CHALMERS
UNIVERSITY OF TECHNOLOGY

Obstacle circumvention by automated guided vehicles in industrial environments

Master's thesis in Systems, Control and Mechatronics

Matilda Halldén
Filip Saltvik

Department of Electrical Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2018

Master's thesis EX039/2018

Obstacle circumvention by automated guided vehicles in an industrial environment

Matilda Halldén
Filip Saltvik

Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden 2018

Obstacle circumvention by automated guided vehicles in an industrial environment
Matilda Halldén
Filip Saltvik

Supervisor: Fredrik Hagebring, Electrical Engineering; Jonas Martner, AGVE
Examiner: Petter Falkman, Electrical Engineering

Master's Thesis EX039/2018
Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover:

Typeset in LaTeX
Gothenburg, Sweden 2018

Obstacle circumvention by automated guided vehicles in an industrial environment
Matilda Halldén, Filip Saltvik
Department of Electrical Engineering
Chalmers University of Technology

Abstract

The intralogistics of companies has been the subject of improvement and increased efficiency. The Automated Guided Vehicle (AGV, plural AGVs) has been used for decades instead of having manual labour transporting products or material around the site. The demand for these types of robots are increasing, and their autonomy is continuously developed. There is a wish to increase autonomy such as avoiding obstructions or collaboration between robots. Robotics research has proven that there are numerous ways to get a robot to overtake an obstacle. How should it be made in an industrial environment? These are uncontrolled environments with other workers present who should not be injured or even startled by AGVs crawling up behind them. This master thesis aims to develop a functional way to increase autonomy in industrial AGVs by enabling them to circumvent obstacles that block their intended path. The result is promising but raises the question of when more safety scanners is needed.

Keywords: Automated guided vehicles, AGV, industrial environment, obstacle circumvention, autonomous system, LIDAR, autonomous fork lift

Acknowledgements

We would like to thank our supervisor at Chalmers University of Technology: Fredrik Hagebring for his constant encouragement. We would like to thank Jonas Martner for his help at AGVE, his intimate knowledge of the system, and simple solutions to our coding issues.

Matilda Halldén
Filip Saltvik

Gothenburg, June 2018

Contents

List of Figures

1
Introduction

Transportation of goods is a big part of any production company and is also an industry in itself. It has been part of production and trade in history and today it is an essential part that can be the difference between unsuccessful and prosperous business. In any production company there is a need to transport raw material, production material and finished products within the factory. In all these cases the transports is usually subject to optimization, streamlining and automation. Automatic transport can vary from a rail or conveyor belt to a more recent introduction of Automated Guided Vehicle (AGV). AGVs were introduced in the 50s[1] to serve intralogistics.
Transportation of goods is the most common purpose for use of AGVs[1], e.g. a forklift transporting pallets of goods, and they are often designated to carry heavy loads. Another example of usage is carrying a car through an assembly line. An AGV can offer more freedom than conveyor belts by being less reliant on creating expensive and invasive infrastructure and can be placed in almost any industry setting where transportation of goods is made by workers.
The industry setting differs depending on the company, and the type of goods that is transported. AGVs often coexist or collaborate with workers in the area, and space available is limited. AGV systems has to meet the requirements that comes with these environments and need to work with limitations of the company. For example the perception of the surrounding world on an AGV is usually limited.
In the book Automated Guided Vehicle Systems by Günter Ullrich[1], several guidance methods for AGVs are described e.g. physical guidelines, magnetic anchoring, GPS and LIDAR navigation. The latter two are set up by placing reflectors on the premises which provides the AGV with its coordinates by triangulation. In LIDAR navigation a rotating laser transmitter is placed upon an AGV and there are positioned reflectors which sends back the beam to the transmitter and records the distance and angle, giving the relative position to nodes with known position.
Ullrich also describes the safety systems of AGVs which at least in the European Union is well regulated and followed. Since AGVs often share operating space with workers the AGV must be able to recognize possible collisions and brake to avoid person injury and damage to equipment. There are numerous techniques to identify obstacles, some older and some more recent entries to the safety department of AGVs. An old method which still serves a function today is mechanical systems that triggers stop on physical collision. This require speed slow enough to be able to stop at the occurrence. Early versions of AGVs which only had this safety system was moving slow all the time, but with contact free sensors e.g. laser or radar, it is possible to maintain safety at higher speed since its sensors will react early enough

to brake and come to a complete stop before collision. An advanced sensor is the safety laser scanner and Ullrich argues it is most commonly used. It scans the area in front of an AGV, and can give very precise readings of distance and angle to a measured point.
What they do lack however, is ability to detect whether an obstacle is a moving person or object, and if the AGV is able to (or should) pass the obstacle. What happens today is that they will wait indefinitely, until someone removes the obstacle, harming the makespan enormously. If the obstacle is workers they should move out of the way and the AGV can carry on, but if something else is in the way it might be desirable to get around the obstacle to finish the orders. AGVs often lack the autonomy of performing this task.
Several spokespersons in the industry argue that obstacle circumvention is a problem that should not need dealing with in the first place. Instead all areas where AGVs move should be free of non permanent obstacles[2]. They also argue that it is desirable to know exactly how a robot acts. As Ullrich describes in his book [1], a guidance system is usually keeping track on all AGVs and are centrally coordinating all their movements and orders. Thus its easy to raise a warning flag to whomever that needs to see to remove the obstacle. However if there are no workers currently available to remove the obstacle, e.g. during a night shift where the AGVs still are active and no workers are present, circumventing the obstacle could lessen the impact on the system performance. There are also situations where placing things temporarily in the way of AGVs is necessary. It also takes time for workers to get used to new AGV systems and new employees might still make mistakes.
This thesis will create an autonomous obstacle circumvention for AGVs, starting in what previously have been done in the area. This thesis is done together with the company AGVE. AGVE designs and manufacture AGVs as well as traffic controllers which supervise the AGVs. The implementation will be made on AGVE's system on a an AGV with their specific hardware. A picture can be seen in figure 1.1. The focus is finding a solution that would work on AGVEs system and many types of similar existing systems today, and to understand the industrial needs for it.

1.1 Objective

The aim of the project is to make an AGV detect and autonomously search its way around a non permanent obstacle in an industrial environment. The AGV should be able to detect an obstacle and identify the outer edges using a safety laser scanner. After that the AGV should leave the original path and search for an alternative path to circumvent the object. The search finalizes when there is a clear path back to the original path. The new path should also be saved for future usage by other AGVs, as long as the obstacle remains.
The goal is to find an algorithm that deal with the industrial setting and its implications. Should the obstacle be small, only a minor deviation from the controlled path should be made. The aim is not to produce a general circumvention algorithm but something that can be used in industry. It should be simple enough to be used with many different AGV systems. It should work well with limited perception of the world, and it should not be a computational heavy method due to limitations

Figure 1.1: A tricycle AGV on which the implementation has been made, at AGVE

in embedded systems. It should be made in a safe manner where it won't reduce the makespan any more than if the AGV would have remained still in front of the obstacle. The purpose is to raise the level of automation without interfering too much with safety and predictability of the system.

1.2 Scope

The focus of this thesis will be to make an AGV search its way around an obstacle with the help of a safety laser scanner, which is the equipment available for testing. The AGV used in the project is a tricycle fork lift with one steering wheel and is therefor limited to not being able to move sideways. The safety laser scanner, giving range and bearing, that will be used to solve the problem is commonly used on AGVs.
The project is carried out in an industrial environment which means that the environment is restricted and narrow and also includes other moving objects as well as people. This together with heavy duty AGVs result in many constraints on the type of movement that should be allowed. This thesis focus on systems where AGVs can not move freely but are restricted to specific paths. The paths covered are only straight ones. Curved paths requires other functions and implementation.
This thesis relies on that it is possible to get spatial measurements of an obstacle. The detection should work on an obstacle of unknown size and shape, but the

obstacle needs to be detectable by a safety laser scanner. It is also assumed that the object is not moved. The scope is not to understand what type of obstacle but rather the presence and borders of it. Specifically the edges of an obstacle must be detectable by the system. It is also worth noting that the AGV must be able to actually move away from the original path and find its way back. This demands e.g. that it is physically possible for the AGV to move away from it.
Resource allocation will not be a part of the thesis, meaning other AGVs or paths will not be taken into consideration. Instead the focus is on finding a path around an obstacle in order to avoid the situation of a complete stop of an AGV.
The scope is not to evaluate different types of obstacle circumvention or a best way to do it. It is a solution to a problem with many interesting angles to cope with in different scenarios.

1.3 Outline

This thesis has chapters about preliminaries, method, results and discussion. The preliminaries describes the type of AGV used and its limitations, as well as concepts that need specification. The method chapter starts by describing the system used by the company AGVE where the work has been performed. It then gives details about implementation together with arguments for choices made based on industrial needs and restrictions. The result section contains both simulated and real life testing on the system. The final chapter consists of discussion on the functional system, its properties and its drawbacks as well as some thoughts of future work.

2

Preliminaries

In order for an AGV to autonomously circumvent something that currently is obstructing its path it needs to decide which way around it should take, and how to do it without risk of colliding with e.g. other AGVs or human operators. In this chapter some prerequisites and ideas of how to solve an autonomous circumvention are presented. The idea of a robot autonomously avoiding and circumventing a new obstacle is certainly not a new idea, and there are many algorithms and solutions. The solutions all have some things in common though, the robot needs to be able to perceive its surroundings and navigate through it.

2.1 Industrial environment

An industrial environment containing AGVs are often characterized by cramped areas, and areas that are narrow with room for only one AGV to occupy such as different types of corridors. These environments, that typically are logistics or production areas, are shared between the autonomous system and human operators, and may also include manually driven machines. The system are typically indoor but can take place partly outdoor which then holds higher requirements on the parts operating in the system. A main issue is safety, and the system needs reliable safety measures. AGVs operating in an industrial environment often carry heavy loads which makes their movement less flexible.

One way for the AGVs to move in the environment is by following predefined paths, that together creates a roadmap. This type of system constraints the possible motions for the AGVs and makes the system less flexible but the roadmap does however also work as a safety measure since the AGVs will not be able to move in an unpredicted way. A system that follows predefined paths like this can be called a guided system.

Guided systems are the ones most commonly used in industrial environments where laser navigation is the absolute most frequently used method today, and has almost replaced the use of magnets and wires. This gives a more dynamic system since the paths are more easily changed due to them being digital. With help from the exact positioning of the laser navigation it is possible for the AGV to follow the predefined segments. LIDARs are commonly used for the overall navigation while information about the environment around the AGV is given by other sensors, like a laser scanner.

2.2 AGV model and system

This project focuses on an AGV system using tricycle AGVs in an industrial environment and in this section details about how they function is presented. A system often consists of segments, nodes and agents, where agents are the AGVs moving in the system, nodes are points of interest, e.g. a station to place objects, and segments are predefined roads between two nodes.

2.2.1 Tricycle AGV

A tricycle AGV has a steering wheel in front and two support rear wheels. The steering wheel is placed in the center of the AGV and gives a heading and steering angle to the system. The AGV navigates by a laser scanner on top and placed at the front there is one or several safety sensor(s), often laser scanner(s). Using a tricycle AGV affects the possible ways to move in contradiction to e.g. smaller AGVs that move freely in any direction and turn on the spot. In figure 2.1 a model of a tricycle AGV can be seen where wb is the wheelbase, α is the steering angle, i.e. the relative angle between the heading of the AGV and the steering wheel, and θ is the heading angle in the global layout.

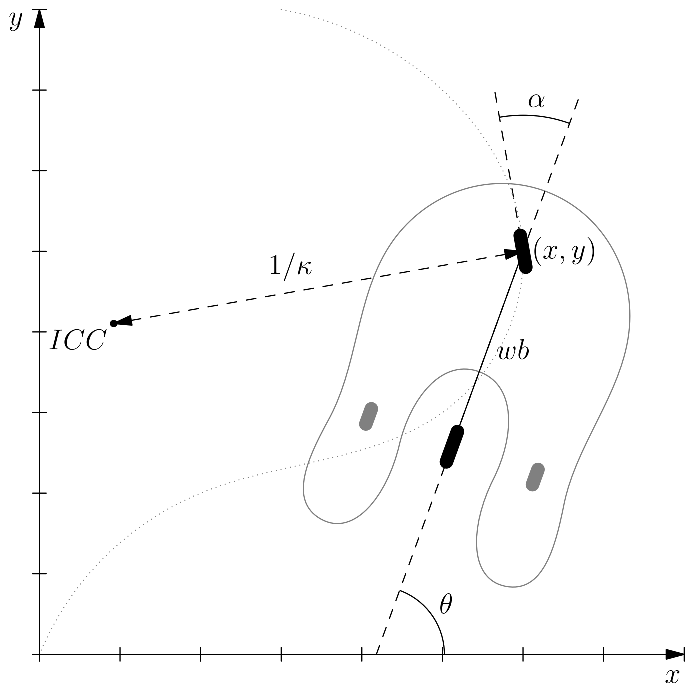


Figure 2.1: Model of a tricycle AGV, by Dahl and Forsberg [3]. ICC is the instantaneous center of curvature, α marks the steering angle. The axes marks the coordinate position in the global layout.

As mentioned in section 2.1 the areas may only have space enough for one AGV to occupy, that combined with the dynamics of an AGV and its limited perception of the surroundings sets certain limitations on implementation of an obstacle circumvention. A front steered tricycle dynamic traits that need to be considered are:

- It rotates around with a notable lever
- Its body straightens up slowly onto a line path
- In order to quickly straighten up it needs to overshoot its path

A forklift's lever together with its perception can be problematic. With a sole rotational movement the sides are at collision risk and might be a blind spot if there are no sensors covering the sides. This typically creates problems in a corridor shaped as an S since there is no room for e.g. overshoots. A change in movement for example, raises the need to be aware of the tail of the truck, especially when loaded, to not overshooting the chosen paths. As seen in figure 2.2, the tail of the truck does not follow the segment which could mean that it will slam into something unintentionally. The slow straightening might also impair the perception of interest if the sensors are placed in a way that their vision might be partly blocked by the AGV itself.

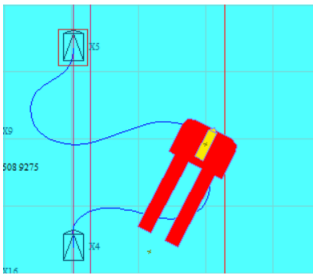


Figure 2.2: The tail of the truck needs to be taken into consideration where there are tight overshoots

It is possible to perform a path change and quickly align the heading of the truck to that of the path with an overshoot, i.e. crossing the path first in order to quickly bring the back part of the truck to the path and then maneuver the front back to the path. This is a common approach but in tight areas this might be impossible. These things need to be considered in approaching the obstacle circumvention problem.

2.2.2 Navigation

For an AGV system to function it has to keep track of where the AGVs are, with millimeter precision. This is done with different kinds of navigation systems, which can be divided into two divisions. One is using unguided techniques such as SLAM (simultaneous localization and mapping) or free navigation while the other one uses guided techniques like laser, IR, magnetic strips or wires. Laser navigation is the one most widely used for AGV applications.

Laser navigation is based on triangulation, i.e. there are several reflectors placed in the area where the AGV is to move. On top of the AGV there is a LIDAR sending beams 360 degrees around and registers when the beam has hit a reflector and returns to the device. The reflectors are marked in the map of the system, and by comparing the distance to the reflectors the AGV's current position can be

calculated. It grants a position estimate with a high accuracy that is continuously updates.
The millimeter precision given by using laser navigation gives some advantages over the other systems since it can act in a more constricted area. The precision from the navigation makes it possible to leave a pre-defined segment and still keep track of where the AGV is positioned. Since the position is precise the safety functions can also be more secure and adaptable, which is desirable when leaving the pre-defined segments and expected behaviour.

2.2.3 Safety systems

The safety systems are crucial for the system to both work as expected as well as making sure no person or property gets injured. The safety system can be of different kinds, e.g. having physical bumpers that needs to be pressed or by using contact free sensors. It gives the possibility to react to the environment.
Using the output from contact free sensors to define fields of interest, a reaction can be triggered if somethings appears in the fields. The fields can be used to indicate that the AGV should slow down since something is approaching or that it should stop since an object is too close to the AGV. The size of these fields affects the possible maximum speed of each AGV since it has to be able to stop within the length of the field to not slam into the detected object.
When creating segments and turns for the AGVs, their range of movement needs to be considered as well as the area needed for it to drive.

2.2.4 Embedded systems

AGV systems are often built with embedded systems. It is designated to control the movement and communication of respective AGV. That usually means most of the components on an AGV: control of drive and steering, loading of goods e.g. forklift control, navigation and communication. It grants a small, cheap and low power consuming dedicated function, but it comes with restricted processor power and memory which implies that the calculations made has to be designed and adapted to function properly.

2.3 Path planning

There are several ways of performing path planning for the AGVs and the best choice of algorithm generates paths that are feasible, i.e. driveable and obstacle free [4]. Obstacles are often other agents in the system but could also be other objects. The choice of how to perform the path planning is also dependent on which computational power available and needed accuracy in combination with the complexity of the layout [5].
The path planning is made for the whole system meaning it is performed for multiple AGVs at the same time since collaboration between the AGVs is crucial when they operate in a confined area with a completion objective.

The planning of the paths and the traverse along them are usually made by an overall planning system, a traffic controller, which sends instructions to the AGVs and keeps track of their positions. The traffic controller is responsible to make sure the total throughput of the system is met and is optimizing the system. The throughput is often measured in number of delivered orders over an amount of time but could be measured in other ways. All are different measures of economics, and if the throughput is too low the system will not deliver enough to be economically sustainable.
The available paths in the system can be built in different ways by segments and have different properties such as direction(s). It is not suitable for all AGVs to be able to travel in both directions at all segments, therefor some are uni-direct and some are two-ways.
If there is enough space, there can be two parallel uni-direct parallel segments creating a two-way direction that can be traversed by two AGVs at the same time. This however requires more available space but has the advantage of handling several AGVs.
If it is possible to travel other segments to reach the intended goal and avoid the blocked segment, a rescheduling of the path can be made. Some paths lead to dead ends which means that it might not be possible to take other segments to reach the goal and will result in a non-solvable problem if the system has no way to dynamically change or add segments. If there are other segments that leads to the goal the path is updated and the AGV is receiving the new path plan.

2.4 Obstacle detection

An AGV can only react to obstacles if there is some form of sensor that can identify an obstruction. As mentioned in section 2.2.3 there are several types of sensors. The distinct information given by different kind of sensors has to be regarded when it comes to understanding and identifying an obstacle. There is also a discrepancy if the sensor is placed on an AGV or if it is placed stationary in the area. A sensor on the AGV gives a relative measurement to the AGV, depending on the placement. This is interesting in the obstacle detection implementation since an obstacle is something that is in the AGVs intended path and thus it is easier to understand what is an obstacle and what is not, if the sensor is mainly reading the intended path.

2.4.1 Perception

Using a laser scanner as perception, a lot of interpretation is needed to identify and understand the environment. Its output is the distance and angle to every measured point, which is a planar view of the environment. The height of the given view is dependent on the placement of the laser scanner, often is it placed low on the AGV. If the perception is delivered by another type of scanner such as a camera or a 3D laser scanner which delivers more than one plane of view, a better distinction between objects could be made and it would also be possible to increase the understanding of an obstacle.

It is easy finding obstructions that are in the immediate path, given some standard trigonometry the area which it will traverse can be computed and compared to the measuring points. Given the AGVs global position and heading (including the heading of the scanner), all the measurement points can be translated to a global coordinate. Given these points it is not always that points in front of an AGV should be considered an obstacle, it can be a wall just before a turn, another AGV etc. By comparing with a layout of the system it is possible to more certainly distinguish an obstacle from the surrounding. Given these points it is a filtering implementation of choice that decides what constitutes an obstacle, which can be made in more or less clever ways. A typical scenario where the robot uses the only perception it has is presented by [6] and [7]. The perception is often a laser scanner or ultrasonic ranging methods that gives range and bearing measurements which output then is used to perceive the obstacle.

2.4.2 Identification of points

To detect the obstacle and identify its properties such as size and shape, some processing of the data from the AGV's sensor(s) is needed and several measurements are needed. This can be done by different heuristics to remove data artifacts and then classify the remaining points to find which belong to the obstacle [8]. The classification of the points can then be based on probability and assumptions such as that the ground is uniform and that the obstacle is detected by changes of curvature. The measured points relative the AGV can be used to distinguish the obstacle, by being in the movement direction of the AGV.

2.5 Obstacle circumvention

Given knowledge of the layout together with knowledge of the front end of the obstacle, it is possible to find a guessed shortest path by converting the layout with the obstacle into a grid and then apply some shortest path algorithm, e.g. A* [7]. Nelson describes in the article a way to create smooth curves for an AGV to follow which are described by end point parameters. Any known arbitrary obstacle can be circumvented by planning a smooth segment around it[9]. The central controller system, the traffic control system, can also provide detail about the presence of any other permanent obstacles or other AGVs in the way.

If the shape of the obstacle is not known a method to circumvent an obstacle could be based on making local deviations from the planned path where the obstacle is present. The deviation is made by creating an S-shaped curve from the blocked segment and then moving parallel to it. When the obstacle is completely passed, another S-shaped curve is made leading back to the original segment and the original path plan is continued [10]. The process can be seen in figure 2.3, where the orange rectangle models an AGV moving along the blue segment, detects and circumvents the obstacle by creating S-shaped curves and moves in parallel to the obstacle.

Given the size and shape of the obstacle and an AGV with free range of movement gives the option to use other methods to circumvent the obstacle. One uses artificial potential field and its gradient that produces a repulsive force [11]. The repulsive

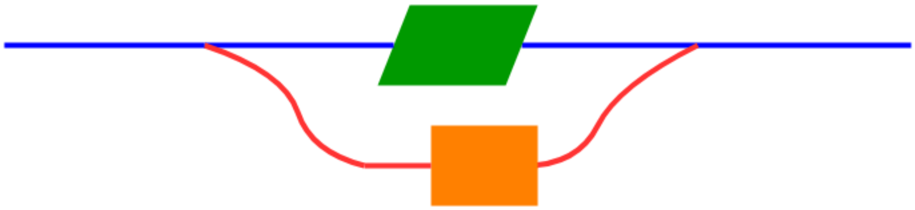


Figure 2.3: Circumvention of an obstacle

force can then be used to make the AGV drive away from the obstacle by creating new segments based on the force. Another way to avoid the obstacle is to use gyroscopic force that drives the vehicle, without colliding with the obstacle, towards a goal point [12]. The method is based on creating shells around the vehicle, an outer detection shell and an inner safety shell where the goal is to avoid getting the obstacle in the safety shell. It uses the potential force for convergence and the gyroscopic one for avoiding the obstacle. Using the procedure in [6] it is possible to plan a curve that bypasses a visible edge of an obstacle without colliding with it if the AGV is equipped with sensors that measure range and bearing. As stated in the article it is proven to work, but not without any drawbacks. It does not plan optimal paths around for any arbitrary obstacle and the chosen path (left or right) might not be globally optimal. There also exists a risk of collision since creating the path does not regard anything that might be behind the end points of the obstacle.

3
Methods

The thesis is done in collaboration with the company AGVE, where tests and implementation has been carried out which gives some restrictions to the work. The method chapter has five separate sections. The first section details the system that has been used at AGVE. Then follows two sections with general ideas and restrictions of a solution. Section 3.4 describes the made implementation and 3.5 describes what has been used for result generation.

3.1 AGVE system

The AGV system used at AGVE works in general as described in section 2.2 but has some specifics that affects the development and implementation of automatic circumvention. The layout of the system where the AGVs are supposed to work is created in a CAD software and contains all predefined parts of the system such as segments, programmed turns and nodes at points of interest. Every AGV has its own processor for running different applications and the implementation is made and tested on a processor with approximately 1 GHz [13].

3.1.1 Path planning

The path planning in the system of AGVE is executed by a traffic control system and then sent as instructions to the AGVs in the system. There might just be instructions to travel one node ahead, depending on the density of the nodes since the allocation is based on time. Instructions holds information like move from one node to another via a certain segment or to pick something at a station. The instructions are performed by executing binary tables which contains information generated from the layout of the system such as speed and coordinates. The tables are decoded by an interpreter at each AGV and stored in a FIFO buffer where the AGV pulls instructions one by one until the buffer is empty.

All segments used in the system are defined when the system is created, and the traffic controller can not make an AGV drive outside of a defined segment. This means that the path planning has a limited number of total paths that fulfills the system objective and if one segment for some reason is blocked or unavailable it either reduces the number of feasible solutions or blocks the system totally.

3.1.2 Control system

AGVE has an implemented control system that works on several levels. The control feedback consists of a low level loop where the current steer angle and speed are fed back to the system to control the same to wanted values. There also exists a high level feedback loop which considers navigation parameters. Main focus is on making sure the navigation in the overall system is prioritized. The navigation parameters that are fed back to the system are the current position in the layout given by the laser navigation by either laser deflectors or contour navigation. If an AGV is outside its path, a cost function makes sure the AGV prioritizes getting back to the segment over aligning itself according to the heading of the segment.

3.1.3 Navigation

Navigation is performed by a LIDAR and reflectors which gives high accuracy of the AGV's position. The position is continuously calculated at each AGV and then communicated to the traffic controller to keep track of all agents in the system. Each AGV has an expected position on determined segments and if the communicated position deviates from it, outside a given range, an error signal is created. This error helps to prevent any off-track AGVs that for some reason has driven too far away from the planned path and its expected behaviour.

3.1.4 Safety

The safety systems in the AGV system is placed locally at each AGV and the traffic controller has no element of safety apart from planning the paths in a way such that collisions should be avoided.

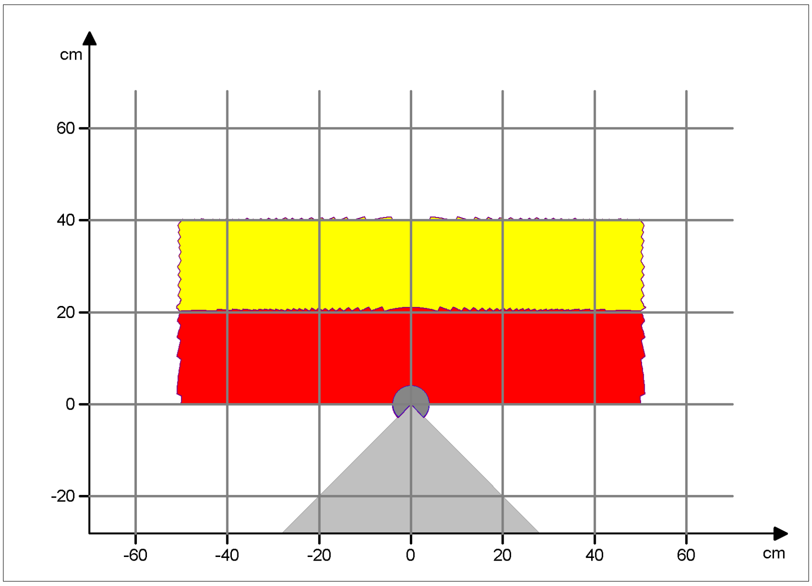


Figure 3.1: An example of fields of the laser scanner, the yellow represent a slow down-field and the red a stop-field

The safety systems consists of one or multiple safety laser scanner(s) with a scan view of 270 degrees and 540 points and are placed in the front of the AGV at the same height. The scanners have fields for detecting obstacles appearing in the contrived path in front of and around the AGV, see section 2.2.3. The size and shape of the fields are individual for each layout and in figure 3.1 an example of the fields are shown.

3.1.5 Dynamic turns

When the system is running, dynamic turns can be added to the layout to replace a segment.The dynamic turns are created by manually driving the AGV along the wanted path and taking samples along the way. The samples are then used to create a new, feasible segment. The function of manually being able to change the layout without making changes in the CAD software makes the system more dynamic and is specially useful for longtime, non-permanent changes e.g. when there is construction work taking place that affects the layout and blocks segments and/or nodes.
Sample points, chosen coordinates in the global layout, needs to be collected in order to make a dynamic turn, which is performed in two ways. The first is automatically collecting samples when the truck is manually driven and the other is based on manually collected samples. The created segments are based on different kind of curves and splines, such as Bézier, NURBS(NonUniform Rational B-Splines) and clothoids[3]. The number of collected sample points and in which way they are collected, together works as input for the function of dynamic turns to decide which type of curve most reasonable and best for the circumvention. It is done together with respect to creating a feasible curve that does not increase the wear to a bad level. The path created does not necessarily intersect the given coordinates, i.e. the sampled points, but tries to match a feasible curve to the coordinates. Detailed information about how the dynamic turns are created and implemented can be found in the work of Dahl and Forsberg [3].

3.2 Detecting obstacle

As mentioned in the preliminaries, the perception of an AGV is imperative for the understanding of an obstacle. The object is to identify the obstacle and be able to circumvent it. As described in section 2.4 there are several ways to identify it. But only identifying it would need more guess work and generate a trial and error approach to solving the problem of circumventing it. For example, a bumper collision with an object is an identification of an obstacle, but it gives very little information. Better information is the kind that can be given by ranged sensors such as laser, ultrasound or vision systems like cameras.
The readings of the sensors results in a point cloud, with points given relative positions of the AGV. With these points it is possible to identify edges of an obstacle, if there is a gap and if there are several objects. A vision system, e.g. a camera, is of course the most versatile and could give a 3D reading of the object. As mentioned in section 3.1.4 the safety laser scanners gives a 2D representation of the environment. The scanners have a very high accuracy and many scan points gives a detailed

view of the surroundings. It should be pointed out that the sensor only gives one perspective of the obstacle. There is no information what lies beyond the wall of perception or on other height levels, as illustrated in figure 3.2 the scanner of the AGV will not detect the raised forks due to the height.

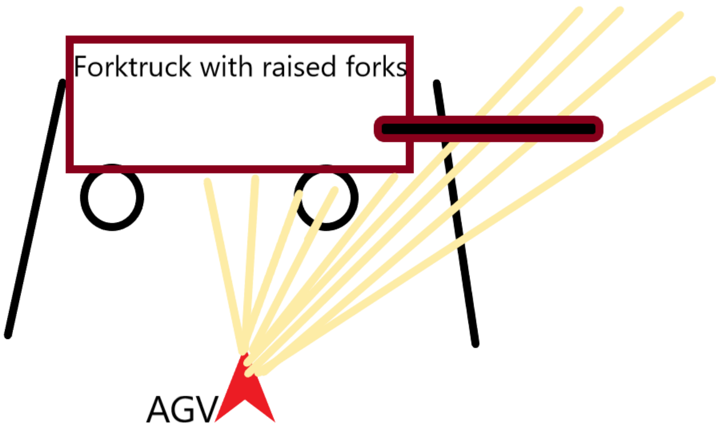


Figure 3.2: Perception of a scanner

Since the AGV in this case move in an industrial environment it limits the scope of points of interest, i.e. points that are not directly in front of the AGV should not be considered, since implementing the obstacle detection when driving through curves are not a part of the scope. This results in a relatively small span of interesting points; the width of the AGV including load limits the span of transverse points whilst the limit in lengthwise direction depends on how far away an object should be considered. This could be well described by a rectangle. Given that the load might affect the transverse span and that in curves the lengthwise distance limits should be different, a dynamic rectangle is reasonable to implement. Filtering the points that lies within the rectangle is made by vector algebra. It is done by projecting them on to the lines that are in lengthwise and transverse direction of the AGV.
In industrial sites some obstacles should not be overtaken. In order to prevent a false and unnecessary circumvention there should be a delay between detection of obstacle and circumvention. This would give plenty of time to read the obstacle. As mentioned in section 1.2 only static obstacles are considered in this work, meaning the necessity to do advanced filtering is low. In order to reduce the risk of bad readings, a simple median filtering can be used to get rid of e.g. outliers as used by Peng et al. [14]
During a circumvention, the goal is to get back to the path, and in order to do that the surroundings and obstacle again has to be considered. Now the interesting view for the AGV is not only what is directly in front of it but also in the angle leading back to the original path. By considering the points which lies within that angle and the distance between the AGV and the original path, the points detected could be considered as the obstacle.
A dynamic tracking could be made with e.g Kalman filter or particle filter but due to only consider static obstacles and performing all calculations online on the limited processor power of the AGV, the more demanding tracking methods are discarded.

In the developed function it means that once the obstacle is identified it is not considered again until a new scan is made when the AGV has arrived at the given coordinate.

3.3 Autonomous circumvention

As presented in section 2.5 there are several ways to plan a trajectory around an obstacle but most requires total knowledge of the shape of the obstacle for the method to be a success. When an obstacle appears in the way of an AGV the only available information to use as guidance is its perception of the obstacle.
Due to the 2D perception of the obstacle, information is insufficient to determine the shape of the obstacle. Assumptions could be made that the obstacle is uniform based on information from the 2D scan or determine a fixed length that will be the same for all obstacles. The risk of making an assumption like this is that the guessed shape will not be enough to cover the actual obstacle and then the AGV will run into it, making the system stop and no improvement of the system is achieved. An alternative is to not create the circumvention all at once but instead split it and use the information available at the moment.
The new available information will always be in 2D, but a picture of the obstacle can be continuously constructed by moving the AGV forward one step at a time. Starting with the width of the obstacle a decision can be made of which direction around the obstacle to perform the circumvention. After that new information needs to be collected to make a new decision of what to do next. If the method presented in [10] is used, their proposed next step after the first S-shaped turn is made to leave the segment, is to drive pass the obstacle in parallel with the original path. After the obstacle is completely passed, the return S-shaped curve should be created back to the original path. Using this method to circumvent the unknown obstacle requires the first turn to be to the outer edge of the obstacle meaning the area next to it has to be clean. However, following this means that decisions have to be made at each step, if it is possible to continue the circumvention.

3.3.1 Limitations of the industrial environment

A circumvention of an obstacle blocking the usual path requires the AGV to leave a pre-determined "safe" segment. In order to overtake obstacles it needs to drive into unknown territory, for which the AGV is not dedicated to be in. The known layout guarantees some understanding of the surroundings, but not everything. This means that finding an outbreak from the original trajectory is the first task at hand.
There are several ways to circumvent a detected obstacle, but in an industrial environment which requires guided paths, it is harder to achieve than in an area where AGVs move freely since new paths have to be created and added online. Also, the robustness and accuracy of the solution has to be considered. An AGV transporting heavy loads puts extra constraints on the movements in order to not harm or drop the load. This means that the movements has to be as smooth as possible, many fast changes and turns are preferably avoided. Since most AGV systems operate nearby people it is of high importance that the circumvention solution looks stable

and predictable, there should be no surprises in the movement. Considering this, solutions that includes high number of iterations of searching and moving a little bit are neglected since they will result in jerky moves, due to limitations of acceleration.

There are some key aspects that affects the circumvention that needs to be considered at all times. Due to the physical properties of the tricycle AGV it is impossible to e.g. do a sideway translation. Another very important thing is that the vision is fastened on the AGV. This means in order to see, it is imperative to keep the direction of the AGV in the heading of interest to not block a part of the view with e.g. the corners and wheels of the AGV. Another aspect is the navigation of the AGV, the possibility to navigate is imperative, and should not be compromised if an circumvention is to be made. This might be the case if the only navigational system is the actual segments, such as magnetic rails, or if reflector posts become invisible. A dead reckoning might serve but it might be undesired for AGVs in order to avoid damages.

Another important aspect to consider is if some reflectors for some reason can not be detected by the AGV. The loss of detected reflectors can be due to something blocking the perception of the laser scanner or the reflector, it can also occur if the AGV is too far away from the defined segments and therefor will not be able to detect the needed amount of reflectors. Loss of detected reflectors can result in lower accuracy on calculating the position of the AGV.

3.3.2 Choosing direction

To make the decision of how to circumvent the obstacle, information about the obstacle and the environment needs to be taken into consideration. The decision making can be based on information or the lack of information, e.g. having information about the layout from the map but no additional information from sensors about the environment. The same for the obstacle, if not every property of it is known, decisions about circumventing it has to be made solely on information from the layout and the current perspective of the obstacle and environment.

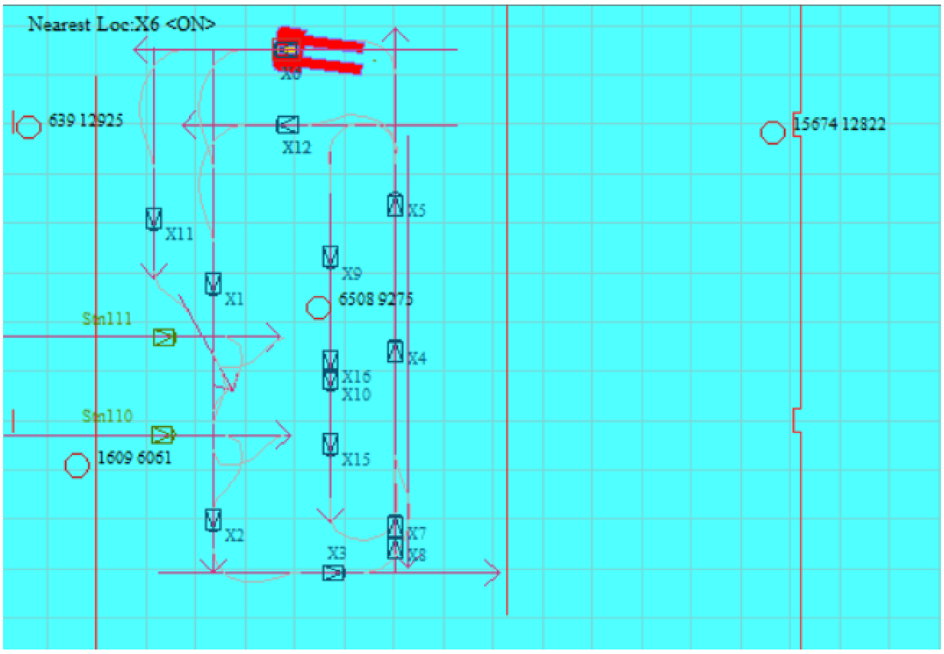


Figure 3.3: A layout with marked segments(arrows), nodes, walls and reflectors(circles)

If an obstacle should present itself on the path of an AGV, the sensors will register this and the safety system will halt the AGV. This gives information about the obstacle from the AGV's point of view. The layout together with the known position gives information of what to expect from the immediate surroundings. The layout also gives information of where the obstacle is placed, which gives possible information as distance to walls etc. An example of a layout can be seen in figure 3.3.

The first decision to be made is if it is possible to pass on either sides of the obstacle, given the layout of the system. There might be limited space available on either side of the AGV in which case the circumvention might be impossible. Given all implications from an industrial system with limited perception and safety system, it is easy to argue that the AGV should stay as close to the original controlled path as possible. If it is possible to circumvent the obstacle on both sides, the side which requires the smallest deviation from the original path should be chosen. In figure 3.4 the two possible points of where to circumvent the obstacle are marked. The point represented by the black X is closest to the original path and therefore the circumvention should be made to the left in this case.

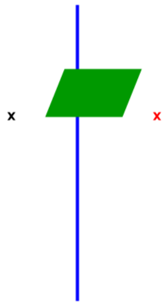


Figure 3.4: The circumvention is made on the side where the deviation from the original path (blue line) will be the smallest. The two X marks the point of where to direct the first step of the circumvention.

A probable scenario is that an obstruction is not directly on the AGV's path but protruding from one side. The workers are aware of the paths of the AGVs and will not deliberately place something in their paths, but might place something just next to the path that is unintentionally hindering the AGV. It could also be preferable to circumvent the obstacle at a certain side, e.g. if there is a "lane" with oncoming traffic on any side this lane could be avoided in order to not cause any other delays on other AGVs in the system.

3.3.3 Leaving the path

Considering the decision making in previous section a destination point can now be chosen. The initial outbreak from the path will aim to overtake the width of the obstacle. In order to get to this point a new segment from the current position is created. With AGVE's system this is made with a dynamic turn which is described in section 3.1. The objective is to get to an advantageous position in order to make new readings of the object and see if it is possible to pass it and get back to the original path. This leaves the AGV in a new position with new conditions. In figure 3.5 the created segment is shown, circumventing on the side that requires the smallest deviation from the original segment and leading to the desired position next to the obstacle.

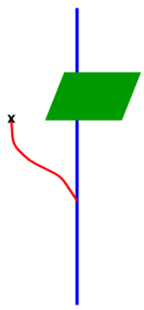


Figure 3.5: Leaving the original path to circumvent the obstacle to the left, and creating the new path to the desired position.

3.3.4 Passing the obstacle

This part aims to pass the obstacle in the direction of the original path. From the new position a new decision needs to be taken. The sensor should now be able to perceive if the obstacle can be overtaken or if even further movement away from the path is needed. This would typically occur if there were unseen part of the obstacle from the previous point of view. Given a clear way in the direction of the path the AGV should drive in that direction until the obstacle has been passed. The sensors needs to read the area between the AGV and the original path in order to see that it is clear for the entire space of the AGV to fit onto the original path. If it is not entirely clear the obstacle has not been passed and the AGV should drive even further. This is done in AGVE's system by creating a new segment between the current position next to the edge of the obstacle and a position straight ahead. The distance to the next position should be quite small but be long enough for the AGV to align better with the original path.

3.3.5 Return to path

For the AGV to find a feasible way back to the original path, it needs to be sure that the obstacle is avoided and passed. To be able to do that, the convex hull of the obstacle should be known since it affects the shape and length of the suggested total circumvention. Given readings from the sensor stating there is nothing in between the sensor and the original path, a safe return trail should be made. Now it is imperative that the load is regarded. Depending on the placement of the sensor it might see that the front of the AGV has passed the obstacle, or the entire AGV has passed it. No part of the AGV should collide with the obstacle whilst returning. This can be made sure by passing the obstacle by a safety margin, or understanding the movements of the AGV. A suggestion is to use the distance from the AGV to the path but in perpendicular direction in the paths direction from the projected AGVs position on the path, see figure 3.6. If the distance to the original path is longer than the sensor range, the new segment should be created as long as the range and then repeat the process until the original path can be reached.

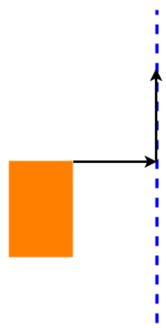


Figure 3.6: Deciding the distance of which to pass the obstacle. The black arrow illustrates the distance from the AGV to the original path(dotted line) and its projection along the path.

3.4 Program flow and algorithms

To circumvent an unknown obstacle, a general program flow has been produced and can be seen in figure 3.7. The sequence will be started when a security check is triggered, i.e. that something blocks the current segment. The obstacle is defined by its outer edges with help of the safety laser scanner, and then the direction of the turn is decided. In reality most AGV systems operate in right-hand traffic, considering that together with the narrow industrial environments they are located in, one reasonable option will be to bypass the obstacle to the left. Before the bypass, a check with the overall layout could be made to see if there is enough space and no wall or other fixed objects. This is done by using the map of the layout each AGV has stored locally. The point will be placed at a security distance from the discovered edge to make sure no collisions occurs.

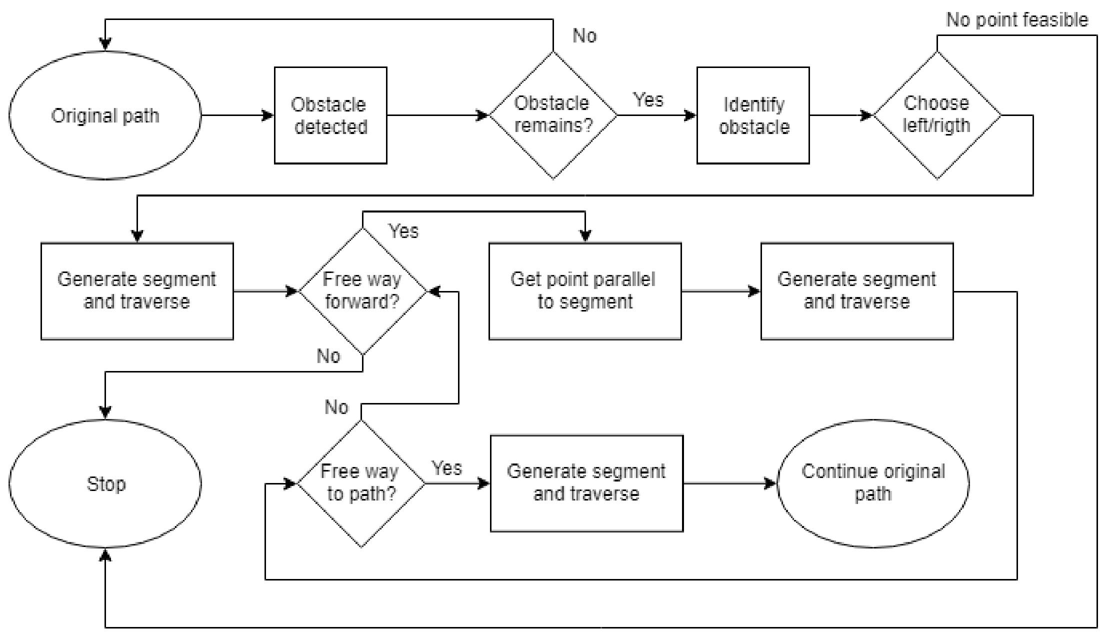


Figure 3.7: Program flow for autonomous circumvention of unknown obstacle

When the point is decided, a segment is created to it and the AGV drives to that position and tries to place itself parallel to the original segment. After that a new scan of the area is made to see if the way is free. If not, it should stop and stand still. If the area is clear, a point e.g. 1 meter ahead is chosen and another segment is created in order for the AGV to move forward, in parallel to the obstacle. When the new position is reached, a scan is made to see if the obstacle is passed and it is possible to return to the original segment. If it is possible, a new segment is created to make the AGV go back to the original path. When the AGV is back at the original path, the total created segment is added and saved to the layout.
The total flow of the autonomous circumvention is achieved by all individual parts working together and passing the right information on between each other. In order for the system to not start any section too early, each section has to signal that it has finished before the algorithm is allowed to move on.

3.4.1 Detecting obstacle

To detect an obstacle appearing in front of the AGV, the area in the AGV's heading direction needs to be continuously scanned and evaluated. Only things that does not belong to the known and expected environment should be considered, meaning no walls or such should be considered, and only points in the direction of the AGVs movement which affects the ability to continuing the movement. The environment is continuously interpreted using the safety laser scanner, and a field in the shape of a rectangle is created in front of the AGV. Points that are detected inside the rectangle are considered as interesting points, and if interesting points appear a signal that alerts the identification function is triggered and the AGV should stop its movement in order to continue with identification of what is detected.
The rectangle field can be seen in Figure 3.8 and has the width of the AGV and a length of 2 m. The scanning rectangle is not wider than the AGV hence no points outside that span affects the AGV's possibility to continue the current path, provided that the AGV is on a straight segment. However, meaning if the obstacle is wider than the rectangle, the whole width of the obstacle will not be measured using this approach. The length of 2 m is to detect the obstacle in time to have enough physical space to make a feasible outrun without having to make a reverse move.

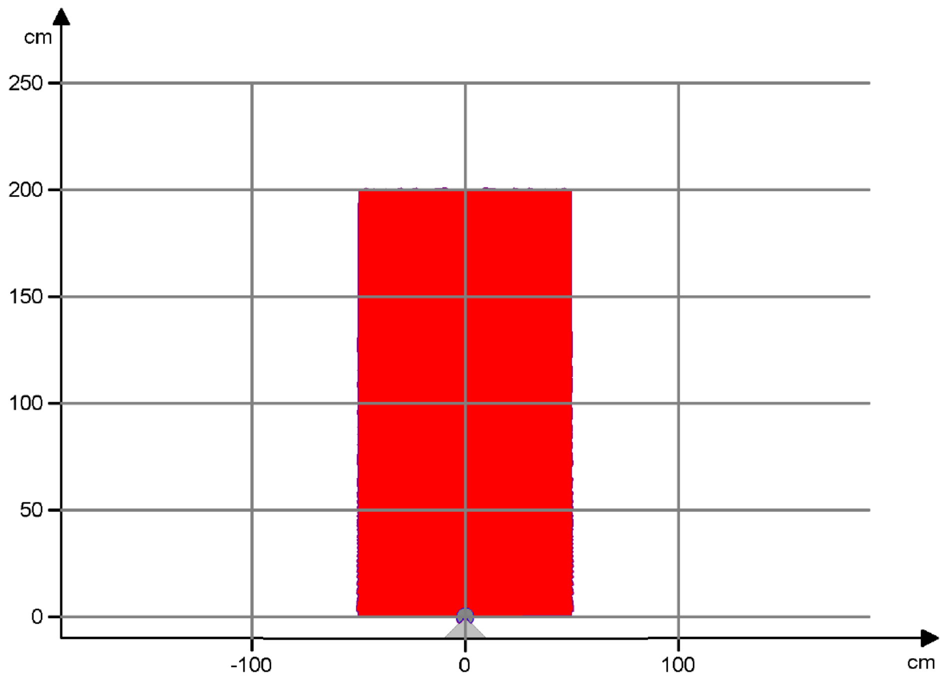


Figure 3.8: The scanning rectangle is centered to align with the center line of the AGV, where the center of the AGV and the safety laser scanner is represented by the grey circle. In this example the width of the AGV is 1 m.

The rectangle should only be activated on straight segments of a certain length and not at any predefined turns to avoid fault stops in the system. This means that when driving a segment that is a curve, the obstacle detection will not be activated, and if there is an obstacle the usual security measures of the safety scanner is used. The AGV will then stop without running into the obstacle but not start the process to circumvent it.

3.4.2 Identifying obstacle

When the identification is called the points from a reading made by the scanner needs to be examined. In order to make sure that the scans are correct, the obstacle actually exists and that no fault points affected the scan, a safety check is made by an median filter. The median filter compares readings of three time steps and chooses the median value, which means that outliers are likely to be removed. The main reason to do this is not to get a fully perfect scan of the obstacle but to make sure there actually is something to circumvent, based on this the median filtering will be accurate enough.

The points from the scanner are filtered by their position relative the AGV, to find those inside the rectangle field presented in figure 3.8. All other points are discarded. For the remaining points, the respective angle relative the center line of the AGV is calculated. Since the choice is to define the obstacle by its edges, only the most outer points are interesting, meaning they can be identified by comparing the angles of the points. The points with angles that deviates the most from the center line are saved to represent the edges. In the case where the obstacle does not pass the center line, the points with the smallest and the biggest angle from the center line are chosen to determine the obstacle. The points that represents the outer edges of the obstacle is then passed to the next function to choose the best direction to make the circumvention. The implementation of the algorithm to identify the edges of the obstacle is shown in Algorithm 1.

3.4.3 Choosing direction

Since the points given can be used to define the obstacle, the points need to be moved in order to be the end of a feasible segment. To do this, a safety margin is added to not create a segment leading directly into the edge of the obstacle. The safety margin will be based on half the width of the AGV (with load) to make sure it will fit and an additional margin that is decided by parameters of the system, e.g. the size of the stop field.

Depending on the layout of the environment, a suggestion for the best choice of point to circumvent is the one closes to the original segment, hence it will require as little extra space as possible. That point is the first to look at, and the position is compared to map of the layout. If there is enough space to circumvent in a safe way that point is chosen, if not the next point is compared to the map in the same way. If there is not enough space to circumvent by any of the points, the AGV stands still. The best point will serve as input to the next function which creates the segment.

Algorithm 1 Identifying obstacle

```
Scan area and save points 3 times
Perform median filtering // Now one set of measurement points remain
for all points do
  b = projected point on AGVs heading direction
  if  distance of b < 2 m then
    a = projected point on perpendicular vector of heading
    if  distance of a < 0.5 * AGV width then
      add point to list
    end if
  end if
end for
for Points within rectangle do
  phi = bearing to point // bearings are measured in relation to heading of AGV,
  where heading = 0
  if phi > bigAng then
    bigAng = phi
    leftEdge = point
  end if
  if phi<smallAng then
    smallAng = phi
    rightEdge = point
  end if
end for
```

3.4.4 Generate segment

As described in section 3.1, dynamic turns are used to create new segments online, and using this will make sure the created segment is feasible and therefor in no need of extra security checks. In order to create a segment by dynamic turns, points of interest is fed to the generating function, in this case the point to reach. Since no automatic sampled points are generated, only manual ones, the created segment is based on clothoids. The dynamic turn can be altered to either do a segment from the starting node to a given point or from the starting node to the end node, bypassing the given point(s). By saving the points of interest at each stage, the segment can be continuously built by generating a new segment each time a new point is found. This means that only one new segment in total is added to the layout independent of the number of iterations and there is no need to add new nodes to circumvent the obstacle.

3.4.5 Get point parallel to segment

In order to pass an obstacle the AGV has to drive by it and recognize that it has been passed. After leaving the original path a new scanning of the area is made. If the scan detects points of interest i.e. points in the heading of the AGV, a new identification of the obstacle should not be made and the AGV should stop the circumvention process. If no points are detected the area in front of the AGV is considered as empty and clear to enter, a segment parallel to the original path is created. The segment is created to a point 1 m straight ahead, by adding the point to the list of points which is used to create the segment. To make a straight segment using the dynamic turns function, the distance is divided into smaller parts which means that the new segment consists of two or more points that are fed to the function.
Since the AGV might not be heading exactly parallel to the original segment, the placement of the point is based on angle information from the original path instead of using the AGV's heading. The distance is measured from where the AGV is positioned at the moment. To guarantee that the new point is feasible to reach for the AGV, it has to not appear in the scan that is performed, meaning if the heading of the AGV is such that the point parallel to the segment is not included in the scan the scan has to be widened more than the rectangle.

3.4.6 Return to segment

To get back to the original path the obstacle needs to be passed by the AGV before a return to it can be made. A new identification of the obstacle needs to be executed to find any points blocking the way back to the original path. The return function is based on detecting points in the direction between the AGV and a certain point on the original path, creating a triangle which is visualized in figure 3.9. The point on the original segment is chosen 1 meter ahead of the current position of the AGV, but placed at the segment.
If no points are detected inside the triangle, the way back to the original path is considered clear to run. The total new segment is created by using the previous

collected and saved points of interest. To create the complete circumvention the start and goal node is connected by the total segment. Once the way back to the original segment is clear, no more points are collected to the total dynamic turn since that function aims to return to the predefined segments as soon as possible.

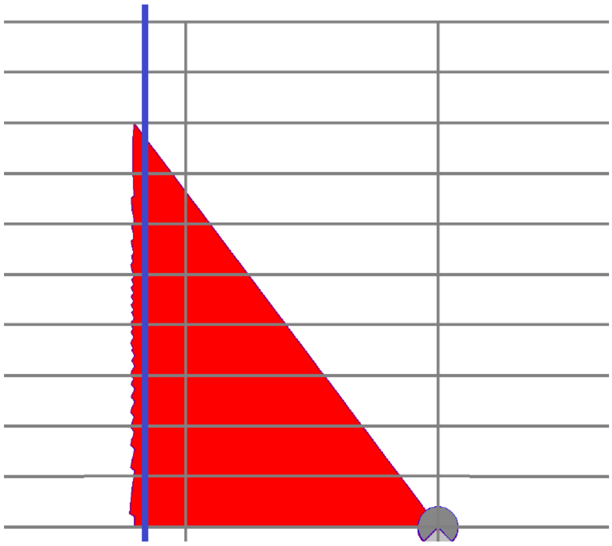


Figure 3.9: The triangle marks where points of interest can be found when checking if the way back to the original segment is clear. The segment is represented by the blue line and the AGV and its safety laser scanner is marked as the grey circle.

If points are detected inside the triangle, the AGV should continue straight ahead and get another point parallel to the original segment. The cases where there is an obstacle placed by the other side of the segment or directly after the end of the triangle are not handled in this circumvention solution.

3.5 Work method and result generation

The implemented functions has been tested in a simulation environment before real life testing. AGVE have a simulation tool that simulates the AGV. This can be run on both an ordinary computer as well as the embedded computer system the AGVs have. It simulates only the truck whereas the rest of the system is the same as used for commercial purposes. The test simulation and proof of concept of function has been made on one of AGVEs embedded computer. It is from these simulations some results have been described. After a function, e.g. object detection, functioned in the simulation environment it was tested on a real AGV with the commercial system.

4

Results

The testing and evaluation of the implemented program and algorithms is performed in a simulated environment as well as on a real tricycle AGV. It is tested at AGVE in their test facility which is an industrial environment in the sense that it has small margins to other equipment as well as workers in the same area. Thus the real testing results is very informative on how the system would work in a live system.

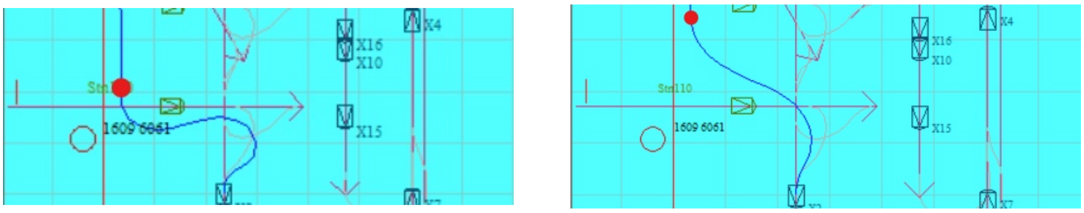
4.1 Simulations

The simulation environment is developed by the engineers at AGVE and makes it possible to implement and evaluate changes and additions in the software that runs on each AGV. The simulation has an instance which makes it possible to send instructions in the exact same way as a traffic controller does in a live running system. Though, there is no implementation of safety sensors in the environment meaning their functionality can not be tested and evaluated here. In the developed method for circumventing an obstacle, the segment generation is suitable for testing in the simulation environment by passing fixed coordinates to the functions since the generation this way does not rely on object identification.

4.1.1 Generating segments

By using pre-decided coordinates to represent points from the obstacle identification, segments can be generated to exactly that point. Generating a dynamic curve to a specific point without tying the start and end node together results in a segment from the start node to the given point and its curvature will vary depending on the distance. The smaller the distance between the current position of the AGV and the point, the turns of the generated segment will be more distinct, while if there is a large distance, the generated segment will be more or less straight.

The properties of the segment are dependent of the given heading and in the case of a cramped area and small distance it is hard to generate a dynamic turn with as few and small turns as possible since the heading and position of the AGV has to end up in a close span of the given point and start heading. If the obstacle is too close to the AGV's current position the segment will have an overshoot when aiming to reach the desired point, meaning more space is required for the movement.



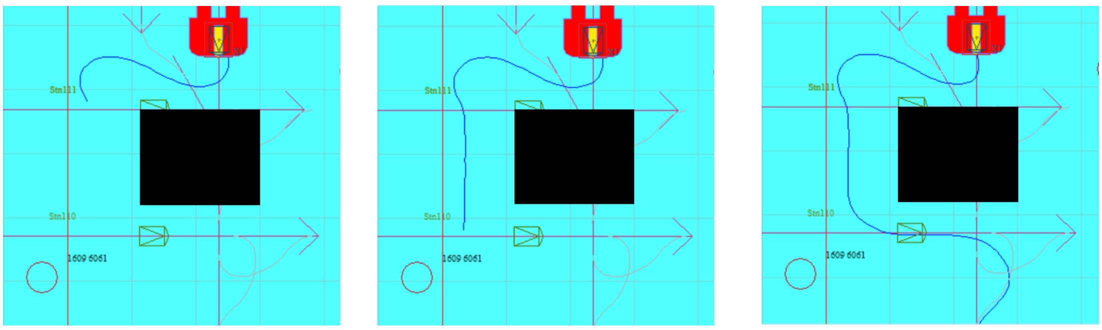
(a) A sharp return to the goal node **(b)** A smoother return to the goal node

Figure 4.1: The curves of the segment defining the return to the original path is dependent of the distance to the goal node

In figure 4.1 two segments are shown, where the red dot marks the position of the AGV when the segment to return was created. As seen in figure 4.1a the distance from the AGV's position to the goal node is smaller than in figure 4.1b which results in a more heavy overshoot of the suggested segment.

4.1.2 Circumvention

The total flow for the circumvention process that can be simulated is shown step by step in figure 4.2. The obstacle visualized in the figure by the black box is only help to the reader to easier understand the movement of the AGV. In the given example the AGV is moving from one node heading towards the next when an obstacle is detected.



(a) The outrun **(b)** The bypass **(c)** The circumvention

Figure 4.2: The stages of creating the new segment that circumvents an obstacle

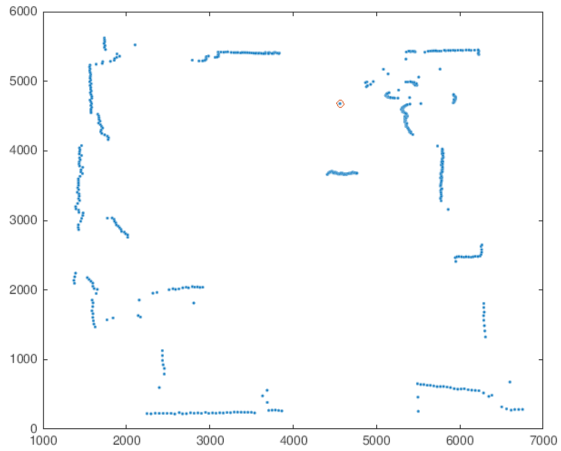
The first segment is created to a point two meters to the right of the original path and one meter ahead, seen from the AGV's point of view. This point is added to the list of interesting points to adapt the circumvention segment to. The generated segment can be seen in figure 4.2a, and is a quite heavy turn making a sharp outrun. In the next step where the bypassing of the obstacle is made, a point straight ahead of the AGV parallel to the original segment is decided. The segment consists of two points, one in the middle of the segment and the end point, these are together with the first point fed to the dynamic turn. The new dynamic turn replaces the old one, and can be seen in 4.2b. The points of interest given to the turn are represented as parts of the segment. When the AGV has passed the obstacle and it is clear to return to the original segment, the dynamic turn is created to connect the start node and the goal node, using the previous captured points of interests as input to the dynamic turn. As seen in figure 4.2c the total dynamic turn replaces the previous one, but this time it cuts the corner when returning to the original path. The points given as input will not exactly be matched in the dynamic turn, due to the clothoid generated will not be able to create a feasible segment that passes by the exact coordinates. The total segment is adapted to reach the points as well as possible but is in some cases not possible, resulting in it might interfere with the area where the obstacle is present or get to close to it resulting in a stop. This has to be taken into consideration when choosing safety margin and is due to the implementation of the dynamic turns and therefor not inside the scope of the thesis.

4.2 Live testing

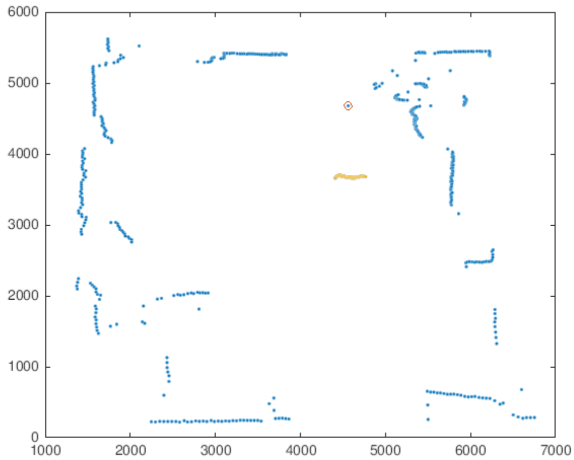
The test of implementation in the real life system is made in two ways; using a test bench with a safety laser scanner and on a real AGV. The obstacle identification was tested on a test bench before moving on to be tested on the AGV.

4.2.1 Detecting and identifying obstacle

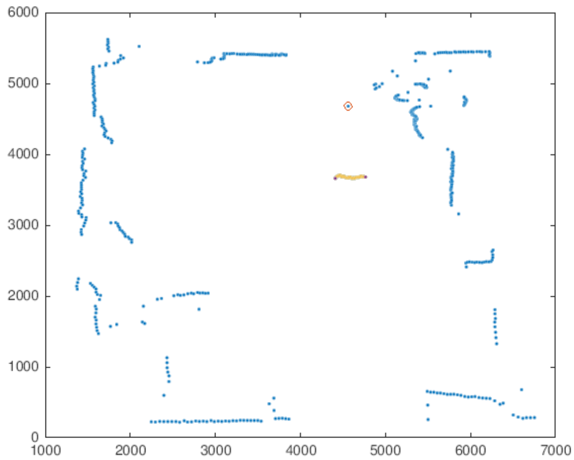
The AGV is able to both detect and identify an obstacle using its safety laser scanner and the detected points. Since the developed method only works with instantaneous readings and no continuous tracking the identification is easily disturbed by the environment. If there is an object identified in the left part of the vision of the AGV and there is an operator standing to the right, not thinking about being detected since not standing close to the segment, the algorithm will try to circumvent the operator as well. This is due to only using the most outer points to define the obstacle, and no other method like clustering of points is being used to identify the obstacle which would make it possible to disregard parts that are detected but probably not belonging to the obstacle.
In figure 4.3a a scan of an area is visualized, with the position of the scanner marked with a red circle heading south. When the rectangle is activated to detect obstacles, the obstacle marked with yellow in figure 4.3b is detected. All other points remain blue, meaning they are not a part of the set of interesting points. Continuing the identification by finding the outer most points to define the edges, the result of figure 4.3c are obtained. The identified points are marked in purple and are considered as the edges of the obstacle. These represent the points that are used to decide which edge to circumvent.
The rectangle boundaries, see section 3.4.2, can be changed to any size but the chosen, the width of AGV, works well by not responding to or considering contours like walls and such at the sides. In the case where the AGV approaches a wall in front of it, e.g. before a planned turn, it will slow down and when close enough start the obstacle detection since no function to decide whether or not it is appropriate to start the process is implemented.
An important part of the obstacle detection is to choose the best way to circumvent the obstacle, based on the available information. The testing is mostly carried out by using an isolated laser scanner on a test bench connected to the simulation



(a) Point cloud representing an area scan, the scanner marked with a red circle



(b) The obstacle is detected, marked with yellow



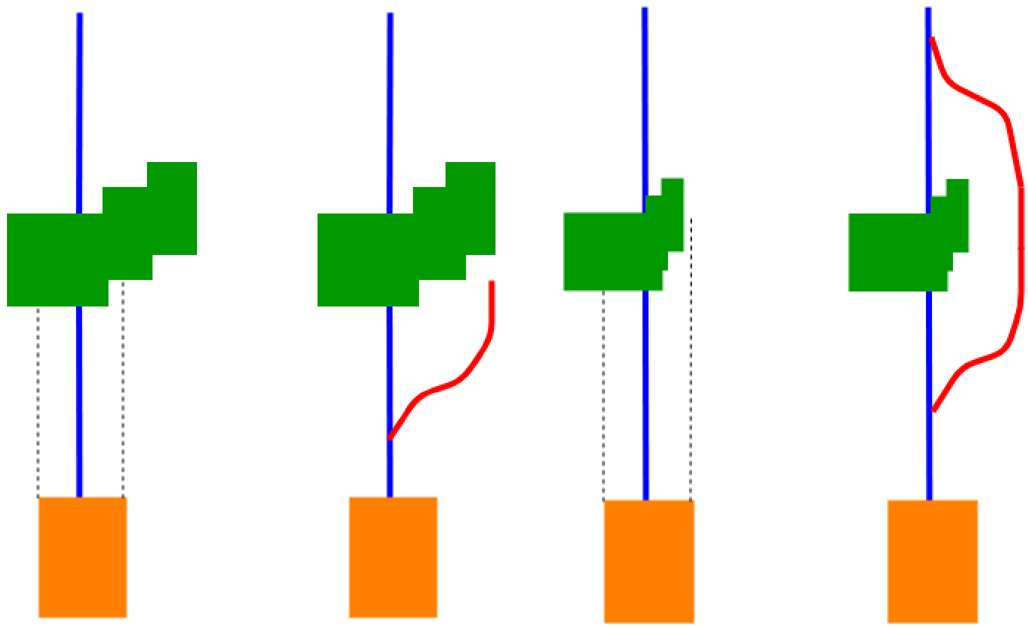
(c) The edges of the obstacle are identified, marked with purple

Figure 4.3: Detection and identification of an obstacle, seen in a global coordinates

environment. Different kinds of objects were placed at several positions to examine the determination of the position of the initial trajectory. The safety scanner does well in detecting and identifying most objects, but there are some types that does not function very well, where one example is a cylinder coated with non-reflective material, which the AGV collided with.

4.2.2 Generating segment

The generation of segments by the dynamic turns function works in reality as in the simulated environment, including cutting corners when creating the segment that connects the start node to the goal node, as presented in figure 4.2.



(a) Failed circumvention **(b)** Successful circumvention

Figure 4.4: The result of circumventing an obstacle of irregular shape

When generating segments around an obstacle of irregular shape, as seen in figure 4.4, the total outcome is dependent of if the whole obstacle fits in the rectangle or not. If the obstacle bulges out of the rectangle on both sides, as in figure 4.4a, the segment leaving the original path will lead the AGV standing right in front of the obstacle and the circumvention process will not continue any further. In the case where the obstacle fits inside the rectangle, on at least one side as in figure 4.4b, the circumvention process will be successful. This means that the circumvention will not detect the obstacle again, once the AGV has left the original path.

4.2.3 Returning to path

The creation of the total segment connecting the nodes is dependent of the instantaneous readings of the scanner being correct. It is deeper described in section 4.2.1, but in this case uses a projection of a triangle instead of a rectangle. If no points are detected the total segment is created, otherwise the AGV continues parallel to the original segment. In the case where there is another obstacle just outside of the triangle, the AGV will stop when it gets close to it since it does not concern anything outside the boundaries of the triangle.

4.2.4 Autonomous circumvention

If the obstacle is placed on a node, or is placed just in front of one, the circumvention process will encounter a problem when generating the final circumvention segment which connects the nodes. The detection and identification of the obstacle will work properly as well as the bypass part, but when the return to the original path is made and the AGV has passed the node or stands besides the node, it will not know where or how to connect the nodes in a reasonable way. The behaviour is, as previously described in section 4.1.1, due to the implementation of dynamic turns to reach the node with a certain heading angle. If there is not enough space between the obstacle and the node, the last part of the segment might be a heavy curve with the result of the forks of the AGV possibly running in to the obstacle due to its rigid body.

5

Discussion

The key findings presented in this master thesis is about making a heavy duty AGV circumvent an obstacle that has presented itself in the way of the AGV. This has been successful given some restrictions such as implemented on straight segments and no nodes being blocked. This worked with the AGVs at AGVE and should be replicable to most AGV systems easily enough. There are however important aspects to consider before implementing this approach. The discussion will focus on the result and the work itself, but it will also be about the choices made and how important they are in industrial environments.

5.1 Method choice

There are some downsides to this type of implementation. Most of all, it is basic compared to other algorithms available today. There are many refined algorithms for scanning obstacles and getting a lot of information about the room. Though, in industry it is a matter of cost and time, and embedded processor chips has historically not been very fast and cheap. AGVE are improving their hardware continuously and the 1 GHz processor is disappearing, but other sensors, such as 3D LIDAR scanners, are still expensive compared to more basic ones. The simplicity gives a functioning implementation but is, as noted in previous section, situational. For broader applications with more advanced detection algorithms additional sensors are recommended be used.

5.2 Perceived result

The final result of the test AGV gives a controlled feeling. It successfully manages to drive around an obstacle without colliding the AGV and the load.

The AGV follows the segment generated by the dynamic turns function as well as it follows the pre-defined segments, except for in situations where the generated curve is heavy and tight. Deviation is continuously calculated by the AGV with help from its laser navigation and can be used as a measure of how easy the given AGV follow the given segments. Though, all segments generated by the dynamic turn are feasible for the AGV and improving them are outside of this thesis.

5.3 A more effective system

It is hard to evaluate the increase in effectiveness, but in the case of AGVE's system it wont become completely blocked in certain situations. This implementation increases the level of automation and removes certain blocked states. As mentioned in chapter 1 this problem should not arise, and in the event it happens when there are workers present they should remove the obstacle. The most valuable improvement would be in a system where workers are not present or when workers are only present during e.g. a day shift with AGVs operating at night. If an obstacle can be circumvented it might mean a difference in one nights operation. This is of course something that needs to be weighed with the traffic controller's performance. If rerouting is possible that might serve some situations better. A suggestion could be to implement a rerouting possibility also, and even compare the two options for best output.

5.4 Spatial occupancy

The extra occupancy that is needed by an AGV for a circumvention is quite apparent. Given the circumstances from section 2.1 this poses quite a problem. Regardless of how small an obstacle is, it will still need quite a lot of space to pass it. Consider a thin wall blocking just a little bit on the side of the path, an AGV will still need considerable space before and after the obstacle in the uncontrolled area, the bigger the AGV the more space required. Together with a load it might require even more space. The load must be considered when planning new trajectories for the AGV in order to prevent collisions with anything in the surroundings. The potential risk of the load colliding with the obstacle is most significant in the outrun and return to the path of the algorithm, and could be dealt with by including more of the movement pattern for the AGV with tail and load when choosing safety measure.

5.5 Implement with caution

The implementation was made using one safety scanner. Even though it works, there are several situations where additional sensors would come in handy. The safety scanner might itself be a source of error, such as misreading points. One problem is if a point is misread as the edge of an obstacle, due to no redundancy of the sensors, a circumvention process could be initiated, or erroneously made. The safety scanner is a good option, but some redundancy might be positive. It has been touched upon in the previous parts of the discussion that there are several *ifs* for this implementation. Even though the numerous examples given throughout the report it is the authors firm view that there are dozens of more scenarios that need special tweaking. This implementation should be considered together with the system, the types of AGVs and the load in mind. It might be impossible for an AGV to deviate even by a small distance from it is original path e.g. in a warehouse system with narrow passages. An open layout gives good possibility to circumvent an obstacle, which should be treated locally. There are places in a

layout where this algorithm proves itself very useful whereas in some spaces it is merely a subject of more frustration. The window of considered obstacles must also be planned depending on where in the layout the AGV currently is. Corners are especially problematic close to walls. Very heavy loads and heavy duty AGVs are more sensitive to changes in the intended path, due to fragile or unsecured load or to increased wear and tear on the components. It is easier for smaller and more agile robots to circumvent obstacles. In a multi AGV site the system can be more flexible if the tasks can be performed by several AGVs. The risk of blocking other AGVs due to a circumvention might be unsatisfactory, in these systems it might be better to remain blocked.
It often boils down to having this function active on very specific segments. All properties are very environmental dependent and a tailored implementation will work best.

5.6 Future work

This thesis work can be considered as proof of concept, due to all limitations the autonomy could be increased. That needs more sensors, since the understanding of the surrounding is the main problem. Some of the problems stated in the discussion is solvable. Several improvements can be done, but due to the limited CPU power it was chosen not to implement any costly filters. There is room for improvement in order to save CPU power. With some new hardware, new sensors and better computational powers it is possible to get very refined solutions to the problem. This might be to implement SLAM of the layout, and especially together with other AGVs this might prove useful by giving information of other parts of the obstacle that is not visible for one AGV.
Another suggestion is to change the setup of the AGV. A tricycle frontal steered truck has notable driving limitations, where some type of quad AGV or an AGV with e.g. mecanum wheels has very small navigational space.

5.6.1 Outer point detection

As mentioned in section 4.2.1 the obstacle is only defined by the outer points which could mean that the detected obstacle is both an obstacle and a worker. Since the safety laser scanner gives no other way to detect the obstacle then by scan points, the work handling those points needs improvement. This could be done by not only using the most outer points and instead look at all detected points to e.g. identify spaces or holes in the detected obstacle to produce a better estimation of it.

5.6.2 Approaching walls

To handle the case where the AGV gets too close to a wall and the rectangle detects it as an obstacle some improvements can be done, one is to deactivate the rectangle at a certain distance from walls or make the length of the rectangle vary depending on the distance to the wall straight ahead. One other solution could be to first consult the map of the layout before starting the obstacle identification to see if the detected obstacle might be a wall or something permanent marked in the layout.

Bibliography

[1] Günter Ullrich et al. "Automated guided vehicle systems". In: *Springer-Verlag Berlin Heidelberg. doi* 10 (2015), pp. 978–3.

[2] Camilla Feledy and Mark Schiller Luttenberger. *A State of the Art Map of the AGVS Technology and a Guideline for How and Where to Use It*. eng. Master Thesis. 2017.

[3] M. Dahl and A. Forsberg. *User-friendly reconfiguration of AGV layouts*. Master Thesis. 2014.

[4] A. Chauhan et al. "Optimal path planning for a non-holonomic robot using interval analysis". In: *2018 Indian Control Conference (ICC)*. Jan. 2018, pp. 184–189. DOI: 10.1109/INDIANCC.2018.8307975.

[5] Mahidzal Dahari and Yang Liu. "A Review of Auto-Guided-Vehicles Routing Algorithms". In: *Advanced Mechanical Design*. Vol. 479. Advanced Materials Research. Trans Tech Publications, Mar. 2012, pp. 443–456. DOI: 10.4028/www.scientific.net/AMR.479-481.443.

[6] Chaoxia Shi, Yanqing Wang, and Jingyu Yang. "A local obstacle avoidance method for mobile robots in partially known environment". In: *Robotics and Autonomous Systems* 58.5 (2010), pp. 425–434. ISSN: 0921-8890. DOI: https://doi.org/10.1016/j.robot.2010.02.005. URL: http://www.sciencedirect.com/science/article/pii/S0921889010000503.

[7] T. T. Hoang et al. "Proposal of algorithms for navigation and obstacles avoidance of autonomous mobile robot". In: (June 2013), pp. 1308–1313. ISSN: 2156-2318. DOI: 10.1109/ICIEA.2013.6566569.

[8] S. Buck et al. "Generic 3D obstacle detection for AGVs using time-of-flight cameras". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Oct. 2016, pp. 4119–4124. DOI: 10.1109/IROS.2016.7759606.

[9] W. Nelson. "Continuous-curvature paths for autonomous vehicles". In: (May 1989), 1260–1264 vol.3. DOI: 10.1109/ROBOT.1989.100153.

[10] V. Digani et al. "Obstacle avoidance for industrial AGVs". In: *2014 IEEE 10th International Conference on Intelligent Computer Communication and Processing (ICCP)*. Sept. 2014, pp. 227–232. DOI: 10.1109/ICCP.2014.6937001.

[11] E. Rimon and D. E. Koditschek. "Exact robot navigation using artificial potential functions". In: *IEEE Transactions on Robotics and Automation* 8.5 (Oct. 1992), pp. 501–518. ISSN: 1042-296X. DOI: 10.1109/70.163777.

[12] Dong Eui Chang and Jerrold E. Marsden. "Gyroscopic Forces and Collision Avoidance with Convex Obstacles". In: *New Trends in Nonlinear Dynamics*

and Control and their Applications. Ed. by Wei Kang, Carlos Borges, and Mingqing Xiao. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 145–159. ISBN: 978-3-540-45056-6.

[13] Engineers at AGVE. private interview. 2018.

[14] Y. Peng et al. "The obstacle detection and obstacle avoidance algorithm based on 2-D lidar". In: *2015 IEEE International Conference on Information and Automation*. Aug. 2015, pp. 1648–1653. DOI: 10.1109/ICInfA.2015.7279550.