Risk Analysis of an Active Traffic Safety System at a Construction Site

June 23, 2022

1st Carl Larsson
cid: larcarl
larcarl@student.chalmers.se

2nd Mårten Lundström
cid: martelu
martelu@student.chalmers.se

3rd Axel Måneskiöld
cid: axelman
axelman@student.chalmers.se

***Abstract*—In this project the implementation of a digital local traffic supervisor is analysed of its reliability in making the crossing safer. A local system has been proposed and an identification of its components has been made. An analysis of failure modes is performed which then works as grounds for proposing updates to the local system. To enable a more thorough analysis, a test system setup is built similar to the proposed local system. The test system setup is also made functional to test on site, enabling it to be implemented into the proposed local system. The results point towards several ways of making the local system better, however none that truly minimize the severity of the risks, as the nature of the crossing is dangerous. A lot of resources can however be put in, to lower the probability of identified risks with the local system.**

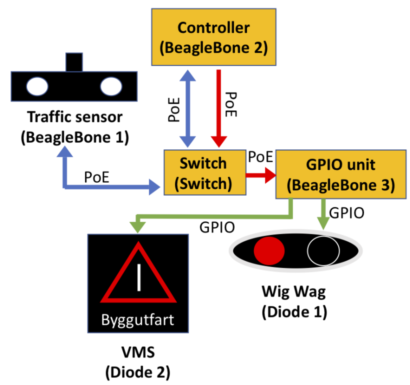

Fig. 1. Local system architecture with associated physical interface. The text within the brackets denotes the corresponding components in the test system setup.

I. Introduction

In an on-going research project between Chalmers Revere, Viscando, Technolution, Ramudden and the Gothenburg Urban Transport Administration, a new type of traffic control system is being developed and tested, aiming at making construction site entry and exit areas safer. The system is based on detecting and locating road users using cameras and machine learning based image processing. So far, a remote server has been used to handle the sensor data, make decisions, and update road signs on site. However, flaws using a remote solution has been identified and an assisting implementation of a local system is of interest. The local system is intended to be more robust and responsive than the original complete system. This project is carried out as a part of the course *SSY226 - Design project in systems, control and mechatronics* at Chalmers University of Technology and has the objective to analyze potential risks with the proposed local system, seen in Figure 1, which is part of a the larger online system of the research project pictured in [Appendix A. Figures 1-2]. Additionally, a sub goal is to build a physical test bench (the test system) of the proposed local system. And lastly, to propose updates for the local system based on the analysis.

A. Related Work

The conflict point of Vulnerable Road Users (VRU) intersecting with heavy vehicles at construction site entrances has been identified as an issue with potential for improvement by the Gothenburg Urban Transport Administration [1]. Based on the Swedish Traffic Accident Data Acquisition (STRADA) database [2], a minimum of 273 traffic accidents between 2010 and 2019 involving vulnerable road users were found to be related to roadworks or construction sites, including one fatal accident in 2018 which increased the efforts put into improving safety related to the entry and exit of construction vehicles in Gothenburg [1].

Gothenburg is not the only municipality working with this issue. Initiatives are for example in place in the United Kingdom, where the Construction Logistics and Community Safety (CLOCS) standard has been put in place to outline the responsibilities for each party involved in traffic safety around construction sites [3]. This includes requiring authorities to allocate enough funds to meet the CLOCS requirements, requiring clients to define last-mile routing for construction vehicles in the Construction Logistics Plan, and requires principal contractors to implement efficient traffic management principles, such as one-way roads, traffic lights and traffic mirrors, as well as appoint traffic marshal(s) to oversee the situation and monitor compliance [3].

The ongoing studies in Gothenburg, which this project is contributing to, focuses primarily on the physical layout of the crossing/intersection. Some actions that have been tested are:

- Portals over bike/pedestrian path with warning signs and warning lights
- Nudging bicyclists to slow down by artificially narrowing bike lanes using road markings
- Flashing lights in conjunction with warning signs
- Illuminated portal intended to warn VRUs

The main purposes of these proposals has been to reduce the velocity of bicyclists, and to increase attention [1]. While some of these actions, such as the nudging markings, did show a slight reduction of bicyclist speed, the report concludes that awareness-raising actions aimed at bicyclists in general has limited potential to improve traffic safety, as there is a risk that warning systems are distracting for the road user, and therefore may reduce the attention on the actual danger in the situation.

B. Local test site

As described earlier, the existing online system is to be complemented with a local controller, which in this project is analysed. The site where the existing online system is implemented is viewed in [Appendix A, Figure 3]. The set up is comprised of traffic sensors that monitor road users in the area, two Wig Wag signs that signal to construction vehicles if they should stop or not, and VMSs that signal to pedestrians and bicyclists. The sensors send HTTP requests with information about all detected road users currently in the area, which the controllers act on and use to update the Wig Wags and VMSs.

TABLE I
ABBREVIATIONS

FMEA	Failure Mode and Effects Analysis
GPIO	General Purpose Input/Output
SQL	Structured Query Language
TTC	Time To Crossing
VMS	Variable Message Sign
VRU	Vulnerable Road User
UPS	Uninterruptable Power Supply

II. METHOD/PROPOSED SOLUTION

The project was carried out in three different parts resulting in proposed updates. The first part established how the local system architecture was setup. This prepared for the second, and third step of identifying risks in the local system, and directing the approach for a physical testing environment, respectively. Lastly, a clustered description of failure modes and their possible solutions are presented.

A. Identification of the local system

For effectively conducting a Failure Mode and Effects Analysis (FMEA), a description of the local system is made. The analysis can be divided in two different sub parts where the first look at overall functions of the local system, e.g. ”warn VRUs”, and the second looks at individual components and the interfaces between them. The identified functions are *Detect object*, *Track object*, *Warn VRUs*, *Stop VRUs* and *Stop construction vehicle*. A list of identified components for the complete (online) system can be seen in [Appendix A, Table 1]. It is however only the Local Controller, the Traffic Sensors, the Traffic Signs, and the Power Grid that concerns the local implementation.

B. FMEA

The analysis was carried out by brainstorming possible failure modes, and their possible causes for different parts of the local system. For each possible cause of failure a severity rating, a probability rating, and a detectability rating is judged.

1) Severity: The severity rating tries to capture how severe the effect of the fault would be. In the crossing people might get hurt if it goes wrong, which is more severe than for example a truck having to wait for a long time. This is the most important measure to reduce.

2) Probability: Probability is the next measure which tries to describe how likely the failure mode is to occur for each potential cause. It is desirable to have a low probability of failure modes to occur, however not as important as lowering the severity of the different effects. A low probability index would indicate that it happens rarely, and a high that it happens often. This measure is also important for the safety of the system, but only second to severity, and accepted to be larger, as long as the severity is low.

3) Detectability: The third measure, detectability, describes how unlikely the fault is to be detected, a higher index means harder to detect. This is mainly used to propose solutions, and could for example determine if a solution can be reactive or needs to be proactive.

The analysis in its entirety tries to capture scenarios where some part of the local system fails, and detect what risks are happening in the traffic area. It does so by looking at each identified function and part of the system, in turn, and allows for brainstorming predictions of what failure modes and scenarios can occur. Added to each failure mode is also Effects and Potential Causes, which describes what can happen if the failure mode presents itself and what could be the underlying cause. All these possible failure modes that are generated gets scored for Severity, Probability and Detectability. These ratings are viewed as a means to direct focus for describing solutions. In this work, most of the attention is put to reduce the severity and likelihood of failure modes, but detectability also influence the shape of the proposed solutions. Modes with high severity, above 2 on a scale from 1 to 5, are seen as unacceptable and solutions to reduce the rating are thought of. However, all severity ratings cannot be reduced below 3 and in these cases the probability of it happening is instead targeted.

The analysis also takes into account how the road users may perceive different scenarios and what actions they might risk taking. These failure modes and effects might be impossible to cover in its entirety and harder to prevent but still needs consideration.

C. Test System Setup

The test system is built for experimenting with and invoke imagination of different failure modes that could present themselves. It is thus mainly used as a tool in the analysis. There is however a second sub goal for building the test system which is to test the idea of a local controller on site, a proof of concept. It will not pose as a prototype for further production, but rather a testing platform.

The test system setup is built to represent the different functions in the proposed system shown in Figure 1, using emulated data, with the option of using real data. The actual setup of this test system is briefly described in Table II and visualized in Figure 2.

TABLE II
TEST SYSTEM SETUP COMPONENTS.

Physical part	**Functions**
Switch	Provides a http network to send data between the different components of the test system.
Beaglebone 1	Serves as emulating the traffic sensor and sends HTTP requests to the controller.
Beaglebone 2	Serves as the main controller receiving sensor data, computing trajectories, hosting an SQL server, and sending state updates to Beaglebone 3.
Beaglebone 3	Serves as a GPIO interface to the traffic signals. It takes HTTP requests and updates the signs accordingly.
Computer	Connected to the network to update parameters of the Beaglebones and read streamed data.
Relay board	If the test system is to be used for testing on site and connected to the different digital signals, then this is needed.

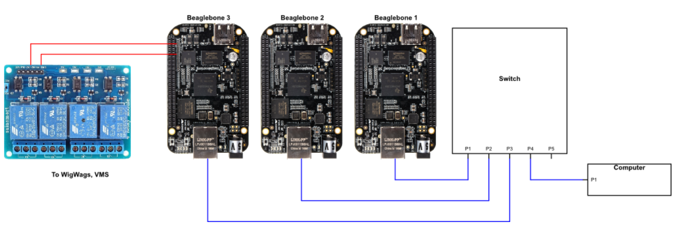

Fig. 2. Physical setup of test system (power excluded)

1) Traffic Sensor Emulator: For the test system setup, several functions were constructed to represent the complete local system. The first part consisted of a traffic sensor emulator generating realistic data for the system. The functionality was to generate road users with different properties, i.e. class, speed, heading. The data is then transmitted to the controller by an HTTP POST request made by the traffic sensor emulator.

2) Controller: A controller was developed to execute the system functions based on the input about the traffic situation received from the traffic sensor emulator. The controller consists of two software components - the trajectory engine and the rule engine.

The trajectory engine performs a coordinate shift and rotation in order to improve the usefulness of the locational data received. It then creates an object of the object class `vehicle` (pedestrians can also be created as this object class). The parameters and functions of this object class are as follows:

- `objType`: The type of object, for example ”truck”, ”bike” or ”pedestrian”.
- `x, y`: Position data, where the origin is located in the center of the crossing. As the x-axis is defined along the bike/pedestrian path, a positive y-value indicates the truck is on the road and may intend to enter the construction site.
- `v`: Absolute velocity of the vehicle.
- `vin`: Vehicle velocity relative to the crossing. Will be negative if the vehicle is travelling away from the crossing.
- `distance()`: The distance from the vehicle to the center of the crossing. Calculated using the vehicle’s position data and the Pythagorean theorem.
- `ttc()`: Time-To-Crossing, the estimated time until the vehicle reaches the center of the crossing. Calculated using the vehicle’s `distance()` and `vin`.

Lastly, a function returns data for the truck (if present), as well as the nearest VRU in terms of time-to-crossing (if present). This function is used by the rule engine.

The rule engine continously envokes the trajectory engine to retrieve the most recent data about the traffic objects in the crossing, and a timed automaton is then used to decide which state should be selected by the system. The decision making is based on the current state, waiting time for the truck and the movement of VRUs in the intersection. For example, it will not let the truck wait indefinitely even if there are many VRUs crossing the intersection, instead it will after a set time transition to perform the *Stop VRUs* function in order to let the truck pass.

If the state is changed, the output to the VMS and WigWag systems will be updated accordingly by sending an HTTP POST request to Beaglebone 3 (GPIO unit). The trajectory engine as well as the rule engine are realized using a Python script running on Beaglebone 2, and the rule engine behaviour is described by a Moore machine which can be found in [Appendix B].

3) Database: Information about the road users recognized by the traffic sensor gets transferred to a local database. The real-time operation gets performed when the trajectory engine has laid out a trajectory for a road user. The table in the database contains information about the road users given by the traffic sensor and calculations from the trajectory engine. The database management system is MariaDB and hence Structured Query Language (SQL) is used for managing the data.

4) Visualization and User Interface: To visualize information from the database a frontend interface is used. The information shown is the road user currently in the crossing, statistics about the number of users per hour and weekday. The information is collected in real-time from the database through PHP and Javascript. Moreover, the information gets displayed by using the JavaScript Graphing Library Plotly. This consists of a scatter plot for the current traffic situation in the crossing and bar charts for the hour and weekday statistics. The information are displayed through a HTML page hosted by Beaglebone 2. Accessing the page is made through connecting a computer by ethernet to the switch and enter the Beaglebone 2 IP-address in a web browser. A view of the page can be seen in [Appendix C].

Furthermore, for easier usage of the interface, the different parameters of the system are stored in the local database. During the start of the system, the user can choose to load the different parameters of the system from previously saved settings.

5) GPIO unit: This part represents the general purpose input/output component of the system. A solution would need to take the state of the local system's controller and display this on the WigWag and VMS. For this, a Beaglebone receives the HTTP POST request made by the controller when the state is changed. When the POST request received the Beaglebone 3 check if the state is the same as before. If this is not the case the Beaglebone sends out a 3V signal. In the test system, this is represented by diodes that corresponds to different states being active. However, the system also have the option of being implemented into the local site setup. When this is made the test system sends out a digital three voltage output to a relay, which in turn shorts the signs' individual signal cables to ground.

III. Results

The main interest of this project is to identify risks with the local system and propose possible solutions to the discoveries. Through identifying the components, performing an FMEA, and thinking about the test system, the following subsections each describe a failure mode and proposed methods that could lower the severity and likelihood of the identified risks. The most severe risk identified is a situation where the local system's signs suddenly goes dark, indirectly indicating that it is safe to pass the crossing. What follows is clustered failure modes and their possible mitigating solutions. Some of them reference a 'safe state' which is a system state in which the least amount of risk is experienced. This state is possibly a state where all road users are instructed to stay put, informed of system failure, and/or to proceed with caution. The safe state also assumes a defined safe transition to the safe state.

A. System does not show any warning or stop signal

The most common risk is if the warnings do not show correctly or the truck stop light is not working properly. There is a multitude of reasons that it could happen and it is not possible to envision them all. Examples of dangerous situations are lights going dark suddenly and road users believe they function as usual and everyone enters the crossing simultaneously, and road users getting accustomed to dark signals signalling that there is no danger to cross. This can also be a result of false negatives of the sensors, i.e. no signal is shown as road users are not detected properly.

1) Complementary regular signs: Complementary to the digital signs, regular traffic signs that describe the situation that the road users are approaching, e.g. signs showing a crossing, or warn about crossing construction vehicles with limited view.

2) Complementary traffic mirrors: When the system does not warn road users of other road users, they need to perceive them themselves. Using mirrors to reach around corners and the often obstructing walls of the construction site works in tandem with the digital sign system.

3) Use UPS: An Uninterruptible Power Supply (UPS) could be used in parts of the system that is critical, e.g. the digital signs or the GPIO unit, or on the entire system all together. A power outage, either in some parts of the system or everything, can leave a hazardous situation in its wake. A UPS could either uphold the entire system, or atleast uphold default, safer states of the signs.

4) Double signs: In a situation where only part of the system breaks, in this case a signal or power cable to a digital sign, a supplementary sign with different cable routing can make for redundancy.

B. Network and connection failures

The local system relies on having an uninterrupted flow of information between the sensors and the signs. When any or all of these connections go bad, either momentarily or for longer times, the road users are at risk of not getting the correct information.

1) Safe default states: All components should be able to default to a safe state, in a safe way, if it finds any crucial part of the network not responding properly. An example would be the GPIO unit that should default, in a safe manner, to a state which gives a regular warning or the information that the system is offline.

2) Notify a supervisor: When a part of the system is unresponsive, a supervisor should be notified automatically.

3) Use a backup WIFI connection: Let a separate WiFi connection run in tandem at all times when the system is running. If something goes wrong with the ethernet network, then the WiFi could take over and secure the system's functionality. A notification to a supervisor about the switch over could also be used.

4) Double GPIO units: The GPIO has been identified as the most critical part of the system. If it fails, or connections to the signs fail, no functionality in the system can replace it. For more redundancy in this case, another GPIO unit could be used in tandem to ensure functionality.

C. Irrational behaviour of road users is detected

The sensors are required to give an accurate representation of what happens in the crossing. A road user could however be detected not behaving as usual, due to bad sensor data or just irrational behaviour from the individual road user. The local system needs to be adapting to unintentional situations, e.g. where a cyclist ignores warnings and pedals towards an incoming truck. Another possible fault could be bad sensor data that thinks a pedestrian is standing in harms way even though they are in a safe zone. This situation should probably default to trigger a safe state.

1) Detect unlikely behaviour: Implement limits for likely behaviour of road users, e.g. setting a limit on bikes to 50km/h, if they are faster than that they are probably identified wrong. When a limit is reached a safe state could be set while waiting out the abnormality.

2) Override current state: Implement transitions to safe states if irrational behaviour is detected. For example, if a bicyclist ignores warnings and stop messages a truck should not be let in to the crossing. The truck might have had the right of way, but as the the bicyclist did not slow down according to the warnings, the system should change state and stop the truck from going in to the crossing.

D. Bad sensor data

The local system relies on correct data from the sensors and dangerous situations could present themselves if the data is wrong.

1) Use overlapping sensors: If a sensor is obstructed or damaged, a redundant system of multiple sensors with overlapping fields of view could be used.

2) Regular inspection: The system is sensitive to blocked sensors and signs. A regular inspection of these to remove dust, snow and obstructing objects could reduce unsafe system behaviour.

3) Learn to recognize faults: Sensors could be used to also read conditions of the site and sensor health. If bad weather is detected, measures could be taken. If snow obstruction is detected, a supervisor could be notified.

IV. Discussion

A main conclusion that was drawn from the FMEA was that it proved difficult to reduce the severity of the identified risks of failure. The most severe risk, where the local system suddenly goes dark, was also not eliminated.

The proposed actions rather work to reduce the probability of the failure occurring, which is important for improving the reliability of the system. This has been identified as an issue with warning systems tested in the past, with false warning being a specific issue to be addressed [1]. By the proposed actions, the risk for this should be reduced. However, improving the reliability will only work to reduce the likelihood of a severe accident.

In the end, it is however very difficult to eliminate the risk of collision completely with a warning system, as this would likely require a physical barrier similar to a railway crossing, or another form of obstruction. One of the proposed actions, adding mirrors to the crossing, could have some reducing effect on the severity as it allows the truck driver to see road users it would not otherwise have seen.

However, the individual road user still has a responsibility to act safely in traffic and obey by traffic laws. Future studies could therefore investigate the effect of which message is displayed on the VMS, for example if a stop sign was displayed instead of a warning sign. It could also be investigated if other actions, such as using road markings as a way of nudging, which have in previous studies shown positive results, could be used in combination with the active warning system.

V. Conclusion

Through analysing a test system of a local controller that can be implemented in the complete local system, a resulting list of proposed additions to the system is made. The solutions however, fail to reduce the severity of the identified risks to an acceptable level. All solutions, except mirrors to some extent, can be viewed only reducing the likelihood of risk events. We find this to be a challenging problem when the situation itself, i.e. the crossing, is an environment heavily dependant on the individuals therein. The augmented crossing with a supervising system can only do so much as to support awareness.

References

[1] A. Niska, J. Karlström, and J. Eriksson, *Construction traffic in Gothenburg. Planning and implementation for increased safety for cyclists and pedestrians*, ser. VTI rapport 1094. Swedish National Road and Transport Research Institute (VTI), 2021.

[2] Transportstyrelsen, "Om olycksdatabasen Strada," October 2021. [Online]. Available: https://www.transportstyrelsen.se/sv/vagtrafik/statistik/olycksstatistik/om-strada/

[3] Construction Logistics and Community Safety, "CLOCS Standard Version 3," January 2019. [Online]. Available: https://www.clocs.org.uk/resources/clocs_standard.pdf

Appendix A: Complete System Overview

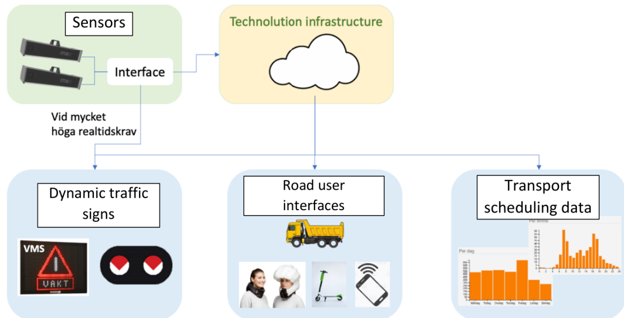

Figure 1: A simplified presentation of the complete online system.

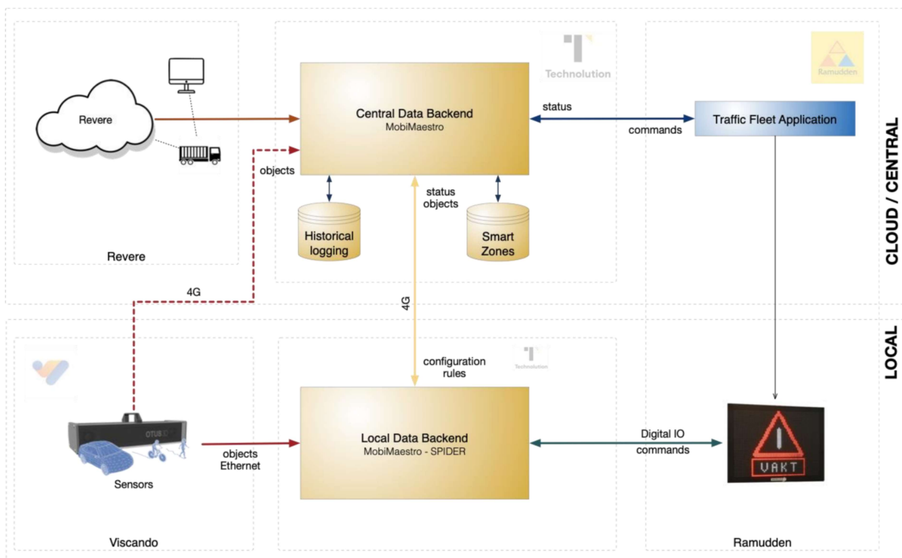

Figure 2: A more detailed representation of the complete online system.

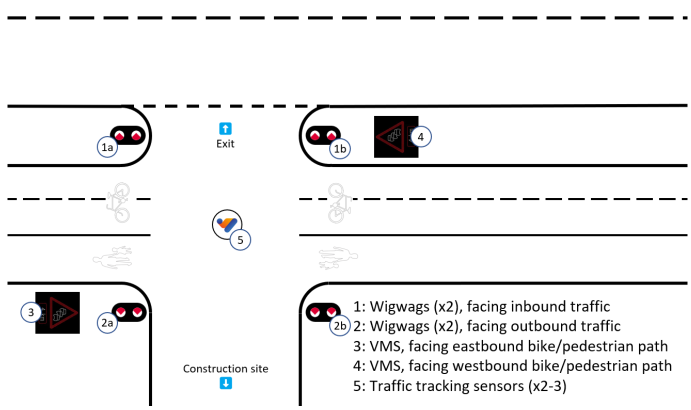

Figure 3: Overview of the local test site.

Table 1: Short description of the complete online system's components.

Part number	Name	Description
1	Local controller	Consists of a computer, switch and GPIO board and is the main focus of this project.
1.1	Controller (computer)	The local Linux computer. It runs the local controlling of the traffic signs, using sensor data. It is managed by MobiMaestro SPIDER and runs a trajectories engine, a rule engine, and an API for the GPIO board.
1.2	Switch	The local switch connecting the computer, trafic sensors and GPIO board together over ethernet. The switch is POE with a minimum of 2 powered connections and 8 in total. The POE standard of IEEE802.3at needs to be supported.
1.3	GPIO unit	A MOXA IOLOGIC E1211 (or similar) universal I/O of sink type, that takes a http request and controls sink value pins connected to the traffic signs. It has a minimum of 8 output connections. GPIO id [0] is for image A of the VMS displays, id [1] for image B, and id [2] for controlling the Wig Wags.
2	Traffic sensors	Uses live imaging to detect road users and their positions. May consist of several sensors, all connected to the local controller.
3	Remote server	The central cloud environment handling the controlling of the traffic signals originally. Will serve as a backup and central control agent in the new system to be tested. Also handles communication with other remote services such as Revere and Traffic Fleet Application.
3.1	Historical logging	A sub part of the remote server which keeps a record of the traffic situation state.
4	Traffic Fleet Application	Ramudden's current local driver for the VMS displays and Wig Wags.
5	Traffic signs	All of the different traffic signs in the entry exit area. Consists of VMS displays and Wig Wags.
5.1	VMS displays	Variable Message Sign display. Can view a set of images and animations. In this projects working example there are two of them aimed in opposite directions in the bike path.
5.2	Wig wags	A set of red signals that can either be on or off. They are aimed at the heavy traffic and in the working example consists of 2 pairs in opposite direction.
6	Revere	Vehicle data service that provides gps positions of the construction vehicles.
7	Power Grid	The local power grid supplying all of the local components on site with electricity

Appendix B: Test system overview

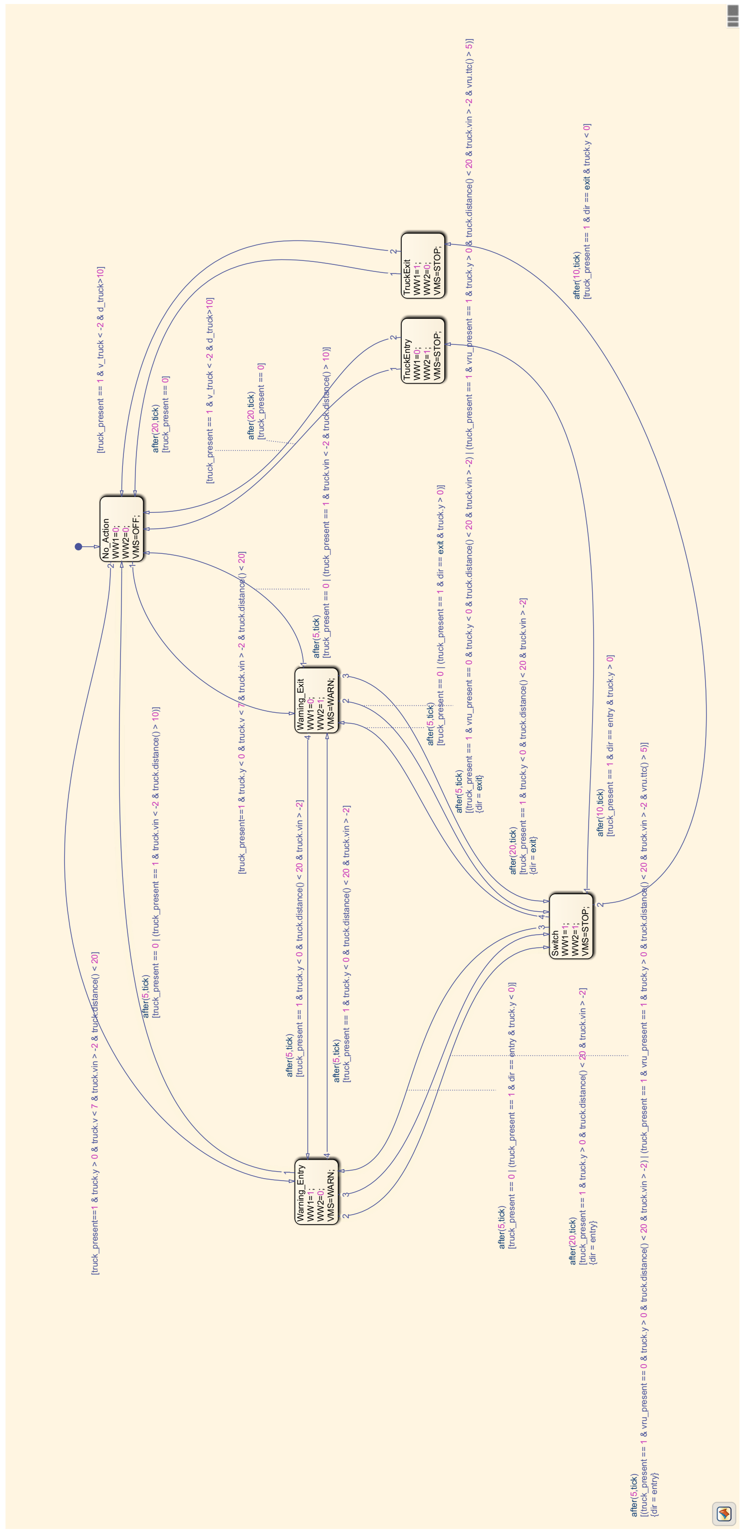

Figure 1: Moore machine describing the rule engine

Appendix C: Overview of the interface

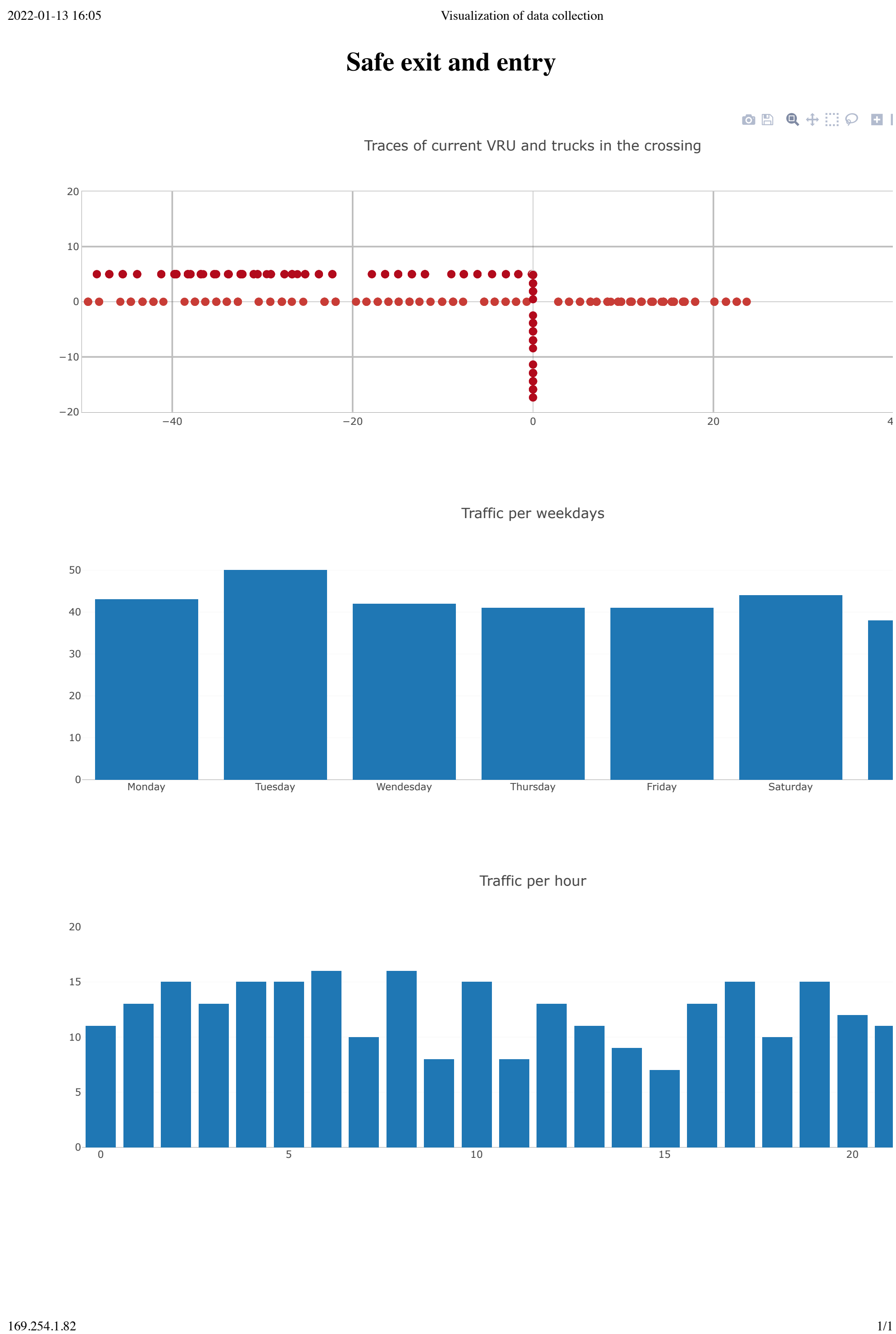

Figure 1: Website containing real-time information about the road users in the crossing and statistics about previous traffic in the crossing