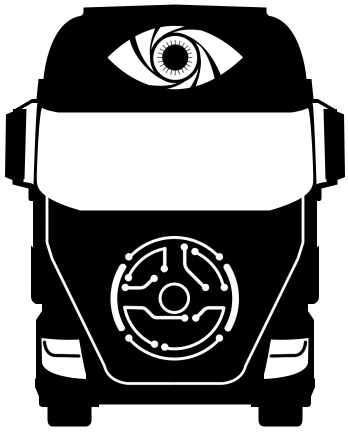
Requirement Specification

Autonomous Truck With a Trailer

November 17, 2020

Version 2.1



Status

Reviewed	Filip Jussila	2020-11-17
Approved	Carl Hynén	2020-11-17

Project Identity

Group E-mail: vituv953@student.liu.se

Homepage: https://tsrt10.gitlab-pages.liu.se/2020/rev_truck

Orderer: Carl Hynén, Linköping University
Phone: +46 73-562 34 34
E-mail: carl.hynen@liu.se

Supervisor: Daniel Arnström, Linköping University
Phone: +46 76-831 24 09
E-mail: daniel.arnstrom@liu.se

Customer / Course Responsible: Daniel Axehill, Linköping University
Phone: +46 13-28 40 42
E-mail: daniel.axehill@liu.se

Participants of the group

Name	Responsibility	E-mail
Per Antonsson		peran080@student.liu.se
Edvin Bourelius		edvbo949@student.liu.se
Gustav Erbing		guser450@student.liu.se
Johan Gustafsson		johgu927@student.liu.se
Anton Holgersson	Information Manager (IM)	antho223@student.liu.se
Ofa Ismail	Software Architect (SW)	abdis077@student.liu.se
Filip Jussila	Document Manager (DM)	filju425@student.liu.se
Per Liljeström	Test Leader (TL)	perli397@student.liu.se
Katherine Rajala		katra756@student.liu.se
David Salomonsson	Design Manager (DeM)	davsa358@student.liu.se
Viktor Uvesten	Project Leader (PL)	vituv953@student.liu.se

CONTENTS

DOCUMENT HISTORY

Version	Date	Changes made	Made by	Reviewer
0.1	2020-09-17	First draft	Project Group	Ofa Ismail
0.2	2020-09-21	Modified and added requirements	Project Group	Filip Jussila
1.0	2020-10-01	Small changes made on behalf of the orderer	Project Group	Filip Jussila
2.0	2020-11-09	Final draft	Project Group	Filip Jussila
2.1	2020-11-17	Revised requirement 14 to priority 2	Project Group	Per Liljeström

1 INTRODUCTION

This is the requirement specification for the project *Autonomous Truck With a Trailer* in the automatic control project course TSRT10 given at Linköping University during the fall of 2020.

The project regarding autonomous vehicles has been performed at Linköping University for a few years. Each project being a continuation of the previous. This year's main goal is to improve the stability and robustness of the automatic control system by implementing a Model Predictive Control (MPC) controller. A Raspberry Pi (RPi) will also be integrated to the already existing truck to replace the need of an external computer during run-time. The purpose of this document is to specify the requirements for the final results of the project.

1.1 Requirement description

The requirements will have the format shown in Table 1. Each requirement will either be an original requirement with the number **X** or a revised version **Xa** which is renegotiated in version x.x of this document. Furthermore, each requirement will have a description and a priority; 1,2 or 3 – given in descending order. Requirements of priority 1 should be fulfilled at the final delivery of the project. Requirements of priority 2 and 3 will be fulfilled if there is any time left when all requirements of priority 1 is fulfilled.

Table 1: Example of how the requirements are formatted.

Requirement	Revising	Description	Priority
X	Original	Description of requirement X.	1/2/3
Xa	Renegotiated: version x.x	Description of revision **a** of requirement X.	1/2/3

1.2 Partners

The following parties are involved in this year's project:

- The project group, which consists of eleven students studying applied physics and electrical engineering or mechanical engineering at Linköping University.
- The orderer Carl Hynén, department of automatic control at Linköping University.
- The customer and examiner Daniel Axehill, department of automatic control at Linköping University.
- The advisor Daniel Arnström, department of automatic control at Linköping University.

1.3 Definition of terms

Below are some terms and abbreviations defined and explained.

- **APPAS** – Autonomous Path Planning & Parking Assistance System, the name of the product developed during the project.
- **TG** – Tollgate.
- **Git** – Distributed version control system used for software.
- **MPC** – Model Predictive Control, optimization based control method used to control a process while satisfying a set of constrains.
- **QualiSys** – Positioning system used in Visionen.
- **ROS** – Robotic Operative System, an open source software library which simplifies writing modular code for robotic applications.
- **RPi** – Raspberry Pi, a small single-board computer.
- **Visionen** – A research arena for robotics applications at Linköping University.

1.4 Aims and goals

The long term goal of the this project is to create a platform for research and teaching on autonomous vehicles at Linköping University. Last year, the focus was to autonomously drive the truck backwards to a loading bay and improve the motion planning by implementing a more efficient search heuristic and including a post optimization step of the motion planning.

This year, the aim is to improve the stability and robustness of the system by implementing an MPC controller and make the use of an external computer obsolete by integrating an RPi to the already existing system. Another aim is to improve the visualization of the system using the projector in Visionen. Bug fixes of last year's implementation are also planned along with a simple initial calibration phase of the steering angle of the front wheels that will be examined and possibly implemented as well.

1.5 Use

The final product will be used in both research and teaching courses in advanced control theory at Linköping University as well as marketing and recruiting for the university.

1.6 Background information

Autonomy is a growing subject in our society, therefore are autonomous vehicles of great interest for the industry and thus a relevant research area [1]. A difficult task that truckers face daily is reversing their truck with a trailer, this tricky and arduous maneuver can be simplified through a driver assistance system such as APPAS. APPAS is not only used for assisting in reversing but also with general path planning, as the name implies.

2 SYSTEM OVERVIEW

The main system of last year consists of a LEGO truck with a trailer, a standard laptop as a computing unit and a loading bay with included trailer tracking system developed and provided by last year's project group [2]. This year's main system is a continuation of last year's main system but the controller used will be an MPC controller, the computing unit will be an RPi and the loading bay will not be used. Another parallel main system this year will be a visualization system that uses the existing positioning system installed in Visionen and its projectors to visualize obstacles and the truck's planned path.

The goal of the system is to have a truck move between two points autonomously. The truck should be able to drive the path both forward and reversed while avoiding obstacles from a predefined map. An example of this is shown in Figure 1.

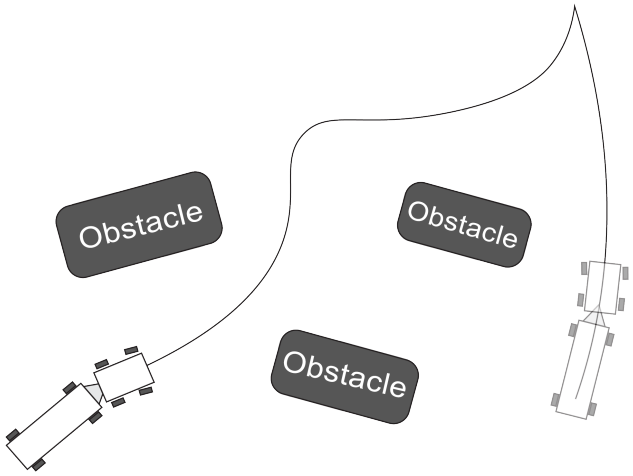


Figure 1: Illustration of the driving phase.

General requirements regarding the workflow and the overall system are described in Table 2.

Table 2: General requirements on the system.

Requirement	Revising	Description	Priority
1	Original	Requirements that the project group are unable to fulfill should be renegotiated with the orderer well before the deadline.	1
2	Original	The obstacles should be able to be placed arbitrary.	1
3	Original	No computations should be done by an external computer.	1

2.1 Included subsystems

The entire system will consist of three different subsystems shown in Figure 2. The subsystems colored grey are already implemented from last year. The state observer will receive measurements in form of position and various angles of the truck. The motion planner will use the states to determine a path. The truck system is simply the physical truck itself.

The subsystems colored blue will be implemented by the project group. The MPC controller, explained in section 3, will calculate control signals based on the motion planner's calculated path. The communications system, explained in section 4, will consist of an RPi attached to the truck. This system will handle the communication between other subsystems via ROS. The visualization system, explained in section 5, will display the path along with the obstacles on the floor in Visionen by using existing projectors.

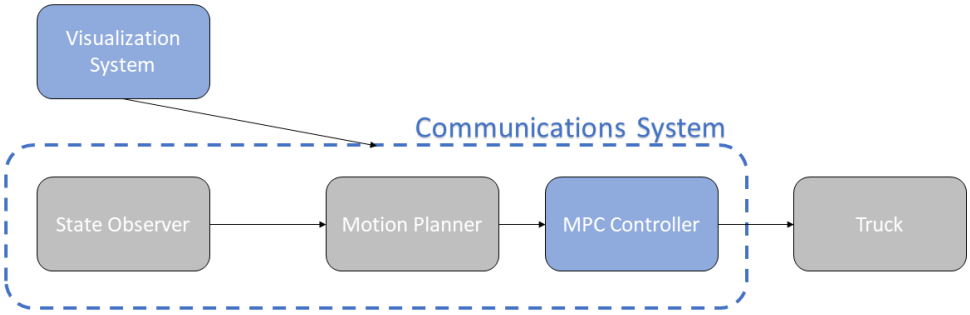


Figure 2: Overview of the subsystems and how they communicate with each other.

2.2 Limitations

To be able to reach the requirements of this project there had to be some limitations and simplifications made.

- All obstacles in the operating area are static.
- The operating area is flat.
- A low enough velocity is guaranteed to justify the usage of a kinematic vehicle model.

3 CONTROLLER SYSTEM

The controller system's task is to follow the path that is calculated by the motion planner which uses state parameters from the state observer along with the QualiSys system. The controller should be able to follow the path when moving forward as well as when reversing.

3.1 General requirements

The general requirements for the MPC controller are listed in Table 3.

Table 3: General requirements for the MPC controller.

Requirement	Revising	Description	Priority
4	Original	The controller should be of type MPC.	1
5	Original	The controller should be implemented on the RPi.	1
6	Original	The controller should be implement on the EV3.	3
7	Original	The controller should be a node in ROS.	1
8	Original	The controller should work in a simulation environment.	1
9	Original	A simple calibration phase of the steering angle of the front wheels should be implemented in the start of deployment.	1
10	Original	An improved calibration phase should be implemented that has a maximum execution time of 15 seconds.	3
11	Original	The controller should work together with other parts of the system.	1

3.2 Interfaces

Requirements regarding the communication between the MPC controller and the other subsystems are listed in Table 4.

Table 4: Requirements regarding communication between the MPC controller program and the other systems.

Requirement	Revising	Description	Priority
12	Original	The controller should be compatible with the existing motion planner.	1
13	Original	The controller node should publish the planned route on a topic.	1
14	Revised: 2020-11-17	The parameters in the controller should be easy to change via a laptop that is remotely connected to the RPi during development.	2
15	Original	The controller should use ROS to communicate with the other systems.	1

3.3 Design requirements

Requirements for how the MPC controller should be designed and which extensions that should be used are listed in Table 5. These will most likely be reiterated multiple times as the design of the MPC controller will evolve throughout the project.

Table 5: Requirements regarding the design of the MPC controller.

Requirement	Revising	Description	Priority
16	Original	The controller should work in real time.	1
17	Original	The controller should have integral action.	1
18	Original	The controller should avoid obstacles.	1
19	Original	There should be appropriate constraints on the control signal (sent to the truck) to ensure that the physical constraints of the actuators are not exceeded.	1

3.4 Functional requirements

Requirements for how well the MPC controller should work are listed in Table 6.

Table 6: Functional requirements for the MPC controller.

Requirement	Revising	Description	Priority
20	Original	The truck should be able to follow a straight path, without deviating with more than 10 cm from the path, when moving forward.	1
21	Original	The truck should be able to follow a straight path, without deviating with more than 5 cm from the path, when moving forward.	2
22	Original	The truck should be able to follow a straight path, without deviating with more than 1 cm from the path, when moving forward.	3
23	Original	The truck should be able to follow a straight path, without deviating with more than 10 cm from the path, when reversing.	1
24	Original	The truck should be able to follow a straight path, without deviating with more than 5 cm from the path, when reversing.	2
25	Original	The truck should be able to follow a straight path, without deviating with more than 1 cm from the path, when reversing.	3
26	Original	The truck should be able to follow a curved path, without deviating with more than 10 cm from the path, when moving forward.	1
27	Original	The truck should be able to follow a curved path, without deviating with more than 5 cm from the path, when moving forward.	2
28	Original	The truck should be able to follow a curved path, without deviating with more than 1 cm from the path, when moving forward.	3

cont. on next page

cont. from previous page			
Requirement	Version	Description	Priority
29	Original	The truck should be able to follow a curved path, without deviating with more than 10 cm from the path, when reversing.	1
30	Original	The truck should be able to follow a curved path, without deviating with more than 5 cm from the path, when reversing.	2
31	Original	The truck should be able to follow a curved path, without deviating with more than 1 cm from the path, when reversing.	3
32	Original	The truck should be able to follow a path, without deviating with more than 10 cm from the path, generated by the motion planner when moving forward.	1
33	Original	The truck should be able to follow a path, without deviating with more than 5 cm from the path, generated by the motion planner when moving forward.	2
34	Original	The truck should be able to follow a path, without deviating with more than 1 cm from the path, generated by the motion planner when moving forward.	3
35	Original	The truck should be able to follow a path, without deviating with more than 10 cm from the path, generated by the motion planner when reversing.	1
36	Original	The truck should be able to follow a path, without deviating with more than 5 cm from the path, generated by the motion planner when reversing.	2
37	Original	The truck should be able to follow a path, without deviating with more than 1 cm from the path, generated by the motion planner when reversing.	3
38	Original	The truck should be able to finish an entire mission, that includes both driving forward and reversing, without deviating from the generated path with more than 10 cm.	1
39	Original	The truck should be able to finish an entire mission, that includes both driving forward and reversing, without deviating from the generated path with more than 5 cm.	2
40	Original	The truck should be able to finish an entire mission, that includes both driving forward and reversing, without deviating from the generated path with more than 1 cm.	3

4 COMMUNICATIONS SYSTEM

The communications system is the brain of the complete system. It will consist of an RPi attached to the truck which will replace the need of an external computer and thus make the truck entirely mobile. The communications system will be used to host the ROS master node and perform the calculations needed for the motion planning in a separate node. The requirements for the communications system are listed in Table 7.

Table 7: Requirements on the communications system.

Requirement	Revising	Description	Priority
41	Original	An RPi should be attached to the truck.	1
42	Original	A separate power supply should be attached on the truck used to power the RPi.	1
43	Original	ROS should be installed on the RPi.	1
44	Original	The RPi should host the ROS master node.	1
45	Original	A node used for motion planning should be used by the RPi.	1
46	Original	A script running on start-up should be established where the ROS system runs automatically when the RPi is plugged into the power supply.	2
47	Original	The RPi should have a wired connection to the EV3.	1

5 VISUALIZATION SYSTEM

To better use APPAS in teaching and advertising purposes, should APPAS be visualized using one of the projectors in Visionen. Using the projector the planned path and the path followed by the truck will be displayed on the floor during the entire operation. To receive information about the path the visualization system will have wireless communication with the communications system.

Figure 3: The visualization system.

5.1 Design requirements

Requirements regarding the design of the visualization system are listed in Table 8.

Table 8: Design requirements on the visualization system.

Requirement	Revising	Description	Priority
48	Original	The visualization system should run on the computer in Visionen connected to the projector.	1
49	Original	The visualization should display the offline planned path.	1
50	Original	The visualization should display the online improvements of the path.	1
51	Original	The visualization should display the driven path during the operation.	1
52	Original	The visualization should display the obstacles.	1

5.2 External interfaces

Requirements regarding the external interfaces of the visualization system are listed in Table 9.

Table 9: Requirements regarding external interfaces on the visualization system.

Requirement	Revising	Description	Priority
53	Original	There should be wireless communication to and from the RPi.	1
54	Original	The external computer should be connected to the projector with an HDMI cable.	1

6 SIMULATION ENVIRONMENT

For faster development of the planner and controller, a simulation environment will be used. Requirements on this system are listed in Table 10.

Table 10: Requirements on the simulation environment.

Requirement	Revising	Description	Priority
55	Original	The simulation environment should display the planned path of a mission.	1
56	Original	The simulation environment should be able to run the planner.	1
57	Original	The simulation environment should display the obstacles.	1
58	Original	The simulator should simulate the truck's movements.	1
59	Original	The simulation environment should display the simulated path of the truck's movements.	1
60	Original	The simulation environment should be able to run the truck's MPC controller.	1
61	Original	The simulation environment should simulate the truck in 3D.	3

7 SOFTWARE QUALITY

To be able to guarantee a certain form of standard in the software implemented, requirements are established that keeps a consistency in the code. The requirements on the software implementation are listed in Table 11.

Table 11: Requirements regarding the software quality.

Requirement	Revising	Description	Priority
62	Original	All new code written by the group should comply with the formatting used previous years, which is the Google code standard for C++ and Python.	1
63	Original	All new modules should be compatible with the existing ROS structure.	1
64	Original	All known bugs and issues with the code should be documented before the project ends.	1
65	Original	Git should be used for the written code.	1
66	Original	No development should be done in the master branch on Git.	1
67	Original	All code should be merged through a merge request where another person has reviewed and accepted the changes.	1

8 ECONOMY

The economy requirements on the project are presented in Table 12.

Table 12: Economy requirements.

Requirement	Revising	Description	Priority
68	Original	Each group member shall spend 240 hours on the project.	1
69	Original	The group shall receive a total of 40 hours of tutoring time.	1
70	Original	The group shall be provided with a project room.	1
71	Original	Necessary hardware for the project shall be provided by the customer.	1

9 DELIVERY

Requirements for the necessary deliveries are listed in Table 13.

Table 13: Requirements on the deliveries.

Requirement	Revising	Description	Priority
72	Original	The following should be delivered by TG2 (2020-09-22): requirement specification, project plan, time plan, an oral presentation of the system and a draft of the design specification.	1
73	Original	The following should be delivered by TG3 (2020-10-06): final design specification and a test plan.	1
74	Original	The following should be delivered by TG4 (2020-10-29): all subsystems should be individually tested in simulation or in Visionen.	1
75	Original	The following should be delivered by TG5 (2020-12-01): fully functional system, user manual, test protocol and a presentation where it is shown that the requirements are fulfilled.	1
76	Original	The following should be delivered by TG6 (2020-12-14): technical report, after study, movie, poster presentation and a website describing the project.	1

10 DOCUMENTATION

Table 14 lists all documents that shall be produced during the project, where the different targets for these documents are the project group (PG), orderer (O) and customer (C).

Table 14: Documents to be produced during the course of the project.

Document	Language	Aim	Target	Format
Project plan	English	Formulates how the work during the project is organized, including milestones and deliveries.	PG, O	Electronic/PDF
Requirement specification	English	Describes the different requirements of the product which has to be fulfilled by the project group.	PG, O, C	Electronic/PDF
Time plan	English	Specifies how the resources shall be distributed between the activities.	PG, O	Electronic/PDF
Design specification	English	It will contain an thorough overview of the whole system and also of the three subsystems.	PG, O	Electronic/PDF
Test plan	English	Describes the tests that are supposed to be performed and the requirements they are suppose to fulfill.	O	Electronic/PDF
Test protocol	English	A protocol which present a list of all the tests which were made and the requirement they fulfilled.	O	Electronic/PDF
User manual	English	Describes how the end product should be used.	C	Electronic/PDF
Technical documentation	English	A documentation that will contain all the technical results during the project.	O	Electronic/PDF
After study	English	A reflection of what went well during the project and what could be improved in future work.	O	Electronic/PDF
Poster presentation	English	A poster which will summarize the whole project, mainly the results and conclusion.	C	Poster
Web page	English	A web page that will present the project and contain all the relevant documents.	C	Electronic/HTML
Movie	English	A short movie presenting the final system and how it works.	C	Electronic/Video
Meeting minutes	English	The agenda of each meeting which also contains the notes during the meeting.	PG, O	Electronic/PDF
Time report	Swedish	An accurate time report which will present how much time each project member has spent on the different activities each week.	PG, O	Electronic/PDF
Status report	Swedish	Report of what each member have done during the week.	O, PG, C	Electronic/PDF

REFERENCES

[1] K. Stubbs, P. J. Hinds, and D. Wettergreen, "Autonomy and common ground in human-robot interaction: A field study," *IEEE Intelligent Systems*, vol. 22, no. 2, pp. 42–50, 2007, [Online accessed: 21 September 2020].

[2] T. Fridén, L. Junler, A. Källström, O. L. Jonsson, T. Nyberg, and T. Westny, "Technical documentation, autonomous reversing truck," http://www.isy.liu.se/edu/projekt/tsrt10/2019/rev_truck/, 2019, [online accessed: 16 September 2020].