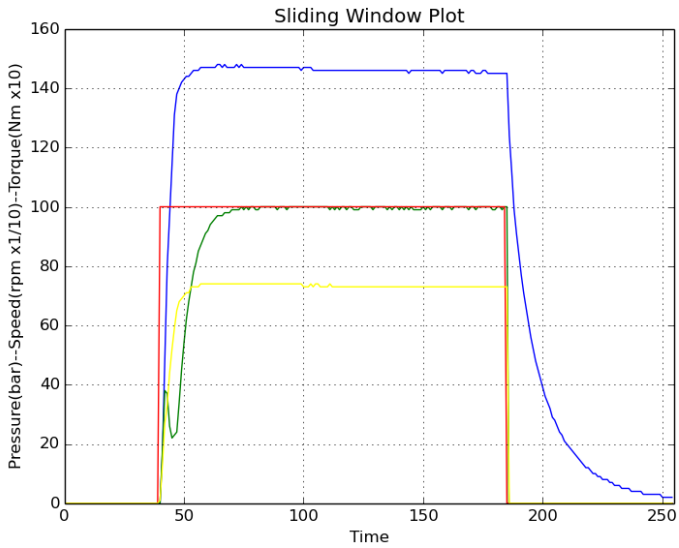


Figure 14. Pump running at 1000 rpm.

Furthermore, the hardware performance was also validated by running the pump at constant pressure (Figure 15). These three experiments (Figures 13–15) were performed to validate the hardware

and highlight any inherent delays in the pump before validating the controller. With the understanding of these results, we are able to validate better the robustness of bi-state controller.

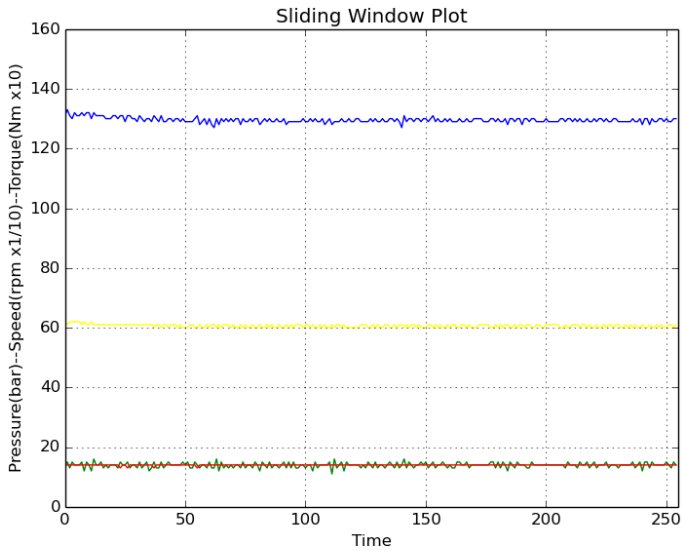


Figure 15. Pump running at constant pressure (130 bar).

The typical results collected from the pressure loop are presented in Figures 16 and 17. To validate the pressure control loop, two separate experiments were performed at different pressure set points (100 bar and 120 bar). The satisfactory results obtained from Figures 16 and 17 enhance the reliability of the PID pressure loop controller, thereby increasing its robustness.

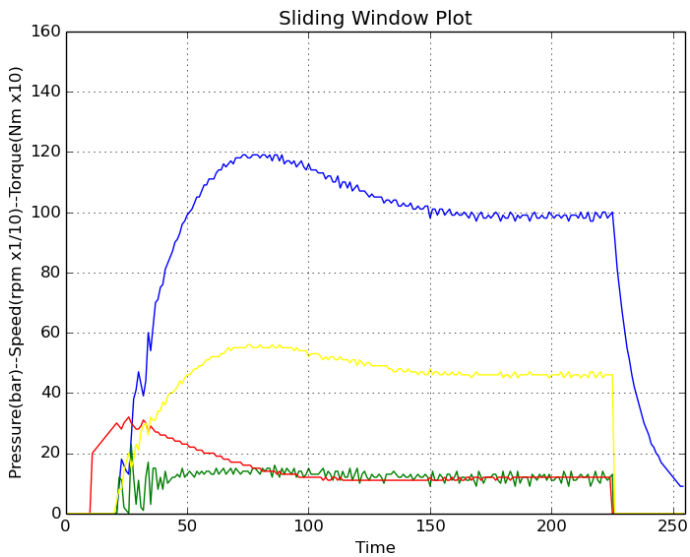


Figure 16. Pressure loop at 100 bar.

While performing the speed test (Figure 18), a jitter was observed at low rpm. It appears that the jitter near the setpoint was due to bad encoder settings for the servomotor: the latter used the UVW phase signals for the synchronous motor instead of the signal from the encoder to get the positions. Thus, there were less coding points (ticks), which is detrimental to speed control for low speeds. Apart from the jitters at low speeds for the reason mentioned above, the controller performance in the speed test (Figure 18) was well within the acceptable range when keeping in mind the inherent lags of the system.

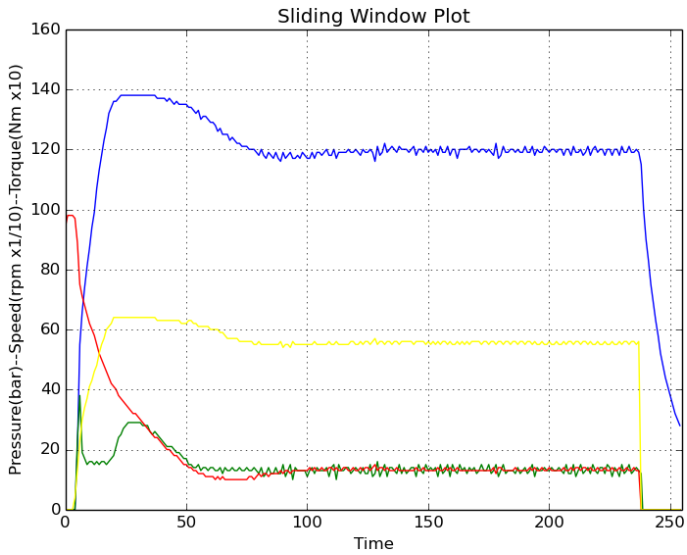


Figure 17. Pressure loop at 120 bar.

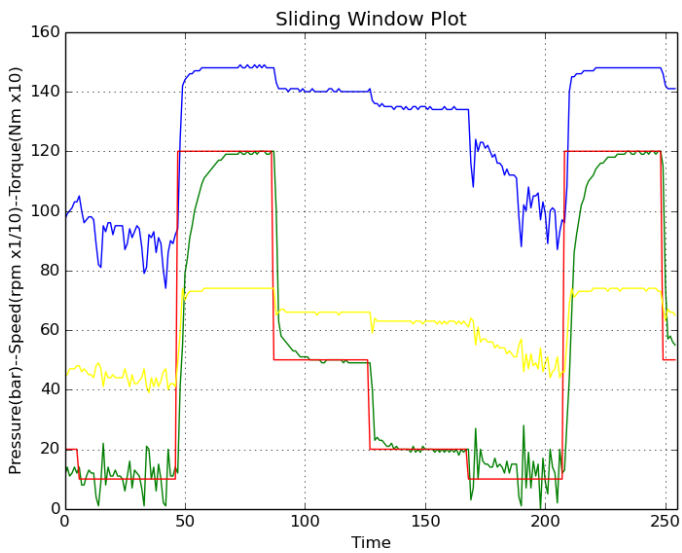


Figure 18. Speed step test.

It was thus observed that at low speed, the pump was uncontrollable. This was because the efficiency of the pump was very low at low speeds. The gear pump had an interval of regulation and for lower speeds the internal leaks made it difficult to control, since flow sometimes leaked at low speed. This particular problem was not resolved even after installing a small tank of 0.35 L to damp the system and avoid oscillations. During the acquisition, the bypass valve was closed and then was opened; still, the pump was not able to cope with increasing pressure without creating huge torque, making the pump inefficient.

7.2. Final Algorithm Test

The results of pressure control were again collected with the inefficient pump at low pressure, as shown in Figure 19. From Figure 19, one can observe that the actual speed does not follow the set speed, hence the pressure ripple. For the results of the bi-state regulator (Figure 20), a horizontal piston of the forklift system was actuated (which was responsible for its horizontal movement), and we set the nominal speed to 300 rpm, and the maximal pressure to 25 bars. The flow limiter was completely open for that cylinder.

Figure 19. Pressure loop at 70 bar with an inefficient pump.

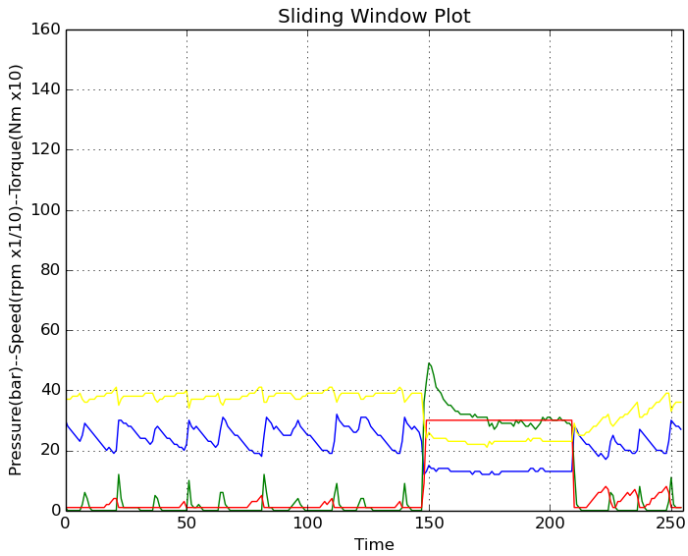


Figure 20. Bi-state regulator results.

To validate the results of the bi-state regulator and to check its performance, another experiment was performed (Figure 21) with the same conditions. The results of both experiments (Figures 20 and 21) show consistency in the performance enhancing robustness of our bi-state regulator. Furthermore, the results validate our bi-state controller and its performance; actual speed (green) followed the set speed (red) in both of the experiments.

In spite of the presence of ripples in the pressure control mode, it was observed that the piston moved slowly till it reached an end-position and became pressure controlled. An effort was made to block the piston and limit the pressure. The piston was blocked with a chain, which moved slowly with the load release. Additionally, the speed was limited without the flow limiter, which was wasting energy; and the hydraulic pressure regulator was not used. The motor with a properly functioning pump should be low in pressure regulation mode. It is nevertheless possible to activate the pump when an actuator needs to move and to stop it on another condition, i.e., low speed of the pump or timer (tri-state machine).

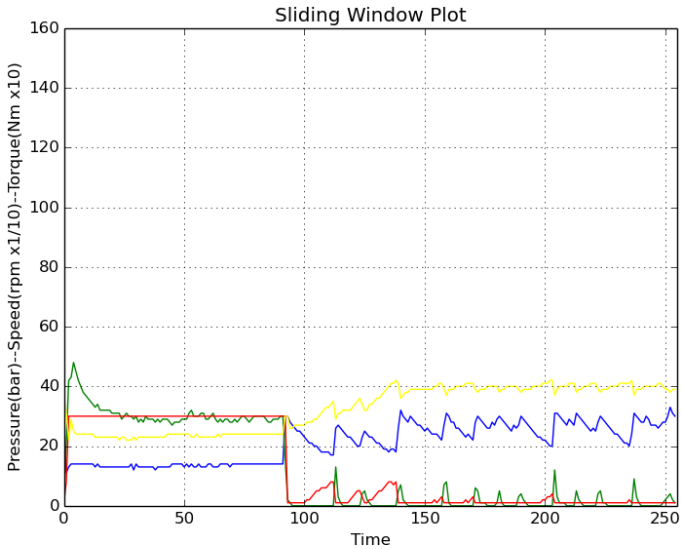


Figure 21. Second experiment of the bi-state regulator.

8. Conclusions

The overall hydraulic system of FURBOT was upgraded and optimized in terms of energy and safety. An innovative controller was implemented for pressure regulation and flow regulation which enhanced the overall performance of the installed forklift on FURBOT. The whole operation was conceived conceptually and simulated in Simulink before actual implementation. Perfect actuation of the forklift was not achieved due to hardware limitations, but the whole concept was electronically and physically implemented. Additional work is required to achieve better results and remove the remaining issues in the development of actuation system of the forklift. Nevertheless, the achieved results are very promising and of practical use. These results are very attractive in the automation process of project FURBOT.

Furthermore, we have shown that we were able to move the cylinder pump at constant speed through constant flow rate while controlling the maximum force exerted at all times. This was achieved with a single constant flow rate pump and a pressure sensor ending power wastage with pressure reducers. Additionally, at end positions, switching to pressure regulations eliminated the need for limit switches. The drawback of this scheme is that it is not easily scalable to multiple and simultaneous actuators control. This is because the flow of the constant flow pump will be divided through all actuators and the force exerted depends on the section of the cylinder pump; thus, a single pressure setpoint will not be enough.

Author Contributions: Conceptualization, X.D.; methodology, X.D.; software, X.D. and K.M.; validation, X.D. and M.Z.; formal analysis, X.D. and K.M.; investigation, X.D. and M.Z.; resources, X.D. and M.Z.; data curation, X.D. and K.M.; writing—original draft preparation, K.M.; writing—review and editing, K.M., X.D. and M.Z.; visualization, K.M., X.D. and M.Z.; supervision, R.M. and M.Z.; project administration, R.M. and M.Z.; funding acquisition, R.M. and M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was supported by European Union's Horizon 2020 research and innovation programme under grant agreement number 875530, project SHOW (SHared automation Operating models for Worldwide adoption).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

FURBOT	Freight Urban Robotic Vehicle
SHOW	SHared automation Operating models for Worldwide adoption

References

1. Masood, K.; Molfino, R.; Zoppi, M. Simulated Sensor Based Strategies for Obstacle Avoidance Using Velocity Profiling for Autonomous Vehicle FURBOT. *Electronics* **2020**, *9*, 883. [CrossRef]
2. Masood, K.; Zoppi, M.; Molfino, R. Mathematical Modelling for Performance Evaluation using Velocity Control for Semi-Autonomous Vehicle. In Proceedings of the 15th International Conference on Soft Computing Models in Industrial and Environmental Applications, Burgos, Spain, 16–18 September 2020.
3. Wang, J.Y.; Zhao, J.S.; Chu, F.L.; Feng, Z.J. Innovative design of the lifting mechanisms for forklift trucks. *Mech. Mach. Theory* **2010**, *45*, 1892–1896. [CrossRef]
4. Minav, T.A.; Laurila, L.I.E.; Pyrhönen, J.J. Relative Position Control in an Electro-Hydraulic Forklift. *Int. Rev. Autom. Control* **2010**, *6*, 54–61
5. Minav, T.; Filatov, D.M.; Laurila, L.; Vtorov, V. Self-tuning-parameter fuzzy PID speed control in an electro-hydraulic forklift. *Int. Rev. Autom. Control* **2012**, *5*, 9–15.
6. Minav, T.A.; Filatov, D.M.; Laurila, L.I.; Pyrhönen, J.J.; Vtorov, V.B. Modelling of an electro-hydraulic forklift in Matlab Simulink. *Int. Rev. Model. Simul.* **2011**, *4*, 640–647.
7. Pan, L.; Du, Q.; He, C. Design Research on Hydraulic System of Working Device of a Forklift. In Proceedings of the 5th International Conference on Advanced Design and Manufacturing Engineering, Shenzhen, China, 9–20 September 2015; pp. 1813–1817.
8. Reutov, A.A.; Kobishchanov, V.; Sakalo, V. Dynamic Modeling of Lift Hoisting Mechanism Block Pulley. *Procedia Eng.* **2016**, *150*, 1303–1310. [CrossRef]
9. Ivantysynova, M. Innovations in pump design—What are future directions? In Proceedings of the 7th JFPS International Symposium on Fluid Power, Toyama, Japan, 15–18 September 2008; pp. 59–64.
10. Batdorff, M.; Lumkes, J. Virtually Variable Displacement Hydraulic Pump Including Compressability and Switching Losses. In Proceedings of the ASME 2006 International Mechanical Engineering Congress and Exposition, Chicago, IL, USA, 5–10 November 2006.
11. Tu, H.C.; Rannow, M.B.; Van de Ven, J.D.; Wang, M.; Li, P.Y.; Chase, T.R. High Speed Rotary Pulse Width Modulated On/Off Valve. In Proceedings of the ASME 2007 International Mechanical Engineering Congress and Exposition, Seattle, WA, USA, 11–15 November 2007.
12. Manring, N.; Williamson, C. The Theoretical Volumetric Displacement of a Check-Valve Type, Digital Displacement Pump. *J. Dyn. Syst. Meas. Control* **2018**, *141*, 031014. [CrossRef]
13. Drumea, P.; Dumitrescu, C.; Hristea, A.; Popescu, A. Energy Use in Hydraulic Drive Systems Equipped with Fixed Displacement Pumps. *Hidraulica* **2016**, 48–57.
14. Van de Ven, J.D.; Olson, M.W.; Li, P.Y. Development of a Hydro-Mechanical Hydraulic Hybrid Drive Train with Independent Wheel Torque Control for an Urban Passenger Vehicle. *Int. Fluid Power Expos.* **2008**, *51*, 503–514.
15. Minav, T.; Heikkinen, J.; Schimmel, T.; Pietola, M. Direct Driven Hydraulic Drive: Effect of Oil on Efficiency in Sub-Zero Conditions. *Energies* **2019**, *12*, 219. [CrossRef]
16. Roskam, R. Development of a Forklift for Research and Education in Mechatronics. In Proceedings of the 2018 2nd International Conference on Mechatronics Systems and Control Engineering, ICMSCE, Amsterdam, The Netherlands, 21–23 February 2018; pp. 17–21. [CrossRef]
17. Minav, T.; Lehmuspelto, T.; Sainio, P.; Pietola, M. Series Hybrid Mining Loader with Zonal Hydraulics. *Int. Fluid Power Conf.* **2016**, *3*, 211–224.
18. You, Z.; Wang, L.; Han, Y.; Zare, F. System Design and Energy Management for a Fuel Cell/Battery Hybrid Forklift. *Energies* **2018**, *11*, 3440. [CrossRef]
19. Nagase, M.; Hirata, K.; Dohi, R.; Nishino, K.; Ikeda, N. Pressure-Type Flow Rate Control Device. U.S. Patent US10,261,522B2, 6 April 2019.
20. Hirose, T.; Yoshida, T.; Matsumoto, A.; Hirata, K.; Ikeda, N.; Nishino, K.; Dohi, R.; Sugita, K. Pressure-Type Flow Rate Control Device. U.S. Patent US10372145B2, 6 August 2019.
21. Serbin, A.; Petlenkov, E.; Vodovozov, V. Energy saving in centrifugal pumps using mutual pressure and flowrate control. In Proceedings of the 2017 IEEE 58th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, Latvia, 12–13 October 2017. [CrossRef]

22. Jian, H.; Wei, W.; Li, H.; Yan, Q. Optimization of a pressure control valve for high power automatic transmission considering stability. *Mech. Syst. Signal Process.* **2018**, *101*, 182–196. [CrossRef]
23. Durfee, W.; Sun, Z.; de Ven, J.V. *Fluid Power System Dynamics*; Center for Compact and Efficient Fluid Power, Department of Mechanical Engineering, University of Minnesota: Minneapolis, MN, USA, 2015.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

Mathematical Modelling for Performance Evaluation Using Velocity Control for Semi-autonomous Vehicle

Khayyam Masood(✉), Matteo Zoppi, and Rezia Molfino

PMAR Robotics, DIME, University of Genova, Genova, Italy
khayyam.masood@edu.unige.it

Abstract. Freight Urban RoBOTic vehicle (FURBOT) is a semi autonomous vehicle for which it is desired that it could deliver freight autonomously from one destination to another. The vehicle is required to operate in Genova, Italy which in general has steep slopes. Additionally, safety of this vehicle and of the environment is of critical importance for urban autonomous driving thus the need for having a simulation model arises. Furthermore, the vehicle is expected to perform last mile freight delivery in European H2020 project SHOW for which highest autonomy is required. For these purposes, a mathematical model is constructed for autonomous velocity control over gradient varying hilly terrain. Autonomous traction and braking of the vehicle is introduced for catering for gradient varying terrain. The model built for this vehicle will serve as basis for embedding new sensors in future, tracking their performance and overall creating a safe environment for the vehicle to operate.

Keywords: Mathematical modeling · Performance evaluation · Mobile robots · Freight vehicle · Velocity control

1 Introduction

Freight Urban RoBOTic vehicle (FURBOT) is light weight, fully electronic vehicle designed for sustainable freight transport in urban areas. It is one of pioneering autonomous vehicles in freight delivery sector. The vehicle is expected to handle first and last mile freight delivery in an urban environment setting for the European H2020 project SHOW (SHared automation Operating models for Worldwide adoption). For the project SHOW, FURBOT is expected to attain maximum autonomy in its drive. Due to the autonomy requirements of SHOW project, it is essential for FURBOT to be modelled and simulated prior at length. For this purpose, it is very essential to build a custom-made simulation platform where automation testing and vehicle performance could be judged prior to experiments. This work is an effort to create such simulation platform in order to enhance the performance of the vehicle when integrated with new sensors and in general observing the performance anomalies if any.

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Switzerland AG 2021
Á. Herrero et al. (Eds.): SOCO 2020, AISC 1268, pp. 617–626, 2021.
https://doi.org/10.1007/978-3-030-57802-2_59

Dynamic modelling of new types of robots is becoming an essential research platform and is a required essential tool for developing better performing robots in current research. From Bi-pedal robots [1] to Hexa-slide robots [2] to parallel manipulators [3], dynamic modeling is becoming the key for better understanding the robots behavior in the natural environment.

Motion dynamics and custom control in robotics is essential for performance evaluation. Mobile robots specially wheeled robots which have vast human interaction and unknown environment requires simulation testing. Reference [4] uses sliding mode control for trajectory tracking for wheeled robot and [5] uses control to move their robot on slippery downhill. Mathematical modeling thus mainly revolves around modeling Newton-Euler equations [6,7] and using MATLAB as a tool for simulation.

This work, thus is essence of aforementioned researches in the field of mobile robotics. Due to need of modelling of vehicle, motion dynamics and controls are modelled for simulating our robot in MATLAB coupled with Simulink environment. As FURBOT is unique freight vehicle, modeling and simulating it is key for performance evaluation for vehicle and environmental safety.

2 Vehicle Dynamics

The current mathematical model of FURBOT is being developed for judging the performance of the vehicle before embedding any new hardware (sensors) or implementing any new automated technique which could result in an accident if not simulated prior to the experiments being conducted.

2.1 Constraints

The mathematical model is able to perform well on an empty straight road. Additionally, it is assumed that the vehicle does not have any inherent errors which can deviate it from the straight path e.g. tyre misalignment/balancing, lateral center of gravity shift etc. The road geometry consists of a straight road with null radius of curvature. The only varying feature of the road is the gradient of the road, which is extracted from Google Maps and simulated for traction power and braking control of the vehicle while driven on a hilly terrain.

2.2 Mathematical Modeling

An effort is made to make the mathematical model as precise as possible. The current coordinate system has its x-axis along the nose of the vehicle, y-axis towards left and z-axis is upwards [7]. It is considered that the vehicle is unable to roll, move along z-axis or pitch (unless due to road gradient change), making this mathematical model as a primarily three degree of freedom model. Both forces acting along x and y axis are modelled with moments acting along z-axis. However due to constraints, the motion of the vehicle is along a straight line.

Traction Force. Most of the definitions of forces acting along the body x-axis are taken from Ref. [8]. The forward force generated due to the torque acting on the driven wheels is given by Eq. 1

$$F_t = \frac{T_p \iota_g \iota_o \eta_t}{r_d} \tag{1}$$

Where F_t is the traction force and T_p is the torque output of the power plant and in our case is the output of the throttle controller. ι_g is the transmission gear ratio, ι_o is the final drive gear ratio, η_t is the final efficiency of the driveline from the wheels to the power plant and r_d is the radius of the wheels respectively.

Drag Force. The drag force calculation is straight forward and is estimated with the shape of FURBOT in mind. At present, drag coefficient (C_d) of 0.5 is selected which is in reference to usual drag coefficient of such shape vehicle. Equation 2 is the equation used for drag force calculation.

$$F_d = 0.5\rho V^2 A_f C_d \tag{2}$$

Where F_d is the drag force acting on the body, ρ is the air density, A_f is the vehicle frontal area and V is the total velocity of the vehicle.

Gradient Force. Although the gradient force calculation is comparatively simpler i.e. dependent only on weight of the vehicle M_v and gradient angle α (Eq. 3, [8]), but the amplitude of this force can be significantly higher compared to other resistive forces. Also since in our case, there is no traffic or hurdle, this is the only force which cause braking to come in action (in case of negative gradient).

$$F_g = M_v g sin\alpha \tag{3}$$

Rolling Resistance Force. Rolling resistance force is due to the friction between tyres of the vehicle and the surface of road. The two main components attributing to this force are the normal force acting on the vehicle tyres and rolling resistance coefficient. Calculation of normal force is pretty straight forward, however there are number of different ways to calculate rolling resistance coefficient specially with varying velocity of the vehicle. The main equation for rolling force is given by Eq. 4 given in Ref. [8].

$$F_r = (M_v g cos\alpha) f_r \tag{4}$$

Where f_r is the rolling force coefficient and remaining is the normal force acting on the vehicle. For calculation of f_r numerous techniques are found in literature, however for this work, calculation of rolling force coefficient is taken from the work of Brian [9] which is also an extension of his work in [10] and is given in Eq. 5.

$$f_r = C_{sr} + \{3.24 C_{dr} (\frac{V}{100})^{2.5}\} \tag{5}$$

Where C_{sr} and C_{dr} represents the static and dynamic components of rolling resistance coefficient. In [11], variation of both, C_{sr} and C_{dr} are plotted against tyre pressure and Brian [9] used these graphs to extract polynomial expressions for C_{sr} and C_{dr} which are given in Eq. 6 and 7. These equations are thus taken from the work of Brian [9] and their validity is discussed in his work.

$$(C_{sr})_{ref} = -0.0000001687P_i^3 + 0.0000255349P_i^2 - 0.0012944847P_i + 0.0305104628 \quad (6)$$

$$(C_{dr})_{ref} = -0.0000002636P_i^3 + 0.0000404822P_i^2 - 0.0020812137P_i + 0.0381150798 \quad (7)$$

Where P_i denotes the tyre pressure. The above equations are considered for calculating the rolling resistance force coefficient for this work.

Forces Along y-Axis and Moment Along z-Axis. The Newton-Euler equations of motion for forces along y-axis and moment along z-axis are used for calculating respective forces and moments and are given by Eq. 8 and 9 [12].

$$\dot{v_y} = \frac{1}{M_v v_x}(-a_1 C_{af} + a_2 C_{ar})r - \frac{1}{M_v v_x}(C_{af} + C_{ar})v_y + \frac{1}{M_v}C_{af}\delta - rv_x \quad (8)$$

$$\dot{r} = \frac{1}{I_z v_x}(-a_1^2 C_{af} - a_2^2 C_{ar})r - \frac{1}{I_z v_x}(a_1 C_{af} - a_2 C_{ar})v_y + \frac{1}{I_z}a_1 C_{af}\delta \quad (9)$$

These equations are expressed in the body coordinate frame for the planar rigid vehicle [7]. C_{af} and C_{ar} are the cornering stiffness of the front and rear wheels respectively. δ is the steering angle and a_1/a_2 are the distances of the rear/front wheels from the CG of the vehicle. Since our steering angle for this work is considered zero, the forces acting along y-axis and moments acting along z-axis yield very negligible values which are not enough for moving the vehicle considerably in y-axis.

Longitudinal Equation of Motion Along x-Axis. The longitudinal equation of motion is relatively simple as the problem is reduced to forces along x-axis and is given by Eq. 10.

$$\dot{v_x} = \frac{\sum F_x}{M_v} = \frac{F_t + F_d + F_g + F_r}{M_v} \quad (10)$$

2.3 Velocity Control

Two separate controllers are designed for motion control for the autonomous vehicle FURBOT. One for traction power control and one for braking power

control. The reason for using such cascade controller is because our requirements for braking and acceleration are different. For traction power control, we require a smooth robust controller whereas for braking power controller, we require sharp responses addition to smooth behavior. Traction power control is a PD controller with an error amplification factor, whereas braking power control is a simple proportional error control with a self-defined operational dead-band of 1 km/h speed, thus it is only initiated if there is a difference of at least 1 km/h speed between reference and actual speed. Reason for not including integral component for the controller designs was because of the overshoot integral values were causing in actuation values. The designing criteria for both the controllers was to keep the velocity error < 2 km/h. Inputs of both controllers is the difference between reference and actual velocity in km/h. Details of these controllers are given in Table 1.

Table 1. Velocity controllers

Controller type	Controller values		
	Error amplification	Proportional gain	Derivative gain
Traction power control	500	20	1
Braking power control	1	80	0

A switch is placed between traction controller and braking controller. If the signal of braking control is greater than 0 i.e. it is active, the traction controller is automatically turned off. The controller design is kept simple in order to build the mathematical model.

3 Simulation Setup and Results

The whole mathematical model was built in MATLAB and SimuLink and simulated for obtaining outputs for validating the model under genuine conditions.

3.1 Simulation Setup

Genova, Italy has one of the most unique terrain topology. Not only does it have beaches and mountain ranges, it is also quite dense in population. Driving an autonomous vehicle on such a terrain for the first phase of testing requires extensive simulation testing for environment and vehicle safety. A potential route is selected in Genova for FURBOT for its performance evaluation. The route selected for the FURBOT mathematical model verification and for elevation data is taken from Google Maps [13] and is shown in Fig. 1

For results evaluation, only the elevation profile is extracted. The current scope of work is to evaluate FURBOT capability over hilly terrain thus the selected path served as an ideal candidate for performance evaluation. The elevation profile extracted from the Fig. 1 is plotted in Fig. 2.

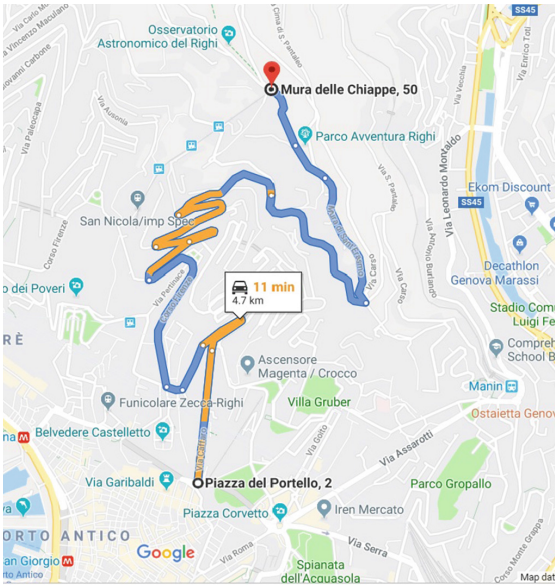


Fig. 1. Selected elevation route

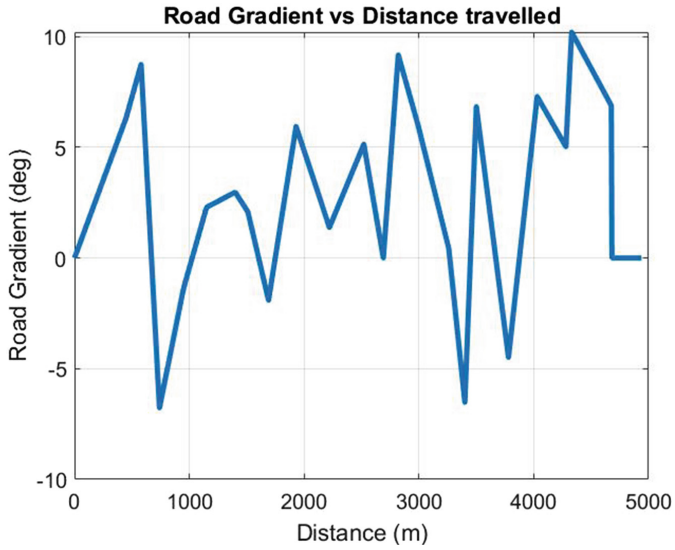


Fig. 2. Extracted elevation data

3.2 Results

In the current simulated scenario, the vehicle behaved comparatively well. The reference velocity was set to 40 km/h for the vehicle and velocity control contained the velocity error well within acceptable 1 km/h bound. Figure 3 shows the overall velocity of the vehicle and Fig. 4 shows the relative error in velocity in the whole simulation.

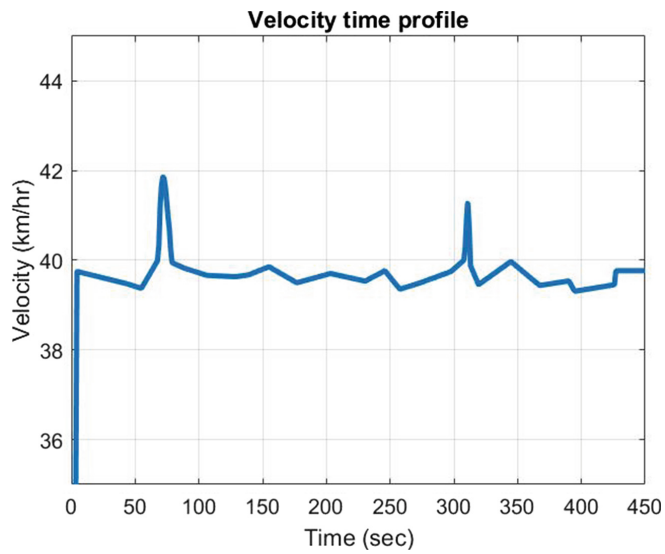


Fig. 3. Velocity profile over time

It is observed from the velocity profiles that it is easier for controlling the vehicle uphill compared to downhill as there are smooth transitions on positive gradient. The net effective traction force profile is given in Fig. 5. It is observed from the traction force profile that it follows the profile of the road gradient. Additionally, acceleration is zero in the negative gradient zone of the road which is also conceptually correct. Overall this shows satisfactory performance of the traction force controller.

The braking of the vehicle is triggered twice in the current scenario on the downhill journey of the vehicle. When compared with the gradient of the road, it shows coherence of braking with negative road gradient. A dead-band of 1 km/h is deliberately selected for avoiding any unnecessary use of braking. Figure 6 shows the comparison of braking force with the velocity error. This shows that braking is only triggered when higher than reference velocity is attained.

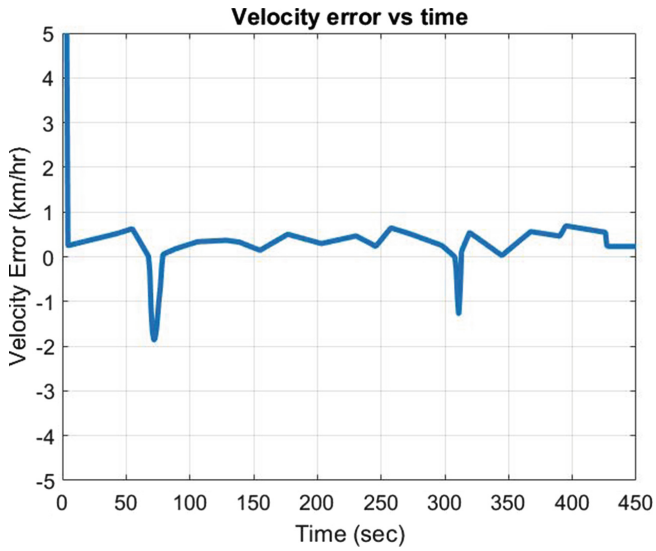


Fig. 4. Velocity error profile

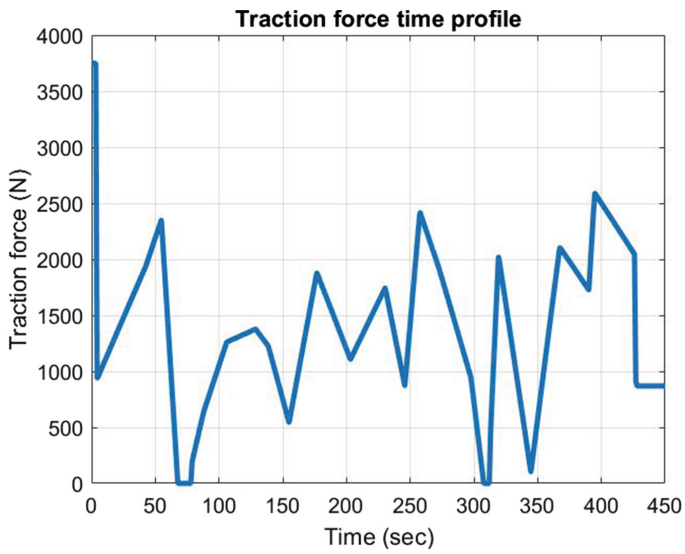
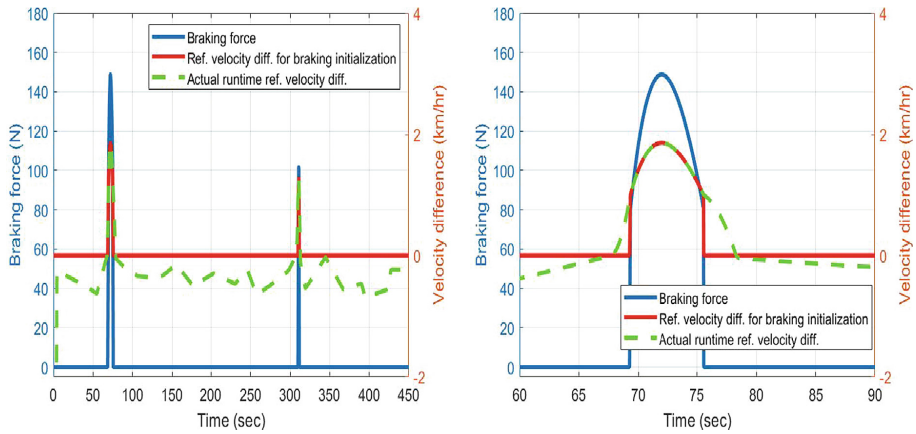


Fig. 5. Traction force profile

If we zoom-in Fig. 6 a) at one of the braking instances, as shown in Fig. 6 b), we can clearly observe the application of the velocity dead-band implemented for braking. Where green shows the velocity difference profile and red denotes the velocity difference profile input to the braking controller.



(a) Full profile (b) Zoomed profile

Fig. 6. Braking force and velocity error profile

4 Conclusion

The mathematical model for FURBOT worked as per requirements. The velocity control for the vehicle created nominal errors which were within tolerable range of 2 km/h. The switching between the cascade controller for velocity control also behaved as per need. The vehicle was able to maintain its velocity over the uneven hilly terrain which was the goal of the research. Furthermore, the whole mathematical model generated realistic results.

After modeling complete road topology and embedding traffic data into the simulation, steering control can be incorporated. This can make the vehicle's mathematical model complete and autonomous which will be of critical importance for selecting and testing new sensors for the vehicle. This model will additionally serve as a platform for the future work on this autonomous vehicle. Number of safety enhancements can be incorporated in the vehicle after simulating its behavior. Some future outputs of this system include path planning, safe parking, cargo alignment and enhancing the safety of the vehicle and the environment which includes bounds on top speed, radial velocity, minimum safe distance and operational battery life before recharge is required.

Acknowledgment. This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 875530, project SHOW (SHared automation Operating models for Worldwide adoption).

References

1. Westervelt, E.R., Grizzle, J.W., Chevallereau, C., Choi, J.H., Morris, B.: Feedback control of dynamic bipedal robot locomotion. CRC Press, Boca Raton (2018)

2. Fiore, E., Giberti, H., Ferrari, D.: Dynamics modeling and accuracy evaluation of a 6-DoF Hexaslide robot. In: Nonlinear Dynamics. Conference Proceedings of the Society for Experimental Mechanics Series, vol. 1, pp. 473–479 (2016). https://doi.org/10.1007/978-3-319-15221-9_41
3. Pedrammehr, S., Qazani, M.R.C., Abdi, H., Nahavandi, S.: Mathematical modelling of linear motion error for Hexarot parallel manipulators. Appl. Math. Model. **40**(2), 942–954 (2016). https://doi.org/10.1016/j.apm.2015.07.004
4. Esmaeili, N., Alfi, A., Khosravi, H.: Balancing and trajectory tracking of two-wheeled mobile robot using backstepping sliding mode control: design and experiments. J. Intell. Robot. Syst. **87**(3–4), 601–613 (2017). https://doi.org/10.1007/s10846-017-0486-9
5. Asano, F., Seino, T., Tokuda, I., Harata, Y.: A novel locomotion robot that slides and rotates on slippery downhill. In: 2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM) (2016). https://doi.org/10.1109/aim.2016.7576804
6. Rodriguez, R., Ardila, D.L., Cardozo, T., Perdomo, C.A.C.: A consistent methodology for the development of inverse and direct kinematics of robust industrial robots. J. Eng. Appl. Sci. **13**(1), 293–301 (2018)
7. Marzbani, H., Khayyam, H., To, C.N., Quoc, D.V., Jazar, R.N.: Autonomous vehicles: autodriver algorithm and vehicle dynamics. IEEE Trans. Veh. Technol. **68**(4), 3201–3211 (2019). https://doi.org/10.1109/tvt.2019.2895297
8. Ehsani, M., Gao, Y., Longo, S., Ebrahimi, K.: Modern Electric, Hybrid Electric, and Fuel Cell Vehicles. CRC Press, Taylor & Francis Group, Boca Raton (2019)
9. Wiegand, B.P.: Estimation of the Rolling Resistance of Tires. SAE Technical Paper Series (2016). https://doi.org/10.4271/2016-01-0445
10. Wiegand, B.P.: Mass Properties and Advanced Automotive Design. SAWE Technical Paper 3602, 74th SAWE International Conference on Mass Properties Engineering; Alexandria, VA (2015)
11. Dixon, J.C.: Suspension Geometry and Computation. John Wiley & Sons Ltd., Chichester, UK (2009). ISBN 978-0-470-51021-6
12. Fu, C., Hoseinnezhad, R., Bab-Hadiashar, A., Jazar, R.N.: Electric vehicle side-slip control via electronic differential. Int. J. Veh. Auton. Syst. **6**, 1–26 (2014)
13. Google (n.d.): Google Maps directions for driving from Piazza del Portello, Genova to Righi, Genova. https://www.google.com/maps/dir/44.4114759,8.9345774/44.4241951,8.9379112/@44.4185213,8.9331592,15z/data=!4m2!4m1!3e0?hl=en. Accessed 12 Sept 2019

electronics

MDPI

Article

Simulated Sensor Based Strategies for Obstacle Avoidance Using Velocity Profiling for Autonomous Vehicle FURBOT

Khayyam Masood *,†, Rezia Molfino and Matteo Zoppi

PMAR Robotics, DIME, University of Genova, 16126 Genova, Italy; molfino@dimec.unige.it (R.M.); zoppi@dimec.unige.it (M.Z.)

* Correspondence: khayyam.masood@edu.unige.it; Tel.: +39-3205304054

† Current address: Università di Genova, PMAR Robotics Group, DIME—Dipartimento di Ingegneria Meccanica, Energetica, Gestionale e dei Trasporti, Via all'Opera Pia, 15/a, 16145 Genova, Italy.

Received: 21 April 2020; Accepted: 22 May 2020; Published: 26 May 2020

Abstract: Freight Urban Robotic Vehicle (FURBOT) is an autonomous vehicle designed to transport last mile freight to designated urban stations. It is a slow vehicle designed to tackle urban environment with complete autonomy. A slow vehicle may have slightly different strategies for avoiding obstacles. Unlike on a highway, it has to deal with pedestrians, traffic lights and slower vehicles while maintaining smoothness in its drive. To tackle obstacle avoidance for this vehicle, sensor feedback based strategies have been formulated for smooth drive and obstacle avoidance. A full mathematical model for the vehicle is formulated and simulated in MATLAB environment. The mathematical model uses velocity control for obstacle avoidance without steering control. The obstacle avoidance is attained through velocity control and strategies are formulated with velocity profiling. Innovative techniques are formulated in creating the simulated sensory feed-backs of the environment. Using these feed-backs, correct velocity profiling is autonomously created for giving velocity profile input to the velocity controller. Proximity measurements are assumed to be available for the vehicle in its given range of drive. Novelty is attained by manipulating velocity profile without prior knowledge of the environment. Four different type of obstacles are modeled for simulated environment of the vehicle. These obstacles are randomly placed in the path of the vehicle and autonomous velocity profiling is verified in simulated environment. The simulated results obtained show satisfactory velocity profiling for controller input. The current technique helps to tune the existing controller and in designing of a better velocity controller for the autonomous vehicle and bridges the gap between sensor feed-back and controller input. Moreover, accurate input profiling creates less strain on the system and brings smoothness in drive for an overall safer environment.

Keywords: velocity profiling; sensor feedback; autonomous drive; obstacle avoidance; hurdle detection; FURBOT

1. Introduction

Work on autonomous vehicle and specially for obstacle avoidance is being studied in depth in recent times due to high demand of autonomous vehicles. Within literature, obstacle avoidance for autonomous vehicles mainly fall in the category of path planning and steering control [1–8]. Very few of the articles have mentioned obstacle avoidance taking into account the velocity adjustment to the obstacle [2,4,9]. Steering control for hurdle avoidance is a good technique specially for relatively high speed vehicles or for the vehicles which require to avoid obstacles on highway [1,3,6] but for slow moving vehicles, where there is usually only one designated lane, this approach is not viable.

From the research on autonomous vehicle it is clear that the controller choice for autonomous vehicles is model predictive controller or MPC [2,3,5–11]. This is because MPC has the ability to anticipate future events and can take control actions accordingly, unlike PID controllers which do not have this predictive ability. Many researchers used MPC for obstacle avoidance with steering control [2,3,5–8], however there is an exception in reference [4] where MPC is used to not only avoid obstacles through steering control, but also for control velocity of the vehicle in case of unexpected maneuver of the vehicle ahead. Although MPC is famous choice for many for control of autonomous vehicles [10]; many other types of control algorithms are also used for velocity and steering control. Among them sliding mode controller [12] and fuzzy logic controller [13,14] are highlighted. Similarly other control theories are also used for autonomous vehicles depending on choice and situation of user.

The outlook of the past decade in autonomous vehicles given in [10] also shows that it is difficult to seamlessly integrate motion prediction with vehicle control. Not only that, MPC has reported issues with steering control [10]. So the use of PID for our vehicle seems as a good enough option only if we can predict our motion computationally without the help of MPC. The study on autonomous vehicles in the last decade also shows that there has been very little work done on unexpected hurdles or tackling of usual urban obstacles, e.g., traffic lights.

Much of the autonomy of autonomous vehicles revolves around cruise control [1,3,15]. This is very essential for vehicles driving at high speed or at highway. Reference [6] used additional constraints for high-speed vehicles in the sense of toppling of the vehicle. The relatively less constrained autonomous vehicles, are not only dangerous for themselves, but are also dangerous for the environment. Another interesting work which is done recently is to maintain center-line i.e., center-line control [3,9,15]. This is another very key aspect for autonomous driving specially in cruise control. Lane discipline is one of the primary requirements for autonomous vehicles. Experimental obstacle avoidance is rarely found in literature for a full autonomous ground vehicle (AGV) however, in [16] obstacle avoidance with experimental results was achieved with AGV using steering control.

Freight Urban Robotic vehicle is an autonomous vehicle which is required to operate autonomously in urban environment for last mile freight delivery. It has to achieve this goal for the European H2020 project SHOW (SHared automation Operating models for Worldwide adoption) in 2021 in Trikala/Greece. In the demonstration, it needs to operate in bicycle lane with no lane change option available. Figure 1 shows the demonstration area for FURBOT in Trikala for SHOW project. This work is an attempt to achieve the goals of automation of FURBOT set by project SHOW. The velocity controller for FURBOT requires true classification of velocity profile in order for correct automation. Since the vehicle is required to perform in urban environment at low speeds, it is imperative that it will encounter hurdles within its path. Thus, the need for obstacle avoidance is generated.

(**a**) Demonstration area first angle (**b**) Demonstration area second angle

Figure 1. Demonstration area for Freight Urban Robotic Vehicle (FURBOT) in Trikala/Greece (Source street view, google maps [17]).

Very few articles solve the issue of obstacle avoidance in urban setting. Reference [8] is one of the few studies which conducted obstacle avoidance in urban setting. However, only vehicle avoidance was observed. Other obstacles which vehicles may encounter were not observed in this study.

For slow moving vehicles, obstacle avoidance is usually limited to autonomous parking problem [18,19]. This is the usual setting where vehicle speeds are slow and speed control is also necessary in addition to steering control. However, this problem is very limited in path driven of the vehicle yet it has positive practical applications in autonomous drive. Using mathematical models for trajectory generation is also noted work in autonomous vehicles [11,16] yet is one of the initial steps for autonomous driving modules.

INRIA team RITS (Robotics and Intelligent Transportation Systems) have been working on autonomous vehicles for more than 15 years and is working on similar type of slow vehicles for obstacle avoidance. They have published on pedestrian detection [20,21] and pedestrian avoidance [22]. The approach used for pedestrian hurdle avoidance is the same which is used in this research i.e., slowing down of vehicle until the hurdle goes away from the path thus avoiding the hurdle. Their research is also a benchmark for this research. Velocity profiling for obstacle avoidance or navigation adjustment due to obstacles is still a novel concept and is being under study. Apart from INRIA's work in [22] for pedestrian obstacle avoidance; another use of velocity profiling for cruise control behind a slow moving object is shown in [23]. Velocity profiling is not just essential for avoiding obstacles, but it is also necessary to ensure smooth drive and drive comfort [24].

Our problem revolves around FURBOT which has to tackle obstacles and adjust itself while maintaining its lane. The best option to do so is not through steering control due to lane limitation, but through velocity control. Moreover, it has to tackle other type of hurdles within its path except moving vehicles e.g., traffic lights. Little work is found in literature where traffic signal is treated as an obstacle for autonomous vehicles. The usual work found on traffic lights is usually on its recognition system [25]. Obstacle avoidance for a slow moving autonomous ground vehicle which has its dedicated lane is not found in literature. Thus the need of our work is generated. This is emphasized for the vehicles which has to navigate in dense environment to be prepared for all types of obstacles, whether it is a slow vehicle, traffic signal or human obstruction; the vehicle in such urban environment needs to be able to adjust according to all such obstacles. In Addition to above, FURBOT has to manage obstacle avoidance by adjusting its speed as it has to operate in single lane. Thus, autonomous velocity profiling is very essential for its drive. Keeping lane restriction in mind and goal of avoiding obstacles, the best option is to cater with autonomous velocity profiling. This need sets the tone for technical requirement for our work that is carried out in this research.

2. Lateral and Longitudinal Equation of Motion

A summarized mathematical model that is used is presented using Newton-Euler equations of motion for forces along x-axis, y-axis and moment along z-axis and are given by Equations (1)–(3).

$$\dot{v_x} = \frac{\sum F_x}{M_v} = \frac{F_t + F_d + F_g + F_r}{M_v} \tag{1}$$

$$\dot{v_y} = \frac{1}{M_v v_x}(-a_1 C_{af} + a_2 C_{ar})r - \frac{1}{M_v v_x}(C_{af} + C_{ar})v_y + \frac{1}{M_v} C_{af}\delta - r v_x \tag{2}$$

$$\dot{r} = \frac{1}{I_z v_x}(-a_1^2 C_{af} - a_2^2 C_{ar})r - \frac{1}{I_z v_x}(a_1 C_{af} - a_2 C_{ar})v_y + \frac{1}{I_z} a_1 C_{af}\delta \tag{3}$$

In above equations; M_v is mass of vehicle, F_t is the traction force, F_d is drag, F_g is gravitational and F_r is the rolling friction force acting on the vehicle. Similarly, I_z is moment of inertia of vehicle and C_{af}/C_{ar} are the cornering stiffness of front/rear wheels respectively. δ is the steering angle and a_1/a_2 are the distances of the rear/front wheels from the CG of the vehicle. The detailed mathematical custom made model for FURBOT is previously published in [26]. This model is used as a platform for

modelling the obstacles in the same MATLAB/SimuLink environment and implementing the obstacle avoidance techniques presented in this research.

3. Obstacles Characteristics and Modelling

Obstacle characterization is very vital for autonomous control of the vehicle. Four obstacles of different nature are studied and modelled for velocity profiling for FURBOT. The mathematical modelling and approach to the obstacles for simulation are discussed in their respective section below. The four different nature of obstacles modelled are

- Traffic signal
- Human/Stationary object
- Speed breaker
- Slow vehicle interaction

3.1. Traffic Signal

Creating the mathematical signal for identification of traffic signal and the color change posed three main hurdles; Signalling out current color of the light, approaching yellow light after green or red light and finally approach velocity to the light wrt current traffic light color. For signalling out the current color of the traffic light, each color detection is equated with a numeric value termed as t_l; 0 for green, 1 for yellow and 2 for red. With the help of three pulse generators, a desired output for traffic signal was generated. The signal attributes chosen were 60 s for red, 10 s for yellow and 40 s for green. The numeric signal out for the traffic light is shown in Figure 2.

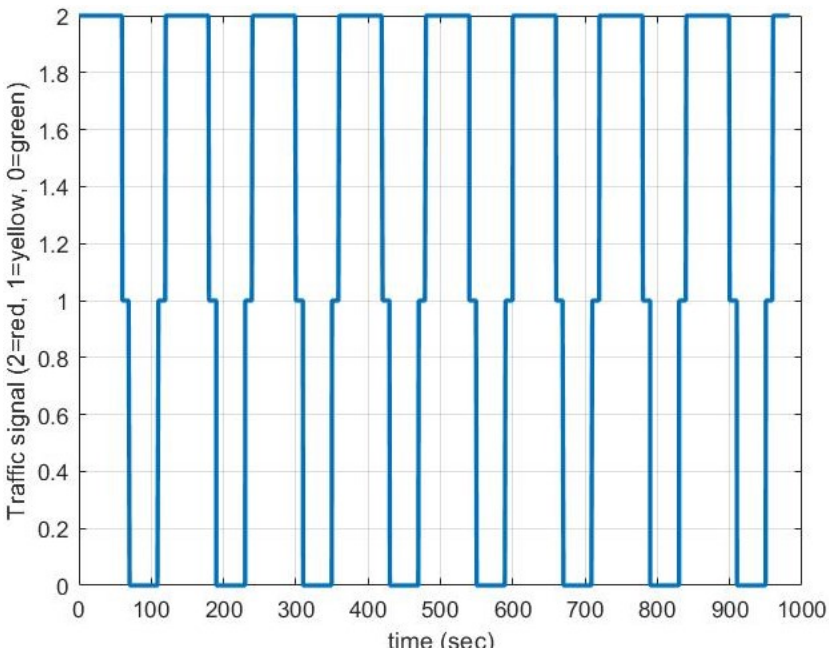
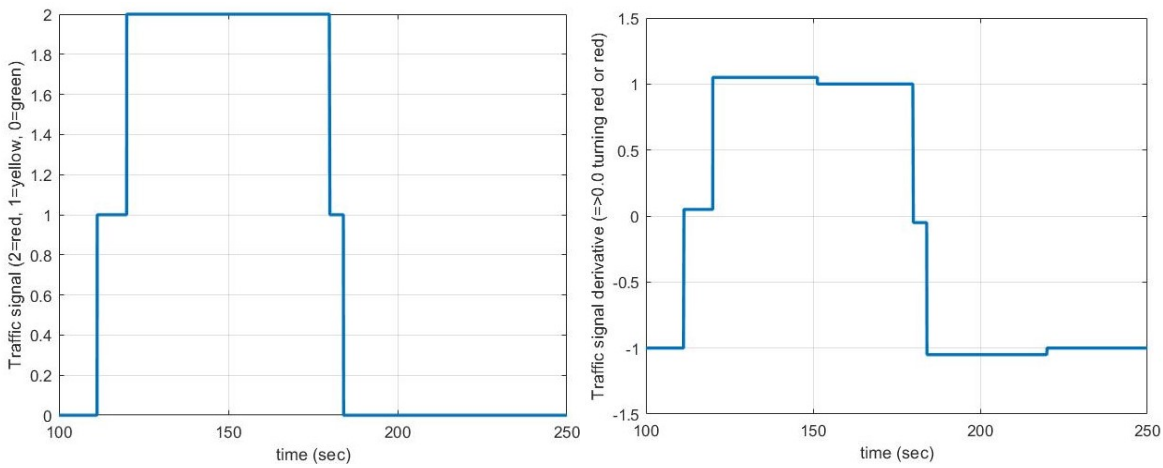


Figure 2. Traffic signal output.

An unfamiliar problem arose while modelling the vehicle velocity profile when approaching traffic light. The issue was how to approach yellow light when it is turning from red to yellow or from green to yellow. The idea is to stop before the signal in case light is turning yellow after green and to keep driving if it is turning from red to yellow or switch to cruise if the car is stopped at signal. An intuitive approach is used to tackle this issue; since numeric values were assigned to colors of signal, a derivative coupled with rate limiter of traffic light numeric value wrt time could define whether the signal is turning red or turning green after yellow signal. Using this ideology, an intuitive conditional equation for reference velocity $V_{ref_{sig}}$ as given by Equation (4) is formulated.

$$V_{ref_{sig}} = \begin{cases} v_{ref_{red}}, & \text{if } \frac{d}{dt}t_{lr} + t_l - 1 \geq 0 \\ v_{ref_{green}}, & \text{if } \frac{d}{dt}t_{lr} + t_l - 1 < 0 \end{cases} \quad : \quad t_{lr} = \begin{cases} R_t\Delta t + t_l(i-1), & \text{if } \frac{d}{dt}t_{lr} \geq R_t \\ t_l, & \text{if } R_t > \frac{d}{dt}t_{lr} > -R_t \\ -R_t\Delta t + t_l(i-1), & \text{if } -R_t \geq \frac{d}{dt}t_{lr} \end{cases} \tag{4}$$

where R_t is the rate limiter valued at 0.05 and $v_{ref_{red}}$ / $v_{ref_{green}}$ are the reference velocities for red and green light respectively. Using this equation, we can stop the vehicle if the value is greater than 0 or keep cruising if it is less than 0 keeping in view the yellow traffic light signal condition. Figure 3 shows vehicle approach to the signal. The start of the signal corresponds to the signal capturing by sensors of the traffic light. Figure 3a reflects the sensor output of the traffic light detection whereas Figure 3b outs the conditional derivative value of the signal sensor.



(**a**) Traffic light sensor feedback (**b**) Sensor feedback derivative

Figure 3. Traffic light sensor and derivative feedback.

The reference velocity profiling for traffic signal is kept as to steadily decrease the velocity of vehicle to zero on approach to the traffic light. This is achieved by considering current velocity of vehicle and distance to the light as given in Equation (5) where $V_{ref_{sig}}$ is reference velocity due to traffic light, d_{obs} is remaining distance to the obstacle, d_{band} is stop criteria for vehicle, d_{init} is initial detection of the obstacle and V_{ref} is desired reference velocity of the vehicle i.e., 20 km/h in our case. An example velocity profile using this formula is thus shown in Figure 4. The criteria for stopping the vehicle is to stop 2 m before the traffic light i.e., d_{band}. The same formula is used for velocity profiling in other obstacles as well except speed bump where simple switch between the velocity profiles is used.

$$V_{ref_{sig}} = \frac{d_{obs} - d_{band}}{d_{init}} \times V_{ref} \tag{5}$$

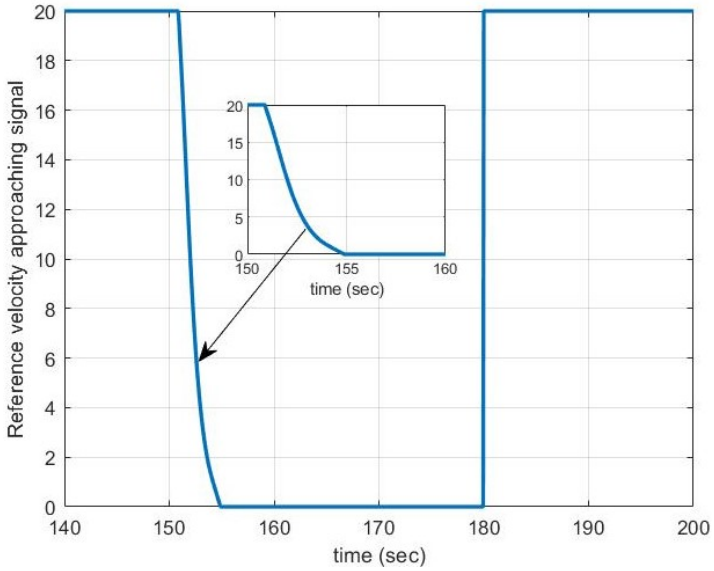


Figure 4. Traffic signal output.

3.2. Human/Stationary Object

Unlike in airborne vehicles, road vehicles can encounter sudden unknown obstacles e.g., humans or animals. These types of obstacles require sudden and immediate braking or if the object is farther away from the vehicle, a quick decrease in the overall velocity of the vehicle. This need is further explained in reference [22]. This aspect of driving phenomenon is taken into consideration in order to model such obstacles for better controlling the vehicle.

The stationary object/obstacle/human detection was randomly placed between 2∼20 m ahead of the vehicle. The duration of the object in the navigation path was randomly fixed between 2∼30 s. If the object moved away before the arrival of the vehicle, the vehicle continues with its reference velocity otherwise it adjusts its velocity to be zero 2 m ahead of the obstacle. This helps in gradually decreasing the velocity of the vehicle in case the object is away from the vehicle or to apply sharp and sudden brakes if it is in the vicinity of the obstacle. The reference velocity profile adjustment for a single such obstacle detection is shown in Figure 5.

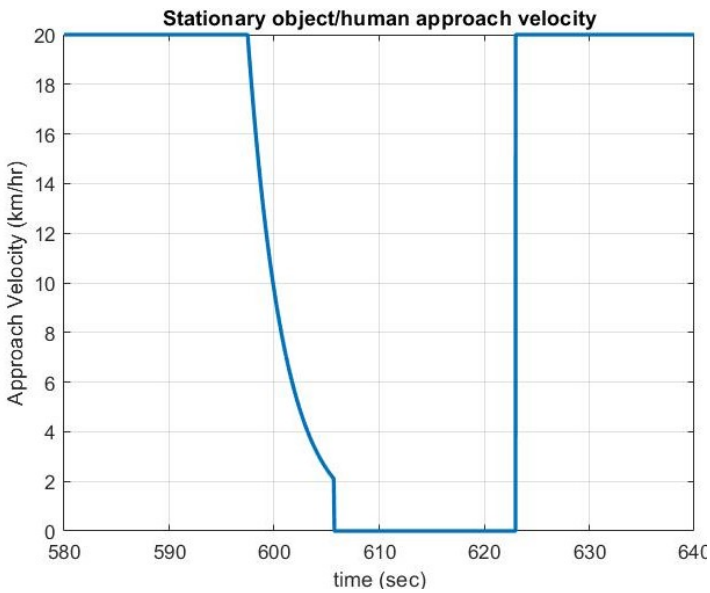


Figure 5. Velocity profiling for stationary object.

From observing the Figure 5 we see how the velocity is gradually adjusted to keep the object safe and to avoid a collision. During a single 5 km patch of simulation, five stationary objects with random times in-front of the vehicle are simulated. The placement of such hurdles was random in the 5 km patch. Additionally, the timing in-front of the vehicle of the obstacles and their distance immediate to the vehicle was also random between the limits mentioned above.

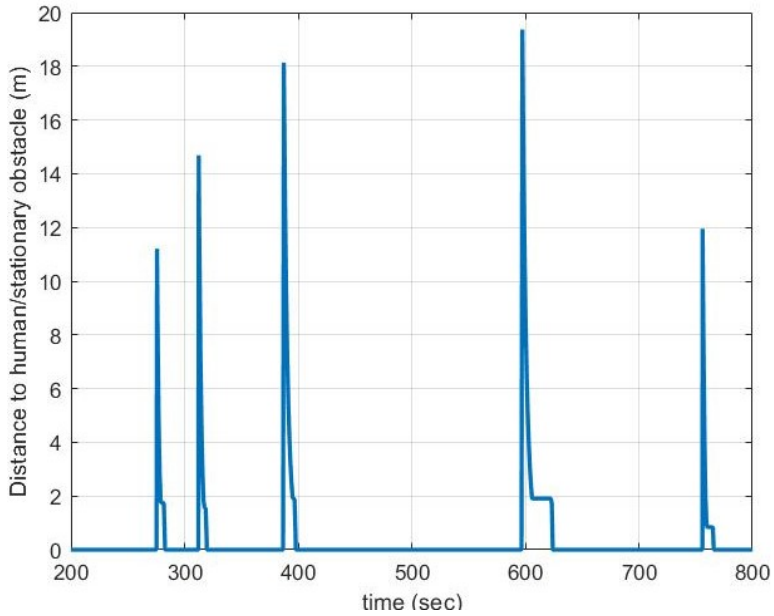
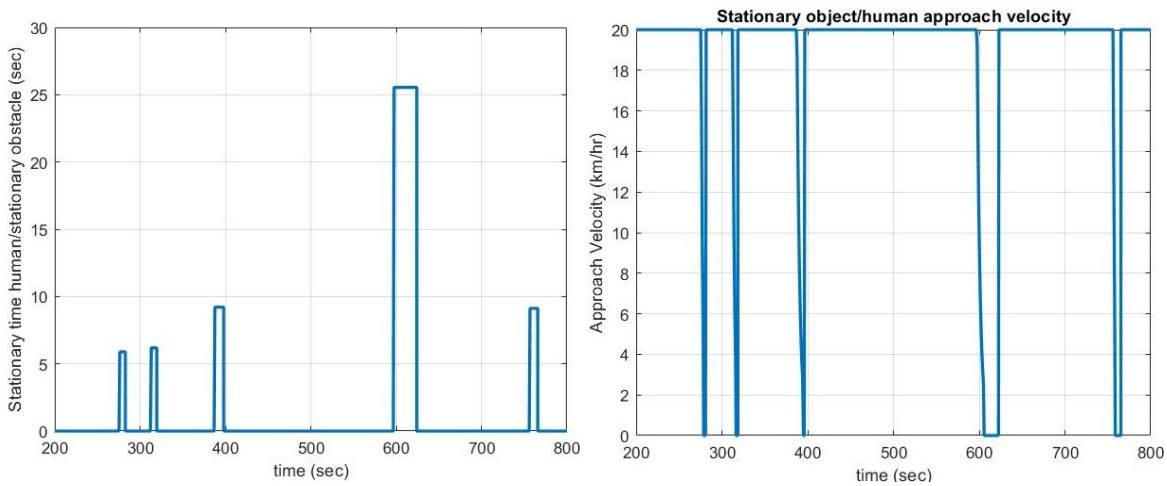


Figure 6. Vehicle distance to obstacle.

Figure 6 shows the detected object in-front of the vehicle. It is noted that the condition for stopping 2 m before the obstacle was not always met. In the last hurdle, the distance reduced to 1 m which was due to the fact that the car was on a slope to the approach. However, in all the cases, the vehicle stopped before colliding with the stationary hurdle. The approach velocity and the time of the obstacle in-front of the vehicle are is plotted in Figure 7.



(**a**) Time in-front of the vehicle (**b**) Reference velocity adjustment

Figure 7. Stationary obstacle collision avoidance.

3.3. Speed Breaker

The velocity adjustment in case of speed breaker obstacle was kept simple as the velocity drop is relatively small and there is no behavioral change in the obstacle. On detection of speed bump,

the velocity of FURBOT is adjusted to 10 km/h. The overall velocity adjustment on the detection of speed breaker is shown in Figure 8.

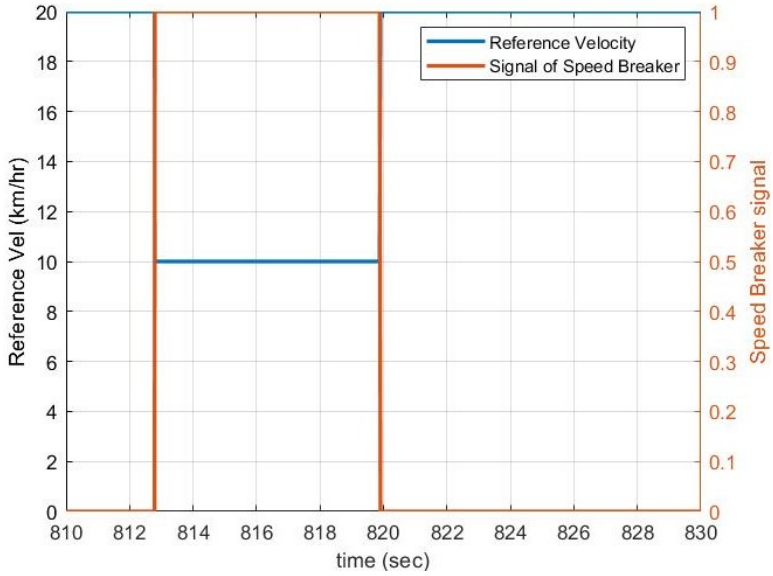


Figure 8. Speed breaker hurdle avoidance.

3.4. Slow Vehicle Interaction

One of the biggest hurdles/obstacles which are faced by driving vehicles is the encounter of slow vehicles and thus adjusting the velocity of the vehicle to that of the slow vehicle in-front. For this purpose, a slow vehicle is modelled in the path of the vehicle which comes randomly in its 5 km journey. The number of slow vehicles, their velocity profiles and any additional behaviors could be adjusted in the simulation. However, to test the controller, there is set to be a single slow vehicle which has a continuous velocity profile of 10 km/h. A random signal is generated for simulation purposes which shows that the vehicle has now moved away from the path of the vehicle. This signal is randomly generated between 5∼60 s of encountering the vehicle. After the slow vehicle signal disappears, the vehicle continues with it's default reference velocity profile.

Figure 9 shows the overall interaction of FURBOT with the slow vehicle. The velocity profile of the vehicle is gradually adjusted with time as to minimize the distance between the two vehicles and for not applying sudden brakes. The reference velocity adjustment takes the same approach of decreasing velocity gradually with the distance to the obstacle as was similar with the other obstacles. In Figure 9 the blue line represents the current position of the slow vehicle wrt time, the dotted red line shows the reference velocity of the FURBOT and the orange line shows the distance to the vehicle. The detection of slow vehicle is set to be 20 m away. We can see that the reference velocity starts to gradually decrease at 20 m markup and attains the velocity of the vehicle 2 m before it, thus avoiding the collision. As soon as the signal of the vehicle moving away from the path of the vehicle is received, the vehicle reference velocity is restored to its original value.

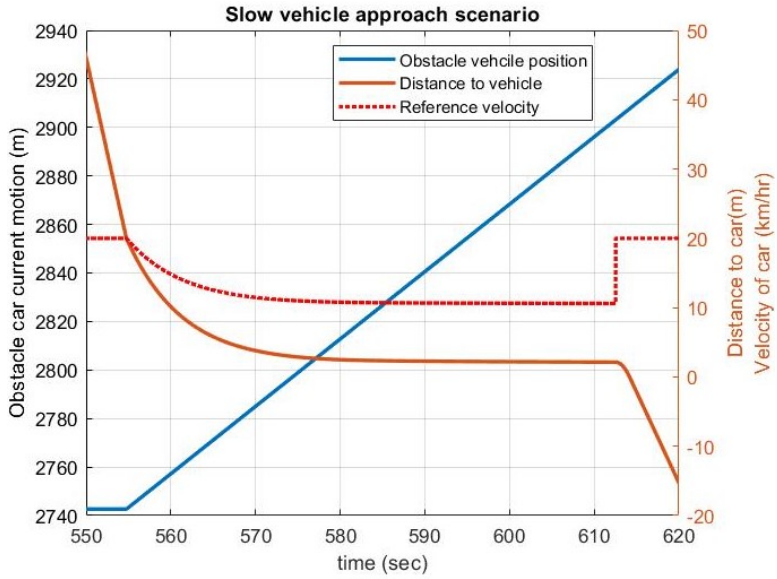


Figure 9. Slow vehicle hurdle avoidance.

4. Simulation Platform and Results

All the obstacles mentioned were modelled and simulated in a single run for a 5 km drive with varying road elevation profile chosen arbitrarily from an expected logistic path of the vehicle. The logistic path and the elevation profile chosen are available in reference [26]. The obstacles present in the simulation are given in Table 1. Furthermore, the whole obstacle avoidance technique is further explained in the flowchart in Figure 10. One additional condition from flowchart information is also placed which is given in Equation (6) in case of multiple obstacle detection.

$$V_{ref} = min(V_{ref_{ped}}, V_{ref_{sig}}, V_{ref_{car}}, V_{ref_{bump}}) \tag{6}$$

where $V_{ref_{ped}}$ is the reference velocity generated due to pedestrian, $V_{ref_{car}}$ is reference velocity from moving car obstacle and $V_{ref_{bump}}$ is reference velocity from speed bump obstacle. The above condition is additionally placed in case multiple obstacles are detected in the presence of each other.

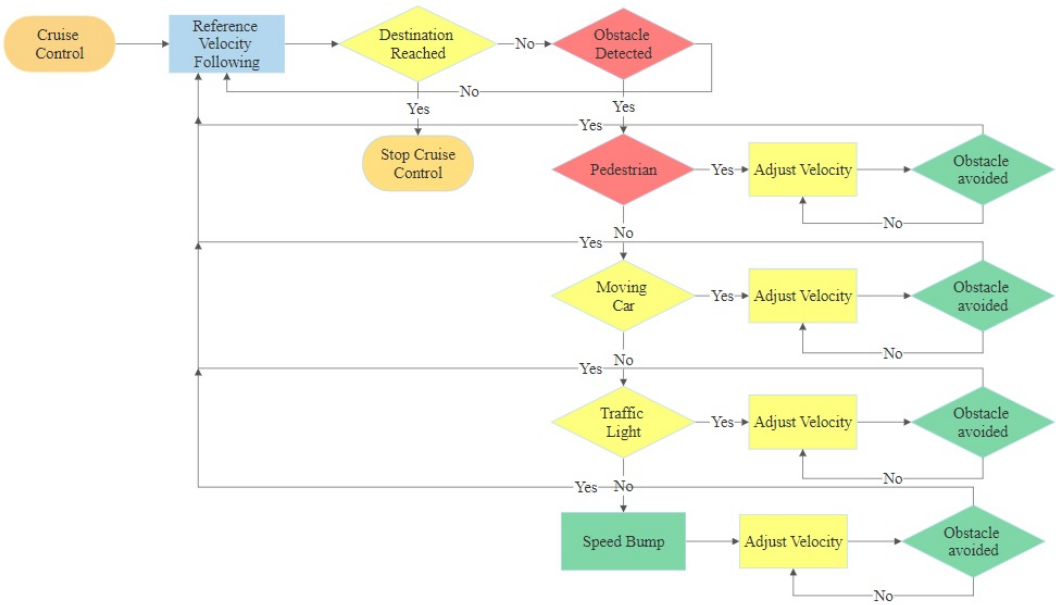


Figure 10. Flowchart of obstacle avoidance algorithm.

Table 1. Type of obstacles present in simulated drive.

Obstacle Type	Number of Obstacles	Time In-Front of Vehicle (s)	Distance In-Front of Vehicle (m)
Traffic light	1	Dependent on light	From 20
Stationary/Human	5	2∼30	2∼20
Speed Breaker	1	NA	From 20
Slow vehicle	1	5∼60	From 20

In case two or more obstacles are encountered simultaneously within the path of the vehicle; the vehicle adjusts its reference velocity in accordance with Equation (6). Figure 11 is shown as an example where two obstacles i.e., moving car and stationary person coincides during the passage of vehicle. The vehicle first adjust its reference velocity wrt slow vehicle but when human obstacle is detected, the reference velocity is automatically adjusted to the lower reference velocity.

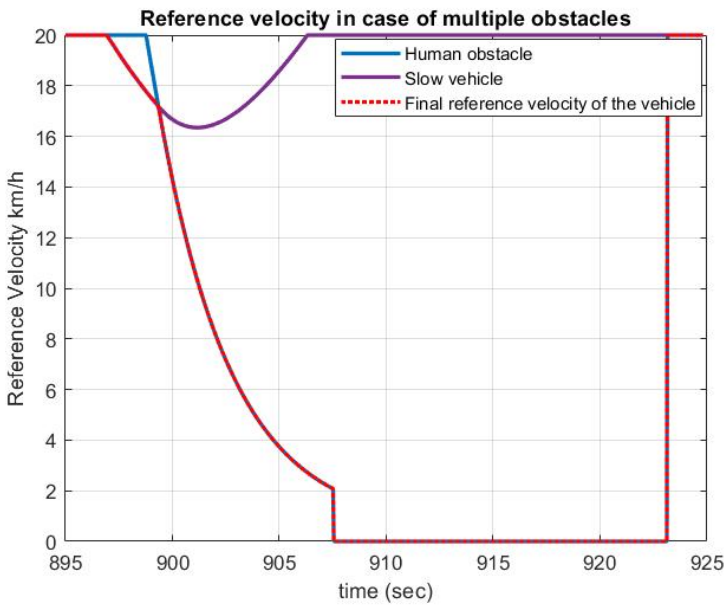


Figure 11. Reference velocity of the vehicle.

Two separate controllers are designed for velocity control of the vehicle; one for braking force and one for the traction force. This type of cascade controller is chosen deliberately as the requirements of acceleration and brakes are different for the vehicle. Additionally, when braking is applied, traction power is automatically shut down as in conventional vehicles. The idea of a low-level speed controller is taken from the idea introduced in Cycab [22] where the speed is regulated as a function of vehicle current speed and the target speed. The sampling time for the simulation is chosen as 1 ms and the reference velocity is updated at every sampling period. The controller for the vehicle had to be altered to adjust for obstacle avoidance. For braking control a PI controller was chosen whereas for traction force a PD controller was chosen. The traction force controller had an additional error amplification factor to adjust for power control. The overall controller configuration is given in Table 2.

Table 2. Velocity controllers.

Controller Type	Controller Values			
	Error Amplification	Proportional Gain	Integral Gain	Derivative Gain
Traction power control	500	20	0	1
Braking power control	1	550	6	0

The mathematical model for the vehicle was previously developed and behavior checked and verified in [26]. The reference velocity for the vehicle was set at 20 km/h unless the need to adjust reference to obstacles was required. The minimum reference velocity received either from the obstacles or in general is the final reference velocity of the vehicle in accordance with Equation (6). Figure 12 shows the reference velocity for the vehicle in the presence of aforementioned obstacles.

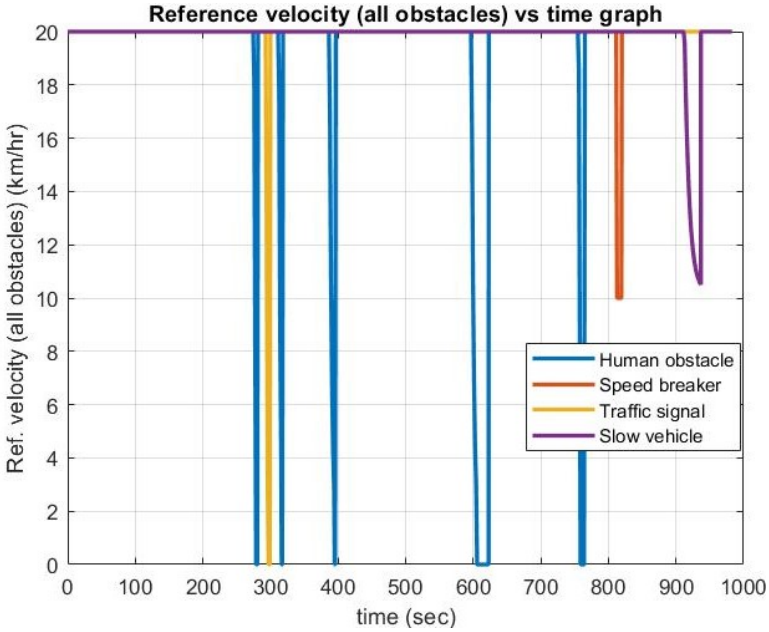


Figure 12. Reference velocity of the vehicle.

The braking control is only activated when there is positive difference between actual and reference velocities as shown in Figure 13. Moreover, the actual velocity of the vehicle and distance travelled by the vehicle are given in Figure 14. The bumps in the velocity profile when compared to reference velocity are due to the fact that there is a gradient profile of the road and the vehicle velocity changes due to weight shift of the vehicle. The simulation results displayed satisfactory acknowledgement of hurdles by the algorithm and the controller took correct preventive measures to avoid collision or adjusting to traffic light/speed breaker.

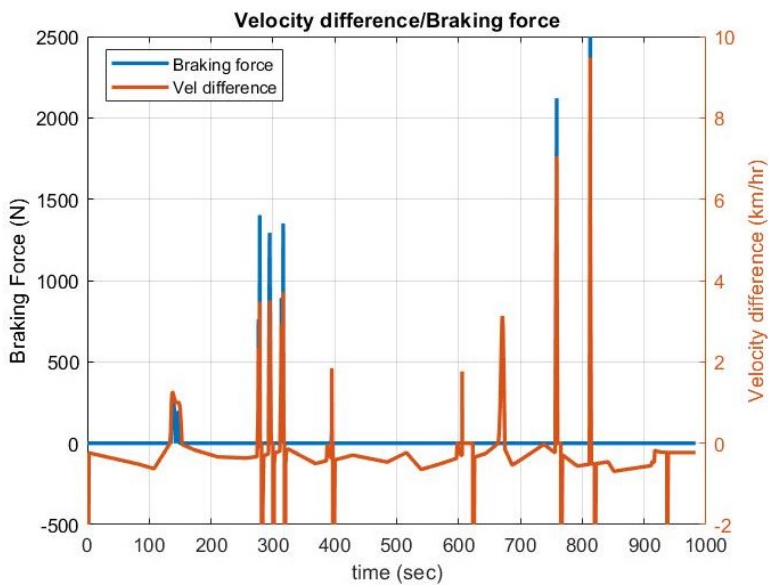
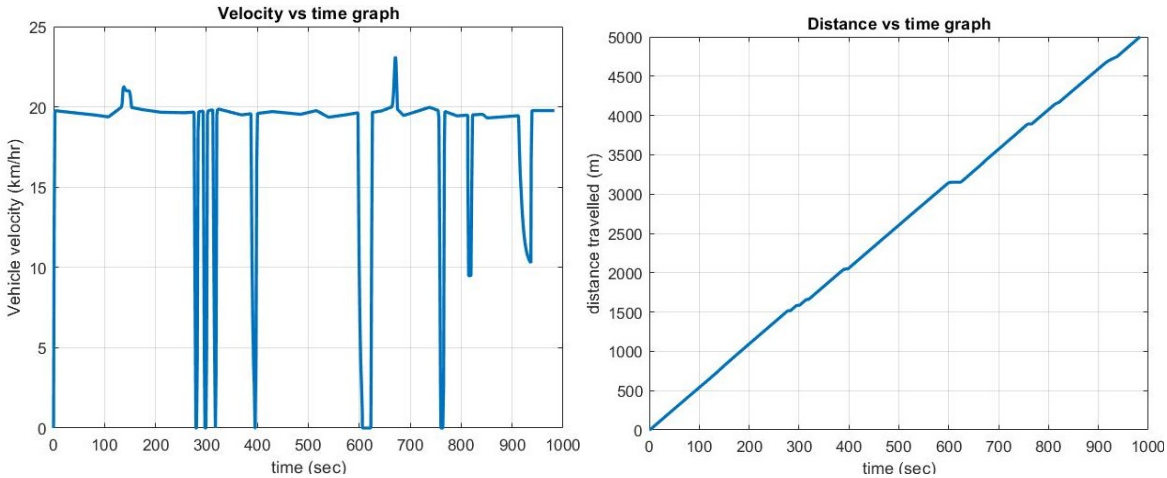


Figure 13. Braking force wrt velocity difference.



(**a**) Ground velocity of FURBOT (**b**) Distance travelled

Figure 14. Stationary obstacle collision avoidance.

5. Conclusions

Obstacle avoidance for lane-restricted slow vehicles is studied and results are shared. Velocity profiling was formulated in this research and implemented in the simulated environment for the project FURBOT. The correct velocity profiling successfully helped in controlling the vehicle and avoiding the obstacles and collisions. The approach in itself is novel as it creates less strain on the controller with multiple inputs. All four hurdles are successfully modelled and avoided in random fashion within the course of drive for the vehicle FURBOT in the simulated environment. Additionally, this research will further help the autonomy of autonomous vehicles while they are commuting in a semi autonomous environment i.e., where manual and autonomous vehicles drive together. True velocity profiling in urban environment can also take away another driver based decision for manual cars. Furthermore, with this technique being implemented, we are one step closer to driving our autonomous vehicle FURBOT in urban environment setting for project SHOW.

Author Contributions: Conceptualization, K.M.; methodology, K.M.; software, K.M.; validation, K.M. and M.Z.; formal analysis, K.M.; investigation, K.M.; resources, K.M.; data curation, K.M.; writing—original draft preparation, K.M.; writing—review and editing, K.M. and M.Z.; visualization, K.M.; supervision, R.M. and M.Z.; project administration, M.Z.; funding acquisition, R.M. and M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 875530, project SHOW (SHared automation Operating models for Worldwide adoption).

Conflicts of Interest: The authors declare no conflict of interest.The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

FURBOT	Freight Urban Robotic Vehicle
SHOW	SHared automation Operating models for Worldwide adoption

References

1. Brown, M.; Funke, J.; Erlien, S.; Gerdes, J.C. Safe driving envelopes for path tracking in autonomous vehicles. *Control Eng. Pract.* **2017**, *61*, 307–316. [CrossRef]
2. Ji, J.; Khajepour, A.; Melek, W.W.; Huang, Y. Path Planning and Tracking for Vehicle Collision Avoidance Based on Model Predictive Control with Multiconstraints. *IEEE Trans. Veh. Technol.* **2017**, *66*, 952–964. [CrossRef]
3. Rosolia, U.; Bruyne, S.D.; Alleyne, A.G. Autonomous Vehicle Control: A Nonconvex Approach for Obstacle Avoidance. *IEEE Trans. Control Syst. Technol.* **2017**, *25*, 469–484. [CrossRef]
4. Magdici, S.; Althoff, M. Fail-safe motion planning of autonomous vehicles. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016. [CrossRef]
5. Abbas, M.A.; Milman, R.; Eklund, J.M. Obstacle avoidance in real time with Nonlinear Model Predictive Control of autonomous vehicles. In Proceedings of the 2014 IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE), Toronto, ON, Canada, 5–8 May 2014. [CrossRef]
6. Liu, J.; Jayakumar, P.; Stein, J.L.; Ersal, T. Combined Speed and Steering Control in High-Speed Autonomous Ground Vehicles for Obstacle Avoidance Using Model Predictive Control. *IEEE Trans. Veh. Technol.* **2017**, *66*, 8746–8763. [CrossRef]
7. Febbo, H.; Liu, J.; Jayakumar, P.; Stein, J.L.; Ersal, T. Moving obstacle avoidance for large, high-speed autonomous ground vehicles. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017. [CrossRef]
8. Schwarting, W.; Alonso-Mora, J.; Paull, L.; Karaman, S.; Rus, D. Safe Nonlinear Trajectory Generation for Parallel Autonomy With a Dynamic Vehicle Model. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2994–3008. [CrossRef]
9. Altche, F.; Polack, P.; de la Fortelle, L. High-speed trajectory planning for autonomous vehicles using a simple dynamic model. In Proceedings of the 2017 IEEE 20th Internationalaaonference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017. [CrossRef]
10. Berntorp, K. Path planning and integrated collision avoidance for autonomous vehicles. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017. [CrossRef]
11. Li, B.; Du, H.; Li, W.; Zhang, B. Integrated trajectory planning and control for obstacle avoidance manoeuvre using non-linear vehicle MP algorithm. *IET Intell. Transp. Syst.* **2019**, *13*, 385–397. [CrossRef]
12. Hu, C.; Wang, R.; Yan, F. Integral Sliding Mode-Based Composite Nonlinear Feedback Control for Path Following of Four-Wheel Independently Actuated Autonomous Vehicles. *IEEE Trans. Transp. Electrif.* **2016**, *2*, 221–230. [CrossRef]
13. Guo, J.; Hu, P.; Wang, R. Nonlinear Coordinated Steering and Braking Control of Vision-Based Autonomous Vehicles in Emergency Obstacle Avoidance. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3230–3240. [CrossRef]
14. Benloucif, A.; Nguyen, A.-T.; Sentouh, C.; Popieul, J.-C. Cooperative Trajectory Planning for Haptic Shared Control Between Driver and Automation in Highway Driving. *IEEE Trans. Ind. Electron.* **2019**, *66*, 9846–9857. [CrossRef]
15. Plessen, M.G.; Bernardini, D.; Esen, H.; Bemporad, A. Spatial-Based Predictive Control and Geometric Corridor Planning for Adaptive Cruise Control Coupled With Obstacle Avoidance. *IEEE Trans. Control Syst. Technol.* **2018**, *26*, 38–50. [CrossRef]
16. Laghmara, H.; Boudali, M.-T.; Laurain, T.; Ledy, J.; Orjuela, R.; Lauffenburger, J.-P.; Basset, M. Obstacle Avoidance, Path Planning and Control for Autonomous Vehicles. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019. [CrossRef]
17. Google. "Steetview", Digital Images, Google Maps. Photograph of 30 Geor. Kondili, Trikala, Greece, Taken in 2019. 2020. Available online: http://maps.google.com (accessed on 17 April 2020).
18. Zhang, X.; Liniger, A.; Sakai, A.; Borrelli, F. Autonomous Parking Using Optimization-Based Collision Avoidance. In Proceedings of the 2018 IEEE Conference on Decision and Control (CDC), Miami Beach, FL, USA, 14–19 December 2018. [CrossRef]
19. Siedentop, C.; Laukart, V.; Krastev, B.; Kasper, D.; Wedel, A.; Breuel, G.; Stachniss, C. Autonomous Parking Using Previous Paths. In *Advanced Microsystems for Automotive Applications 2015*; Lecture Notes in Mobility; Springer: Cham, Switzerland, 2015; pp. 3–14. [CrossRef]

20. Pop, D. Detection of Pedestrian Actions Based on Deep Learning Approach. *Studia Universitatis Babeș-Bolyai Informatica* **2019**, *64*, 5–13. [CrossRef]
21. Pop, D.O.; Rogozan, A.; Nashashibi, F.; Bensrhair, A. Pedestrian Recognition using Cross-Modality Learning in Convolutional Neural Networks. *IEEE Intell. Transp. Syst. Mag.* **2020**. [CrossRef]
22. Flores, C.; Merdrignac, P.; Charette, R.D.; Navas, F.; Milanes, V.; Nashashibi, F. A Cooperative Car-Following/Emergency Braking System With Prediction-Based Pedestrian Avoidance Capabilities. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 1837–1846. [CrossRef]
23. Jacobsen, S.E.T.; Gustafsson, A.; Vu, N.; Madhusudhana, S.; Hamednia, A.; Sharma, N.K.; Murgovski, N. Predictive cruise control behind a stationary or slow moving object. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019. [CrossRef]
24. Du, Y.; Liu, C.; Li, Y. Velocity Control Strategies to Improve Automated Vehicle Driving Comfort. *IEEE Intell. Transp. Syst. Mag.* **2018**, *10*, 8–18. [CrossRef]
25. Eickeler, S.; Valdenegro, M.; Werner, T.; Kieninger, M. Future Computer Vision Algorithms for Traffic Sign Recognition Systems. In *Advanced Microsystems for Automotive Applications 2015*; Lecture Notes in Mobility; Springer: Cham, Switzerland, 2015; pp. 69–77. [CrossRef]
26. Masood, K.; Zoppi, M.; Molfino, R. Mathematical Modelling for Performance Evaluation using Velocity Control for Semi-Autonomous Vehicle. In Proceedings of the 15th International Conference on Soft Computing Models in Industrial and Environmental Applications, Seville, Spain, 13–15 May 2020.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

sustainability

MDPI

Article

From Drive-By-Wire to Autonomous Vehicle: Urban Freight Vehicle Perspectives

Khayyam Masood [1,2,*], Matteo Zoppi [1], Vincent Fremont [2] and Rezia M. Molfino [1]

[1] PMAR Robotics, DIME, University of Genova, 16126 Genova, Italy; zoppi@dimec.unige.it (M.Z.); molfino@dimec.unige.it (R.M.M.)
[2] LS2N, CNRS, Ecole Centrale de Nantes, 44300 Nantes, France; vincent.fremont@ec-nantes.fr
* Correspondence: khayyam.masood@edu.unige.it; Tel.: +39-3205304054

Abstract: Freight Urban RoBOTic vehicle (FURBOT) is a complete drive-by-wire vehicle expected to perform autonomously in an urban setting. This upgrading has raised issues that need to be resolved/addressed for the vehicle to achieve a higher level of autonomy. This research addresses two of these main issues. The First is the legal framework/licensing issue necessary to be addressed for the vehicle to be insured and legally drive on public roads. The second is the changes and upgrading the vehicle must go through to become a complete autonomous freight handling vehicle. The outcome of this research led to the decision for correct categorization of the vehicle for resolving its licensing issue and its legal status on the European roads by understanding the limitations of the vehicle, which includes vehicle current state and its structural properties. An additional contribution of this research is identifying the software and hardware changes the vehicle has to go through to be fully autonomous. This includes identification of correct sensors and their placement and quantities. In addition, in-depth study for software identification for the vehicle is provided resulting in favorable choice for an off-the-shelf software. Additionally, foreseeable issues, expectations from the vehicle and requirements (considering its demonstration as an autonomous vehicle) that need to be fulfilled are also highlighted. For demonstration site, use cases and site dynamics are also studied for achieving autonomy. Fulfillment of these requirements is necessary for the vehicle to demonstrate autonomous navigation and freight handling for SHOW (SHared automation Operating models for Worldwide adoption) H2020 project for delivering freight in an urban setting.

Keywords: freight vehicle licensing; urban deliveries; autonomous vehicles; logistics; driverless; policies management

check for updates

Citation: Masood, K.; Zoppi, M.; Fremont, V.; Molfino, R.M. From Drive-By-Wire to Autonomous Vehicle: Urban Freight Vehicle Perspectives. *Sustainability* **2021**, *13*, 1169. https://doi.org/10.3390/su13031169

Academic Editor: Mehdi Nourinejad
Received: 21 December 2020
Accepted: 20 January 2021
Published: 22 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

FURBOT (Freight Urban RoBOTic vehicle) is an urban freight delivering vehicle designed for last-mile delivery in an urban environment. The vehicle was part of European Green Vehicles Initiative (EGVI) funded under the umbrella of FP7-Transport European project ending in December 2015 [1,2]. It was completed as a complete drive-by-wire vehicle and conceptually designed to be converted to an autonomous vehicle. Recently, FURBOT has taken part in SHOW (SHared automation Operating models for Worldwide adoption) project where 69 partners from all across the EU will take part in the adaptation of autonomous vehicles in 20 cities across Europe where autonomous vehicles will demonstrate their autonomy for 24 months (https://show-project.eu/). For a full demonstration, the vehicle needs to be adapted in an autonomous environment with full legislative support. For homologation of the vehicle, two separate unique hurdles are being faced. The first one is the vehicle approval under EU regulations for the vehicle to operate on public roads for delivering goods. The second is the upgrading of the vehicle to turn it from a drive-by-wire vehicle into a completely autonomous vehicle. The homologation of FURBOT is thus only possible if we resolve both issues. Thus, the need for our work is generated.

Several freight delivery options are available within urban setting for last mile delivery, which include advance delivery locations (reception boxes (parcel boxes), delivery boxes, controlled axes points, collection points and locker banks) and advanced delivery vehicles (electric delivery vehicles, electric cargo bicycles and autonomous driving vehicles) [3]. However, any single solution is not enough for smart last mile delivery, as every option has its own advantages and disadvantages [3]. However, road-based transportation using AVs can significantly reduce accidents and operational time of vehicles [4]. Parcel lockers do not seem to provide less congestion for last mile delivery if their utilization is less than 93% [5]. Among advance delivery locations, reception boxes have the greatest usefulness, flexibility, punctuality, security, privacy, reliability and lack of liability issues [6]. In our case, FURBOT provides the unique solution of integrating AVs with delivery boxes. This unique solution eliminates disadvantage of customer unavailability on reception with smart delivery locations, which is close to customer [7].

Bringing autonomous vehicles on the road is still an enormous challenge for companies. In March 2018, Tesla's vehicle had an accident killing its passenger while on autopilot [8]. Autonomous vehicles are extremely expensive and yet not completely reliable [9], thus a good start for autonomous vehicles would be to introduce them in last mile delivery option where risk is minimum [9]. For first stage deployment of autonomous vehicles, slow speed last mile freight delivery is an excellent opportunity [9]. With over 1.7 billion parcels delivered per annum [10], optimization in this business sector is a very fruitful idea [11]. In addition, inaccessible curb space, improper off-loading facilities for the vehicles and lack of elevators waste a lot of energy in last mile delivery [12]. In such scenario, delivery boxes through vehicles best suits the need for the customer and operators alike [13]. To justify such need, FURBOT is an in-line vehicle to solve the last mile delivery problem in urban setting.

Bringing autonomous vehicles on the road is not a new concept now; in fact, exactly a decade ago Google announced that they had built a car capable of driving itself [14]. Since then, autonomous vehicles have come a long way from driving themselves [15] to delivering freight [16]. Recently, JD.com (https://www.chinamoneynetwork.com/2017/06/19/jd-com-launches-robot-delivery-services-in-chinese-universities) (http://www.ecns.cn/news/sci-tech/2018-06-20/detail-ifyvmiee7350792.shtml) used robot delivery services in Beijing and universities of China [17,18] for delivery of personal packages. Similarly, Starship technologies (https://www.theverge.com/2019/8/20/20812184/starship-delivery-robot-expansion-college-campus) is also going to deploy its six-wheeled delivery robots in the universities of the US for delivery services within university campuses [19]. It is now clearly envisioned that delivery systems are specially targeted for autonomous vehicles [17–19]. However, thus far, heavy freight delivery is not completely possible in an urban setting through autonomous vehicles. FURBOT will attempt to achieve this milestone for autonomous freight delivery within an urban environment. The concept of using heavy autonomous vehicles is still under study [20], but these concepts are still in the conceptual phase and use mini autonomous vehicles for personal package delivery within urban environments. The idea is to use large vehicles (LV) to deliver and deploy small autonomous vehicles (SAV) for personalized deliveries of freight [20,21]. The novelty of FURBOT is that it is not only a freight delivering autonomous vehicle, but it is also equipped with fully functional forklifts, capable of loading and unloading freight autonomously.

Requirement formalization for autonomous vehicles is a very crucial need. Whether it is for bringing the vehicle on the roads [22] or to formalize the requirements for localization [23], addressing the requirements is a must for advancing the vehicle upgrading process. For this purpose, requirement formulation for conversion from drive-by-wire vehicle to a completely autonomous vehicle driven on roads is necessary. This need thus essentially created the necessity of our work.

There is a lack of research published for policy and regulations formation for autonomous trucks–large vehicles [24]—because of the belief that it will be a long time until

they are formally allowed on roads [25]. This natural belief creates increased confusion regarding the policies governing freight vehicles and legalizing them for the roads. Thus, the need for requirement assessments for policy and regulations for bringing the vehicle on the roads for individual cases as FURBOT arises. For autonomous vehicles to reach their full potential, the policies should be ready for the interests of manufacturers and potential plaintiffs alike [26]. The current system, however, does not provide such an environment [26].

In addition, sensor identification for autonomous vehicles is also an important topic. As the identification of sensors for autonomous vehicles is an important issue, there is research available for identification of sensors [27], their fusion within the system [28] and their use in autonomous vehicles [29]. As FURBOT is upgrading itself from a drive-by-wire vehicle to a fully autonomous vehicle within a short period, it is essential for its upgrading that identification, fusion and use of the new sensors which will enable it to navigate and handle freight autonomously be carefully selected. It is also necessary to identify an open platform as in [30] which can help in bonding the new sensor hardware with the mission computer of the vehicle.

This research is thus an attempt to address two main core issues for FURBOT for it to be upgraded from its current state to an autonomous vehicle. The first issue revolves around the correct categorization of the vehicle so it can earn legal status on the road. Second, the vehicle must also be upgraded with hardware and software for it to be a fully autonomous vehicle. For that, we should identify the required sensors and the complete platform where the hardware can be easily incorporated in the software.

Research Methodology

The two main core issues this paper addresses (mentioned above) arose for the vehicle's amalgamation in the SHOW project. The methodology developed to address these issues is discussed in this section. Two key questions are answered: (1) How were the data collected/generated? (2) How the situation to come up with the solutions analyzed?

One focus for this study is to develop a strategy to secure licensing for the vehicle to be insured properly. To resolve this issue, first, the current state of vehicle is distinguished, highlighting its feature. Second, a study of the European type approval is carried out, identifying the vehicle categories [31] approved by European commission. Comparing the vehicle design with the approved vehicle categories by European commission, best suited vehicle categories are identified. Afterwards, requirements in context with the current state of the vehicle are analyzed, keeping in view selected vehicle categories. Hurdles to categorize our vehicle in the shortlisted categories are highlighted and analyzed. Thus, we were finally able to resolve the vehicle categorization issue for licensing and insurance purposes.

The second focus of this research is to identify hardware and software changes the vehicle requires in order to upgrade itself from a drive by wire to an autonomous vehicle. The methodology opted for resolution of this issue involved a literature search in the type of sensors required by the vehicle to be upgraded to autonomous vehicle. Similar autonomous vehicles are studied to identify sensor requirements. Study participants involved vehicles with similar shape as FURBOT. Several study participants are chosen to identify software required for the upgrade of the vehicle. The inclusion criterion for software is ease of changing/development of autonomous navigation modules, whereas the exclusion criterion is the rigidity of change in software, i.e., being a black box. Further inclusion criteria for software are for it to be open sourced, accessible and have a proven track record in autonomous industry. Vision check for 3D LiDAR is performed through CAD software and the results are analyzed to identify the right sensors and their quantity.

Regarding the demo site, in-depth site study is performed collecting data from the site. Afterwards, use cases are developed considering the environmental constraints. Step-by-step freight collection for the vehicle is envisaged. The autonomy level required is analyzed and completed through project constraints. Environmental problems and advantages are

studied and highlighted. Complete demonstration is studied and analyzed, helping in clearly constraining the performance requirements.

The paper is further organized into five additional sections. In Section 2, the current state of the vehicle, its mass distribution and current features of FURBOT are discussed. In Section 3, we discuss the work progress in terms of current achievements and future goals. In Section 4, the technical classifications of how the vehicle can be correctly categorized is debated. The licensing issues are additionally highlighted. In Section 5, we discuss the foreseeable issues about the demonstration in the actual site. Finally, the vehicle upgrading requirements and expectations are highlighted in Section 6 before the conclusion.

2. State of the Vehicle

The vehicle is expected to transport two types of freights i.e., pallets and small parcels, as a feeder service from retail urban distribution service in late nights or early mornings. It is expected that the vehicle will collect freight from the loading bay, navigate to the designated store and deliver the freight. The expected workload per day for the vehicle during the full demo is around 5 h or 5–6 autonomous deliveries in full use case demo of the vehicle.

The vehicle at its current state is a complete drive-by-wire vehicle. Loading and unloading functionality for the cargo pallet is in working condition for the vehicle. The vehicle has two primary modes, i.e., fork mode and drive mode. While in drive mode, the vehicle cannot operate a forklift for loading/unloading of freight, and, while in fork mode, it cannot drive.

The vehicle is also equipped with multiple stereo cameras (side cameras, front and back) for the driver to see around the vehicle (with screens in front of the driver). A pair of single-plane LiDARs (front and back) are also mounted for emergency braking. The vehicle is also equipped with driving assistance, including emergency braking and adaptive speed control. However, in its current state, it lacks the sensors for complete autonomous driving. At the current stage of the project, it can be categorized as a fully functional drive-by-wire vehicle with loading/unloading capability of freight. The current state of the vehicle is shown in Figure 1.

Figure 1. FURBOT while loading freight.

2.1. Vehicle Components Breakdown

The overall mass breakdown considering its different components is mentioned in Table 1. The vehicle dimensions are given in Table 2. The passenger and cargo of the vehicle do not make up the weight of the vehicle for classification; thus, the mass of these components is ignored. The tires of the vehicle take up approximately 25 kg of weight.

Table 1. Vehicle components mass breakdown.

Vehicle Mass Comp. (kg)	Chassis	Forklift	Body cover	Aux. comp.	Battery	Tires	Total Mass (kg)	1300
	500	180	300	110	200	25		

Table 2. Vehicle dimensions in meters.

Vehicle Dimensions	Length	Width	Height
(meters)	4	1.5	2

Apart from these three components, other components of the vehicle are discussed and detailed below. Additionally, a full CAD drawing of FURBOT is shown in Figure 2. The vehicle is comprised of eight major components which are discussed below. These components are:

1. Chassis
2. Forklifts
3. Vehicle cover
4. Batteries
5. Installed sensors and additional hardware
6. Cargo
7. Passenger
8. Tires

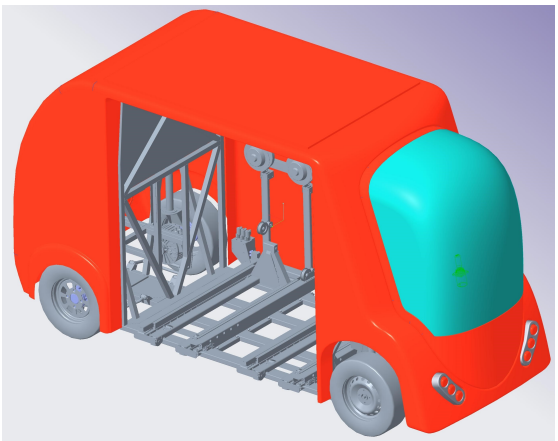


Figure 2. FURBOT CAD model.

2.1.1. Chassis

The chassis, as shown in Figure 3, is made of steel and is the heaviest part of the vehicle, weighing 500 kg. Because of this steel frame, it is very difficult to trim down the weight of the vehicle. The structural mass of the steel frame is unavoidable unless redesigning of the vehicle is carried out.

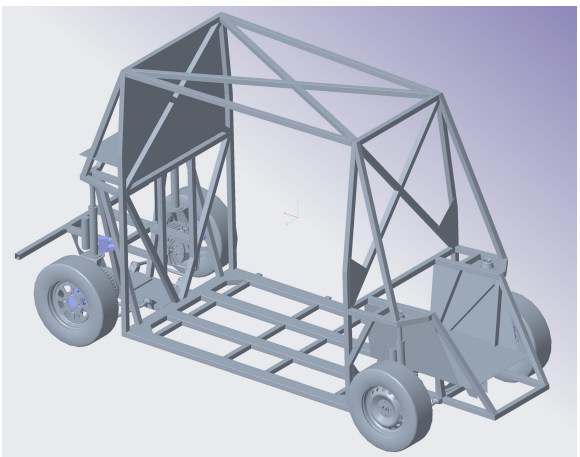


Figure 3. Chassis of FURBOT.

2.1.2. Forklifts

Two separate forklifts are part of the total mass of the vehicle. The mass of each forklift is roughly 90 kg. Thus, these two forklifts combined make up about 180 kg of FURBOT's mass. The forklift design is shown in Figure 4. It is also discussed that the mass of forklifts could be part of the payload/cargo of the vehicle and can be excluded from the total mass of the vehicle.

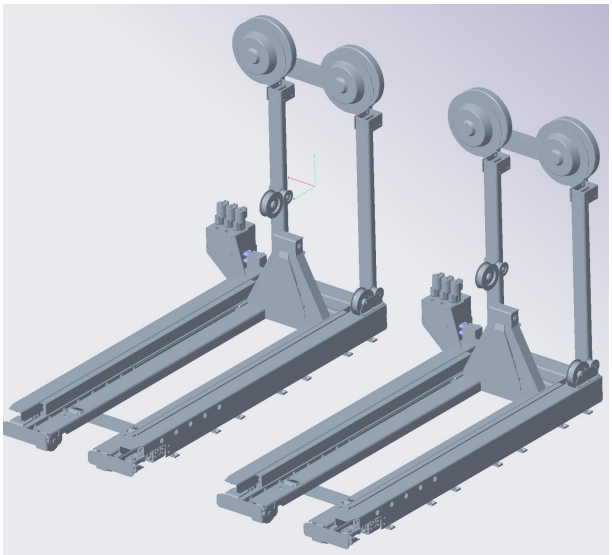


Figure 4. CAD model of forklifts.

2.1.3. Vehicle cover

FURBOT body cover comprises the orange cover which fits on the chassis of the vehicle and gives it the overall shape of the vehicle. The cover of FURBOT is roughly 300 kg and includes almost all the vehicle's outer body. The CAD model for the body cover is given in Figure 5.

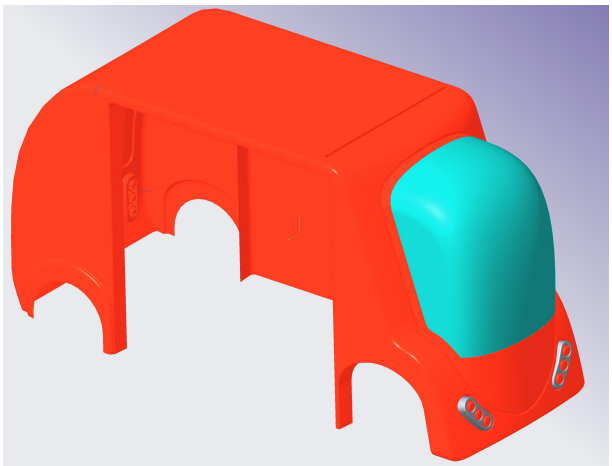


Figure 5. FURBOT cover.

2.1.4. Batteries

The mass of the batteries is not included for L7E category vehicles, but it is included for N1 category vehicles (vehicle categorization is further discussed in Section 4), thus this mass is highlighted. The batteries have custom made power management with distributed drivers. The battery power pack is shown in Figure 6. The battery management cabinet on the battery frame and the complete power unit can be easily assembled or removed from the back of the vehicle.

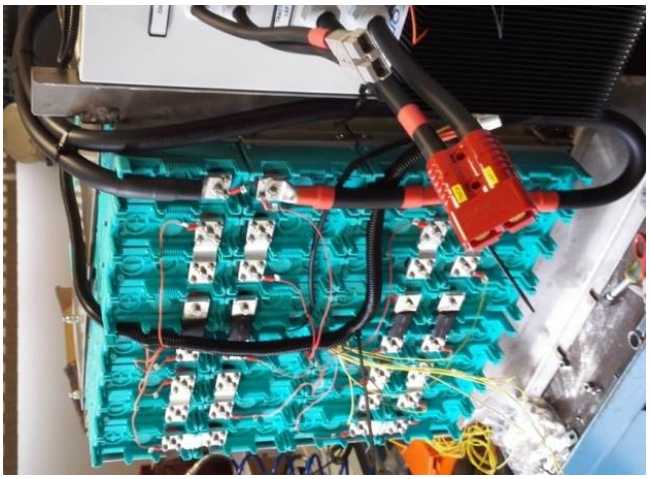


Figure 6. Electronic batteries.

2.1.5. Installed Sensors and Additional Hardware

At present, FURBOT is installed with minimal sensors. However, after the upgrading of the vehicle takes place for autonomous driving, the increase in mass is expected to be roughly 200 kg or less.

At this moment, the mass of auxiliary components and computational hardware cannot be calculated, thus no exact figure can be given for these components which are already installed in FURBOT.

The auxiliary components whose mass is not included in the overall breakdown of the mass of FURBOT are as under:

- Hydraulic services
- Suspensions
- Active braking
- Control hardware
- Seating and wiring

It is estimated that the overall mass of these components makes up about 110 kg of the vehicle's mass.

2.2. Current Features of FURBOT

The vehicle is currently a full drive-by-wire vehicle. The highlighted features of FURBOT are discussed below:

- Adaptive speed control
- Emergency braking (pure electrical): The hydraulic braking available is on front wheels only with a single line circuit. The pump is automotive standard but using only one piston line: no crossed braking circuit. Electrical braking on rear wheels is through the traction motors. The presence of the gearboxes makes maximum braking torque above the minimum requirement for the overall N1 vehicle category, but with control or electrical fail this reduces to a viscous braking torque lower than the requirement and not controllable.
- Joystick for steering, i.e., no steering wheel attached: the by-wire steering system works for a prototype, but it does not have the features of an automotive standard by-wire setup. It has no hardware redundancy and the software and method of operation are not certified for by-wire driving in road condition. Doing this certification is not workable without a major restructuring of the HW and SW. For certification, the joystick needs to be replaced with the steering wheel.
- Hydraulic-based controls of suspension and forklift: They are not involved in vehicle homologation and classification.
- Driver assistance: It is not involved in vehicle homologation and classification.

- Pedal and joystick-based braking: The pedal is the hardware on the braking pump, with the limitations exposed above concerning homologation; the stick is by-wire operating on the electric motors, again with the limitations explained above.

3. Work Progress for Automation

Continuous efforts are placed for the upgrading of FURBOT so it can perform all tasks autonomously for successful contribution and complete all UCs (Use Cases) associated with it. A brief description of how proceedings are moving forward with FURBOT per requirements is provided below.

3.1. Current Achievements

Considerable developments were done in the first six months of the upgrading project for FURBOT. The tasks carried out are detailed below.

- In the technical domain, mathematical modeling for FURBOT is carried out devising obstacle avoidance techniques/strategies [32,33]. Further explanation of this work is given in Section 3.1. The multi-body model is also created for performance evaluation to analyze extreme cases [34].
- Identification of shortcomings in terms of sensors required and upgrading needed is carried out.
- Identification of issues concerning freight approach and autonomous drive within an urban environment are highlighted. Strategies on how to tackle the respective issues are formalized. Furthermore, the hardware of forklifts is also upgraded [35].
- Development of a plan for upgrading is carried out where hardware and software requirements for the vehicle are completed.

Mathematical Modeling and Obstacle Avoidance

Mathematical model of the vehicle was developed as part of development of its digital twin for critical analysis [32]. The longitudinal equation of motion used and developed is given in Equation (1).

$$\dot{v_x} = \frac{\sum F_x}{M_v} = \frac{F_t + F_d + F_g + F_r}{M_v} \tag{1}$$

Newton–Euler equations of motions used for vehicle lateral model are further given in Equations (2) and (3). The development, expansion and annotations of these equations are presented in [32].

$$\dot{v_y} = \frac{1}{M_v v_x}(-a_1 C_{af} + a_2 C_{ar})r - \frac{1}{M_v v_x}(C_{af} + C_{ar})v_y + \frac{1}{M_v} C_{af}\delta - r v_x \tag{2}$$

$$\dot{r} = \frac{1}{I_z v_x}(-a_1^2 C_{af} - a_2^2 C_{ar})r - \frac{1}{I_z v_x}(a_1 C_{af} - a_2 C_{ar})v_y + \frac{1}{I_z} a_1 C_{af}\delta \tag{3}$$

To control the vehicle, three different PID controllers are developed for traction power control, braking power control and steering control [33]. The values of the respective controllers are given in Table 3. Using the mathematical model, obstacle avoidance techniques are developed using mainly velocity control [33]. Using velocity control, the velocity of the vehicle is adjusted whenever interacting with obstacles. A reference of how velocity of the vehicle is adjusted when facing an obstacle is shown in Figure 7. The complete workflow can be found in [33].

Table 3. Velocity and steering controllers.

Controller Type	Controller Values			
	Error Amplification	*Proportional Gain*	*Integral Gain*	*Derivative gain*
Traction power control	500	20	0	1
Braking power control	1	550	6	0
Steering control	1	10	2	0

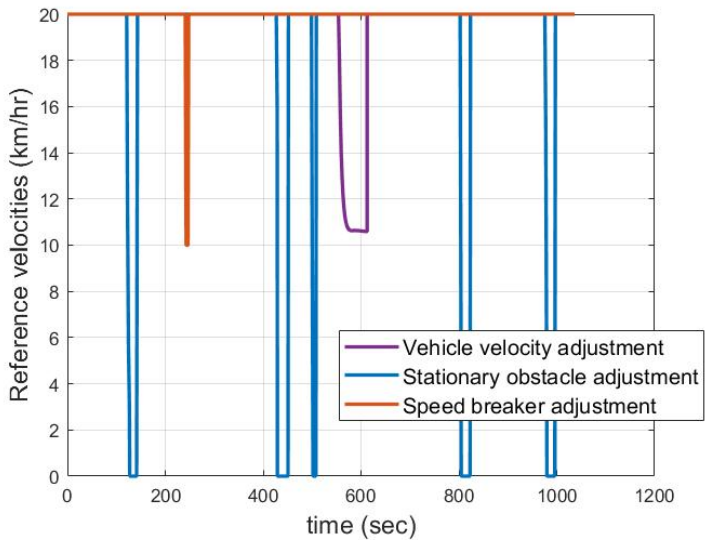


Figure 7. Vehicle velocity adjustment.

3.2. Future Goals

The future/upcoming goals for FURBOT are list below.

- Initialize the upgrading process, including software upgrading of the vehicle
- Incorporate relevant sensors for autonomous driving and autonomous handling of freight, including relevant sensors for identification of freight
- Strategies for freight approach and cargo loading are being visualized and simulations are to be developed for the collection of freight, navigation to the drop-off location and drop-off of freight

The way forward towards these goals are further discussed in Section 6.

4. Technical Classification for Vehicle

Currently, there are no existing legal basis for the autonomous freight vehicle operation in a real traffic environment in Europe. The best way to legalize the freight vehicle is to categorize it into one of the already approved categories of EU vehicle definitions. Among the many approved vehicle categories available, FURBOT can be considered in two categories according to its size, weight and type. These two categories of EU vehicle definitions and the issues governing in opting for them are further discussed below

1. L7E-CU: L7E vehicle is a vehicle with four wheels, whose unladen mass is not over 400 kg (550 kg for vehicles intended for carrying goods), not including the mass of batteries with electric vehicles and whose maximum continuous rated power does not exceed 15 kW. Category L7E-CU is a sub-category that is a heavy Quadri-mobile for utility (utility vehicle only designed for the carriage of goods).

2. N1-1: N1 vehicle is designed for the carriage of goods and having a maximum mass not exceeding 3.5 t. N1 vehicles are further classified into three sub-classes, which are discussed in their respective subsections.

4.1. L7E-CU

L7E-CU is defined as a Heavy Quadri-mobile for utility purposes. Further details for this classification of the vehicle are taken from [36] and are given in Table 4.

Table 4. L7E-CU Classification.

Category	Category Name	Common Classification Criteria
L7E-CU	Heavy Quadri-mobile for utility purposes	only designed for the carriage of goods with an open or enclosed loading bed, virtually even and horizontal loading bed that meets the following criteria: 1. Length loading bed x width loading bed $\geq$ 0.3 x Length vehicle x Width vehicle or 2. An equivalent loading bed area as defined above designed to install machines and/or equipment and 3. Designed with a loading bed area which is separated by a rigid partition from the area reserved for the vehicle occupants 4. The loading bed area shall be able to carry a minimum volume represented by a 600 mm cube Maximum two non-straddle seats, including the seating position for the driver.

Some other technical aspects of L7E-CU as taken from [37] are given in Table 5.

Table 5. Technical aspects of L7E-CU.

Weight Limit	Max. Velocity Limit	Max. Engine Power	Dimensions L $\times$ W $\times$ H
$\leq$600 kg capable of maximum 1000 kg load. This mass limit excludes the weight of the batteries	$\leq$90 km h $^{-1}$	$\leq$15 kW	$\leq$3700 mm 1500 mm 2500 mm

According to the classification of L7E-CU, FURBOT falls well within this category except for its overall weight, i.e., 1300 kg excluding batteries, cargo and passenger, which is 700 kg more than the designed limit of 600 kg. The excess weight reduces to 500 kg if considering the forks as part of the cargo; further reduction could be achieved with the redesign of structural parts of the vehicle, but it would be better to rebuild the entire vehicle if weight limitation criteria have to be met. Apart from weight issues, the vehicle falls well within the limits of the L7E-CU category.

Another consideration in terms of safety is that, as FURBOT is a prototype vehicle, it cannot take part in impact testing, which is usually necessary for vehicles for granting minimum road safety. However, L7E category vehicles were exempted from impact test criteria [37], thus FURBOT might qualify for safety in impact testing.

4.2. N1-1

As mentioned above, N1 category vehicles are those which are designed for carrying goods and have a maximum mass of 3.5 t. There are three sub-classes of N1 type vehicle which are given in Table 6 and are taken from [38]. The classes are based on reference mass, which is defined as the mass of the vehicle in running order, less the mass of the driver.

Table 6. N1 sub-classes definitions.

Class	Reference Mass, RM	
	Euro 1–2	**Euro 3**
I	RM $\leq$ 1250 kg	RM $\leq$ 1350 kg
II	1250 kg $<$ RM $\leq$ 1700 kg	1350 kg $<$ RM $\leq$ 1760 kg
III	1700 kg $<$ RM	1760 kg $<$ RM

Further details on N1 type vehicles can be found in [39]. Apart from impact testing requirements, which cannot be managed for FURBOT as it is a prototype vehicle, there are no other requirements that FURBOT does not fall in.

4.3. Possible Licensing Issues

The major hurdle which is foreseen for licensing for the vehicle in the category of L7E-CU is:

- Overweight issue of the vehicle

For licensing the vehicle under the N1 category, hurdles are:

- Absence of steering wheel
- No impact testing possible as it is a prototype vehicle

5. Foreseeable Issues

As FURBOT is expected to operate at late night or early morning, some concerns regarding the demo area are highlighted.

1. Illegally parked cars: Vehicles that will be parked illegally in the demo area might become an obstacle for FURBOT, which can require additional help during the drive.
2. Expected U-turn: The vehicle should have enough room to make a U-turn maneuver efficiently for a return route at a designated area. The demo area should thus be equipped with such a spot. It is preferable if the vehicle is not required to cross busy roads for U-turn rather stays within the dedicated lane.
3. Freight placement for designated shops: While placing the freight at the designated stores/clients, vehicle needs to be sure that the area where the freight is being placed is clear from any obstacles or human interference. As the freight needs to be laterally extruded from the vehicle, it must not hit any obstacle. A supervised drop-off of the freight might solve this issue, otherwise appropriate sensors need to be placed which can take an automated decision on freight placement.
4. Empty pallet collection: A strategy for the collection of empty pallets by the vehicle also needs to be planned. Empty freight collection has almost the same concerns as the freight placement as the forklift has to be laterally displaced from the vehicle to collect the empty pallets from the collection points.

6. Upgrading of Vehicle (Requirements and Expectations)

Work on the upgrading of FURBOT has already started in terms of automation requirements. For this purpose, necessary sensors are being identified and shortlisted for purchase. The guideline for the required sensors is taken from the Apollo Auto hardware installation guide [40] and further modified keeping in view the vehicle. The initial identification of sensors has placed the requirements of the following sensors for the vehicle:

1. 3D LiDAR at the top (for 360° view of the surrounding)
2. Three Leopard Cameras with respective lens and trigger cables [41] (for traffic light detection and freight handling)
3. Two single plane LiDARs (front and rear)
4. To RADARs (front and rear)

5. Industrial computer (for processing of sensor data)
6. LTE/5G modem (for data communication between vehicle and remote user)
7. GNSS for localization
8. IMU for orientation sensing
9. Medium-range RFID readers (for freight recognition)

It is expected to upgrade the vehicle software and hardware architecture for its complete autonomy.

6.1. Resolution of 3D LiDAR Purchase/Placement/Quantity Issues

3D LiDAR is one of the most expensive sensors that is being purchased for the vehicle. As the vehicle is unconventional compared to typical Apollo Auto vehicle [40], typical Apollo Auto conventions cannot be followed. For considerations, three 3D LiDARs have been shortlisted keeping in mind the cost, availability and compatibility with Apollo software [42–44]. The specifications which are affecting the decision of the LiDAR are mentioned in Table 7.

Table 7. Comparison of 3D LiDARs.

Sr. No	Company	Product	FOV (Vertical)	FOV (Horizontal)	Frame/Rotation Rate (Hz)	Range (m)	Price
I	Velodyne	PUCK	30° (+15° to −15°)	360°	5–20	<100	$$
II	Velodyne	Ultra PUCK	40° (+15° to −25°)	360°	5–20	<200	$$$
III	Hesai	PANDARXT	31° (+15° to −16°)	360°	5–10–20	<120	$

As shown in the figures in Section 2 and Table 2, FURBOT has a box shape figure and thus looking down from the mid-top of the vehicle through 3D LiDAR is very difficult. In that respect, the most crucial aspect for consideration is the negative values of the vertical field of view for each LiDAR. Since Velodyne PUCK is not better than HESAI PANDARXT in price or vertical FOV, it is not further considered. The primary advantage of choosing Ultra PUCK over PANDARXT is the better vertical FOV values. While placing Hesai PANDARXT on top middle of the vehicle and trying to yield the maximum field of view, the LiDAR has to be placed almost 0.5 m above the vehicle, as shown in Figure 8. This shows that, although Ultra PUCK is better than PANDARXT, we cannot use the additional vertical field of view as 0.5 m above the vehicle is already out of bounds for placing 3D LiDAR stably.

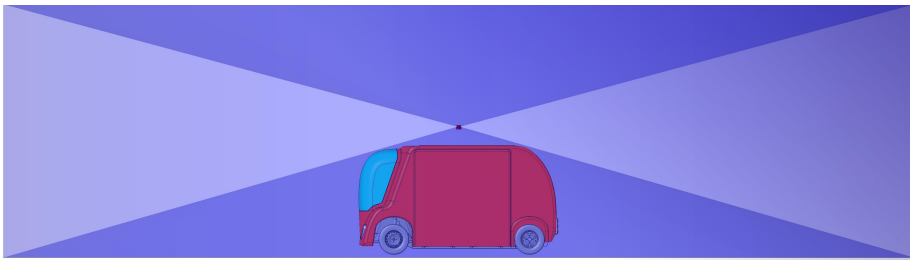


Figure 8. PANDARXT LiDAR placed on top middle with 0.5 m elevation from vehicle top-base.

The above insight into placement and type of LiDAR puts Hesai PANDARXT as an optimum option under given constraints. However, even choosing PANDARXT does not solve the problem of 7 m undetectable shadow in front of and behind the vehicle (Figure 8). To solve this issue, inspiration is taken from a similar structure autonomous vehicle Olli [45]. Because of the type of structure Olli and FURBOT has, it is better to place two LiDARs, as shown in Figure 9.

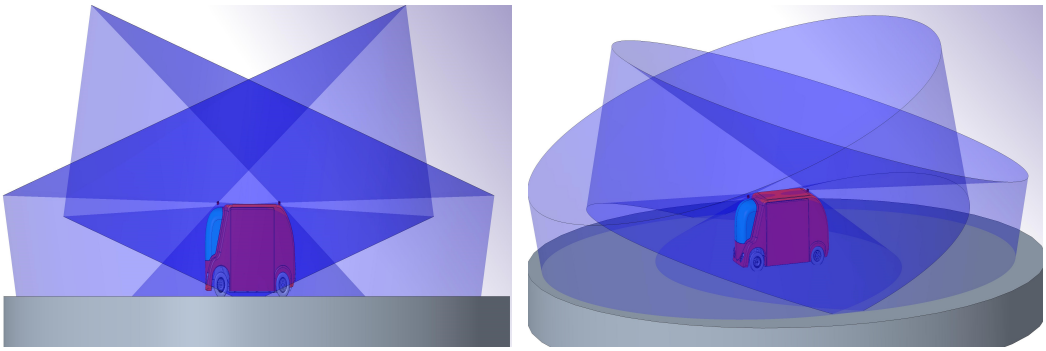


Figure (a) Figure (b)

Figure 9. (**a**) Two LiDARs placed front/back (side view); and (**b**) two LiDARs placed front/back (isometric view).

Having the above configuration for FURBOT gives sturdier placement for LiDARs and decreases the blind spots for LiDAR detection. This configuration does however come with a drawback that it increases the cost of installation as the need is increased from one 3D LiDAR to two. However, selection of economical LiDAR gives us advantage to go ahead with this configuration. The remaining blind spots are further removed by placing single plane LiDARs and RADARs at the front and back of the vehicle (same configuration as Olli [45]).

6.2. Requirements

The vehicle is expected to navigate in an urban environment at low speed, i.e., 15 km h $^{-1}$, in a dedicated lane for the autonomous vehicles. Although the lane for autonomous vehicles is dedicated, it is also a lane for cyclists, and people are expected to cross the lane. Figure 10 shows the expected lane in the demonstration area where the vehicle is expected to navigate with autonomy, addressing few hurdles within its path. Figure 11 presents the clear setting of the demonstration area and the environment.

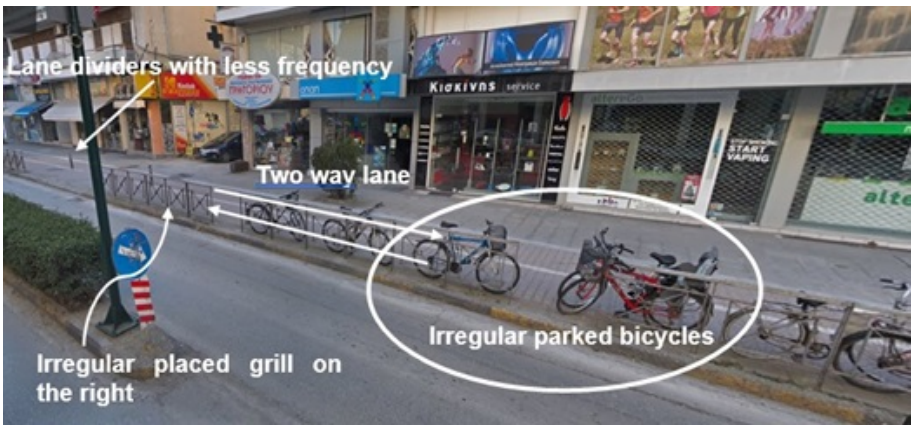


Figure 10. Street view of demonstration area with few identifiable hurdles (source street view, Google Maps [46]).

Currently, the vehicle operates with Level 3 automation. However; it is expected that the vehicle needs to be upgraded to Level 4 automation (according to SAE J3016 standards [47]) during pre-demo and full demo. The vehicle is expected to demonstrate TRL (technology readiness level) 6 for pre-demo and 7 for a full demonstration. Table 8 further summarizes the automation/performance level required for real life demonstration.

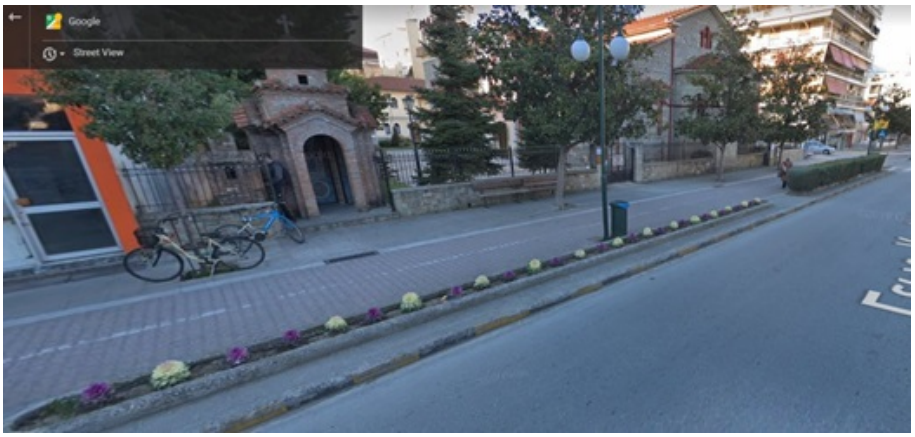


Figure 11. Street view of clear setting of demonstration area in Trikala/Greece (source street view, Google Maps [46]).

Table 8. Vehicle infrastructure for demonstration.

Demonstration Location	Operational Speed	Connectivity	Automation Level	TRL (pre-demo-full demo)
Trikala/Greece	15 km h^{-1}	5G/4G OR Fiber optic network	4 (within the project, from current 3)	6 → 7

In terms of connectivity, two service areas are proposed. The first service area will be covered by 5G/4G, whereas the second service area will be covered by a proprietary fiber-optic network. The infrastructure availability for FURBOT is still uncertain and will depend on the service area where autonomy demonstration is required.The proprietary fiber-optic network availability and its use is only considered if a major client is in an area which is not covered by 5G/4G network.

6.2.1. Use Cases for Demonstration Site

The vehicle is expected to cover five use cases. For successfully demonstrating the autonomy of the vehicle, these use cases have to be performed. The use cases are further enumerated below:

1. Autonomous cargo vehicle operation in real urban pedestrian city-center environment
2. Autonomous cargo vehicle operation and parking in real urban pedestrian city-center environment
3. Autonomous cargo vehicle operation, smooth braking and immobilization in real urban pedestrian city-center environment
4. This is combined with Use Case 2, as part of the routes will be performed in mixed lanes with other vehicles
5. Autonomous cargo vehicle remote monitoring and emergency braking for immobilization mechanism via the connection with the remote control center

The flow of Use Case 3 combined with Use Case 5 is enumerated below to understand the demonstration requirement better:

1. The FURBOT vehicle load is packaged in freights boxes with the help of the operator.
2. The safety driver on board monitors the vehicle's route.
3. The FURBOT follows its pre-defined route and stops at the fixed location in order to unload part of its cargo.
4. The vehicle parks safely autonomously.
5. The local business stakeholder picks up the load via the robotized freight boxes.
6. The vehicle continues its route, but a pedestrian is crossing the road.
7. The vehicle detects the pedestrian, adjusts its speed and stops smoothly.

8. The safety person on board also activates the emergency brake.
9. After the pedestrian moves and the road is unblocked, the vehicle continues its route towards every delivery location until all the goods are delivered.
10. The vehicle parks at the depot area.

6.2.2. Software Requirements

Many software architectures are available for autonomous vehicles to interface the hardware with the software. Some software architectures available for engineers are given in [48]. Under the current software architecture developments, autonomous vehicle software architectures are under constant scrutiny from a security perspective [49] and few security concerns are repeatedly highlighted [50,51]. Choosing the right software architecture can be an arduous task for an autonomous vehicle designer. The considerations for our team before selecting the software architecture for FURBOT were that the software should be:

- Open sourced
- Easily accessible
- Modifiable
- Have a proven track record in the industry

Considering the above requirements, Apollo Auto fulfills our requirements, and it is considered as the "Android for autonomous vehicles" [52]. It has intelligent components such as perception, simulation, planning and intelligent control [52]. These features help in developing the interface with vehicle's hardware and thus achieving autonomous navigation. Since FURBOT is a freight vehicle and the autonomous freight handling part has to be developed, it is imperative that the selection of our software architecture is programmable and open sourced so that changes in the software could be easily implementable.

Considering factors highlighted above, it is thus decided to interface the entire vehicle through Apollo Auto for autonomous and remote controlling for the vehicle. Moreover, the vehicle needs to have a remote control for the vehicle besides on-board autonomous control.

The latest launch of Apollo is Apollo 5.0. Apollo 5.0 ushers in a brand-new data pipeline service with per-vehicle calibration options, along with spruced-up prediction evaluators and map data verification tools [53]. The highlights are Open Space Planner and improved support for parking and intersections, including stop signs and traffic lights and those without signage. Finally, Dreamland, Apollo's web-based simulation platform for model validation and testing, has been updated with a more robust scenario editor and control-in-the-loop simulation [53].

Already many auto companies are on board with Apollo Auto to integrate it with their autonomous vehicles. Toyota Motor already signed on to the Apollo self-driving platform led by Baidu. Toyota will be the second Japanese automaker to take part, following Honda Motor [54].

Not content to let Toyota be the only tech giant to get their hands on Apollo, Geely—the company that owns both Volvo and Lotus—will also work with Baidu, but it will be less focused on self-driving hardware and more focused on artificial intelligence [55]. Presently, Apollo has dozens of partner companies, including plenty of Chinese domestic companies as well as global companies such as Honda, BMW, Daimler, Volkswagen, Intel and Nvidia. The project was started in 2017 and is already in its fifth iteration with plans to reach full autonomy by 2021 [55].

Interfacing of the entire vehicle simultaneously with Apollo Auto is also essential in the development of the autonomous vehicle. All sensors should be able to interface with Apollo Auto for seamless automation of the vehicle. Integration and interfacing with Apollo Auto are also essential for off-line testing of the vehicle.

6.2.3. Hardware Identification and Installation

Hardware installation is required for the overall upgrading of FURBOT for autonomous navigation and freight handling in an urban environment. Among the major

hardware changes which are requested are the integration of new sensors, connecting and interfacing the installed sensors with MC and establishing the connection with Apollo Auto.

Apart from new sensor requirements, their integration with MC is also essential. The cable/connectors requirements also need to be identified. In short, the transmission of sensor output data to MC whether wired or wireless needs to be identified.

6.3. Operative Modes of the Vehicle

There are mainly two operative modes for the vehicle: "fork mode" for freight handling and normal mode for vehicle navigation. When "fork mode" is activated, the vehicle cannot drive. Thus, the exact identification of freight must be performed before fork mode is started. The two operative modes, i.e., autonomous urban navigation and autonomous freight handling, are further discussed in their respective sections below.

1. Autonomous urban navigation: Within this mode, the vehicle is supposed to navigate within the urban environment with the help of newly installed sensors in all weather and light conditions, i.e., in daylight and nighttime. It is also supposed to identify obstacles within its path and avoid them during its urban navigation.
2. Autonomous freight handling: For autonomous freight handling, the vehicle is expected to autonomously load the pallet (freight) at the loading bay and then deliver it to the expected shop in an urban setting through autonomous urban navigation. Afterward, it should deliver the pallet, collect the empty pallet and then return to the loading bay. For the vehicle to perform this task seamlessly, the vehicle is expected to:
   - Approach the pallet using GNSS, RFID and LiDARs autonomously in the loading bay
   - Collect the pallet by parking right next to the pallet and initiating fork mode for collection
   - Deliver the freight at designated undocking location, mainly with the GNSS location of the designated shop
   - Collect the empty pallet at the designated shop using GNSS, RFID and LiDARs same way as in the loading bay
   - Unload the empty pallet at the designated location in the loading bay

   The process is further explained in Figure 12.

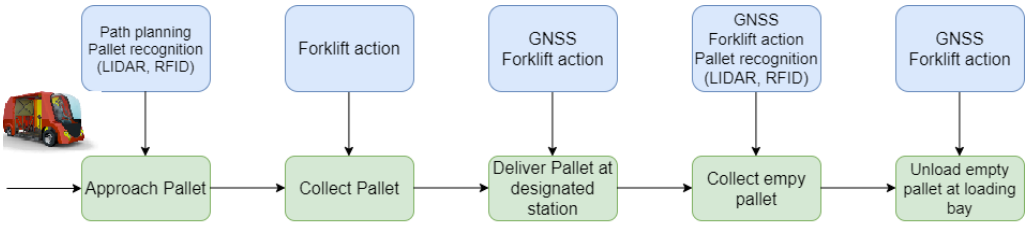


Figure 12. Autonomous freight handling flow.

6.4. Expectations

The automation of FURBOT needs to fulfill additional expectations than just the automation requirements set forth. Among the expectations, the expectations at pre-demo and full demo are important besides the performance expectations and the timelines associated with them.

6.4.1. Demonstration expectations

The demonstration requirements are summarized in Table 8. The next is for full functionality for full demo for six months to two years, servicing the transport of goods within an urban environment continuously within the next four years. The expectations for the pre-demo are listed below:

- Automation Level 4 required with TRL 6 at least

- Full connectivity required for seamless integration in an urban environment
- Requirement of a passive driver who can remotely or physically takeover the vehicle in the case of requirements and needs

For full demonstration of the vehicle, the expectations are:

- Automation Level 4 with TRL 7 at least
- Sustainable connectivity for a longer duration of operations
- Expected six daily runs with expected 1920 annual freight delivery runs
- Sustainable automation plan for years to come
- Requirement of a passive driver (off-site) who can remotely take over the vehicle

6.4.2. Performance Expectations

The performance expectations for the vehicle during the upgrading are:

- Hurdle detection and avoidance during autonomous urban navigation
- Fully autonomous driving with embedded fail-safe strategies
- Freight detection and alignment with freight for collection of freight and delivery
- Re-routing for return journey, including taking U-turn

The journey for FURBOT is expected in the following way:

Freight loading at loading bay → Delivery of freight at designated shop → Collection of empty pallets from shops → Return journey planning and Return to the loading bay.

6.5. Recommendations

Future goals for amalgamation of the vehicle are mentioned in Section 3.2. However, few recommendations are highlighted on licensing and hardware/software upgrading for further research.

- As the driver-less era is approaching, it is crucial that proper regulations (in this case, EU regulations) should be developed for the driver-less vehicles to operate on road for them to qualify for medical/comprehensive insurance from insurance providers.
- Targeted designing of the vehicle in a pre-existing vehicle category can lead to much ease when bringing the autonomous vehicle on the road. Addressing the legal requirements for autonomous vehicles is much easier if the vehicle is pre-tailored, keeping in mind a specific category of the vehicle.
- Hardware and software identification for the vehicle autonomy is crucial and thus the highlight of our research. However, there is no single solution or a specific path to take in order to make the vehicle completely autonomous. There is no specific guideline available to resolve this issue, thus the need for our work. However, further contributions such as these can help set a standard procedure for vehicle autonomy upgrading.
- Hardware/software upgrading for autonomy can be very expensive, especially due to the need of expensive sensors and expensive off-the-shelf software. Thus, sensor identification and budgeting should be a prime consideration during autonomy upgrading.

7. Conclusions

We highlight and address several issues for FURBOT to complete its transition from a drive-by-wire electric vehicle to a fully autonomous one. The research focus is to address the issues concerning licensing and autonomy of the vehicle.

For licensing, the weight difference is enormous for the vehicle to qualify for the category of L7E-CU. The vehicle is overweight for this category, and shedding some weight will not solve the issue. However, qualifying the vehicle for the N1 category seems very promising. The only hurdle which needs to be addressed is the incorporation of a steering wheel instead of joystick, which is viable. Second, for crash testing, as the vehicle is a

prototype and very expensive, it is not possible to do crash testing. However, to resolve this issue, it is possible to do simulated crash testing in structural analysis software. Structural analysis in software for crash testing is now becoming increasingly common [56]. Thus, after considering all the factors, it is decided for FURBOT to gain the license in compliance with the N1 category by installing a steering wheel and doing structural analysis for crash testing.

Further conclusive results of our study showed the relevant hardware changes FURBOT must go through to change from a drive-by-wire vehicle to an autonomous one. For that, incorporating new sensors is of high importance. The identification of the right 3D LiDAR with the presence of single plane LiDARs and Radars will be sufficient for the vehicle to perform autonomous navigation. The identification and finalization of software architecture are also of prime importance, and Apollo architecture serves as an ideal platform.

We identify and highlight foreseeable issues that will help us in our path forward. Additionally, we critically analyze how we need to show autonomy, highlighting our strategy and constraints.

Few issues which are envisioned during the implementation phase of the whole hardware/software upgrade are that we stripped the top 3D LiDAR from 128 L (as recommended by Apollo [40]) to 32 L. This means lower resolution with fewer data points for the perception module to detect shapes on the road. However, this was countered with front and back single plane LiDAR/radar for better perception. According to the manual of Apollo Auto, the backward motion is not developed for autonomous vehicles, which is a very crucial problem for parking for freight vehicle in the constrained urban environment. To resolve this issue, a new module has to be designed and incorporated into the system, which is cumbersome to do for off-the-shelf software. Resolution of these additional issues will have a direct impact on the budgeting of the project, as the manpower requirement for the project will considerably increase.

Changing the environment for the vehicle operations (i.e., city/country) can increase the number of requirements for the vehicle to be operational. Currently, the vehicle is envisaged to operate in a dedicated lane where authorization of other vehicles to operate is restricted (only autonomous vehicles are allowed). Thus, the vehicle is only expected to encounter person or cycles. Putting the vehicle in a lane which also occupies other (manual) vehicles will require additional work on modules built for autonomous navigation. Changing countries for vehicle operations can also bring in additional legislative requirements that vary from country to country. This can lead to delays in vehicle freight delivery application.

Author Contributions: Conceptualization, K.M. V.F. and M.Z.; methodology, K.M. and V.F.; validation, M.Z. and V.F.; formal analysis, K.M and M.Z.; investigation, K.M. and V.F.; resources, M.Z. and V.F.; data curation, K.M.; writing—original draft preparation, K.M. and V.F.; writing—review and editing, K.M. and V.F.; visualization, K.M. and V.F.; supervision, R.M., V.F. and M.Z.; project administration, R.M., V.F. and M.Z.; and funding acquisition, R.M., V.F. and M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: This paper was supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 875530, project SHOW (SHared automation Operating models for Worldwide adoption).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

FURBOT	Freight Urban Robotic Vehicle
SHOW	SHared automation Operating models for Worldwide adoption

References

1. Freight Urban RoBOTic Vehicle. 8 August 2016. Available online: https://cordis.europa.eu/project/id/285055 (accessed on 25 November 2020).
2. FURBOT. 9 January 2018. Available online: https://egvi.eu/research-project/furbot/ (accessed on 25 November 2020).
3. Behnke M. Recent Trends in Last Mile Delivery: Impacts of Fast Fulfillment, Parcel Lockers, Electric or Autonomous Vehicles, and More. In *Logistics Management*; Lecture Notes in Logistics; Bierwirth, C., Kirschstein, T., Sackmann, D., Eds.; Springer: Cham, Switzerland, 2019. [CrossRef]
4. Csiszár, C.; Földes, D. System Model for Autonomous Road Freight Transportation. *Promet-Traffic Transp.* **2018**, *30*, 93–103. [CrossRef]
5. Yetis, H.; Karakose, M. A New Smart Cargo Cabinet Application for Unmanned Delivery in Smart Cities. In Proceedings of the 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), Malatya, Turkey, 28–30 September 2018; pp. 1–5. [CrossRef]
6. Felch, V.; Karl, D.; Asdecker, B.; Niedermaier, A.; Sucky E. Reconfiguration of the Last Mile: Consumer Acceptance of Alternative Delivery Concepts. In *Logistics Management*; Lecture Notes in Logistics; Bierwirth, C., Kirschstein, T., Sackmann, D., Eds.; Springer: Cham, Switzerland, 2019. [CrossRef]
7. Cepolina, E.M.; Farina, A. Optimization of the FURBOT urban freight transport scheme. *Transp. Res. Procedia* **2015**, *5*, 119–131. [CrossRef]
8. Tesla Car That Crashed and Killed Driver Was Running on Autopilot, Firm Says. 31 March 2018. Available online: https://www.theguardian.com/technology/2018/mar/31/tesla-car-crash-autopilot-mountain-view (accessed on 8 January 2021).
9. Xia, H.; Yang, H. Is last-mile delivery a 'killer app' for self-driving vehicles? *Commun. ACM* **2018**, *61*, 70–75. [CrossRef]
10. Postal and Logistics Consulting Worldwide. *Review of the Impact of Competition in the Postal Market on Consumers*; Final Report to Citizens Advice; Postal and Logistics Consulting Worldwide: England, UK, 2015.
11. Allen, J.; Bektas, T.; Cherrett, T.; Bates, O.; Friday, A.; McLeod, F.; Piecyk, M.; Piotrowska, M.; Nguyen, T.; Wise, S. The Scope for Pavement Porters: Addressing the Challenges of Last-Mile Parcel Delivery in London. *Transp. Res. Rec.* **2018**, *2672*, 184–193. [CrossRef]
12. Morris, A.G. *The Last Mile: Developing Efficient Freight Operations for Manhattan's Buildings*; Stephen L. Newman Real Estate Institute, Baruch College: New York, NY, USA, 2009.
13. Conway, A.; Fatisson, P.E.; Eickemeyer, P.; Cheng, J.; Peters, D. Urban micro-consolidation and last mile goods delivery by freight-tricycle in Manhattan: Opportunities and challenges. In Proceedings of the Transportation Research Board 91st Annual Meeting, Washington, DC, USA, 22–26 January 2012.
14. Thrun, S. What We're Driving at, Google Official Blog. 9 October 2010. Available online: https://googleblog.blogspot.com/2010/10/what-were-driving-at.html (accessed on 28 September 2020).
15. Kelly, H. Self-Driving Cars Now Legal in California, CNN. 30 October 2012, 12:30 P.M. Available online: https://edition.cnn.com/2012/09/25/tech/innovation/self-driving-car-california/index.html (accessed on 28 September 2020).
16. Tang, C.S.; Veelenturf, L.P. The strategic role of logistics in the industry 4.0 era. *Transp. Res. Part E Logist. Transp. Rev.* **2019**, *129*, 1–11. [CrossRef]
17. Gu, L. Delivery Robots Hit the Road in Beijing. 2018. Available online: http://www.ecns.cn/news/sci-tech/2018-06-20/detail-ifyvmiee7350792.shtml (accessed on 28 Sepetember 2020).
18. Li, D. Jd.com Launches Robot Delivery Services in Chinese Universities. 2017. Available online: https://www.chinamoneynetwork.com/2017/06/19/jd-com-launches-robot-delivery-services-in-chinese-universities (accessed on 28 Sepetember 2020).
19. Andrew, J.H. Thousands of Autonomous Delivery Robots Are about to Descend on Us College Campuses. 2019. Available online: https://www.theverge.com/2019/8/20/20812184/starship-delivery-robot-expansion-college-campus (accessed on 28 Sepetember 2020).
20. Boysen, N.; Schwerdfeger, S.; Weidinger, F. Scheduling last-mile deliveries with truck-based autonomous robots. *Eur. J. Oper. Res.* **2018**, *271*, 1085–1099. [CrossRef]
21. Yu, S.; Puchinger, J.; Sun, S. Two-echelon urban deliveries using autonomous vehicles. *Transp. Res. Part E Logist. Transp. Rev.* **2020**, *141*, 102018. [CrossRef]
22. Broggi, A.; Bombini, L.; Cattani, S.; Cerri, P.; Fedriga, R.I. Sensing requirements for a 13,000 km intercontinental autonomous drive. In Proceedings of the 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA, 21–24 June 2010; pp. 500–505. [CrossRef]
23. Reid, T.G.; Houts, S.E.; Cammarata, R.; Mills, G.; Agarwal, S.; Vora, A.; Pandey, G. Localization requirements for autonomous vehicles. *arXiv* **2019**, arXiv:1906.01061.
24. Monios, J.; Bergqvist, R. The transport geography of electric and autonomous vehicles in road freight networks. *J. Transp. Geogr.* **2019**, *80*, 102500. [CrossRef]

25. Anderson, J.M.; Kalra, N.; Stanley, K.D.; Sorensen, P.; Samaras, C.; Oluwatola, O.A. *Autonomous Vehicle Technology: A Guide for Policymakers. Rand Transportation, Space, and Technology Program*, 2nd ed; Santa Monica, CA, USA, 2014.
26. Funkhouser, K. Paving the road ahead: autonomous vehicles, products liability, and the need for a new approach. *Utah Law Rev.* **2013**, *1*, 437–462.
27. Campbell, S.; O'Mahony, N.; Krpalcova, L.; Riordan, D.; Walsh, J.; Murphy, A.; Ryan, C. Sensor Technology in Autonomous Vehicles: A review. In Proceedings of the 29th Irish Signals and Systems Conference (ISSC), Belfast, UK, 21–22 June 2018; pp. 1–4. [CrossRef]
28. Kocić, J.; Jovičić, N.; Drndarević, V. Sensors and Sensor Fusion in Autonomous Vehicles. In Proceedings of the 26th Telecommunications Forum (TELFOR), Belgrade, Serbia, 20–21 November 2018; pp. 420–425. [CrossRef]
29. Taraba, M.; Adamec, J.; Danko, M.; Drgona, P. Utilization of modern sensors in autonomous vehicles. In Proceedings of the 2018 ELEKTRO, Mikulov, Czech Republic, 21–23 May 2018; pp. 1–5. [CrossRef]
30. Kato, S.; Takeuchi, E.; Ishiguro, Y.; Ninomiya, Y.; Takeda, K.; Hamada, T. An Open Approach to Autonomous Vehicles. *IEEE Micro* **2015**, *35*, 60–68. [CrossRef]
31. Anonymous. Automotive Industry. 28 June 2017. Available online: https://ec.europa.eu/growth/sectors/automotive_en (accessed on 8 January 2021).
32. Masood, K.; Zoppi, M.; Molfino, R. Mathematical Modelling for Performance Evaluation Using Velocity Control for Semi-autonomous Vehicle. In *International Workshop on Soft Computing Models in Industrial and Environmental Applications, Proceedings of the 15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020), Burgos, Spain, 16–18 September 2020*; Advances in Intelligent Systems and Computing; Herrero, Á., Cambra, C., Urda, D., Sedano, J., Quintián, H., Corchado, E., Eds.; Springer: Cham, Switzerland, 2021; Volume 1268. [CrossRef]
33. Masood, K.; Molfino, R.; Zoppi, M. Simulated Sensor Based Strategies for Obstacle Avoidance Using Velocity Profiling for Autonomous Vehicle FURBOT. *Electronics* **2020**, *9*, 883. [CrossRef]
34. Silvestri, P.; Zoppi, M.; Molfino, R. Dynamic investigation on a new robotized vehicle for urban freight transport. *Simul. Model. Pract. Theory* **2019**, 96, 101938. [CrossRef]
35. Masood, K.; Dauptain, X.; Zoppi, M.; Molfino, R. Hydraulic Pressure-Flow Rate Control of a Pallet Handling Robot for an Autonomous Freight Delivery Vehicle. *Electronics* **2020**, *9*, 1370. [CrossRef]
36. Regulation (EU) No 168/2013 of the European Parliament and of the Council of 15 January 2013 on the approval and market surveillance of two- or three-wheel vehicles and quadricycles Text with EEA relevance. Available online: https://www.eumonitor.eu/9353000/1/j9vvik7m1c3gyxp/vj8bg68zk1zn (accessed on 16 January 2021).
37. Edwards, M.; Seidl, M.; Carroll, J.; Nathanson, A. *Provision of Information and Services to Perform an Initial Assessment of Additional Functional Safety and Vehicle Construction Requirements for L7e-A Heavy on-Road Quads (Rep. No. Ref. Ares(2014)1326718-28/04/2014)*. Transport Research Laboratory: Washington, DC, USA, 2014.
38. EU: Vehicle Definitions. Available online: https://www.transportpolicy.net/standard/eu-vehicle-definitions/ (accessed on 11 July 2020).
39. Directive 2007/46/EC of the European Parliament and of the Council of 5 September 2007 establishing a framework for the approval of motor vehicles and their trailers, and of systems, components and separate technical units intended for such vehicles (Framework Directive) (Text with EEA relevance). Available online: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A32007L0046 (accessed on 16 January 2021).
40. ApolloAuto. (n.d.). ApolloAuto/Apollo. Available online: https://github.com/ApolloAuto/apollo/blob/master/docs/quickstart/apollo_3_5_hardware_system_installation_guide.md (accessed on 5 December 2020).
41. LI-USB30-AR023ZWDRB. 18 November 2020. Available online: https://www.leopardimaging.com/product/usb30-cameras/usb30-box-cameras/li-usb30-ar023zwdrb/ (accessed on 30 November 2020).
42. Puck Lidar Sensor, High-Value Surround Lidar. 13 November 2020. Available online: https://velodynelidar.com/products/puck/ (accessed on 5 December 2020).
43. Ultra Puck Surround View Lidar Sensor. Available online: https://velodynelidar.com/products/ultra-puck/ (accessed on 5 December 2020).
44. High Precision. Minimum Range of Zero. Proprietary LiDAR ASICs.32-Channel Mid-Range Mechanical LiDAR. (n.d.). Available online: https://www.hesaitech.com/en/PandarXT (accessed on 5 December 2020).
45. Meet Olli · Local Motors. 21 January 2020. Available online: https://localmotors.com/meet-olli/ (accessed on 6 December 2020).
46. Google n.d., "Steetview," Digital Images, Google Maps, Photograph of 30 Geor. Kondili, Trikala, Greece, Taken in 2019. 2020. Available online: http://maps.google.com (accessed on 17 April 2020).
47. SAE MOBILUS. Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems J3016_201401. SAE International. January 2014. Available online: https://www.sae.org/standards/content/j3016_201401/ (accessed on 6 December 2020).
48. Marks, J. What Software Do Autonomous Vehicle Engineers Use? Part 1/2. 25 June 2018. Available online: https://medium.com/@olley_io/what-software-do-autonomous-vehicle-engineers-use-part-1-2-275631071199 (accessed on 30 November 2020).
49. Miller, C.; Valasek, C. Securing Self-Driving Cars (One Company at a Time). August 2018. Available online: http://illmatics.com/securing_self_driving_cars.pdf?TrucksFoT (accessed on 30 November 2020).

50. Experimental Security Research of Tesla Autopilot. March 2019. Available online: https://keenlab.tencent.com/en/whitepapers/Experimental_Security_Research_of_Tesla_Autopilot.pdf (accessed on 30 November 2020).
51. Khan, S. Validity of Manual Architecture-Based Threat Modeling—The Case of Apollo Auto Software. Master's Thesis, IOWA State University, Ames, IA, USA, 2019.
52. Apollo Auto: Technology Radar. November 2019. Available online: https://www.thoughtworks.com/radar/platforms/apollo-auto (accessed on 6 May 2020).
53. Wiggers, K. Baidu Releases Apollo 5.0 with Upgrades for 'Complex' Road Scenarios. 1 July 2019. Available online: https://venturebeat.com/2019/07/01/baidu-releases-apollo-5-0-with-upgrades-for-complex-road-scenarios/ (accessed on 6 May 2020).
54. Writer, S. Toyota to Join Baidu's Self-Driving Platform Apollo. 27 June 2019. Available online: https://asia.nikkei.com/Business/Companies/Toyota-to-join-Baidu-s-self-driving-platform-Apollo (accessed on 6 May 2020).
55. Hyatt, K. Toyota and Geely Partner with Baidu's Apollo Self-Driving Car Program. 8 July 2019. Available online: https://www.cnet.com/roadshow/news/toyota-geely-baidu-apollo-partnership/ (accessed on 6 May 2020).
56. Consolazio, G.R.; Chung, J.H.; Gurley, K.R. Impact simulation and full scale crash testing of a low profile concrete work zone barrier. *Comput. Struct.* **2003**, *81*, 1359–1374. [CrossRef]

2021 IEEE Intelligent Transportation Systems Conference (ITSC)
Indianapolis, USA. September 19-21, 2021

A Qualitative Analysis to Develop Handover Strategies for Freight Handling Autonomous Vehicle

Srinivasan Anbarasan [1], Khayyam Masood*[1,2], Matteo Zoppi[1], Vincent Frémont[2], Rezia M. Molfino[1]
[1] PMAR Lab
Università Degli Studi Di Genova - DIME
Via all'Opera Pia 15/A, 16145 Genova, Italy
Emails: seenu0613@outlook.com, zoppi@dimec.unige.it, molfino@dimec.unige.it
[2]ARMEN team
École Centrale de Nantes - LS2N
1 rue de la Noë, 44321 Nantes, France
Emails: khayyam.masood@ls2n.fr, vincent.fremont@ls2n.fr

2021 IEEE International Intelligent Transportation Systems Conference (ITSC) | 978-1-7281-9142-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/ITSC48978.2021.9565043

***Abstract*— Before attaining full autonomy by autonomous vehicles, impact of human interference is significant while achieving full automated driving system. Thus, it creates a fundamental challenge to have a safe and smooth takeover by the driver from the automated system until vehicles are completely autonomous. In this paper, we study two strategies for taking control of autonomous freight vehicle. First being the haptic takeover control method in which drivers cognitive workload and drivers readiness is evaluated during the Take Over Request (TOR). In the haptic takeover feedback, the process overcomes challenges in driver-automation collaboration during handover transitions between human and automated vehicles. In the second strategy, we analyze the need for Take Over Request, system behaviour during the Take Over-Request phasehow the system should react during a TOR phase, and the procedures that should be followed during the TOR to have a safe and comfortable take over the transition.**

I. Introduction

Complete automation describes a level 5 automated vehicle, which has mastered many challenges, including detecting roads/obstacles and monitoring driver behavior, and can decide during an emergency. Table I describes the various levels of automation and depicts how the take over request takes place at every level. However, the key to enhancing road safety is eliminating human interference, but enforcing complete automated vehicles on the roads is a complex multi-disciplinary problem which involves fields such as sensors, machine learning, social acceptability, which are still not solved completely [4].

TABLE I: Automation level defined by SAE [4]

Automation level	Name	Monitoring of Driving environment	TOR of driving task
0	No Automation	Driver	Driver
1	Assisted	Driver	Driver
2	Partial Automation	Driver	Driver
3	Conditional Automation	System	Driver
4	High Automation	System	System
5	Full Automation	System	System

With a constant increase in the production of level 3 automated vehicles, which are well equipped with technologies like Active cruise control (AAC), traffic jam assistance, and lane changing assist, but when the system falls in an emergency, it must rely on the driver to overcome it because the systems are still underdeveloped. The concept of requesting taking over control for automated vehicle is called a Take over Request (TOR). TOR is initiated when the system cannot handle the emergency or exceeds the system's Operational Design Domain (ODD) [1].

Some standard steps pursued during the handover progression are reported. During the introductory phase, the system suspends other active tasks before the system alerts the driver and initiates a program to decelerate the vehicle. In the second phase, the system explains the situation to the driver to take over control, and the takeover transition occurs [2]. During the last phase, after accepting the request, the driver takes control of the vehicle. Once the vehicle overcomes the situation, the system recommends reactivation of automation and gets activated based on driver's need. In another case, if the driver refuses to accept the request, the autonomous system works out the instruction to manage the situation on its own, and it follows different deceleration strategies depending on the situation [3].

When the system generates TOR, the driver needs to deviate their attention from the Non-Driving Related task (NDRT) to the Driving task. Before it generates TOR, both the driver's internal and external information are used to calculate driver's attention based on the driver's concentration, tiredness, workload, neuromuscular activity, and driving trajectory to calculate deviation measure in-formation, which guarantees safe TOR transition [2]. Some studies in Human-machine interaction (HMI) are being performed to estimate the amount of time taken to take over control of the automated vehicle. Preliminary results show that it takes 7 seconds for humans to take over control from a non-driving related task, but research shows that with shorter time taken for take over request, the worse the driver's performance [4].

TABLE II: Functional Failures and Automated Driving System (ADS) Handover scenarios [5]

level	Functional Failure	Handover Scenarios
Actuator	Over-breaking/steering Under-breaking/steering Tire Blowouts Break Failure steering failure Driving System Failure	Speed Limit Roadside Parking Continue driving to safe location Slide till stop Slowly stop in lane Slowly stop in lane
Perception	Loss of location Loss of front perception (in right most lane) Loss of front perception (not in right most lane) Loss of side/ back perception	Roadside parking Roadside Parking Slowly stop in lane Keep in current lane
Decision	ADS loss all redundant path All paths of ADS fail	Continue driving to safe location Emergency stop
Operational Design Domain (ODD)	out of ODD	Roadside parking
Power supply	loss of backup power after initial power failure	Emergency stop
Human Driver Problems	Drowsiness/lack of attention Sudden Disease Tire Blowouts	Drive to safe location Emergency stop Continue driving to safe location

Hence, while designing a handover strategy for an automated vehicle, it must give sufficient time for the driver to take over control.

A new challenge arises in ensuring a safe takeover request transition while switching back to driving task from the different mental state of driver [3]. This problem requires new research and design by considering the driver's readiness, attention, workload, and driving trajectory [2], [3].

To develop a handover strategy, it is important to know failure situations and different scenarios where handover control is most required [2]. Table II describes several system failures and the scenario that will follow [5].

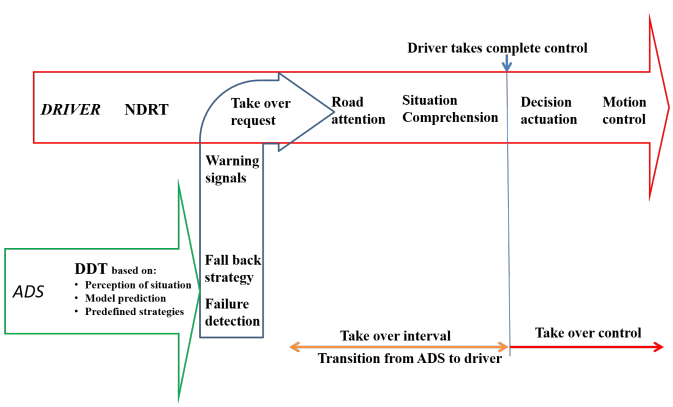


Fig. 1: Handover strategy module in ADS model [5]

In the handover strategy model, input is the detected failure, and the output corresponds to the fallback scenario given to the decision module, which turns out to be a fallback strategy. The refresh is initiated once the error code changes [4]. The figure 1 shows the handover strategy model. This review aims to study the techniques used for handover or take over the suitable request to be implemented on FURBOT.

The work carried out is a step forward in automation for freight handling vehicle FURBOT for its participation as an autonomous vehicle in project SHOW. For vehicle autonomous navigation, a mathematical model has been built [6] which is further developed for sensor-based strategies for obstacle avoidance [7]. Further automation is achieved by controlling the pallet handling robot using hydraulic pressure-flow rate control [8]. Strategies are built to convert the vehicle from drive-by-wire to a completely autonomous vehicle [9]. This work is the next step in creating handover strategies for the autonomous vehicle in case of operation failures.

In the next section we discuss challenges during takeover transition based on the various factors while driving a level 3 automated freight handling vehicle. In section 3, haptic takeover framework is reviewed. Section 4, complex problems and variables faced during an automated driving system and its implication are presented thus leading to the conclusion of our research.

II. Automation Challenges and Complexity

The next milestone in the automobile industry is level 3 automated vehicles. However, few companies like Ford have shifted their focus in developing level 4 or higher-level automated vehicles instead of level 3. The crucial reasons for this decision rely on the interference of humans in level 3 automation [1].

Taking control over an autonomous vehicle is a complex process where human driver is required to be aware of the emergency situation.

This research describe various circumstances that contribute to complexity:

- Reduction in driver awareness due to possible boredom/road dullness associated with autonomous driving.
- The time taken by the driver to react from NDRT (non-driving-related task) to take over control [30].

A. Objective Complexity

Objective complexity refers to system's essential properties. For handover, there are factors that not only depend on the driver's perception but also the objective situation [10]. The factors used to determine objective complexity for takeover request [1] are given below.

- Intensive traffic
- Road condition
- Environmental condition
- System to transfer control to driver from automation.

B. Subjective Complexity

Subjective complexity involves factors affected by the driver's perception adaption process and their influence on the driver's response to a TOR, including the driver's mental state and work load due driver's distraction over NDRT (Non driving related Task).The communication between the

driver and system is set up by the HMI (Human-Machine Interface), which helps in driver's decision-making process. The most important question in recent research is that how TOR can consider subjective complexity [1].The important aspects about subjective complexity are :

- Driver's alertness after TOR is generated.
- System evaluation for interacting with driver for TOR.

These aspects are further discussed in detail below.

1) Driver's readiness Evaluation: When TOR is generated, various approaches are used to evaluate the drivers alertness and readiness to interact with the vehicle.

- First, sensors inside the vehicle can sense regular interaction. However, key problem of this approach is compromised by user experience and there is a chance of cheating the system because of over trust. To solve this problem, evaluation of torque applied by the driver on the steering wheel is considered [1], [11].
- To understand humans alertness over driving task, a glance analysis is performed, which is based on the evaluation of the driver's cognitive engagement during Take over Request. Cognitive engagement is based humans pupil diameter, glance duration, and Area of Interest [1] [12].
- Finally, facial recognition and expression can determine the driver's readiness in Take over request, whereby evaluation of change in facial expression is carried out to understand the driver's attention over TOR [13], [1].

2) TOR Communication Interfaces: The essential communication methods used by Human-Machine Interaction (HMI) are visual, auditory, and vibrotactile. Visual display conveys information where the driver can understand at ease, but few studies prove that the task that requires visual attention has a higher unsettling risk than other tasks performed [14] because the driver's attention is affected by multiple glances between vehicle devices and roads, reducing the ability to maintain vehicle control [15]. Whereas auditory messages are complementary to visual aids. In [15] studies show that speech warning helps in improving the response time of the driver to Takeover request.

The vibrotactile display is a device that communicates to the driver by vibrations for take over request, which is placed under the driver's seat and seat belt [16]. This vibrotactile display is beneficial because it becomes harder for the driver to ignore the message and the message is conveyed privately. The most important aspects while placing vibrotactile are

- Frequency/amplitude set with the driver's comfort and concern [1].
- Location/timing adjusted depending on urgency [1].

Different audiovisual display configurations help determine the urgency of the transition from automated to manual driving [17], [1]. An HMI conveys a timely message to the driver, depending upon the urgency of the request. Different ways by which the system conveys a message to the driver is classified as :

- Visual: Information to the driver is conveyed via the display screen and vehicle's ambient light [18].
- Auditory: HMI uses various sound and intensity to alert the driver of the urgency for TOR [15].
- Haptic: Information is conveyed by a haptic interaction relying on kinesthetic communication through tractors located either on the seat or seat belt [1], [17].

III. Haptic Take-Over System

In a partial automated vehicle, the driver should take over control from NDRT to driving tasks when the system can no longer handle a situation. Thus, this poses a new challenge to transfer control to humans safely and smoothly.

Development based on optimal control and artificial intelligence was carried to examine effective interaction between human drivers and automated systems [2]. Some developed models showing results in improved human and system integration are:

- Optimal control model: This model is proposed to understand the behavior of human driver's [2], [19].
- Stochastic optimization method: To formalize the interactive behavior between drivers and automated systems, Stochastic optimization method was developed [2].
- Game theory method: This method is developed to improve human-machine cooperation during a takeover request or any other communication between humans and the system [2]. That resulted in improved lane-keeping performance.

Haptic takeover controller is being developed to have a secure and comfortable communication during the takeover requests. It gets activated to assist the driver when he perceives a takeover request. The driver's takeover capability determines the required feedback torque [3], [2].

A. Haptic Takeover Control Framework

A haptic takeover control framework is developed in [3], during the take over interval the vehicle control is shared by ADS and driver. Firstly optimal sequence of control input is obtained from the planned trajectory of the vehicle. Simultaneously, the required optimal control task is achieved by both human and machine, that share the vehicle's control authority. The neuromuscular system (NMS) generated the driver's actual inputμ_D, which is $\alpha\%$ of the vehicle's total required input. The remaining portion of the optimal control input is contributed by automated input μ_A, which is compensated for the driver's input. The take over process is completed when $\alpha\%$ increases to 100 [3].

It is impossible to ease the automation and give complete control to the driver, as the transition is expected to complete safely and smoothly. The determination of control authority is decided by assessing the driver's take over capability in real-time. The feedback control is feed to the system in an increasing manner. Driver's optimal control authority α_d is calculated in real-time based on driver's take over capability. [2].

The exact value of driver's participation α and the allowed is α_d compared to determine the optimal required input. The feedback is generated to reduce the difference between real drivers input and the required input torque. It is essential to

make takeover transition take place progressively with the suitable steering torque.

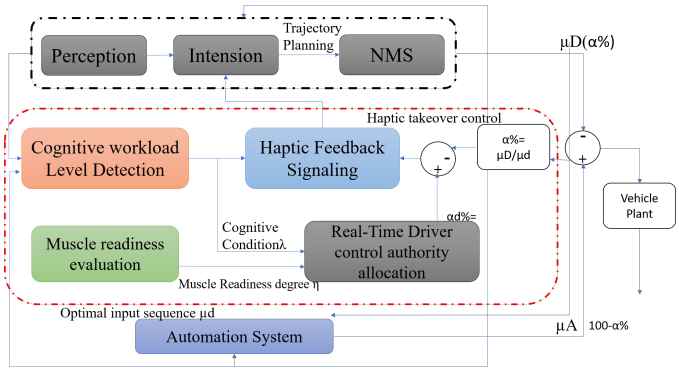
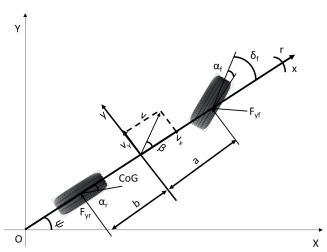


Fig. 2: Haptic Take over Request control framework [2].

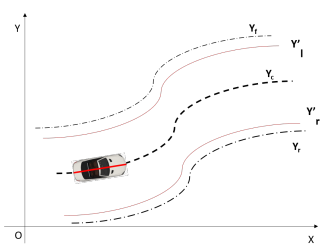
Understanding how to determine the feedback with crucial terms such as the vehicle's planned trajectory, driver's workload, and readiness is necessary. This understanding allows us to understand how the haptic takeover control framework works.

B. Desired Input Sequence Of System

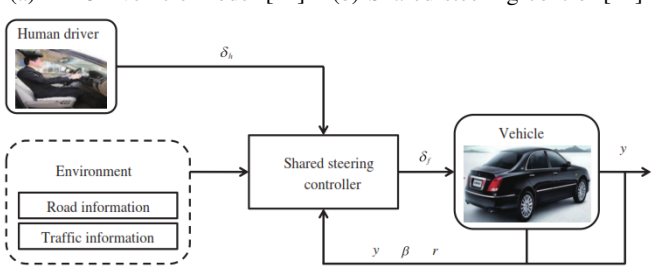
1) Driver-Vehicle System Modelling: The automated driving system with vehicle dynamics is modeled to determine the optimal input sequence. To estimate and model the vehicle's dynamic behavior, 2-DOF bicycle kinematic model is considered [22] [3].



(a) 2-DOF vehicle model [22]



(b) Shared steering control [22]



(c) Control block diagram of the shared steering vehicle [22]

Fig. 3: Kinematic model and control diagram

The figure 3a indicates the amount of assistance required during an overload and underload condition of the driver in a driving assisting system.
Assuming that the longitudinal velocity is constant at V_x, then referred to simplified bicycle model of the vehicle and to the control diagram as represented in Figure 3a and 3b, proposed by [22], the lateral motion of the vehicle in response to the front wheel steering angle δ_f, can be discretized as in :

$$x(k+1) = A_d x(k) + B_d u(k), z(k) = C_d x(k), \quad (1)$$

$A_d = e^{AT_s}$, $B_d = \int_0^{T_s} x^2 \, d\tau . B$ $C_d = C$
The matrix A, B and C are referred from [3].
Where $x(t) = [v_y(t), \omega(t), y(t)\psi(t), \theta_{sw}, \dot{\theta}_{sw}]^T$ is the vehicle model, $v_y(t)$ is the lateral velocity, $y(t)$ is the lateral displacement, $\omega(t)$ is yaw rate, $\psi(t)$ is the yaw angle, θ_{sw} is angular position of steering wheel. z(t)= y(t) is the output and T_s is the sampling time [23] [3].

C. Take over authority determination

Research carried on the alliance between driver's workload based on the driver's cognitive and driving performance depicts a decline in driving performance with driver's workload [24]. Still, there is always a balance between the workload and performance, which are in an acceptable range. Driver's workload is determined by the driver's pupil size, skin contraction, and pulse-to-pulse interval are captured via camera to identify the driver's distraction. The driver's physiological information, the electrocardiogram (ECG) signals, blinking of eye, pupil size, and head turning angles is measured to determine the optimal control authority and drivers cognitive workload [24], [25]. Thus, the control authority required during the take-over transition is determined by a haptic feedback control framework developed based on the driver's cognitive workload condition.

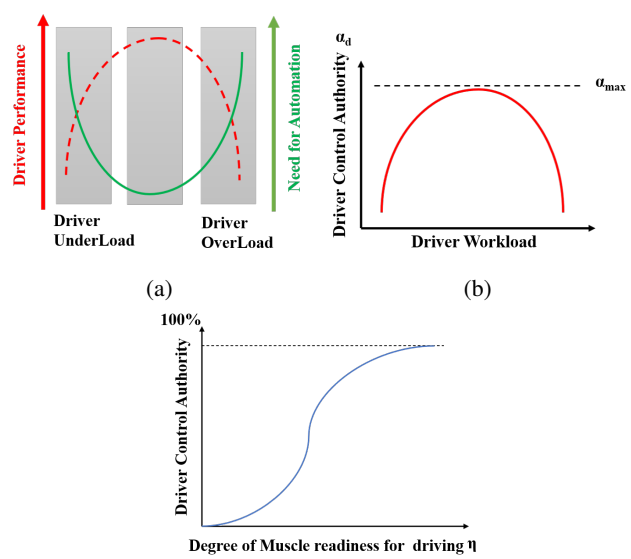


Fig. 4: a) Driver performance and need for automation [2] b) Control authority and driver workload [2] c) Modelled control authority and driver muscle readiness [2]

To understand the logic behind the developed design, the developed design is regulated so that desired control authority α_d is allocated to the driver depending on driver's

real-time workload. The automation should provide continuous assistance to the driver during take over transition by considering the driver's workload. The referenced control authority α_d is modeled as the time-varying function of the workload λ, which corresponds to in U-shape describing the amount of automation assistance required for the driver (figure 4a) [3].

$$\alpha_d(\lambda) = \alpha_{max} - \varphi_1(\lambda - \varphi_2)^2 \tag{2}$$

where $\alpha_d \in [0,1]$ present control authority of the driver, φ_1 and φ_2 are the parameters are designed for haptic take over control, α_{max} is the min value of control authority and λ is the drivers cognitive workload.

D. Muscle Readiness for Takeover Authority Determination

Driver's dynamics related to neuromuscular activities are an essential component in takeover control. During takeover process, driver's hand comes in contact with the steering wheel to control the vehicle, and the driver's biomechanical state affects their manual driving performance. To determine the driver's biomechanical state, evaluation of muscle readiness is carried out in real-time, which is determined by measuring muscle activity with electromyography (ECG) and muscle co-activation during a simulated driving task [26].

To overcome dangerous situation and avoid obstacles , it requires a large steering torque during the takeover process. But if the driver ignores the Take over request situation, and muscles are relaxed, then the driver's readiness for driving is low because it would not allow a smooth takeover process. Considering the above condition of the driver control authority is modeled in figure 4c [27] .

When the handover control is in process, the optimal sequence of control input is derived from the vehicle's planned trajectory. Driver's input is created by haptic feedback system considering the neuromuscular dynamic system, which takes up α % of overall input to the vehicle, and the feedback input is contributed by the automation, which compensates the driver's input until α reaches 100 % which shows hand over control is completed [3].

E. Torque controller for haptic feedback system

In level 3 automation, when the automated system can no longer handle the situation, the auto-driving system deactivates, and the control to the driver is given back. With this haptic feedback control framework, a handover request signal would be generated, assessing the driver's attention level.

The intensity of haptic feedback torque and direction is determined by the driver's attention and participation in the driving task, activated when TOR is requested. Three significant factors decide the level of driver control authority: Required optimal input sequence, cognitive workload, and neuromuscular dynamics. In (equation 3) the driver's current engagement level is evaluated based on the driver's applied steering torque and steering torque required by the system [2].

$$\alpha\% = \frac{u_D}{u_d} * 100 \tag{3}$$

Where u_D is the drivers actual input and u_d is the required input needed from the driver. The feedback torque from the system is generated and exerted on the steering wheel to diminish the difference between α % and desired control authority α_d %, that leads to a safe and smooth take over transition [2], [4]. According to the description from [3] the torque controller is composed of a tracking problem which is combined with feed-forward and feedback. The structure is adopted, as:

$$T_{hpt} = T_{ff} + T_{fb} \tag{4}$$

In which T_{hpt} : haptic feedback torque. T_{ff} is feed-forward and T_{fb} is feed-back component (explained in equations 5-7).

$$T_{ff} = u_d - u_D = u_A \tag{5}$$

$$T_{fb} = -K_p e_c - K_I \int e_c dt \tag{6}$$

$$e_c = \alpha_d\% - \alpha\% \tag{7}$$

Where K_p and K_I in equation 6 are gains of feedback control and e_c from equation 6 is the control error [3], [2].

This study about the haptic feedback take over control, generates the required optimal system input to have a safe and smooth take over transition [3].

IV. Bringing the vehicle to a safe stop within the allotted TOR time

In Level 3 and 4 automation, driver not taking immediate control of vehicle is a major problem. In this section, the implication of this primary drawback on the automated driving system is scrutinized, and a possible solution is stated. The automated driving system state is defined by equation 8 [4].

$$(s\varepsilon S), where(S = S_{Auto}, S_{TOR}, S_{Safe}, S_{Manual}) \tag{8}$$

where S denotes the distance vehicle travelled, S_{Auto} is automated driving, S_{TOR} distance travelled in TOR phase after TOR initiated, S_{Safe} is the safe distance in which vehicle is required to stop in case driver does not re-take control and S_{Manual} is for manual mode.

The key idea of this strategy is very generalized for all TOR situations i.e. to signal the end of the automated driving route e.g. before a vehicle exits the highway, the driver is required to take control of the car when the car cannot handle such a situation. When the driver rejects or does not take control of the vehicle, the system brings the car to a safe halt [28].

A. TOR Time Interval

The principal aim is to build TOR strategy with constraints of TOR, that will be capable to Handel any TOR situation [4]. Time required to take over control is the significant variable in Take over request problem. It is defined as $T_{TOR} = [t_{TOR,s}, t_{TOR,e}]$. Where $t_{TOR,s}$ and $t_{TOR,e}$, are start and end of TOR time phase. The TOR time defines the minimum time, where the driver takes over control [4]. Figure 5 explains the time frame during a takeover process. Preliminary research indicates the take over request transition time is 10s [4].

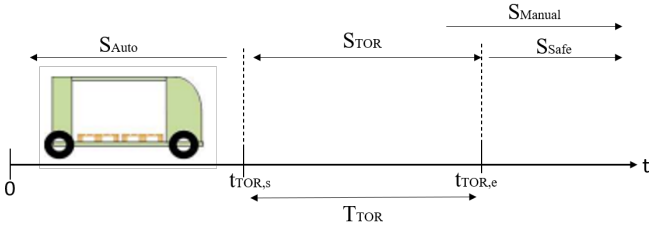


Fig. 5: Take over transition Time line [4]

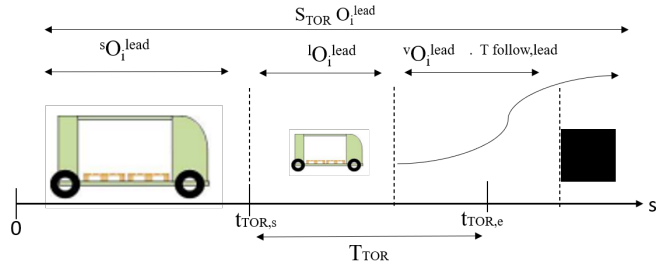


Fig. 6: lead vehicle constraint Time line [4].

Based on the time taken in TOR, two goals are determined, the system should be able to achieve either of one goal at the end of the TOR timeline. These goals are

- The driver must have taken control (during S_{Manual}) of the vehicle.
- If driver does not accept the request in time, then the vehicle comes to a complete stop (during S_{Safe}).

Overall, the first goal must be achieved as much safe and smooth way as possible. This allows the driver to regain control of the vehicle comfortably, but simultaneously make sure the secondary goal can be reached when the driver/human cannot take over control in the time phase allocated i.e. T_{TOR} [4], [13].

B. Stationary Object Constraint

Besides TOR interval, during the take over transition car's enviroment plays a crucial role.. Guaranteeing to achieve the secondary goal that is without colliding, bringing the vehicle to a safe stop. Assuming that the driver can handle a critical situation when an autonomous driving mode starts TOR [4] [29].

During TOR, stationary object constraints is an essential constraint, and d_{TOR} is the distance to such a stationary object.

$$(d_{TOR}) = (S_{TOR,}O_i^{stat}) = (S_{O_i^{stat}}) \tag{9}$$

Where S_{TOR}, O_i^{stat}, is the distance to the object at rest O_i^{stat} coordinate system of the host vehicle. When there is no stationary object is present then the $d_{TOR} = \infty$ [4].

C. Lead Vehicle Constraint

While driving on a highway, active cruise control is activated, which helps in a safer driving environment and keeps a safe distance from lead vehicle. For TOR it is necessary to derive a lead vehicle constraint with the max stopping distance s_{TOR},O_i^{lead} which can be computed from [4].

It is assumed that the lead vehicle's sudden deceleration is automatically handled by normal driving strategy, which is active during S_{Auto}. For lead vehicle constraint, a situation is depicted where a virtual stationary object is placed ahead of the lead vehicle [28].

At any point of time, the lead vehicle may overcome the object by changing the lane, revealing stationary object to the follow vehicle, reducing the problem to the stationary object constraint, such that s_{TOR}, O_i^{lead} is analog to s_{TOR}, O_i^{stat} [29]. This situation is depicted in the figure 6.

For the above-mentioned situation defined, a d_{TOR} is calculated so that a stopping distance for vehicle behind the lead vehicle. The new stopping distance $(d_{TOR}) =(s_{TOR}),(O_i^{lead})$ is calculated as:

$$(d_{TOR}) = (S_{O_i^{lead}}) + (l_{O_i^{lead}}) + (v_{O_i^{lead}})(t_{follow,lead}) \tag{10}$$

$(s_{TOR}),(O_i^{lead})$ is the distance of the vehicle in front,$(v_{O_i^{lead}})$ is the speed of the vehicle and $(t_{follow,lead})$ is the time-gap to the stationary object from the lead vehicle.

D. Environment Observability Constraint

During TOR, the vehicle can observe only a specific area of the environment, which depends on sensors and weather factors. The assumption being made, such that the stationary object lies behind the observability distance. When object is detected, the vehicle must come to a complete during S_{TOR}.

$$(d_{TOR}) = (S_{TOR,obs}) = (S_{obs}) \tag{11}$$

Where $S_{TOR,obs}$ is the necessary stopping distance based on observability and S_{obs} is the observability in the current lane.

E. Applying these constraint

The above discussed constraints are evaluated at every time interval and the minimum distance constraint is used in handover control strategy.

$$(d_{TOR}) = min(s_{TOR}, O_i^{stat}, s_{TOR}, O_i^{lead}, s_{TOR,obs}) \tag{12}$$

As a prime goal, it is required to give sufficient time to drivers for take over requests. The above discussed constraint is used to develop a deceleration profile during S_{TOR} based on the Model Predictive Control (MPC) approach that brings the vehicle to a complete stop under safe travel distance [3].

V. Conclusion

This research mainly focuses on handover scenarios and its challenges and complexity in automated vehicle. This review's key aim is to devise a strategy for minimal risk conditions when encountering emergency situations. To achieve this, we have discussed various functional failures and its proposed handover scenarios and different handover strategies which helps vehicle in safe and smooth TOR transition. Handover scenarios have been discussed for different situations to overcome functional failures.

In the handover strategies, we have considered two fundamental approaches, i.e., Haptic feedback take over control and bringing the vehicle to a safe stop during TOR. Both strategies have their importance and usage. In the future, we will focus on implementing these strategies in the freight handling vehicle FURBOT, which would be essential to have a safe TOR transition and would help the FURBOT travel in an optimized trajectory and safe from its surroundings during its autonomous navigation.

ACKNOWLEDGMENT

This paper was supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 875530, project SHOW (SHared automation Operating models for Worldwide adoption).

REFERENCES

[1] W. Morales-Alvarez, O. Sipele, R. Léberon, H. H. Tadjine, and C. Olaverri-Monreal, "Automated Driving: A Literature Review of the Take over Request in Conditional Automation," Electronics, vol. 9, no. 12, p. 2087, 2020.

[2] Y. Li, C. Lv, and J. Xue, "A Novel Predictive Haptic Control Interface for Automation-to-Human Takeover of Automated Vehicles," 2019 IEEE Intelligent Vehicles Symposium (IV), 2019.

[3] C. Lv, H. Wang, D. Cao, Y. Zhao, M. Sullman, D. J. Auger, J. Brighton, R. Matthias, L. Skrypchuk, and A. Mouzakitis, "A Novel Control Framework of Haptic Take-Over System for Automated Vehicles," 2018 IEEE Intelligent Vehicles Symposium (IV), 2018.

[4] M. Bahram, M. Aeberhard, and D. Wollherr, "Please take over! an analysis and strategy for a driver take over request during autonomous driving," 2015 IEEE Intelligent Vehicles Symposium (IV), 2015.

[5] J. Yu and F. Luo, "Fallback Strategy for Level 4+ Automated Driving System*," 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019.

[6] Masood K., Zoppi M., Molfino R. (2021) Mathematical Modelling for Performance Evaluation Using Velocity Control for Semi-autonomous Vehicle. In: Herrero Á., Cambra C., Urda D., Sedano J., Quintián H., Corchado E. (eds) 15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020). SOCO 2020. Advances in Intelligent Systems and Computing, vol 1268. Springer, Cham. https://doi.org/10.1007/978-3-030-57802-2_59

[7] Masood, K.; Molfino, R.; Zoppi, M. Simulated Sensor Based Strategies for Obstacle Avoidance Using Velocity Profiling for Autonomous Vehicle FURBOT. Electronics 2020, 9, 883. https://doi.org/10.3390/electronics9060883

[8] Masood, K.; Dauptain, X.; Zoppi, M.; Molfino, R. Hydraulic Pressure-Flow Rate Control of a Pallet Handling Robot for an Autonomous Freight Delivery Vehicle. Electronics 2020, 9, 1370. https://doi.org/10.3390/electronics9091370

[9] Masood, K.; Zoppi, M.; Fremont, V.; Molfino, R.M. From Drive-By-Wire to Autonomous Vehicle: Urban Freight Vehicle Perspectives. Sustainability 2021, 13, 1169. https://doi.org/10.3390/su13031169

[10] C. Gold, M. Körber, D. Lechner, and K. Bengler, "Taking Over Control From Highly Automated Vehicles in Complex Traffic Situations," Human Factors: The Journal of the Human Factors and Ergonomics Society, vol. 58, no. 4, pp. 642–652, 2016.

[11] J. Capalar and C. Olaverri-Monreal, "Hypovigilance in limited self-driving automation: Peripheral visual stimulus for a balanced level of automation and cognitive workload," 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), 2017.

[12] H. Hayashi, M. Kamezaki, U. E. Manawadu, T. Kawano, T. Ema, T. Tomita, L. Catherine, and S. Sugano, "A Driver Situational Awareness Estimation System Based on Standard Glance Model for Unscheduled Takeover Situations," 2019 IEEE Intelligent Vehicles Symposium (IV), 2019.

[13] X. Li, R. Schroeter, A. Rakotonirainy, J. Kuo, and M. G. Lenné, "Effects of different non-driving-related-task display modes on drivers' eye-movement patterns during take-over in an automated vehicle," Transportation Research Part F: Traffic Psychology and Behaviour, vol. 70, pp. 135–148, 2020.

[14] C. Olaverri-Monreal, A. E. Hasan, J. Bulut, M. Korber, and K. Bengler, "Impact of In-Vehicle Displays Location Preferences on Drivers' Performance and Gaze," IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 4, pp. 1770–1780, 2014.

[15] Y. Forster, F. Naujoks, A. Neukum, and L. Huestegge, "Driver compliance to take-over requests with different auditory outputs in conditional automation," Accident Analysis amp; Prevention, vol. 109, pp. 18–28, 2017.

[16] S. M. Petermeijer, J. C. de Winter, and K. J. Bengler, "Vibrotactile Displays: A Survey With a View on Highly Automated Driving," IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 4, pp. 897–907, 2016.

[17] I. Politis, S. Brewster, and F. Pollick, "Using Multimodal Displays to Signify Critical Handovers of Control to Distracted Autonomous Car Drivers," International Journal of Mobile Human Computer Interaction, vol. 9, no. 3, pp. 1–16, 2017.

[18] C. Olaverri-Monreal, S. Kumar, and A. Diaz-Alvarez, "Automated Driving: Interactive Automation Control System to Enhance Situational Awareness in Conditional Automation," 2018 IEEE Intelligent Vehicles Symposium (IV), 2018.

[19] J. Radlmayr, C. Gold, L. Lorenz, M. Farid, and K. Bengler, "How Traffic Situations and Non-Driving Related Tasks Affect the Take-Over Quality in Highly Automated Driving," Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 58, no. 1, pp. 2063–2067, 2014.

[20] X. Ji, K. Yang, X. Na, C. Lv, and Y. Liu, "Shared Steering Torque Control for Lane Change Assistance: A Stochastic Game-Theoretic Approach," IEEE Transactions on Industrial Electronics, vol. 66, no. 4, pp. 3093–3105, 2019.

[21] M. Flad, L. Frohlich, and S. Hohmann, "Cooperative Shared Control Driver Assistance Systems Based on Motion Primitives and Differential Games," IEEE Transactions on Human-Machine Systems, vol. 47, no. 5, pp. 711–722, 2017.

[22] R. Rajamani, Vehicle dynamics and control. New York: Springer, 2012. https://doi.org/10.1007/978-1-4614-1433-9

[23] J. Liu, H. Guo, L. Song, Q. Dai, and H. Chen, "Driver-automation shared steering control for highly automated vehicles," Science China Information Sciences, vol. 63, no. 9, 2020.

[24] K. Zeeb, A. Buchner, and M. Schrauf, "Is take-over time all that matters? The impact of visual-cognitive load on driver take-over quality after conditionally automated driving," Accident Analysis amp; Prevention, vol. 92, pp. 230–239, 2016.

[25] Y. Xing, C. Lv, D. Cao, H. Wang, and Y. Zhao, "Driver workload estimation using a novel hybrid method of error reduction ratio causality and support vector machine," Measurement, vol. 114, pp. 390–397, 2018.

[26] Y. H. Liu, X. W. Ji, H. Ryouhei, M. Takahiro, and L. M. Lou, "Function of shoulder muscles of driver in vehicle steering maneuver," Science China Technological Sciences, vol. 55, no. 12, pp. 3445–3454, 2012.

[27] C. Lv, H. Wang, D. Cao, Y. Zhao, D. J. Auger, M. Sullman, R. Matthias, L. Skrypchuk, and A. Mouzakitis, "Characterization of Driver Neuromuscular Dynamics for Human–Automation Collaboration Design of Automated Vehicles," IEEE/ASME Transactions on Mechatronics, vol. 23, no. 6, pp. 2558–2567, 2018.

[28] C. Gold, D. Damböck, L. Lorenz, and K. Bengler, "'Take over!' How long does it take to get the driver back into the loop?," Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 57, no. 1, pp. 1938–1942, 2013.

[29] M. Ardelt, C. Coester, and N. Kaempchen, "Highly Automated Driving on Freeways in Real Traffic Using a Probabilistic Framework," IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 4, pp. 1576–1585, 2012.

[30] O. M. Winzer, A. S. Conti, C. Olaverri-Monreal, and K. Bengler, "Modifications of driver attention post-distraction: A detection response task study," HCI in Business, Government and Organizations. Interacting with Information Systems, pp. 400–410, 2017.

energies

MDPI

Article

Parking Pose Generation for Autonomous Freight Collection by Pallet Handling Car-like Robot

Khayyam Masood [1,2,*,†], David Pérez Morales [2], Vincent Fremont [2], Matteo Zoppi [1] and Rezia Molfino [1]

[1] PMAR Robotics, DIME, University of Genova, 16126 Genova, Italy; zoppi@dimec.unige.it (M.Z.); molfino@dimec.unige.it (R.M.)
[2] LS2N, CNRS, Ecole Centrale de Nantes, 1 Rue de la Noë, 44300 Nantes, France; david.perezmorales@ls2n.fr (D.P.M.); vincent.fremont@ec-nantes.fr (V.F.)
* Correspondence: khayyam.masood@edu.unige.it; Tel.: +39-3205304054
† Current address: Room 213, LS2N, CNRS, Ecole Centrale de Nantes, 1 Rue de la Noë, 44300 Nantes, France.

Abstract: This paper focuses on autonomous navigation for an electric freight vehicle designed to collect freight autonomously using pallet handling robots installed in the vehicle. Apart from autonomous vehicle navigation, the primary hurdle for vehicle autonomy is the autonomous collection of freight irrespective of freight orientation/location. This research focuses on generating parking pose for the vehicle irrespective of the orientation of freight for its autonomous collection. Freight orientation is calculated by capturing the freight through onboard sensors. Afterward, this information creates a parking pose using mathematical equations and knowledge of the vehicle and freight collection limitations. Separate parking spots are generated for separate loading bays of the vehicle depending on the availability of the loading bay. Finally, results are captured and verified for different orientations of freight to conclude the research.

Keywords: parking generation; freight handling; autonomous vehicles; driver-less parking

check for updates

Citation: Masood, K.; Morales, D.P.; Fremont, V.; Zoppi, M.; Molfino, R. Parking Pose Generation for Autonomous Freight Collection by Pallet Handling Car-like Robot. *Energies* **2021**, *14*, 4677. https://doi.org/10.3390/en14154677

Academic Editor: Guzek Marek

Received: 5 July 2021
Accepted: 30 July 2021
Published: 1 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Vehicle FURBOT (Freight Urban RoBOTic vehicle) is a complete drive-by-wire electric freight vehicle designed to operate in last-mile delivery operations in an urban environment. The vehicle was part of the European Green Vehicles Initiative (EGVI) funded under the umbrella of the FP7-Transport European project ending in December 2015 [1,2]. The vehicle was completed as an entire drive-by-wire vehicle with the possibility of being upgraded to an autonomous vehicle. Currently, the project is being developed to convert it from a drive-by-wire to a completely autonomous vehicle. Thus, the necessity of autonomous freight collection is generated.

For autonomous vehicle navigation, a mathematical model has been built for performance evaluation of the vehicle [3] which is later used for developing sensor-based strategies for obstacle avoidance [4]. Further automation is achieved by controlling the pallet handling robot using hydraulic pressure-flow rate control [5]. Strategies are built to convert the vehicle from drive-by-wire to a completely autonomous vehicle [6]. Additionally, work on perception and control strategies for autonomous docking of the freight has also been previously studied [7]. This work is the next step in attaining the automation of the vehicle for it to auto-load its freight.

FURBOT has to take part in SHOW (SHared automation Operating models for Worldwide adoption) project where it is required to deliver freight autonomously to customers across an urban area. Currently, the vehicle is not equipped with proper algorithms for the autonomous collection of freight. However, the vehicle will be equipped with autonomous navigation software, which has its own autonomous parking algorithms [8,9]. The autonomous freight collection requires the vehicle to park next to the freight, such that it can collect freight autonomously. For that, a need for generation of correct parking pose

is required so the already available parking algorithms can maneuver the vehicle to the correct parking pose. Thus, the need for our work.

The role of autonomous vehicles in an urban environment is still being studied, and there is a gap in understanding the role of urban autonomous vehicles in smart urban mobility [10]. Autonomous freight delivery and their adoption in society are in preliminary stages, and surveys are being conducted for their acceptability in the society [11]. Further studies are being conducted to understand the role of autonomous vehicles in logistics [12]. Still, the concept of using manned deliveries for last-mile delivery through autonomous vehicles is being studied [13]. Privately-owned autonomous vehicles not designed for freight delivery are being looked into, and their potential is being discussed [14]. Currently, autonomous freight delivery through a designated autonomous freight vehicle is still a new subject, and the latest research proves that the solution for autonomous freight delivery is yet to be explored.

Freight delivery in an urban environment comes with many predefined constraints, e.g., loading constraints, vehicle routing problems [15] or compartment and order fulfillment related issues [16]. Many types of loading constraints and vehicle routing problems are discussed in [15,17]. Vehicle compartment issues are further divided into the flexibility of compartment sizes, assignment of product types, and share-ability of compartments. In contrast, order fulfillment issues comprise the mode of demand fulfillment and the total number of visits per customer [16]. However, all these logistic issues do not deal with how the freight needs to be loaded in the vehicle, especially dealing with the freight loading autonomously.

The generation of a parking spot for autonomous vehicles has also been studied in-depth in the last decade. Many next-generation vehicles are already equipped with autonomous parking [18]. New research in parking solutions varies in finding niche problems within the vehicle parking solution domain, including vision-based indoor parking [19], parking of fleet of vehicles [20], collision-avoidance-based parking [21] and sensor (Lidar)-based parking solutions [22]. However, it is challenging to find parking solutions for freight vehicles, especially for freight loading. The closest work on autonomous parking for articulate vehicles is studied in [23], but it is still insufficient in resolving freight-based parking solutions. As FURBOT is a designated freight handling vehicle, the parking solution for autonomous loading of freight needs to be resolved.

Path generation and detecting correct parking pose for the vehicle are interesting subjects, especially in autonomous vehicles as they need to park by themselves. Usual work on vehicle parking pose estimation comes from parking pose marking recognition [24] or using a camera to detect these markings [25]. Apart from parking pose recognition from sensor feedback, estimation of parking pose requires solving geometrical equations for path generation or calculating current parking pose [26,27]. This research also exploits geometrical equations after freight detection to generate a parking pose that can solve the autonomous freight collection problem.

Electric freight vehicle poses a unique challenge for the creation of parking pose concerning freight for its collection. As it is not possible to make the freight perfectly align with any known orientation before loading the freight. Furthermore, there is a hassle for aligning freight to an orientation each time before loading the freight, even if a solution exists. It is much easier to consider autonomously loading the freight through aligning the vehicle w.r.t the freight. However, previous research in this domain is not available, i.e., the generation of parking pose for a freight handling vehicle for freight collection. Furthermore, the vehicle FURBOT is unique because it loads the freight sideways within itself through its loading bays, unlike typical forklift trucks.

This research is a step towards an autonomous collection of freight for the vehicle, especially the part of creating autonomous parking pose w.r.t freight keeping in line the constraints posed by vehicle FURBOT. Currently, there is no alternative solution for collecting freight autonomously for such vehicles. The ability to create a parking pose w.r.t an inanimate object, which is subject to change its orientation and position, is not

previously studied and discussed, thus the need of this research. The highlight of this research is creating a parking pose for such a vehicle, keeping in mind the constraints posed by the freight and the vehicle. Finally, testing the results to validate the proposed mathematical modeling for the solution to the unique problem.

In the next section, we discuss the problem and approach to the parallel freight parking issue, discussing the issues concerning freight detection, how the vehicle should be oriented for collection of the freight, and how parking spot should be defined w.r.t freight. Following, in Section 3, the software and control architecture of the vehicle is discussed, focusing on desired parking pose for the collection of freight. In Section 4, a mathematical model for vehicle parking pose is discussed. This section also covers the vehicle kinematic model and mathematical equations and notations for creating the parking pose for each loading bay of the vehicle. Section 5 elaborates on the desired results achieved of the parking pose for each loading bay of the vehicle, and the results are further verified for different freight poses, and the parking pose is validated as per requirement. Finally, in Section 6, the conclusions of the research are discussed.

2. Problem and Approach

The considered vehicle has to park, keeping the freight on the vehicle's right-hand side for loading the freight through the designed loading bays. There are three core steps in defining the parking spot for the vehicle for the autonomous collection of freight. The critical issues in defining the parking spot are freight detection, the vehicle's orientation, and defining parking spot reference to freight. These issues are further elaborated in their sub-sections below.

2.1. Freight Detection

The first step in defining the parking spot for the vehicle for loading freight autonomously is the detection of freight location. The freight vehicle is equipped with 3D LIDAR, which can identify and distinguish freight from the environment as the freight has a unique shape, size and color from the usual environment. Image feedback from an on-board camera is also used to distinguish freight from the environment. The design and dimensions of the vehicle freight box are given in Figure 1 (dimensions in mm), which also show the freight edges (highlighted in red) that need to be captured for correctly identifying the parking spot next to freight. Further dimensions for the freight box are available in [28].

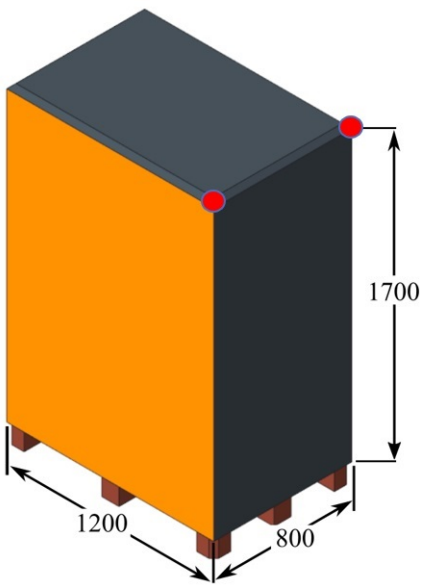


Figure 1. The design of FURBOT freight box.

The vehicle can load two freight boxes simultaneously, and because of the dimensions of the freight, it loads the freight from the depth edge of the freight (measuring 800 mm, as seen in Figure 1), as shown in Figure 2.

Figure 2. Loading of freight.

2.2. Orientation of Vehicle

The next consideration for loading is to park the vehicle next to the freight in such a manner that the vehicle's right side is facing the freight, as shown in Figure 3, since the loading mechanism can only operate from the right-hand side of the vehicle. The maximum tolerance for collection of freight between the freight and vehicle is 300 mm [29]. This limitation, along with the width of the vehicle, defines our parking spot's total width.

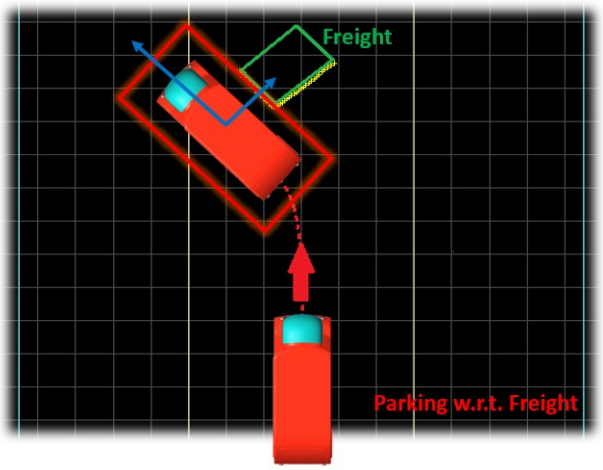


Figure 3. Parking w.r.t freight.

Furthermore, the vehicle is equipped with two separate loading bays, as shown in Figure 4. The freight has to be loaded in the vacant loading bay, i.e., if bay 1 is occupied, then load the freight in bay 2. As the loading bays are not centrally aligned with the freight, the vehicle has to adjust its parking spot accordingly so that the freight can be loaded in the available loading bay. Furthermore, the freight also needs to be aligned with the center of the forklift of the loading bays (as shown in Figure 4) for the forklifts to be inserted into the freight pallet.

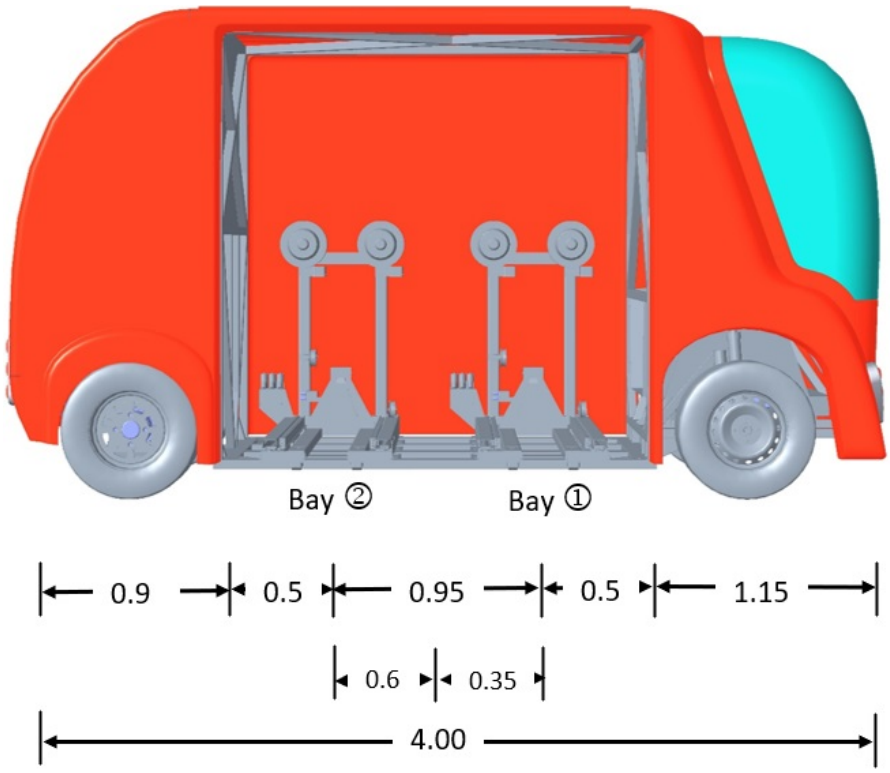


Figure 4. Vehicle dimensions w.r.t loading bays.

2.3. Defining Parking Spot Reference to Freight

The vehicle dimensions are given in Table 1. The width of the vehicle plus the tolerance to the freight defines the width of the parking space. To maintain a reasonable amount of parking maneuvers ($\leq$3 for typical cases) [30], the length of the parking space is chosen to be 2 m more than the length of the vehicle.

Table 1. Vehicle dimensions in meters.

Vehicle Dimensions (m)	Length	Width	Height
	4	1.5	2

Taking the loading of the freight and vehicle dimensions into consideration, the overall parking spot defined for the vehicle is a box of 2.1 $\times$ 6.0 m, as shown in Figure 3.

3. Software and Control Architecture

To speed up the development and integration process with the real experimental vehicle, Robot Operating System (ROS)-based autonomous vehicle's software architecture (called ICARS (Software being developed at LS2N (Laboratoire des Sciences du Numérique de Nantes) www.ls2n.fr (accessed on 15 July 2021)) will be exploited (Figure 5). Considering the task to be accomplished, a particular interest is placed on the Multi-Sensor-Based Predictive Controller (MSBPC) used for parking [31]. As the name of the approach suggests, the parking controller is based on a combination of Model Predictive Control (MPC) and Multi-Sensor-Based Control in order to perform safe (collision-free) parking operations that rely solely on locally perceived sensor features signals and without needing to explicitly plan any path. Moreover, since the technique exploits locally perceived sensor features at each time instant, no localization system is inherently required. The MSBPC approach is now recalled.

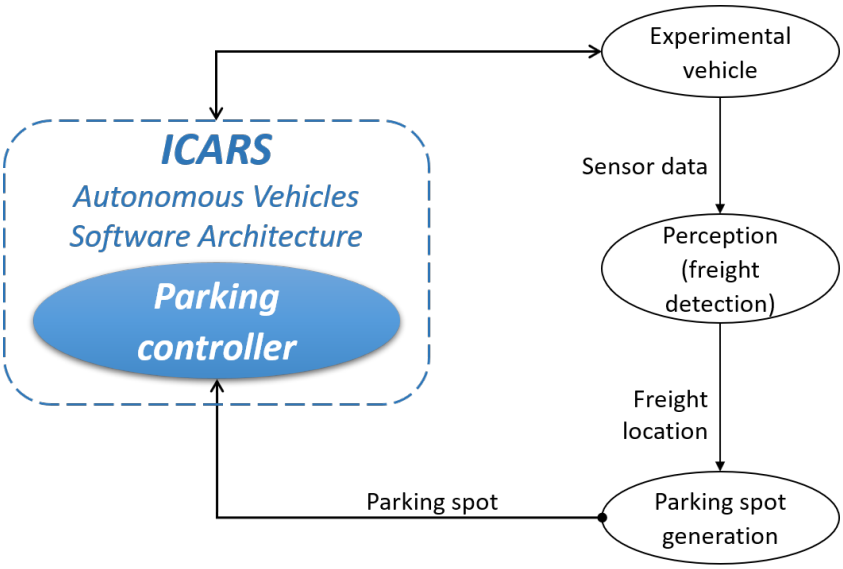


Figure 5. FURBOT software architecture.

An internal-model-control (IMC) structure [32] is used as a basis for formalizing the MSBPC approach (Figure 6). The robotized vehicle and perception system compose the System block. The input to this block is the control variable $\mathbf{v}_r = [v, \delta]^{\mathrm{T}}$, where the longitudinal velocity is denoted by v and the steering angle by δ, while its output is the current value of the sensor features (i.e., corners of the parking spot). The reference signal $\mathbf{s}^*$ is the desired value of the output $\mathbf{s}$. General discrepancies between the current sensor features and the values that were predicted from the model (e.g., modeling errors and disturbances) are represented by the error signal ϵ:

$$\epsilon(n) = \mathbf{s}(n) - \mathbf{s}_{\mathrm{mp}}(n) \tag{1}$$

where n denotes the current time.

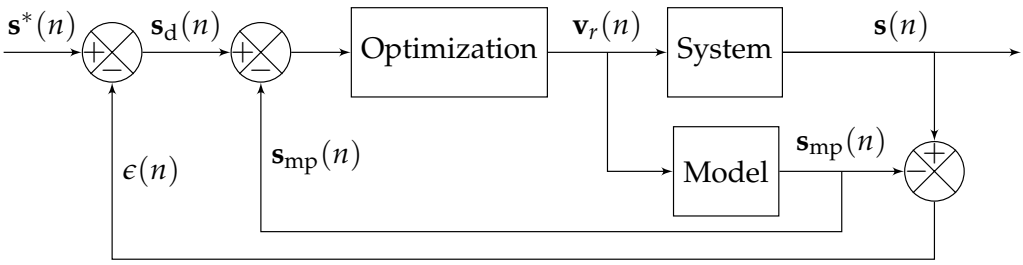


Figure 6. Control structure [31].

The difference between the desired value $\mathbf{s}_{\mathrm{d}}$ and the predicted model output $\mathbf{s}_{\mathrm{mp}}$ is minimized by an optimization algorithm. Following Figure 6:

$$\mathbf{s}_{\mathrm{d}}(n) = \mathbf{s}^*(n) - \epsilon(n) = \mathbf{s}^*(n) - (\mathbf{s}(n) - \mathbf{s}_{\mathrm{mp}}(n)), \tag{2}$$

from where one can deduce

$$\mathbf{s}_{\mathrm{d}}(n) - \mathbf{s}_{\mathrm{mp}}(n) = \mathbf{s}^*(n) - \mathbf{s}(n), \tag{3}$$

thus, to track $\mathbf{s}^*$ by $\mathbf{s}$ is equivalent to track $\mathbf{s}_{\mathrm{d}}$ by $\mathbf{s}_{\mathrm{mp}}$.

The interaction model described in [31] is used in order to predict the evolution of the sensor features $\mathbf{s}_{\mathrm{mp}}$ over a finite horizon N_p. The cost function is to be minimized with respect to a control sequence $\tilde{\mathbf{v}}_r$ over N_p and depends mainly on the difference between $\mathbf{s}_{\mathrm{d}}$

and $\mathbf{s}_{\mathrm{mp}}$. As with any classical Model Predictive Control technique, only the first element $\mathbf{v}_r(n)$ of the optimal control sequence is applied to the system at each iteration.

The controller (implemented in C++) runs online at 10 Hz using the solver NLopt [33] with a Sequential Least Squares Programming (SLSQP) algorithm [34]. Furthermore, it is assumed that both the vehicle's longitudinal velocity and steering angle are controllable; thus, lower-level controllers that directly interface with the actuators are out of the scope of the parking approach.

The simulation inputs are four points generated based on a parked pose and the type of parking spot. This information is then transferred as a topic to the control node, which, in turn, generates a control command as a topic as well. The complete software is constructed in a manner that it is agnostic; it is working in a simulation environment or in a real vehicle.

Therefore, with the experimental vehicle being interfaced with ICARS, two additional ROS nodes would have to be developed: A perception one to extract the freight's pose from sensory data and another one to generate a parking spot next to the freight. Once the parking spot has been successfully generated from the sensory data, the four corners that define it would have to be sent to the ICARS parking controller to park the vehicle in the desired pose to pick up the freight.

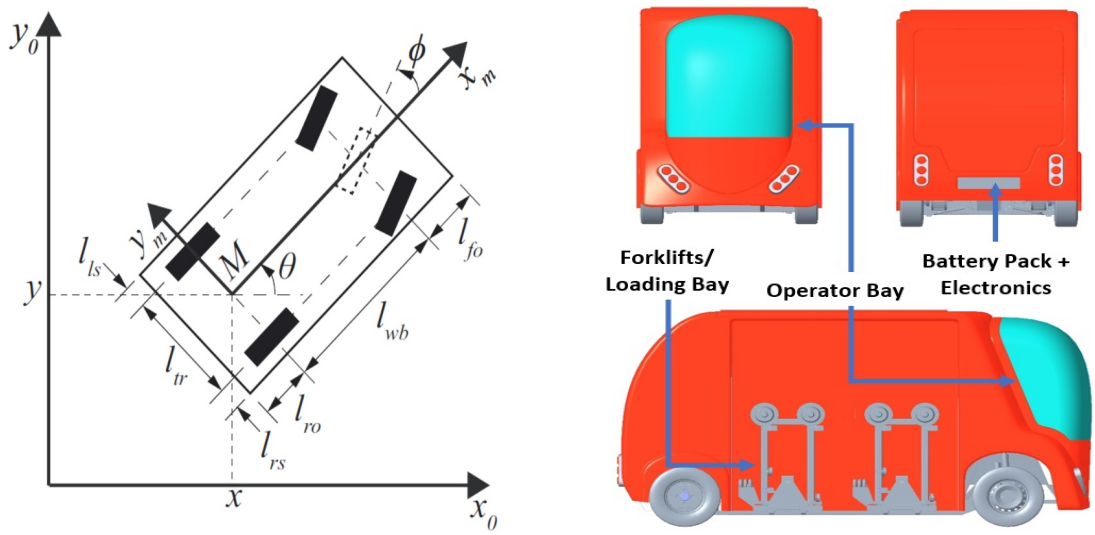
4. Mathematical Modeling and Notation

4.1. Vehicle Kinematic Model

The vehicle kinematic model for a rear-wheel driven vehicle as taken from [8] is presented by Equation (4). The freight vehicle also follows the same kinematic model for its drive.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ \tan\phi / l_{wb} \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \dot{\phi} \tag{4}$$

where v and $\dot{\phi}$ are longitudinal and steering velocities. Since parking maneuvers are performed at low speed, one can consider the kinematic model as accurate enough. Further notations and elaboration of the model as given by Equation (4) are represented in Figure 7a. The vehicle used for parking evaluation is further represented in Figure 7b. There is a designated space for the vehicle operator, as shown in the figure. The battery pack and electronics bay holds the complete electronics for the operation of the vehicle.



(**a**) (**b**)

Figure 7. (**a**) Vehicle kinematic model. (**b**) FURBOT vehicle.

4.2. Points Acquisition for Parking Spot

For correctly identifying parking spots for the vehicle, the acquisition of the freight corners (highlighted by red), as shown in Figure 1, is required through sensor feedback. From there, using these points, the freight center point is calculated using Equation (5).

$$x_{fc} = \frac{(x_{f1} + x_{f2})}{2}, y_{fc} = \frac{(y_{f1} + y_{f2})}{2} \tag{5}$$

Furthermore, knowledge from these two corner points is also used to calculate the inclination angle θ_f of the freight using the Pythagoras theorem (Equation (6)).

$$\theta_f = \tan^{-1}(\frac{y_{f2} - y_{f1}}{x_{f2} - x_{f1}}) \tag{6}$$

Using the width of the parking spot d_w and information of the freight center point, the vehicle center point is calculated from the freight using Equation (7).

$$x_{vc} = x_{fc} + \frac{d_w}{2}\cos(\theta_f + \frac{3\pi}{2}), y_{vc} = y_{fc} + \frac{d_w}{2}\sin(\theta_f + \frac{3\pi}{2}) \tag{7}$$

The above-mentioned points are further explained in Figure 8. Using these points, freight collection parking spots are calculated. This is further explained in the next subsection.

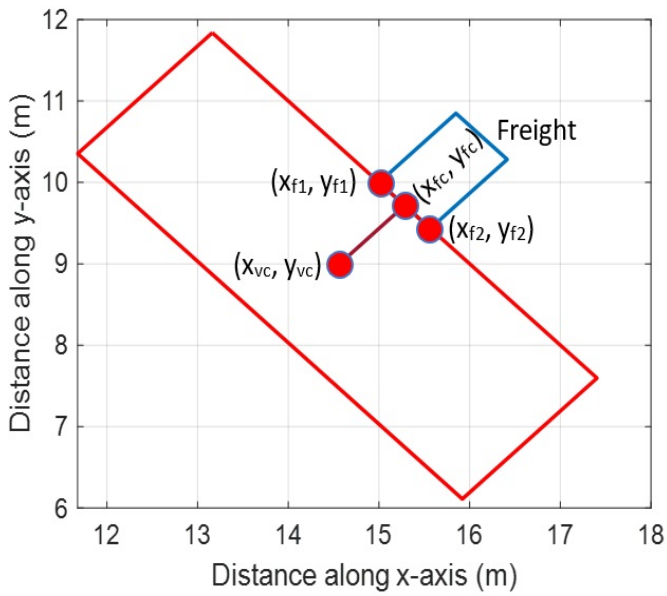


Figure 8. Point definition for parking spot.

4.3. Parking Area with Respect to Loading Bays

As the vehicle is equipped with two loading bays, the parking spot w.r.t each loading bay is different. To load the cargo in the front-loading bay, the vehicle needs to park a certain distance behind its center point, as shown in Figure 4. If the first loading bay is occupied by the previously loaded freight, then the vehicle needs to park a little ahead of its center point by a distance so that the forks are perfectly aligned with the freight to load the freight in the second loading bay. The center point of the parking spot thus varies depending upon the availability of the loading bay. This is further explained in Equation (8).

$$x_{pc} = \begin{cases} x_{vc} + d_{b1}\cos\theta_f, \\ x_{vc} - d_{b2}\cos\theta_f, \end{cases}, y_{pc} = \begin{cases} y_{vc} + d_{b1}\sin\theta_f, & \text{if bay1 = 1} \\ y_{vc} - d_{b2}\sin\theta_f, & \text{otherwise} \end{cases} \tag{8}$$

where d_{b1} and d_{b2} are the positive distances of loading bay 1 and 2 from vehicle center (measurement shown in Figure 4), and (x_{vc}, y_{vc}) is the center-point of parking spot. The condition $bay1 = 1$ denotes the condition of availability of loading bay 1. If bay 1 is available to load the freight, then the first condition applies or else the parking spot is generated w.r.t second loading bay.

After defining the center of the parking spot, the four corners of the parking spot can be defined through Equations (9)–(12) using the knowledge of parking width (d_w), parking length (d_l), angle of the freight (θ_f) and parking center-point (x_{pc}, y_{pc}). The collection of the first corner of the parking pose requires the knowledge of parking pose width + length. The derivation of equations is based on trigonometric relations and point transformation from the known parking center-point to the first edge of the parking pose. Since all four lines of the parking pose are parallel or perpendicular to the freight, the solution is formed by adding the respective angle $(90°, 180°, 270°)$ to the angle of the freight θ_f for calculation of next point of the parking pose.

$$\begin{aligned} x_{p1} &= x_{pc} + \tfrac{d_w}{2}\cos(\theta_f + \tfrac{\pi}{2}) + \tfrac{d_l}{2}\cos(\theta_f + \pi), \\ y_{p1} &= y_{pc} + \tfrac{d_w}{2}\sin(\theta_f + \tfrac{\pi}{2}) + \tfrac{d_l}{2}\sin(\theta_f + \pi) \end{aligned} \tag{9}$$

$$x_{p2} = x_{p1} + d_w \cos(\theta_f + \frac{3\pi}{2}), y_{p2} = y_{p1} + d_w \sin(\theta_f + \frac{3\pi}{2}) \tag{10}$$

$$x_{p3} = x_{p2} + d_l \cos(\theta_f), y_{p3} = y_{p2} + d_l \sin(\theta_f) \tag{11}$$

$$x_{p4} = x_{p3} + d_w \cos(\theta_f + \frac{\pi}{2}), y_{p4} = y_{p3} + d_w \sin(\theta_f + \frac{\pi}{2}) \tag{12}$$

The four corners of the parking pose $(x_{p1}, y_{p1}), (x_{p2}, y_{p2}), (x_{p3}, y_{p3})$ and (x_{p4}, y_{p4}) are the consequent four corners of the parking pose. These points are further explained later.

The summary of the whole pose generation solution is as follows. The vehicle is required to detect freight and acquire the corner points of the freight from the side where it can be loaded. Afterwards, Equation (5)–(12) are solved to get the correct pose for loading freight into the vehicle. The complete process of correct pose generation is further summarized in the flowchart presented in Figure 9.

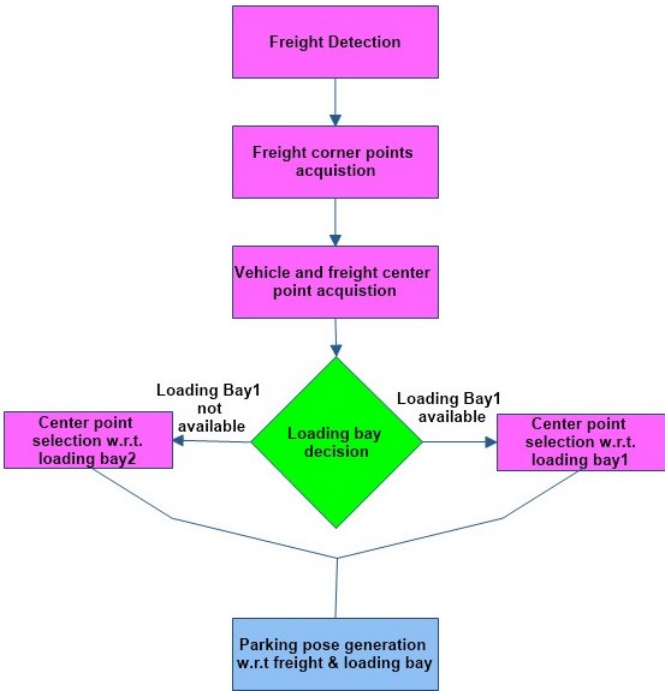


Figure 9. Flowchart for parking pose generation.

5. Results

By using the mathematical equations and notations discussed in the previous section, the node for the vehicle parking spot was run and tested to verify if the parking spot is generated according to the loading bays. The results are further discussed in the sections below for further clarity.

5.1. Parking Spot Definitions w.r.t Loading Bays

The results are first validated for generating the respective parking spot for front and rear loading bays. Figure 10 shows the output of parking pose generated for randomly positioned freight at a 45 degree angle. The blue-colored text represents the points for the parking spot definition concerning the front loading bay, whereas the red-colored text represents the parking spot concerning the rear loading bay. The red line from the parking spot center towards freight represents the right-hand side of the vehicle, thus specifying the heading of the vehicle.

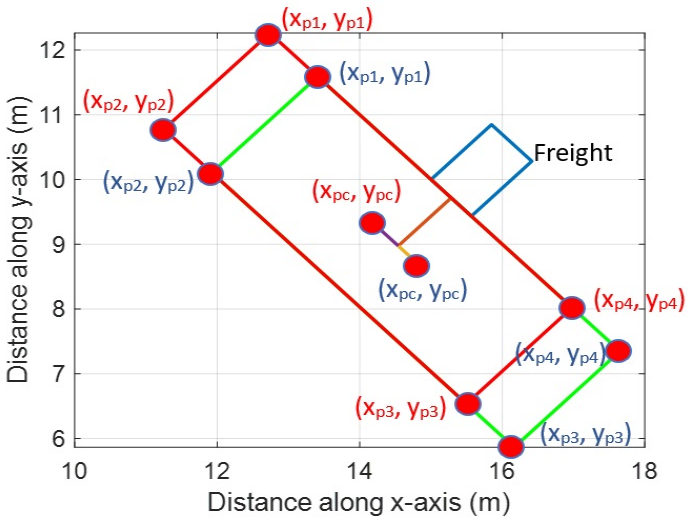


Figure 10. Parking spot with respect to loading bays.

From Figure 10, we can observe that the vehicle parks behind the center-point of the vehicle for loading freight in the first loading bay, whereas it parks a little ahead of its center-point when it is loading freight in the second loading bay. Furthermore, points 1 and 2 show the front end of the parking pose, and points 3 and 4 show the rear end of the pose generated. These results are also in coherence with the physical vehicle anatomy and the mathematical methodology proposed, henceforth validating our results.

5.2. Results for Different Freight Placement

As it is not possible to make sure that the orientation of freight is perfectly aligned with any pre-determined reference, the parking spot for the vehicle must be generated automatically, keeping in view the orientation of the freight. This conditional issue has been taken into consideration prior in the mathematical modeling of the parking spot. The code generated is validated for different orientations of the freight (Figure 11) to verify that the correct parking spot is generated irrespective of the orientation of the freight.

The results achieved, as shown in Figure 11, show four different orientations of freight (225, 0, 90 and 135 degrees, respectively) and the respective parking spot generated for each loading bay. The results show that irrespective of the orientation of the freight, the parking spot is generated accordingly for the vehicle for collecting the freight for the available loading bay.

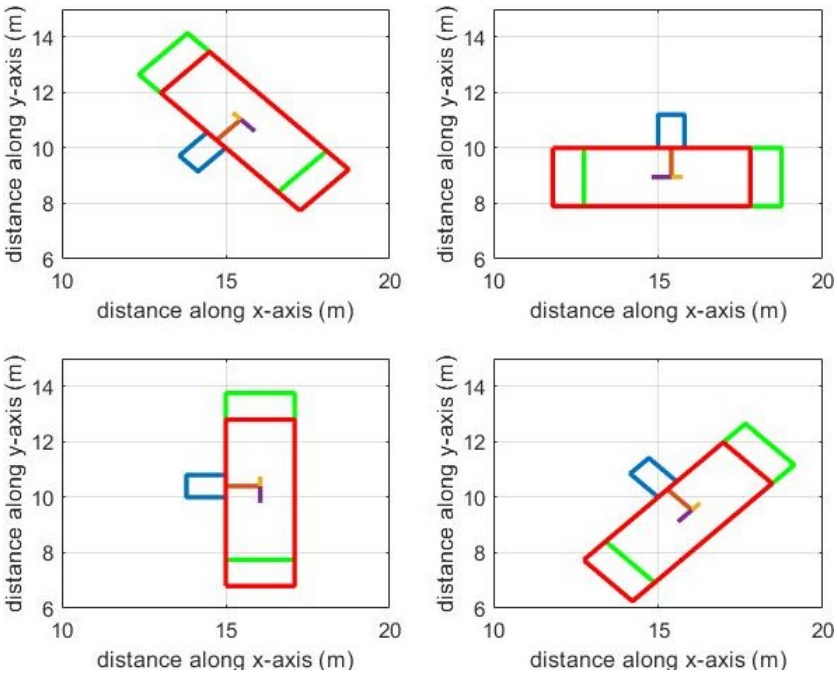


Figure 11. Parking spot relative to oriented loading bays.

Furthermore, the results presented also validate our purpose of generating parking pose autonomously irrespective of the orientation of the freight. The results also show that the vehicle will always park against the freight keeping the freight on the right-hand side of the vehicle so that it can be loaded in the available loading bay, i.e., the heading of the vehicle and loading bay location is always kept into consideration during the autonomous generation of a parking pose. The consideration of the freight w.r.t vehicle is shown by the red line generated from the center of the pose towards the freight (as shown in the figure with the red line). This allows the vehicle to know the direction of the parking pose and the heading angle for the vehicle. The parking pose points generated with the help of Equations (9)–(12) lets the vehicle know the heading of the vehicle as well. Point 1 and 2 depict the points where the front of the vehicle should face, and point 3 and 4 depict the points for the rear end of the vehicle.

5.3. Vehicle Parking in ROS Environment

As discussed earlier in Section 3, ROS environment-based software architecture ICARS is exploited to speed up the actual experiments. Using the already developed parking schemes within the software architecture [8,9,31], we can park the vehicle in the designated parking pose. The already built parking algorithms are sufficient for maneuvering the vehicle within the parking pose generated autonomously for the freight collection. Figure 12 shows the time-wise parking maneuver for vehicle FURBOT in the parking pose generated for the freight. Due to the constraint environment, the vehicle cannot park after detecting the freight because it needs to fulfill the requirement to park the vehicle, keeping the right side of the vehicle towards the freight. Thus, the vehicle has to move ahead of the freight and park while reversing, as shown in Figure 12. These results are generated in ICARS software architecture with input from the simulated sensors mounted on the vehicle.

In Figure 12, the orange box depicts the freight placement and location. The green rectangle represents the parking pose generated w.r.t freight. The red trail left behind the vehicle denotes the performed parking maneuver, and the red lines in front or at the back of the vehicle depict the direction and steering angle of the vehicle.

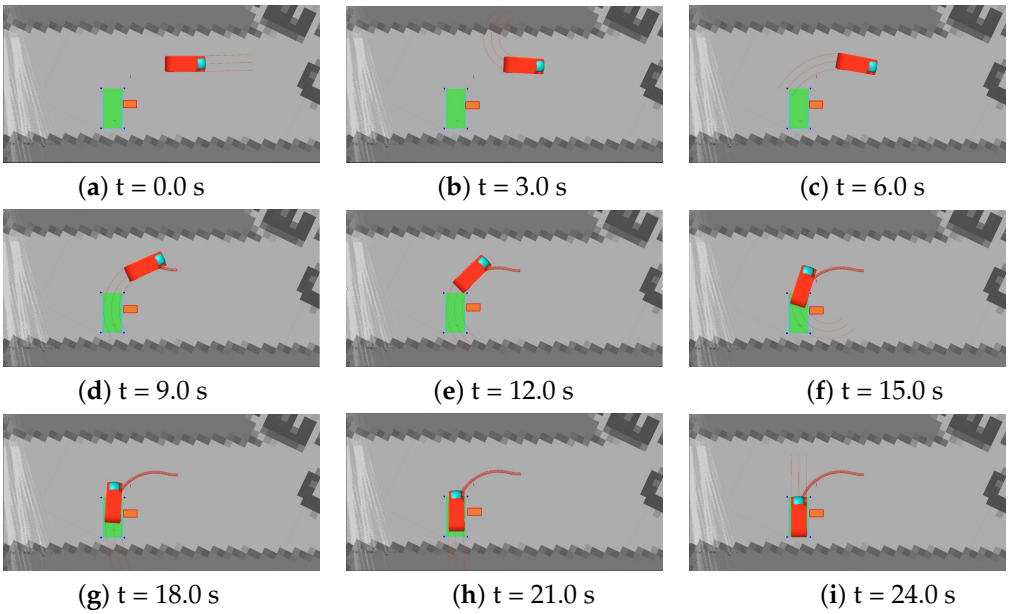


Figure 12. Vehicle parking w.r.t freight at different time intervals.

The virtual experiment is performed in the already installed maps of ICARS software architecture. The environment is constrained by natural buildings surrounding the area. Furthermore, the parking algorithm can detect and avoid obstacles, e.g., pedestrians, in case the vehicle can detect them [31]. The results validated in the ICARS environment prove that the methodology behind generating such parking pose for autonomous freight collection is valid and resolves the issue for autonomous parking pose generation w.r.t freight.

6. Conclusions and Perspectives

Research perspectives for this research were to highlight the gap between currently available research and how to resolve autonomous parking for freight collection. For addressing this issue, collecting the orientation and freight location are key values for the proposed algorithm. Using these key values, we can identify the parking pose for the vehicle for it to load the freight autonomously.

Major findings of this research discuss and resolve the issue related to parking the vehicle next to the freight for loading it into the vehicle. The hurdle of generating a parking spot that could lead to the successful loading of freight is looked into. Control architecture and vehicle dynamics were previously built and checked against different parking conditions, i.e., perpendicular, parallel and angled parking. This enabled the research to focus on the parking pose of the vehicle for freight collection. The solution required an algorithm that can define parking pose for our unique problem of loading the freight autonomously. With this research, this issue is resolved.

Concerning the correct pose for autonomous loading in respective bays of the vehicle, detection of freight from the surroundings is required. Once localization of freight is attained, then the proposed mathematical model suffices to generate the correct parking pose for the vehicle. However, the detection of freight and its localization is out of the scope of this research and will be looked into our future work. The parking pose length and width are defined, keeping minimal maneuvers required and distance to freight into consideration.

Considerations are kept for loading the freight in the correct loading bay. Separate parking spots are generated depending upon which loading bay is to be used for loading the freight into the vehicle. Simulating the freight with different orientations and generating the parking spot accordingly is also verified. The mathematical model verifies that if the freight is correctly identified from the environment, the parking spot generated will be correct irrespective of the orientation or location of the freight.

The main distinctive factor for the vehicle while demonstrating level 4 autonomy is the autonomous collection of freight which sets an electric freight vehicle apart from conventional vehicles. With the help of a previously designed parking controller, the identification of the freight through sensor's feedback and the definition of parking pose through this research, the vehicle can now align itself with the freight for autonomous collection of freight, increasing the level of autonomy of the vehicle.

This research will help to create an alternative approach for parking pose generation, especially in reference to an inanimate object. Researchers working in developing autonomous freight handling issues can directly benefit from this research. This research can further be used in developing parking pose algorithms where it might be necessary to park against an inanimate object, e.g., a bus stop or a door to a facility/delivery location. Currently, there is an unavailability of previous research in this field and solutions, where being studied might require continuous input from sensors in order to achieve the eventual goal where sensor output has to achieve an eventual goal to find a solution. This solution requires comparatively minimal input from sensors (for identification and localization); once attained, it does not require keeping on checking with sensors' feedback.

Future work involves the use of correct sensors (3D-Lidar in particular) for extracting information from the freight from the environment. This includes extracting data through 3D-Lidar's point-cloud data and reconstructing the freight in a virtual environment. Afterwards, the information from point-cloud data will extract the orientation and freight corner points. This perception module will help us extract the freight pose, which will later be used by the parking pose module for the autonomous generation of a parking spot regarding the freight.

Author Contributions: Conceptualization, K.M., V.F. and M.Z.; methodology, K.M., D.P.M. and V.F.; validation, M.Z. and V.F.; formal analysis, K.M. and D.P.M.; investigation, K.M. and V.F.; resources, M.Z. and V.F.; data curation, K.M.; writing—original draft preparation, K.M., D.P.M. and V.F.; writing—review and editing, K.M., D.P.M. and V.F.; visualization, K.M., D.P.M. and V.F.; supervision, R.M., V.F. and M.Z.; project administration, R.M., V.F. and M.Z.; and funding acquisition, R.M., V.F. and M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: This paper was supported by the European Union's Horizon 2020 research and innovation program under grant agreement No. 875530, project SHOW (SHared automation Operating models for Worldwide adoption).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

FURBOT	Freight Urban Robotic Vehicle
SHOW	SHared automation Operating models for Worldwide adoption

References

1. Freight Urban RoBOTic Vehicle. 8 August 2016. Available online: https://cordis.europa.eu/project/id/285055 (accessed on 19 March 2021).
2. FURBOT, EGVI. 9 January 2018. Available online: https://egvi.eu/research-project/furbot/ (accessed on 19 March 2021).

3. Masood, K.; Zoppi, M.; Molfino, R. Mathematical Modelling for Performance Evaluation Using Velocity Control for Semi-autonomous Vehicle. In Proceedings of the 15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020), Burgos, Spain, 16–18 September 2020; Herrero, Á., Cambra, C., Urda, D., Sedano, J., Quintián, H., Corchado, E., Eds.; Advances in Intelligent Systems and Computing; Springer: Cham, Switzerland, 2021; Volume 1268. [CrossRef]
4. Masood, K.; Molfino, R.; Zoppi, M. Simulated Sensor Based Strategies for Obstacle Avoidance Using Velocity Profiling for Autonomous Vehicle FURBOT. *Electronics* **2020**, *9*, 883. [CrossRef]
5. Masood, K.; Dauptain, X.; Zoppi, M.; Molfino, R. Hydraulic Pressure-Flow Rate Control of a Pallet Handling Robot for an Autonomous Freight Delivery Vehicle. *Electronics* **2020**, *9*, 1370. [CrossRef]
6. Masood, K.; Zoppi, M.; Fremont, V.; Molfino, R.M. From Drive-By-Wire to Autonomous Vehicle: Urban Freight Vehicle Perspectives. *Sustainability* **2021**, *13*, 1169. [CrossRef]
7. Clarembaux, L.G.; Pérez, J.; Gonzalez, D.; Nashashibi, F. Perception and Control Strategies for Autonomous Docking for Electric Freight Vehicles. *Transp. Res. Procedia* **2016**, *14*, 1516–1522. [CrossRef]
8. Morales, D.P.; Kermorgant, O.; Quijada, S.D.; Martinet, P. Laser-Based Control Law for Autonomous Parallel and Perpendicular Parking. In Proceedings of the 2018 Second IEEE International Conference on Robotic Computing (IRC), Laguna Hills, CA, USA, 31 January–2 February 2018. [CrossRef]
9. Pérez-Morales, D.; Kermorgant, O.; Domínguez-Quijada, S.; Martinet, P. Autonomous Perpendicular And Parallel Parking Using Multi-Sensor Based Control. In Proceedings of the 9th Workshop on Planning, Perception and Navigation for Intelligent Vehicles at IEEE/RSJ International Conference on Intelligent Robots and Systems, Vancouver, CO, Canada, 24–28 September 2017.
10. Golbabaei, F.; Yigitcanlar, T.; Bunker, J. The role of shared autonomous vehicle systems in delivering smart urban mobility: A systematic review of the literature. *Int. J. Sustain. Transp.* **2020**, 1–18. [CrossRef]
11. Amiri, A.M.; Ferguson, M.R.; Razavi, S. Adoption patterns of autonomous technologies in Logistics: evidence for Niagara Region. *Transp. Lett.* **2021**, 1–12. [CrossRef]
12. Willems, L. Understanding the Impacts of Autonomous Vehicles in Logistics. *Digit. Transform. Logist.* **2021**, 113–127. [CrossRef]
13. Reed, S.; Campbell, A.M.; Thomas, B.W. The Value of Autonomous Vehicles for Last-Mile Deliveries in Urban Environments. *Manag. Sci.* **2021**. [CrossRef]
14. Schlenther, T.; Martins-Turner, K.; Bischoff, J.F.; Nagel, K. Potential of Private Autonomous Vehicles for Parcel Delivery. *Transp. Res. Rec. J. Transp. Res. Board* **2020**, *2674*, 520–531. [CrossRef]
15. Pollaris, H.; Braekers, K.; Caris, A.; Janssens, G.K.; Limbourg, S. Vehicle routing problems with loading constraints: State-of-the-art and future directions. *OR Spectr.* **2014**, *37*, 297–330. [CrossRef]
16. Ostermeier, M.; Henke, T.; Hübner, A.; Wäscher, G. Multi-compartment vehicle routing problems: State-of-the-art, modeling framework and future directions. *Eur. J. Oper. Res.* **2021**, *292*, 799–817. [CrossRef]
17. Bortfeldt, A.; Homberger, J. Packing first, routing second—A heuristic for the vehicle routing and loading problem. *Comput. Oper. Res.* **2013**, *40*, 873–885. [CrossRef]
18. Research, H.A. Self-Parking Cars: Quick Guide, Car and Driver. 30 November 2020. Available online: https://www.caranddriver.com/research/a31995350/self-parking-cars-quick-guide/ (accessed on 20 March 2021). [CrossRef]
19. Kang, D.H.; Kang, C.M.; Kim, J.-S.; Kim, S.; Kim, W.-Y.; Lee, S.-H.; Chung, C.C. Vision-based autonomous indoor valet parking system. In Proceedings of the 2017 17th International Conference on Control, Automation and Systems (ICCAS), Jeju, Korea, 18–21 October 2017.
20. Shen, X.; Zhang, X.; Borrelli, F. Autonomous Parking of Vehicle Fleet in Tight Environments. In Proceedings of the 2020 American Control Conference (ACC), Denver, CO, USA, 1–3 July 2020.
21. Zhang, X.; Liniger, A.; Sakai, A.; Borrelli, F. Autonomous Parking Using Optimization-Based Collision Avoidance. In Proceedings of the 2018 IEEE Conference on Decision and Control (CDC), Miami, FL, USA, 17–19 December 2018.
22. Lee, B.; Wei, Y.; Guo, I.Y. Automatic parking of self-driving car based on Lidar. In Proceedings of the ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Wuhan, China, 18–22 September 2017; Volume XLII-2/W7, pp. 241–246.
23. Kusumakar, R. Autonomous Parking for Articulated Vehicles. Master's Thesis, HAN University of Applied Science, Arnhem, The Netherlands, 2017.
24. Jung, H.G.; Kim, D.S.; Yoon, P.J.; Kim, J. Parking Slot Markings Recognition for Automatic Parking Assist System. In Proceedings of the 2006 IEEE Intelligent Vehicles Symposium, Meguro-Ku, Japan, 13–15 June 2006.
25. Li, S.; Hai, Y. Estimating camera pose from H-pattern of parking lot. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, Alaska, 3–8 May 2010.
26. Du, X.; Tan, K.K. Autonomous Reverse Parking System Based on Robust Path Generation and Improved Sliding Mode Control. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 1225–1237.
27. Vorobieva, H.; Glaser, S.; Minoiu-Enache, N.; Mammar, S. Automatic parallel parking with geometric continuous-curvature path planning. In Proceedings of the 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, USA, 8–11 June 2014. [CrossRef]

28. Cepolina, E.M.; Farina, A. An optimization methodology for the consolidation of urban freight boxes. In Proceedings of the 15th International Conference on Harbor, Maritime and Multimodal Logistics Modeling and Simulation, Athens, Greece, 23–25 September 2013; pp. 46–52.
29. Muscolo, G.G.; Leonardo, L.M.; Pietronave, G.; Zoppi, M.; Molfino, R. Industrial solutions for loading/unloading goods on a full electrical freight urban robotic vehicle. *Int. J. Veh. Syst. Model. Test.* **2015**, *10*, 366.
30. Vorobieva, H.; Glaser, S.; Minoiu-Enache, N.; Mammar, S. Automatic parallel parking in tiny spots: Path planning and control. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 396–410. [CrossRef]
31. Pérez-Morales, D.; Kermorgant, O.; Domínguez-Quijada, S.; Martinet, P. Multi-Sensor-Based Predictive Control for Autonomous Parking in Presence of Pedestrians. In Proceedings of the ICARCV 2020 16th International Conference on Control, Automation, Robotics and Vision, Shenzhen, China, 13–15 December 2020. [CrossRef]
32. Morari, M.; Zafiriou, E. *Robust Process Control*; Prentice Hall: Englewood Cliffs, NJ, USA, 1989.
33. Johnson, S.G. The NLopt Nonlinear-Optimization Package. Available online: http://ab-initio.mit.edu/nlopt (accessed on 1 July 2021).
34. Kraft, D. *A Software Package for Sequential Quadratic Programming*; Wiss. Berichtswesen d. DFVLR: Koeln, Germany, 1988; Volume 88.

Chapter 8

Path Planning for Autonomous Approach to Freight

8.1 Introduction

This chapter discusses the autonomous approach to freight for loading the freight to the vehicle. The vehicle needs to develop an autonomous path to approach the freight for the necessary collection of the load. This chapter is done in collaboration with a Master's Robotics students working at the University of Genova, Italy, who is being supervised and his efforts are appreciated in creating this work.

Path planning for an approach to a target is a known research topic in robotics. Path planning algorithms are mainly divided into two main categories, classical algorithms and heuristic-based algorithms [1]. Classical algorithms are further classified into four major branches; cell decomposition methods, potential field methods, sampling-based methods, and sub-goal networks [1]. The most famous sampling-based methods for path planning algorithms are probabilistic roadmaps (PRM), and rapidly-exploring random trees (RRT) [2]. Both of these techniques are fundamental in their approach, i.e., the idea of connecting randomly sampled points, but they are very different in how the graphs are constructed connecting other points of the proposed path [3]. We selected RRT and RRT*, a derivative of the RRT approach for solving our path planning issue for ease of use. Afterward, bézier curves are fitted on the path planning algorithm for the approach to be more realistic for a four-wheeled mobile robot. All this is further explained in the sections to come.

8.2 Rapidly-exploring random tree (RRT and RRT*)

RRT-based path planning is considered for motion planning for the vehicle for the collection of freight and reaching the requisite parking pose. The rapidly-exploring random tree is designed for problems that have non-holonomic constraints. RRT has some features and properties as a probabilistic roadmap, leading to better performance analysis and consistency of behavior. RRT is a motion planning algorithm based on random sampling. The whole configuration space or the map is rapidly explored by expanding branches from the neighboring node to the random target.

The main disadvantage of using the RRT technique is that due to the randomness of the technique, the proposed solution can be a zigzag solution; thus, it will not be optimal. A tweak in the RRT technique resulting in the RRT* algorithm is more optimal to resolve this issue. RRT* basically tries to smooth the tree branches produced in each step of the RRT

technique. Thus yielding to a much smoother path. In RRT* algorithm, the distance of each vertex is calculated and recorded relative to the parent vertex. After finding the closest node in the path, the neighborhood from the new node is examined with a fixed radius. If a node with a cheaper cost than the proximal node is found, then the proximal node replaces the more inferior node. In this way, we get the shortest path from start to endpoint. This helps in creating a more smooth path for the vehicle to follow. Both (RRT and RRT*) techniques are applied for our mobile robot to get to the freight while avoiding obstacles.

The optimal solution that needs to be selected between RRT and RRT* technique is still under discussion as both methods have their unique advantages. At first, it seems as RRT* technique outperforms the RRT technique. But for formulating a bézier curve on top of the proposed path, it is preferred by the curve fitting method that more points lead to a better vehicle orientation and sometimes a smoother curve. This feature is further discussed in the results presented.

8.2.1 Metric function for RRT and RRT*

While choosing the nearest neighbor q(nearest) as the best input variable, the RRT adopts the nearest neighbor accomplished by the metric function. There are several metric functions used for 2D configuration space. Some of the metric functions, e.g., the euclidean distance method, Manhattan distance metric function, and diagonal distance metric function as taken from reference [4] are given in equations below.

$$
\begin{aligned}
&d_{Euclidean}(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2} \\
&d_{Manhattan}(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| \\
&d_{Diagonal}(i,j) = max(|x_{i1} - x_{j1}|, |x_{i2} - x_{j2}|) + (\sqrt{2} - 1)min(|x_{i1} - x_{j1}|, |x_{i2} - x_{j2}|)
\end{aligned} \tag{8.1}
$$

For ease of use, we have used the Euclidean distance method to calculate the metric function to find the distance between the q(rand) and q(nearest). With the help of the minimum distance to the q(rand), the new q(nearest) is taken into the RRT tree. The complete pseudo-code implemented for our approach is taken from reference [4] and the code implemented uses the same technique as presented in this reference.

8.2.2 Minkowski sum for obstacle avoidance

Minkowski sum is a famous technique used in mathematics for summing two sets of vectors in Euclidean space. We have used this method to avoid any contact with the obstacle from any angle for our collision avoidance scheme. This works because the polygon P, representing vehicle, is placed on the edges of the obstacle polygon Q, and by adding them, we get the vertex of the Minkowski sum of two polygons. This puts a convex hull over the vertex points received from the Minkowski sum. Thus we get the new polygon, and this new polygon is used by the RRT and RRT* algorithm as an object for avoiding a collision. Graphically Minkowski sum can be expressed in the reference [5]. Mathematically, the Minkowski sum is represented by the following equation

$$P + Q = \{\vec{x} + \vec{y} \mid \vec{x} \in P, \vec{y} \in Q\} \tag{8.2}$$

8.3 Results from RRT and RRT* path planning approach

RRT is being used for both online and offline planners, but we have considered offline planners as we have a known map with the obstacle. We consider a map grid from x(0,20) and y(0,20), and defining start point for the vehicle is at (2,2) and goal position, i.e., parking pose location for collection of freight at (19,13) for vehicle path planning. The path and RRT Tree is generated avoiding the set of all obstacle. The obstacle placed is at (10,1) with a height and width of 6 meters each.

RRT is relatively straightforward; the points are randomly generated and connected to the nearest node; in our map, the nodes that are generated randomly are marked as 'x'. As shown in the results, the obstacle is placed in a configuration space indicated by a green box. Every time a new vertex is created, the vertex is checked for no collision with the obstacle coordinates, i.e., check if the vertex lies inside the obstacle. The function no_collision returns 0 or 1 where the 0 is determined no collision and 1 if the vertex lies inside the obstacle, which means a tree that would be newly formed between the vertex and parent vertex would colloid with the obstacle and hence vertex lying inside the obstacle is eliminated.

If no collision is determined, a vertex is added to the RRT tree; if it is determined the vertex lies within the bounds of the obstacle, the tree is not created, and a new vertex is made, and the process continues until the start and goal points are connected via path. The algorithm ends when the tree is built between the start and the goal.

Once the tree is generated between the start and goal points, the path is created by connecting every node that connects the start and goals points. This path is thus highlighted in red and hence defines the path trajectory of the vehicle, which the vehicle needs to follow to reach the goal position for its respective freight collection. The total distance the vehicle needs to travel to reach the goal point is calculated based on the vehicle's travel trajectory. Figure 8.1 shows a typical result from the proposed map grid highlighted in the first para.

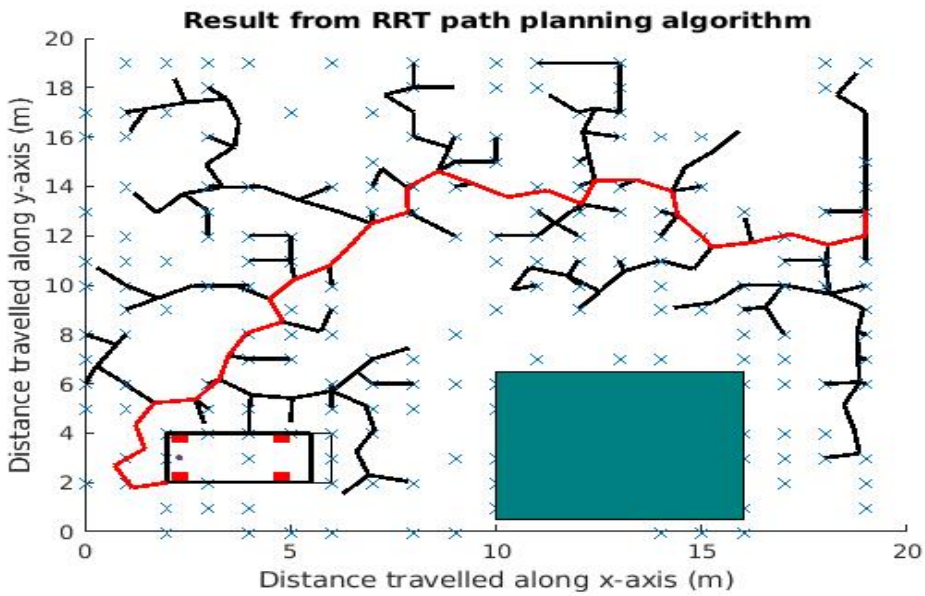


Figure 8.1: Result from RRT path planning approach

As discussed earlier, RRT* is an optimized version of RRT or advancement of the RRT technique. In RRT*, when the number of nodes reaches infinity, the algorithm will deliver the shortest possible path to the goal. The results from the same defined start and goal position and the same configuration map (as was for RRT) show that RRT* found the shortest path to goal compared to RRT. The result achieved from RRT* approach is given in the figure 8.2.

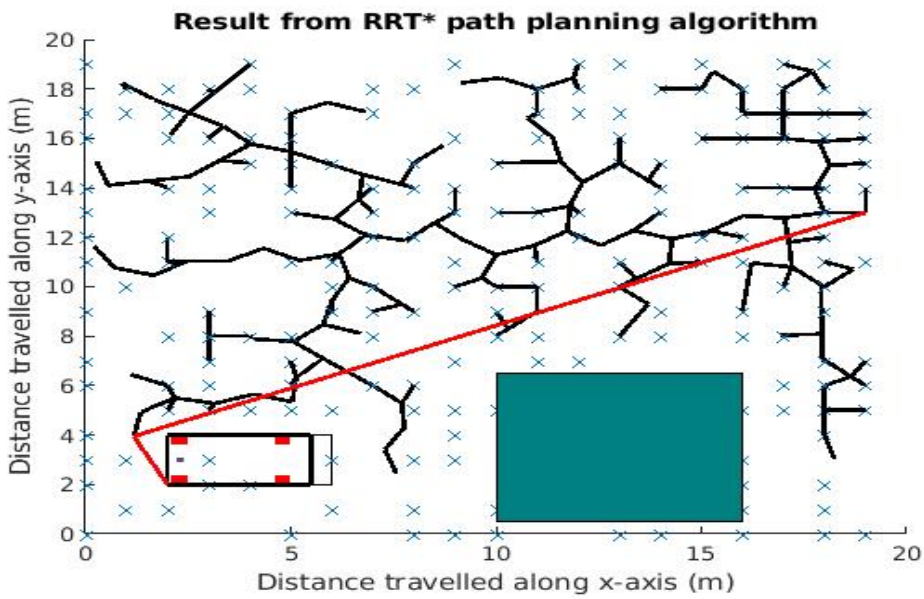


Figure 8.2: Result from RRT* path planning approach

8.4 Bézier curves for mobile robots

Bézier curves are extensively used in path planning for mobile robots as they usually propose a polynomial-based path which can be easily followed by mobile robots [6, 7]. With the help of RRT and RRT* techniques, we were already able to get a path to the goal, which can avoid obstacles using the Minkowski sum. However, the path obtained from the RRT and RRT* techniques is not smooth for a vehicle.

A Path is considered to be comfortable when max curvature value and curvature variations are minimized. Hence the curvature is inversely proportional to angular acceleration and is linked to jerk, a discomfort factor, especially for a vehicle. To eliminate a sharp turn and discomfort factor, splines or curves are introduced in the sharp turns or paths that we have received from the RRT algorithm. This is achieved through Bézier curves for our solution. The mathematical basis for Bézier curves is based on the Bernstein polynomial. The n+1 Bernstein basis polynomials of degree n are defined as

$$B(t) = \binom{n}{i} t^i (1-t)^{n-1} \; where \, t \, varies \; from \; 0 \; to \; 1 \tag{8.3}$$

The Bernstein polynomial is based upon the number of points on which the Bézier curve needs to be fitted. Thus, it can not be predefined as the points to goal vary every time RRT and RRT* finds their relative path. The trajectory points received from the path planning algorithm are fed to the Bézier curve function, which then calculates the Bernstein polynomial and finally outputs x and y points of the Bézier curve which we can use as the final path to the goal for freight collection by the vehicle.

8.5 Application of Bézier curves on RRT and RRT*

The approach was finally applied to the results obtained in section 8.2.2. on RRT and RRT* paths to check whether Bézier curves fitted the proposed path smoothly. The results obtained are shown in figures 8.3 and 8.4. The blue line in the figures denote the Bézier curve fitted on the path, the red line shows the path obtained by either RRT or RRT* path planning approach, and the green box denotes the obstacle avoided by the vehicle to reach its goal.

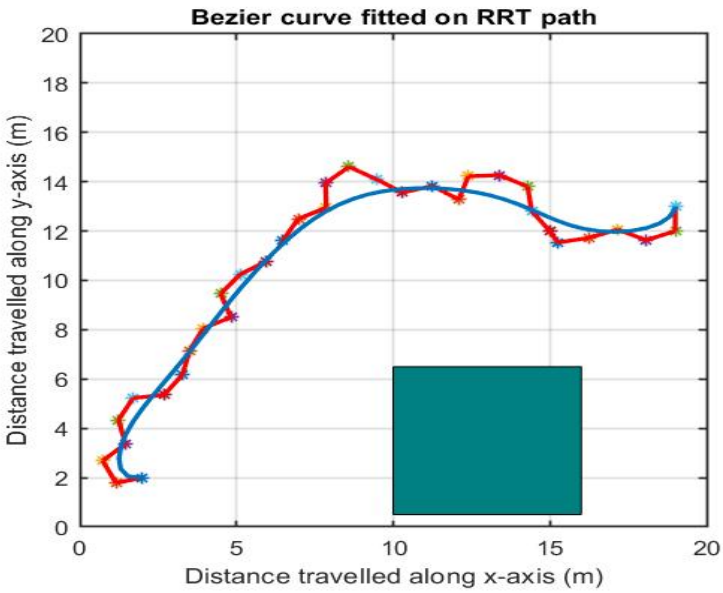


Figure 8.3: Bézier curve fitted on RRT path

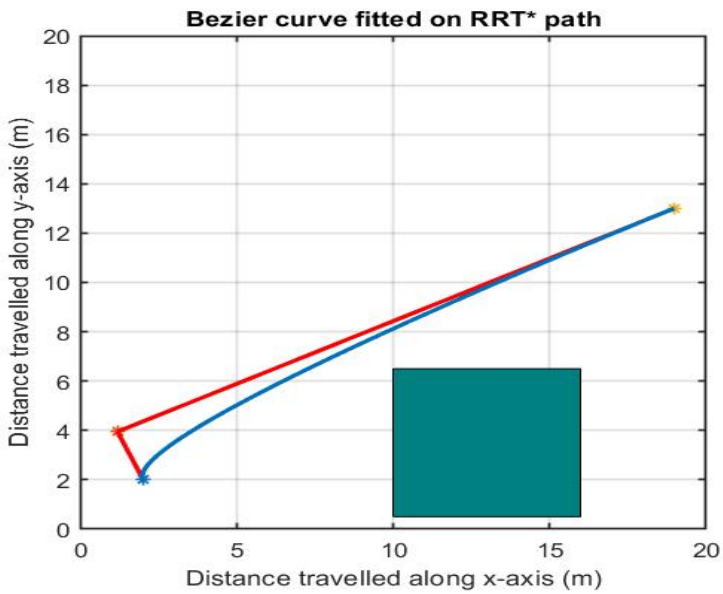


Figure 8.4: Bézier curve fitted on RRT* path

The results depict the successful implementation of Bézier curves on the path proposed by either of the algorithms. The obstacle has also been successfully avoided for the freight approach; however, vehicle orientation at the end of the proposed path still needs to be resolved for the vehicle to successfully load the freight. For this, the end leg of the path needs to satisfy not only the location constraint but also the orientation constraint for the successful collection of freight. This final issue related to the collection of freight is still

being investigated for the completion of this task.

8.6 Conclusion

Path planning algorithms developed through the use of RRT and RRT* algorithms are further approximated using Bézier curves so that our mobile robot can follow that course for autonomous freight collection. However, the issue of reaching the final orientation is still yet to be addressed and is currently being investigated. Yet, the progress made so far is quite promising in attaining the goal of autonomous freight collection, considering the previous work done and current milestones achieved.

8.7 References

Bibliography

[1] T. T. Mac, C. Copot, D. T. Tran, and R. De Keyser, "Heuristic approaches in robot path planning: A survey," Robotics and Autonomous Systems, vol. 86, pp. 13–28, 2016.

[2] J. Lee, O. S. Kwon, L. Zhang, and S.-E. Yoon, "A selective retraction-based rrt planner for various environments," IEEE Transactions on Robotics, vol. 30, no. 4, pp. 1002–1011, 2014.

[3] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning with deterministic μ-calculus specifications," 2012 American Control Conference (ACC), 2012.

[4] J. Li, S. Liu, B. Zhang, and X. Zhao, "Rrt-a* motion planning algorithm for nonholonomic mobile robot," 2014 Proceedings of the SICE Annual Conference (SICE), 2014.

[5] H. Barki, F. Denis, and F. Dupont, "A new algorithm for the computation of the Minkowski difference of convex polyhedra," 2010 Shape Modeling International Conference, 2010.

[6] Jung-Hoon Hwang, R. C. Arkin, and Dong-Soo Kwon, "Mobile robots at YOUR fingertip: Bezier curve on-line trajectory generation FOR supervisory control," Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453), 2003.

[7] G. Kavuran, "Path planning for nonholonomic mobile robot based on BÉZIER CURVE," 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), 2017.

Chapter 9

Conclusion and Perspectives

9.1 Summary of thesis

The software architecture used within the development of the thesis includes extensive use of MATLAB with Simulink environment. The results obtained from the software architecture development are later utilized in ROS (Robot Operating System) environment for developing nodes to perform/calculate specific tasks with the help of Python language. Additionally, for understanding and designing CAD models and to create 3D parts for the vehicle, computer-aided drawing software CREO is used.

In this thesis, we worked on multiple aspects of our urban freight robotic vehicle FURBOT for bringing it on the road as an autonomous vehicle instead of the original drive-by-wire state of the vehicle. Immense effort is placed on upgrading the research for freight handling vehicle-based robots, and techniques developed within the thesis are such that they can be applied to any freight handling autonomous vehicle. The thesis primarily focuses on two significant aspects of the vehicle on which the work is formulated.

1. Upgrading of autonomous navigation plus autonomous freight handling capabilities.

2. Technical evaluation and management for upgrading FURBOT into an autonomous vehicle.

Multiple vehicular aspects are looked into to upgrade the autonomous navigation and freight handling capabilities for the vehicle. For addressing the vehicle's autonomous navigation, a mathematical model of the vehicle was built with velocity and steering control to be used in obstacle avoidance schemes built for the vehicle. For independent freight handling, parking pose was defined concerning detected freight to load the cargo into the vehicle correctly. The path planning to the parking pose was also studied using RRT and RRT* techniques, later adapted with bezier curves for the vehicle to follow. Furthermore, a pallet handling robot installed within the vehicle was upgraded for upgrading the freight capabilities in itself.

Technical management for the vehicle included addressing and performing qualitative analysis on the handover strategies for the autonomous vehicle in case of vehicular system failure. Additionally, a comprehensive study in chapter 5 studies further technical aspects of the vehicle, including its amalgamation in the urban environment, including its licensing issues. Also, an in-depth technical analysis was performed to know the vehicle's current state and what it will take to bring it on the roads as an autonomous vehicle. This included an in-depth study of software and hardware requirements of the vehicle, plus knowing the use cases for the vehicle to evaluate its performance as an autonomous vehicle working in an

urban environment.

The thesis starts by looking at the pre-installed pallet handling robot and addressing the issues of its upgrading. Afterward, the thesis focuses on mathematical modeling of the vehicle for autonomous navigation, followed by devising schemes for obstacle avoidance for slow-moving vehicles based on vehicle velocity control. Then the thesis focuses on the technical management of the thesis focusing on its licensing issues, technical upgrading issues, including software and hardware upgrade, and finally moves on to analysis on handover strategies. Afterward, the thesis focuses on autonomous development of parking spots for freight collection of the vehicle and then finally concluding the thesis with developing schemes for an approach to the freight. Most of the work carried out for this thesis is carried out and validated in a simulated environment.

The conclusive summary of a thesis includes the development of methodologies and strategies for the freight-based robotic vehicle which were not previously present for the system. With the help of the basis provided within this thesis in their specific chapters, upgrading the vehicle towards an autonomous vehicle capable of handling freight autonomously will be more effortless and smooth. The issues addressed within this thesis were necessary for upgrading the vehicle to an autonomous vehicle that can efficiently perform its designated specific task of last-mile delivery. Particular focus is placed within this thesis to keep the vehicle's integrity as freight handling robotic vehicle intact.

9.2 Perspectives

The way forward to achieving an autonomous freight-based vehicle presented in this thesis is not complete. Thus the work presented in this thesis can be considered as work still in progress. We proposed multiple aspects of the freight-based vehicle, discussing how to proceed further to automate the discussed mobile robot. Thus this work is compiled on a different aspect of the said vehicle. Within each chapter, an effort is made to complete a task for the vehicle, which can create a further way forward for the development and enhancement of the vehicle.

In this thesis, we devised methodology and mechanisms for autonomous vehicle navigation and autonomous freight handling capabilities. We are also doing an in-depth analysis in the domain of vehicle technical management. This work can also be considered in the direction of the development of the vehicle's digital twin. In this document, we mainly presented three main ideas on which the work was carried out. It is for certain that as time progresses, new ideas have constantly been generated to complete the vehicle's journey to an autonomous vehicle. The perspective on these ideas is discussed below.

Autonomous Navigation: In the research carried out in this thesis; we focused our work on autonomous navigation of the vehicle. We initially designed the digital twin for the vehicle with the help of the mathematical model proposed. This was also infused with steering and velocity control of the vehicle. Later on, we modeled four different obstacles and developed techniques for avoiding collision or adjusting speed when engaging obstacles using velocity control for the vehicle.

Autonomous Freight Handling: Autonomously handling freight for the vehicle was one of the main focuses of this thesis. We placed special focus on this topic and performed research in multiple aspects to handle this issue. The highlights of our research in this direction are:

- Upgrading the already pallet handling robot installed on the vehicle for loading/unloading the freight into/from the vehicle.
- Generating parking spot/pose for the vehicle next to the freight using information gathered through sensors to collect the freight autonomously.
- Developing path planning algorithms (RRT and RRT* algorithms) that can avoid obstacles and reach the goal (parking spot) for freight collection.
- Smoothening the path received from RRT and RRT* algorithms using Bézier curves so the mobile robot can easily follow the course.

All the work mentioned above was completed/partially completed to increase the vehicle's autonomous freight handling capabilities.

Technical Management of Vehicle: To address the technical management of the vehicle. Two main issues were highlighted during this research. One on how to manage handover techniques in case of vehicular failure. For this, we performed qualitative analysis on the handover techniques which are currently being applied and tempered them into the needs of our vehicle.

Another issue was on how to convert the vehicle from a drive-by-wire vehicle into an autonomous one. For addressing this issue, numerous aspects were looked into. In which the current state of the vehicle was discussed, the expectations and requirements of the vehicle from an autonomous vehicle perspective were looked into. Use cases were studied, plus the terrain in which the vehicle needed to perform was also explored. Most importantly, licensing issue for the vehicle was addressed as the vehicle is unique; it requires a proper license to be able to drive on European roads. Finally, we addressed the hardware and software requirements that the vehicle needs for it to be able to convert itself from drive-by-wire to an autonomous vehicle.

9.3 Future research

As discussed in the perspectives section of this chapter, the work on the vehicle is still ongoing. Thus, a lot of work can be done on the vehicle for its final amalgamation in the urban society. Therefore it is safe to say that several aspects of the project conducted can be improved. Some of these aspects are presented for the vehicle, which can further enhance the vehicle's autonomy.

At first, if we look at the autonomous navigation side of the vehicle, so far in this research, the controllers used are PID controllers, and not a comprehensive study is performed on the selection of the suitable controller. For example, model predictive controllers (MPC) tend to be more favorites for autonomous vehicles. Furthermore, the model built does consider the steering control. Still, a proper path planning algorithm that favors real-life scenarios can

enhance the conceptualization of the vehicle trajectory, which is currently missing in this research. Moreover, with the help of the actual sensor's feedback, e.g., from 3D Lidar, we can construct real scenarios and experiment on the vehicle's autonomous navigation.

Immerse work is already performed on autonomous freight handling capabilities of the vehicle, yet experimental testing on the proposed algorithms is still missing. A lot of work can be done to detect the freight from the environment using Lidar or camera-based data feedback. Correct detection of cargo is crucial in creating the path to the freight for loading freight autonomously. Furthermore, the work presented in this research can also be carried forward for conclusive results on path planning and path following of the vehicle. Experimental evidence on the proposed techniques is also a way forward in this research.

The technical management side of the vehicle is exceptionally vast. Finding the solutions for the vehicle is one part of the research, whereas solving those issues and presenting the vehicle in an urban environmental setting is work in its domain. Applying the proposed techniques to the vehicle will yield additional problems and thus a new research direction.

Continuation of this research will yield more research horizons as it will unfold new problems requiring new innovative solutions. Finally, the list of proposed future research is not exhaustive. There is much more margin of improvement, especially in freight handling robotic vehicles, as this side of research is still comparatively new.