Modernizing process control system in pavement operations

PDEng project final report

Afshin Jamshidi
December 2021

Department of Civil Engineering

UNIVERSITY OF TWENTE.

Type	Project Final Report
Project	Modernizing process control systems in pavement operations
Name	Afshin Jamshidi
Educational institution:	University of Twente
Faculty	Faculty of Engineering Technology
Daily supervisors	Dr. ir. Faridaddin Vahdatikhaki
	Dr. ir. Seirgei Miller
Thesis supervisor	Prof. Dr. ir André Dorée
Company supervisors	ir. Berwich Sluer (Boskalis)
	Marcel Sprenger, ir. Marjolein Galesloot (BAM)
	ir. Jasper Keizer (KWS)
Date	17 December 2021

Acknowledgment

I would like to thank my university and company supervisors for their help and, also my friends in the Netherlands with whom I had great experiences. Also, thank you to my family for their continued support.

LIST OF ACRONYMS

ECTS	European Credit Transfer System
ASPARi	Asphalt Paving Research and innovation
PQi	Process Quality improvement
SDLC	System Development Life Cyle
BLE	Bluetooth low energy
RTK	Real-time kinematic
M2M	Machine-to-Machine
HMA	Hot Mix Asphalt
GPS	Global positioning system
SQL	Structured Query Language
NoSQL	non-SQL (originally), not only SQL (recently)

PREFACE

"Modernizing process control systems in pavement operations" is the PDEng project which is sponsored by ASPARi[1] in cooperation with the University of Twente. ASPARi is a network of construction companies working collaboratively towards improving construction processes and the quality of asphalt roads.

This document serves as a final report of the PDEng project. There are 9 chapters that are described in this document:

Chapter 1: Introduction

Chapter 2: Design methodology

Chapter 3: Function requirement of the system

Chapter 4: Technical design of the system

Chapter 5: Implementation

Chapter 6: Case study and validation

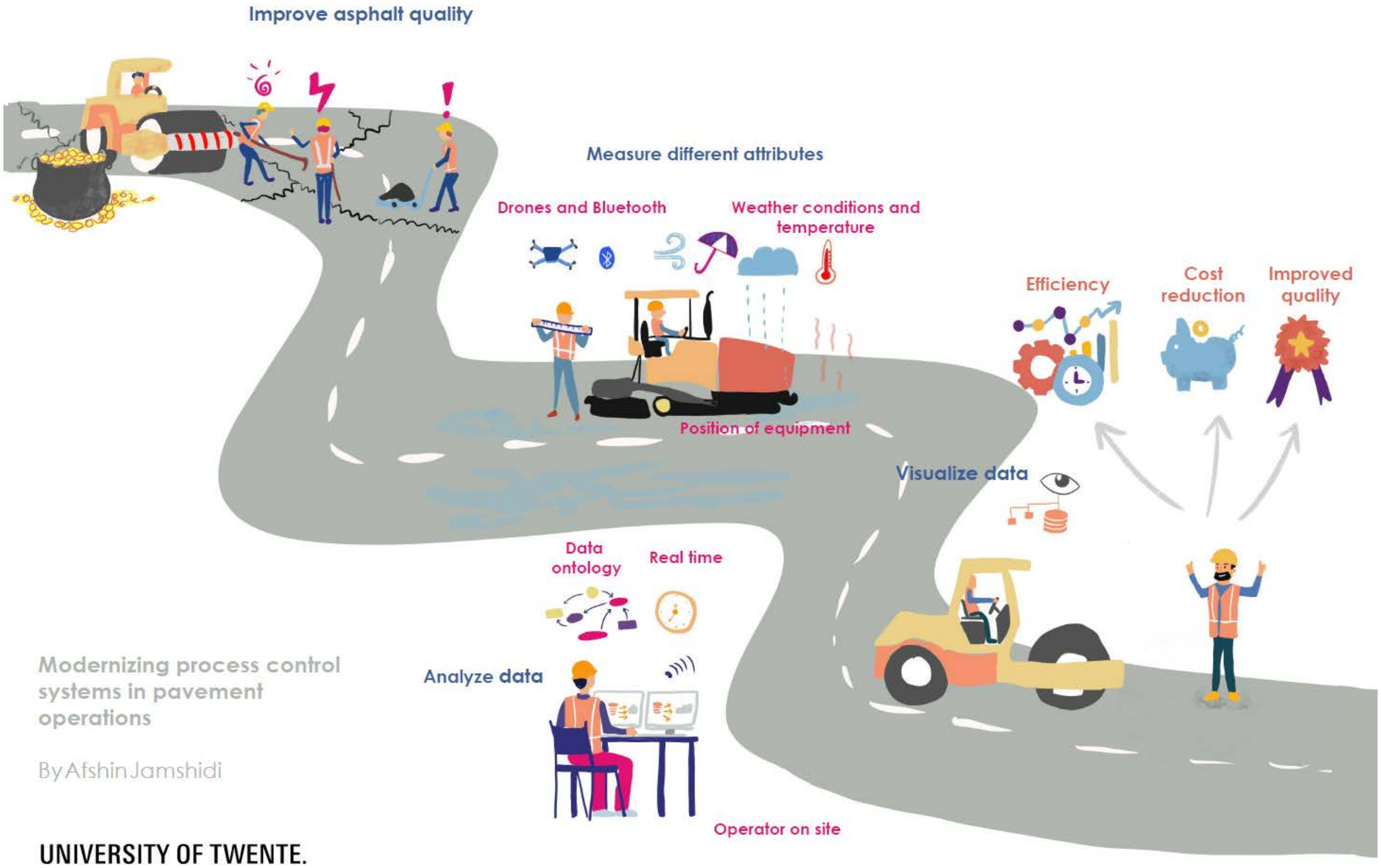
Chapter 7: Exploration of alternative sensors

Chapter 8: Discussion and recommendation

Chapter 9: Conclusion and Future work

[1] **As**phalt **Pa**ving **R**esearch and **I**nnovation

Graphical summary



Executive summary

Asphalt construction is a very complex process where many variables can affect the final quality. Having this in mind, asphalt pavements usually are mostly constructed based on the operator experience and their intuition. ASPARi is a network of asphalt contractors which, with the help of University of Twente studies methods for improving asphalt quality. Process Quality improvement (PQi) is a method, developed in ASPARi for controlling and improving primary processes at an asphalt construction site.

The currently available tools in ASPARi, although working, are mostly used in offline mode and are based on older (outdated) methods. This project provides methods for modernizing hardware and software and helps the operators and onsite managers to better control the construction process in real-time.

As a working prototype, the output and the features of the system can be used as a reference for asphalt contractors to raise their expectations regarding real-time process control system development. They can use the improved ASPARi PQi product to firstly, demonstrate and secondly, request similar or more advanced features from equipment manufactures and other relevant vendors. The adoption of the developed features in future machines should lead to higher asphalt quality.

Product summary

After more than 10 years of research and developments. ASPARi has an established Process Quality Improvement (PQi) methodology that uses a custom-made system, consisting of a set of methods, hardware, and software, to capture, monitor, and analyze paving operations on site. Although the system used for PQi methodology is stable and effective, given that technologies used in this system are nearly 10 years old, it has a number of limitations. These include: (1) the current system has very little support for real-time application, which is essential for providing better support for paving crew on site; (2) a transparent data structure that can support a wider integration of PQi data with other lifecycle pavement management data is missing; (3) logistically, the preparation of PQi system on site is cumbersome, disruptive and labour-intensive; and finally (4) the architecture of the system offers limited extensibility and therefore is not very future-proof. This project aimed for modernizing this system in term of both hardware and software. The main contributions and innovative aspects of this design project can be summarized (in order of impact):

- Improvement of the overall Technology Readiness Level (TRL) of real-time PQi system from TRL 5 to TRL 7, i.e., transition from a validated concept to an established/stable prototype capable of working in operational environment;
- Deployment of ontological modelling and structured relational database for the management of PQi data. This has improved scalability, transparency, extensibility, integrability, and interoperability of the developed system. This was demonstrated through easy conversion of the PQi data into (some of) specific platforms used by different contractors;
- Implementation of multiple levels of automation (i.e., assistance, semi-guidance, and guidance) in the real-time PQi system for the first time;
- Development of user-friendly interface that allows non-experts to use the developed system;

To achieve the above-mentioned contributions several challenges had to be overcome, including but not limited to, (a) lack of trust in the technological tools among the asphalt crew on the construction sites, (b) absence of a clear implementation strategy in the earlier versions of real-time PQi system from the software and hardware development standpoint, (c) insufficient insight into various complex use case scenarios in which a potential stable real-time PQi system should function, and (d) issues regarding the reliability and accuracy of localization components of PQi system under various operational conditions.

With the development done in this research and through the design/implementation of a new system architecture (including IoT nodes, relational database, loose coupling between modules, etc.), the new PQI system can now be used for real-time applications, albeit with caution because the short time of the project did not allow the optimization of the architecture for all possible scenarios. Additionally, this research made the first foray into investigating new alternatives that can improve the PQI system from the perspective of accuracy, practicality, interoperability, and reliability.

An important disclaimer needs to be made about the contribution of this design product. Although this product modernized the PQi system drastically, it mainly does so from the system development perspective. In other words, this design project does not claim any contributions to the development of novel data analytics methods. The ASPARi team at the University of Twente has been developing these methods (e.g., TCP, CCP, priority map, guidance map, cooling curve, rasterization, etc.) over the past few years from the scientific standpoint and mainly at the conceptual level. This design project was the first effort to realize these conceptual ideas and incorporate them into the existing system in a more stable and robust manner, making them an integral and mainstream part of PQi methodology. Therefore, the final product of this research can be considered as a significant development step that allows the ASPARi network to apply state-of-the-art data analytics methods in real-time. Additionally, the ontology-based data structure developed in this research offers a level of extensibility that is new in the ASPARi PQi system. This allows the ASPARi team and contractors to more conveniently explore new technologies and ideas and incorporate them in the system going forward.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

The contractual structure of Dutch construction industry has changed significantly over the last 3 decades (Dorée, 2003). As a part of these changes, contractors have to provide longer guarantee and warranty periods. Contractors have to repair the road and may be penalised for disturbing traffic flow during maintenance (ter Huerne, 2006). As the regulations have changed, seven to ten -year guarantees is now a standard for regular projects. For large highway projects the maintenance period can be even up to twenty years (Arbeider, 2017).

In the light of these changes, contractors sensed the need to focus on improving the construction process and quality control practices to ensure high quality asphalt. This has motivated contractors to scrutinize their current practices, analyze the limitations and shortcomings, and develop improvement strategies. To this end, major contractors in the Netherlands together with the University of Twente formed a large consortium (ASPARi) to build up joint momentum towards advancing the industry as a whole and to consult professionalizing the paving process. The current project falls under the umbrella of the ASPARi initiative and intends to improve/modernize the technological instrumentation used for the monitoring of paving processes.

1.1 Overview of ASPARi history

The ASPARi research unit started from 2007 at the University of Twente. So far more than 60 under – and post – graduate research projects has been undertaken to achieve higher asphalt quality and more than 100 projects have been monitored with the PQi methodology developed by ASPARi researchers.

1.2 The need for PQi measurements

Figure 2 presents an overview of a typical paving operation. In this operation, a paver lays the hot asphalt mix on the base layer and a fleet of rollers apply compaction force on the asphalt layer to ensure that the desired density is achieved.

Previous research stressed the importance of the construction phase on the overall quality of the asphalt layer (Miller S. R., 2010) (Bijleveld F. , 2015) (Vasenev, 2015). Studies also show that asphalt construction is mainly based on the implicit knowledge of the practitioners and the compaction strategy is primarily led by the experience/intuition of equipment operators. The absence of an explicit methodology for asphalt compaction can introduce a high amount of variability to the asphalt layer, which in turn may translate into high costs of repair and maintenance (Bijleveld F. e., 2015).

According to Makarov (2017), the variability in the paving process can be considerably reduced by providing the operators of paving equipment with relevant real-time process quality indicators. By measuring asphalt temperature and equipment location and presenting these data to the operators, operators can base their strategies on the actual data rather than their experience and intuition. At the other end of the spectrum, the real-time data collection and analysis, even if not presented to the operators, can be very useful because it allows to explicate the tacit knowledge of the operators by correlating their experience-driven strategies with the context of paving operation (e.g., weather data, asphalt mix, logistics, etc.).

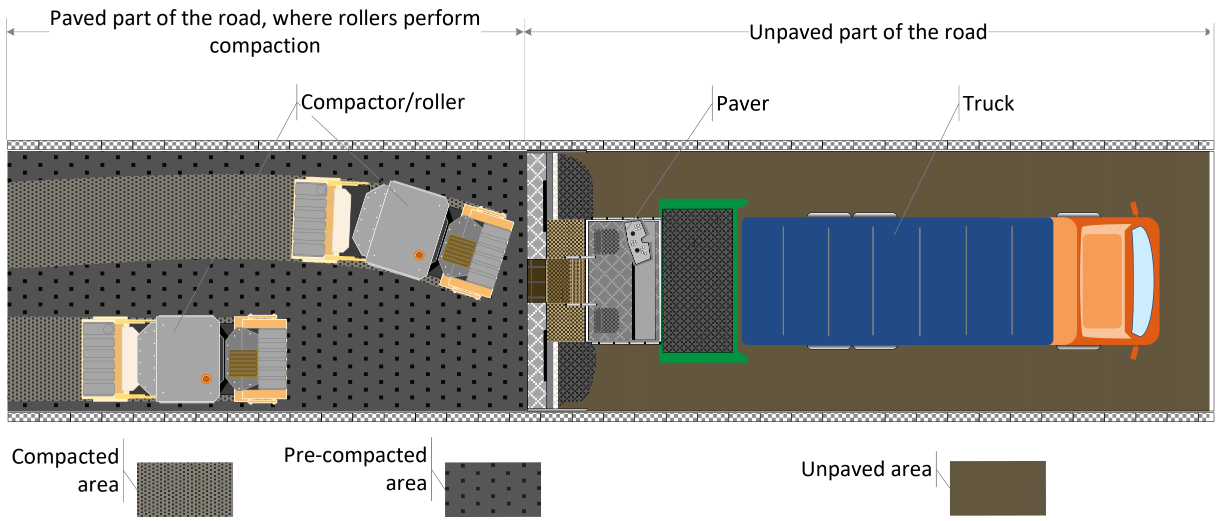


Figure 1: Current asphalt operation

On this premise, the ASPARi research unit at the University of Twente developed the PQi (Miller 2010) methodology that has been successfully used to map operational behaviour during construction using GPS and other sensors. The central goal of the PQi method is to improve process quality by reducing process variability. This is done by, first of all, monitoring the hot mix asphalting process so that the operational behaviour can be made explicit. Secondly, an action research approach is used to work on more process improvement in hot mix asphalt teams. The goal for the HMA teams is to reflect on their work after the project. Here, graphs, visualizations and animations are used to make operational behaviour explicit. In this way, teams are enabled to discuss and analyze what they are doing, how they have done it, and whether there are opportunities to improve their working methods.

Measurement of key process parameters such as temperature homogeneity and compaction consistency is conducted using a range of off-the-shelf sensors. These include differential GPS receivers, laser line scanners, infrared camera, thermocouples and automatic data loggers.

1.3 An overview of PQi System

The current PQi system mainly as shown on Figure 3, has 3 modules for paver, roller and reference temperature. These modules read the data from the sensors and send them to the processing centre which, merges the data and creates the visualizations. Then it will send the visualizations as a video back to the roller and paver module.

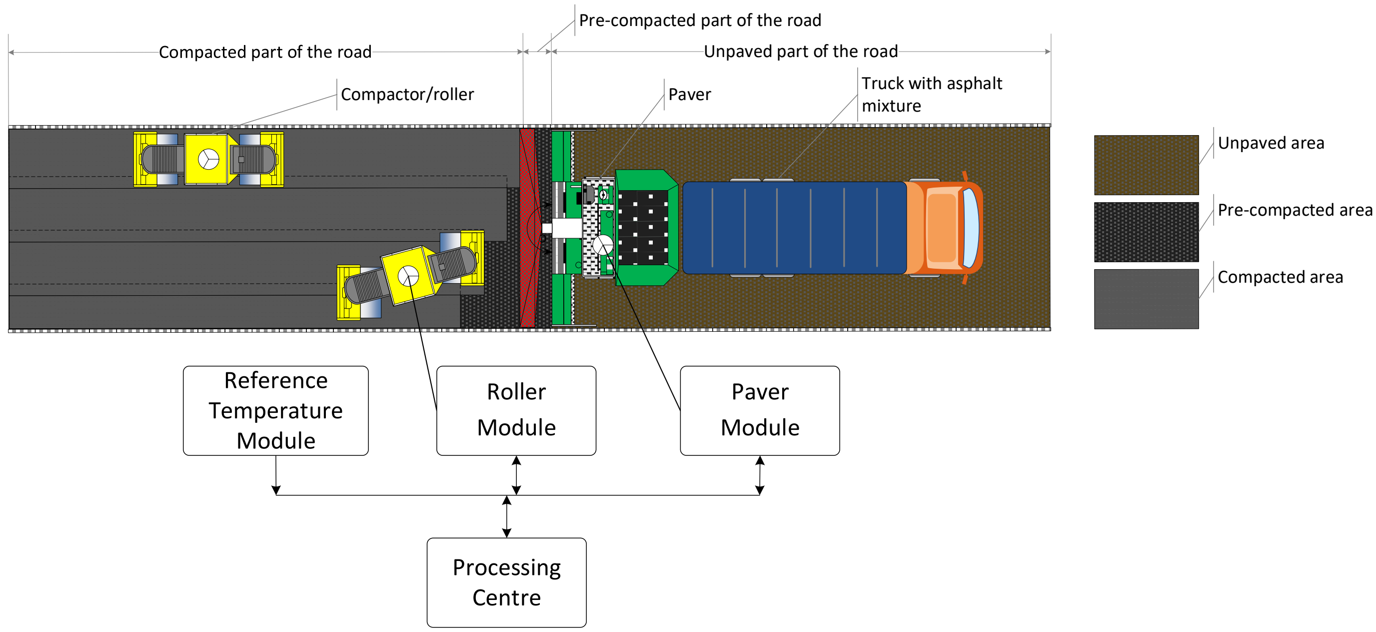


Figure 2: Previous PQi system

The current PQi system also has an offline mode that needs an operator to insert the data into the system and manually fix the errors. Offline mode has been used for most of the PQi measurement so far.

1.4 Problem statement

The current ASPARi PQi measurement equipment and process has 3 major challenges:

1. While functional, the current system is unstable, devoid of a robust structure and designed mainly in an ad-hoc manner.
2. Collected data are stored in proprietary device/sensors formats which in many cases are not common between different manufacturers. This requires considerable post-processing efforts to integrate various types of heterogeneous data type. In other words, the sheer absence of a data structure, or an ontology, for paving process monitoring causes major interoperability issues between front-end applications of the system.
3. The current set of PQi measurements, in terms of sensor acquisition and installation, is expensive and cumbersome. The complex process appears to be intrusive and disruptive to the smooth progression of paving operations and, at times, can create frustration among the paving crew. The high cost of PQi instrumentation renders the applicability of the method for a wider spectrum of end-users, particularly smaller contractors, difficult. Moreover, the functionality of some of the sensors, e.g., GPS, needs to meet strict environmental and contextual requirements, e.g., clear sky view. This poses a major challenge to the widespread application of PQi measurements.

1.5 Design Requirements

Based on the above limitations of the current PQi measurements, ASPARi contractors developed a set of new requirements for the next generation of the PQI measurements:

1- The system must be more robust and reliable, ensuring that it can be setup in different conditions and real-time support is provided consistently;

2- The PQi data must be easily accessible, reusable, and interoperable. The underlying system for PQi measurements must be extensible and reconfigurable; meaning that the system should easily accommodate new technologies as replacement or addition to the current system. This can be translated into the need for a robust, extensible, and well-structured data management system; and
3- The implementation of the PQi process needs to become more applicable, user-friendly, and non-intrusive.

1.6 Design Objectives

To overcome the above-mentioned limitations, this PDEng project aims to develop strategies/methods for the modernization of the PQi process control system in pavement operations, focusing on applicability, practicality, usability, and extensibility. This objective will be achieved by:

1. Developing and fine-tuning the real-time support structure of PQi measurement.
2. Developing a consistent data structure that can accommodate and link various types of data obtained from different sensors in a systematic manner. The data structure must be sensor-independent and customizable to different types of devices.
3. Identifying alternatives for PQi measurements in terms of the implementation of new sensors/devices with major emphasis on ease-of-use and non-intrusiveness of the system set-up.

1.7 Disclaimer

It should be noted that this PDEng project was mainly responsible for the robust implementation of the new design ideas generated and produced by another Ph.D. candidate from the ASPARi network. Therefore, there has been a close collaboration between the PDEng candidate and the aforementioned Ph.D. student. In the light of this disclaimer, while this candidate takes credit for the design, implementation, and testing of the real-time system, no claim is made on the development of the novel theoretical concepts in this system (i.e., methods for the generation of priority mapping and operator guidance). Having said that, for the completeness and readability of the report, these methods are explained in detail (using proper referencing to published work of the Ph.D. candidate).

2 Design methodology

2.1 Design methodology

In order to design the appropriate system for ASPARi, the V-model from SDLC[2] was used as illustrated in Figure 4 (Langroodi, 2020).

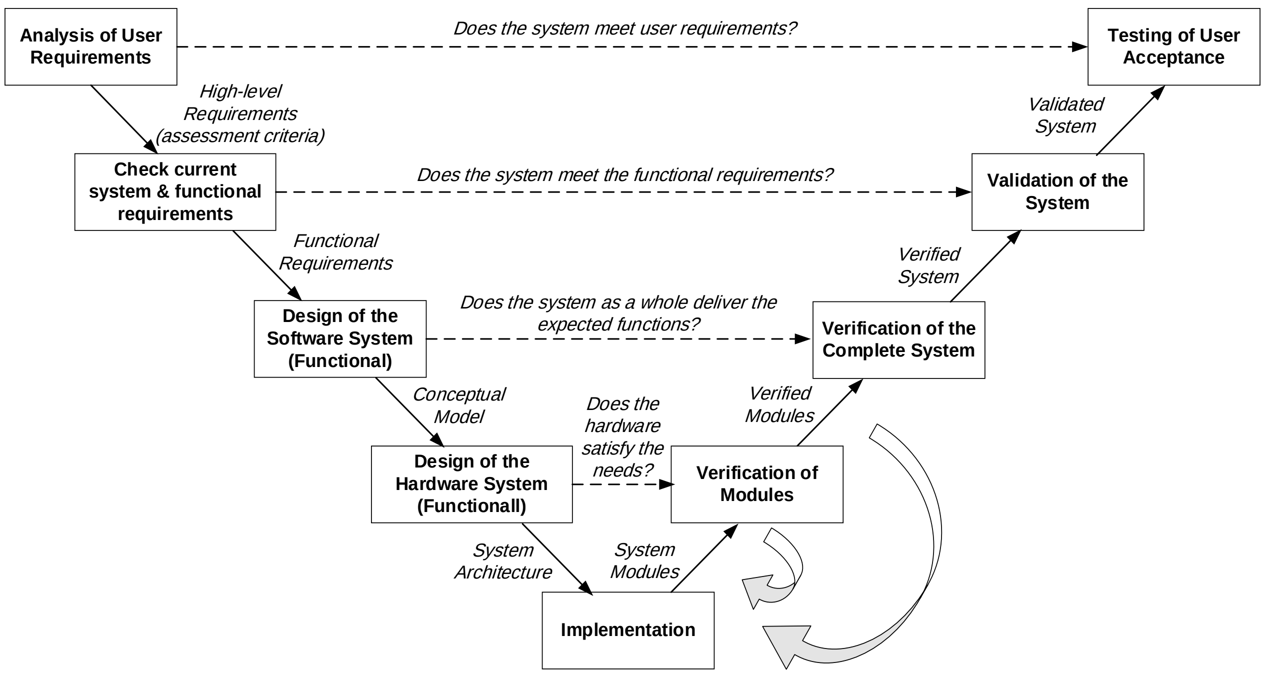


Figure 3: Design methodology of the system (Adapted from Langroodi 2020)

As the first step, the requirements of a "modern PQi system" needed to be identified. In order to extract the requirements, a literature review from the previous ASPARi projects was conducted. Regular meetings with ASPARi company supervisors were held to check if the project is on the right track.

In phase two, the status of the current system and its cons and pros were studied. To be able to fully assess the system, the PQi system was observed in the laboratory with offline data as well as on the construction sites with real-time data.

In the next phase, all the parameters that affect a PQi measurement were investigated, and an ontology was developed to capture the relationship between these parameters. The ontology tries to capture the characteristics of different parameters and how they are related to one another using semantics modelling approaches. The ontology development was done initially by the researcher (based on the input from the previous meetings with the experts) and then validated through several meetings with the company supervisors, i.e., domain experts. Based on the ontology, a database was developed which will be described in section 5.3. Also, based on the stakeholder needs, the functional requirements of the system were identified. These requirements are later used to design and develop the architecture of the real-time system.

[2] System Development Life Cycle

In phase 4, the existing hardware alternatives for collecting the needed data defined in the ontology were investigated. The existing hardware were also re-evaluated to see if they are working as intended and if they need recalibration.

As a next step, the architecture of the system is designed and developed. The implementation started from client components of the system and was tightly intertwined with phase 6 to be able to identify potential problems at an early stage. To do so, each implemented module was verified on a construction site to make sure they are functioning as intended. The implemented modules were used alongside the existing PQi system to cross check their functionality and accuracy. The implementation and validation cycle relies on rapidly implementing and testing new solutions. To be able to test the system in the early stages, the previously captured data were also used and replayed as if the data is being collected in real-time. After reaching certain stability levels, the modules were tested on actual asphalt construction sites.

In phase 7 and 8, after implementing the whole system and checking them individually, the system as a whole was tested, and its functionalities were cross-checked on construction sites to verify if the system satisfies the needs. The criteria for validating the system are described on section 3.2.

Finally, in phase 9, the working prototype was demonstrated to the stakeholders such as roller and paver operators, as well as on-site managers to receive their feedback on the system.

In the end, the trainee reflected on the whole process in this document. Two other documents are presented namely (1) A User Manual which helps the users in installing the prerequisite software as well as guiding them through the hardware installations. There is also a section for guiding the user for interacting with the real-time software; (2) A Road Map which is the trainee’s thoughts about the current state of adoption among competitors and the possible future opportunities . There is also a short-term and long-term plan for ASPARi to remain market relevant from the trainee’s perspective.

3 Functional (User) Requirements of the system

3.1 User Requirements

The first step in the research methodology, as mentioned in the previous chapter, is to identify the high-level user requirements of the system. Through several meetings with the steering committee of this assignment, the following user requirements were identified:

3.1.1 The system needs to be real-time

This means the system should provide the feedback to the operators in a fraction of a second after the data collection. Also, the application shouldn't freeze for a long time to be able to provide updated information to the operators.

3.1.2 The system needs to extensible and future-proof

Given the pace at which the technology is developing and becoming more affordable, it is very likely that new sensors will become available in the near future that can improve the PQi system from the perspectives of accuracy, robustness, ease of use and versatility. Also, new sensors can potentially be deployed that can add new functionalities to the system. Therefore, it is of cardinal importance that the current system offers a level of extensibility that keeps it applicable for a foreseeable future. In other words, the structure of PQi data should be established in such a way that new sensors and methods can be integrated into the system with minimal effort.

3.1.3 The system should be robust

Due to the asphalt heat, vibration of the equipment while paving/compacting, and the smoke from the hot asphalt, the system needs to be robust and resilient. Also, in many instances, the data flow or communication between the components of the system can be disrupted. It is crucial that the system is able to cope with these situations. It is therefore important to deploy heavy-duty industrial equipment to make sure the system can function for long hours on construction sites. If a sensor or any other part of the system stops working for any reason, the other parts of the system should continue working and only the output of the faulty sensor should be affected. For instance, if the IR camera stops working, all the maps will be generated with a static temperature map or if the connection between any devices and server drops, all data will be stored on the clients and can be uploaded to the server at a later stage.

3.1.4 The system should be accurate

Asphalt cracks can be as small as a few centimetres or as big as a few meters. Also, the joint between a newly paved road and the old one can be a few centimetres in length and width. To cover all the scenarios, the information provided by the system should be accurate (within centimetre accuracy).

3.1.5 The system should work under different weather conditions

The weather in the Netherlands is erratic and subject to rapid changes. Although asphalt contractors usually try to pave in stable weather conditions, it is not always feasible. Therefore, the system should be able to operate under different weather conditions.

3.1.6 The system should work for different types of project locations (e.g., highways, rural roads, etc.)

Paving projects take place all over the country in rural districts, urban areas, city centres, densely populated areas filled with trees, tall buildings, tunnels and other potential obstructions that could cause the reflection of GPS signals. Thus, the system should be functional for all the different scenarios.

3.1.7 The system should be interoperable with companies' internal asphalt management systems

Each asphalt contractor works with their internal software and other third-party software available in the market. The system should be able to import their data and export the data to other software.

3.1.8 The system should generate real-time visualizations to (at the least) show asphalt temperature homogeneity and compaction consistency.

ASPARi construction companies fully understand the importance of controlling process parameters such as temperature homogeneity and compaction consistency. Therefore, it is important that visualizations such as the PQi's Temperature Contour Plot (TCP) and Compaction Contour Plot (CCP) be generated and shown to machine operators in as close to real-time as possible. This should provide opportunities for developing better, more method-based operational strategies.

3.1.9 The system should provide after project results within a few minutes after the project is completed

After an asphalt paving project is completed, post-project meetings reflecting on performance should be able to be conducted almost immediately. Hence, the system should be able to provide the information right after the project is completed.

3.1.10 The system should be able to read from different sensors

Each company has their own set of sensors. The sensor technology will also evolve rapidly in the future and the system should be able to operate with all different sensors.

3.1.11 The system should be able to work for a full day's paving

Normally, a full day's paving can last up to 8 hours. Thus, the system should be able to operate at least for 8 hours.

3.1.12 The system should be easily maintained

Using a System Engineering approach, the designer should keep the maintenance of the system in mind. Software and hardware updates should be easy to undertake.

3.2 Functional Requirements

After observing several PQi sessions and investigating the existing system very carefully, the above user requirements can be converted to the following functional requirements:

- The system should be sensor independent; meaning that replacing an existing sensor shouldn't result in changing the whole system.
- The system needs to collect data from sensors, merge data from different sensors, analyze them and show different types of visualizations for different on-site operators.
- The system needs to generate initial TCP and CCP visuals immediately after the project is completed.

- The system needs to replay the equipment movements to be able to have post-project feedback sessions with operators.
- The system needs to be installed and setup on the equipment in less than an hour.
- The data need to be stored systematically and the users should be able to easily retrieve the data afterwards.
- The system should be easy to scale for both using on multiple construction sites and having multiple clients on a single construction project.
- The system needs to show useful information when an error occurs. It should also recover itself whenever possible. Also, the manual recovery by the operators should be easy.
- Defining a new project and setting relevant attributes for a project should be easy and possible to prepare beforehand. Also, there should be an option to change them on a construction site.
- The system should work in real-time which means it should be able to read the data from sensors, process them and visualize them at least every 5 seconds.
- The system should be accurate, and the accuracy should be measurable and reliable.

4 Technical design of the system

Figure 5 presents the overview of the proposed system. At the high level, the system encompasses two main parts, namely server and clients. Clients oversee collecting pertinent data and sending the data to the server. Various types of sensors provide data about weather conditions, equipment movements, asphalt condition (e.g., temperature and compaction), and the human users (i.e., operators and managers). These data are then pushed to the server, which is responsible for (1) structuring the transmitted data in a database, and (2) processing the data and translating them into relevant information that can be pushed back to the clients. Finally, the processed data are transmitted back to the clients. This includes the operators of different pieces of equipment.

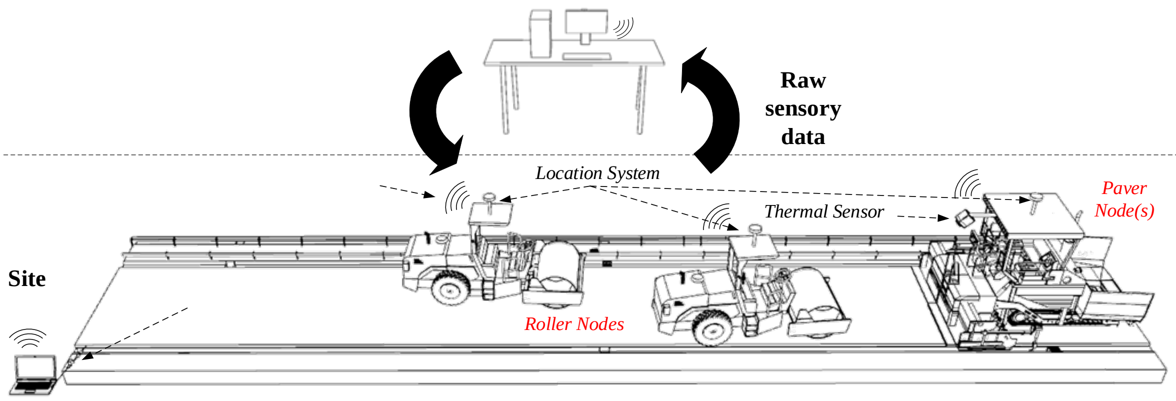


Figure 4: Overview of real-time PQi measurement system

4.1 Clients

As shown in Figure 6, there are at least 3 different types of clients in the proposed system: asphalt node, roller node, and paver node. Each client application has three major tasks, namely collecting data from the sensors, sending them to the server and storing the data locally, and visualizing the received data from the server. Each independent task such as visualization, data storage, sending and receiving data to the server runs independently, meaning failure in any module does not fail the whole application, although it might logically affect the output. For instance, if GPS on a roller fails, the application will still run and show the TPCs but, it will miss the compaction data for that specific roller. The missing data can later be imported into the application and recalibrate the results. Moreover, the clients will detect and report the failures as soon as they happen and will report them to the user. Failures include any errors happening in the application such as problems while reading from sensors or data storage errors. Whenever an error occurs, the affected modules in the setting or information will be shown in red to help the user in running a diagnostic. A more technical message will also be shown in the log section.

Another useful feature of the client components of the system is the ability to run multiple instances of each client over different locations in a network. Each time a client runs, it will register itself as output data listener in the server. As it’s hard to access equipment on site while they are busy paving the road, this feature can be used to see the clients output without physically having access to them. In addition, presenting the outputs and analyzing the operator behaviour would be much easier for onsite managers.

Each client encompasses three main hardware components: (1) sensors, (2) embedded PC, and (3) data transmitter. The functionalities of these components are as follows:

1. **Sensors:** responsible for measuring temperature, location, visual observation, density, etc.
2. **Embedded PC:** responsible for reading data from sensors in real-time, sending and receiving the data to/from the server through a data transmitter, and visualizing the data.
3. **Data transmitter:** data transmitter is responsible for establishing the connection between each client and server. In addition to transmitting the data to the server, each client maintains a local copy of the sensory data. Also, sensors store the collected data on their internal storage. In case of any problems to the server or connection, all the data will be captured for offline processes.

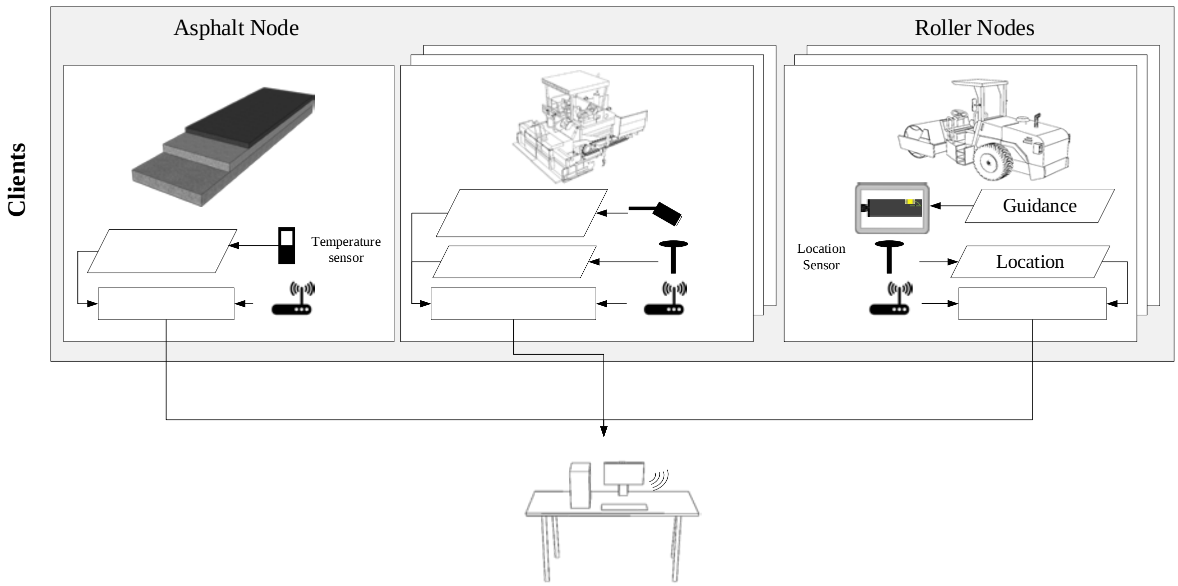


Figure 5: Different types of clients

Each client interface consists of four general parts:

1. ***Data collector***: It consists of a set of configurations that is needed to be set per project such as server IP or GPS rover number. It also shows/visualizes the raw input data from the sensors and shows the application logs.
2. ***Visualizer***: Visualizing the outputs received from the server.
3. ***Server status***: It shows the current status of server connectivity, although the clients will keep trying to reconnect to the servers in case of connection loss.
4. ***Configuration file***: It consists of the configurations that might be needed to change when a change is applied to the hardware or when the user wants the application to behave differently (e.g., the configurations of how to access sensors and the operation frequency of different modules). The list of all configurations is available in the developed user manual.

Reaching the clients on a construction site can be very difficult, if not impossible, without interrupting the workflow. In some extreme cases, the operator might reject the requests for accessing and reconfiguring the client. To solve the issue, a user only needs to set the hardware configurations on the client, such as server IP and GPS rover number, and all the project configurations and other tweaks can be done via the server. To achieve this goal, the clients are stateless and fully data-driven. They are stateless by not keeping any history of data which helps to quickly adapt to the new configuration even after losing the connection to the server. They are also data-driven which means their behaviour can be changed based on the received data.

Finally, to provide more stimulus for the operators to interact with or notice the application (i.e., in-paver and in-roller applications that operators interact with), there is a media player that can play music for the operators. This is important because, based on several site observations, the first thing operators do when entering the equipment is turning on the radio.

4.1.1 Asphalt Node

In the asphalt node, as shown in Figure 7 the temperature of asphalt layers in different locations is measured and sent to the server via the data transmitter module. Therefore, this node requires a temperature sensor and a data transmitter.

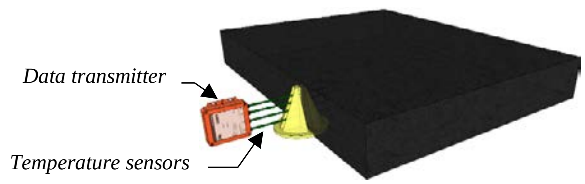


Figure 6: Configuration of Asphalt Node (Makarov, Vahdatikhaki, Miller, Jamshidi, & Dorée, 2021)

As shown in Figure 7, the temperature sensors can be inside the asphalt layer using a stand. This sensor can be placed inside the asphalt before the paving starts. A stand can be used for this purpose to ensure that temperature sensor remain in place all the time. When the paving starts, the temperature sensor gets buried underneath the asphalt and starts collecting the temperature data. This means that it is important for this temperature sensor to withstand the high temperature of hot asphalt. The data transmitter collects the temperature data in real-time and transfer it to the sensor.

4.1.2 Paver Node

The Paver node collects the surface temperature of the asphalt, as shown in Figure 8. This surface temperature is combined with the data collected by location sensors. Finally, paver node sends the asphalt surface temperature with its location to the server. The required components for this node are therefore a location sensor, a contactless temperature sensor, and data transmitter.

The Paver node needs initial temperature and location of each temperature cell to be able to fully function. Beside the general configurations of the clients, since the camera is usually not installed directly on top of the paved road faced down, it is important to address the perspective issue that arises due to imperfect positioning of the infrared camera. This is done with a foursquare whose vertexes can be adjusted by the user. In addition, to reduce the data transfer rate, the user can specify number of sampling points from the foursquare. The paver application will only send the specified number of sampling points to the server. To have the highest accuracy and efficiency, it would be best if the number of sampling points matches number of cells in a row and the foursquare represents a row of temperature data in terms of size. In case the number of data points does not match the required number of points for cells, the server application will interpolate/extrapolate the temperature readings. Moreover, as the camera might be installed at different angles (usually upside down) due to the complexity of the scenarios while installing equipment, the user can also rotate the final image for better representations. Finally, the collected raw data is visualized in the real-time information section.

It should be noted that all the IR images will be saved locally as well as the raw data collected by the paver application. The IR images can later be used for creating animations and checking the logistics for the roller and paver movements.

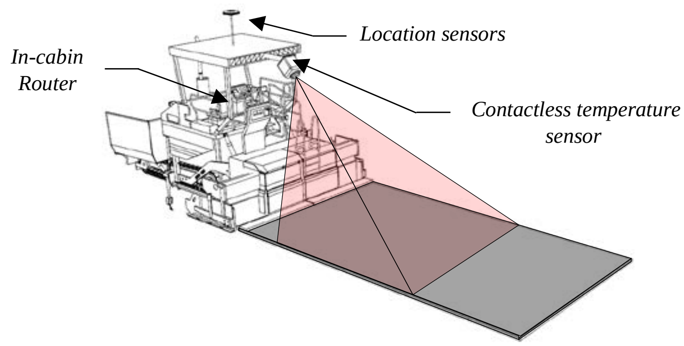


Figure 7: Configuration of Paver Node (Makarov, Vahdatikhaki, Miller, Jamshidi, & Dorée, 2021)

4.1.3 Roller Node

For the roller node, the location data are collected for each roller and sent to the server via the data transmitter. Later, all the data will be combined and processed on the server and different visualizations such as guidance maps, will be shown to roller operators. This node needs an onboard display, a location sensor, and a data transmitter.

4.2 Server

The Server is the core of the system. It is responsible for receiving data from clients, pre-processing them, merge and post processing data to generate the output, save each stage data into the database and send the data back to the clients. In the Same manner as the Clients, the server will also operate in parallel as much as possible, meaning that processing each client, updating the cooling curve, saving data into database and sending data to the clients, operate independently and on different threads. Figure 9 presents the architecture of the server in the proposed system. Once all the data are transmitted to the server, the data processing begins. The data processing consists of several steps, including structuring of the data into a relational database, generation of the cooling curve, generation of the compaction/temperature contour plots, generation of the priority map, and generation of the operator guidance.

At the first step, the raw data transmitted to the server is stored into database. Next, 6 general steps are taken sequentially to generate all the required outputs.

4.2.1 Relational Database

The data received by the server is heterogeneous and asynchronous. That is why these data need to be organized into a relational database. Additionally, the use of a relational database (a) streamlines the implementation of the centralized architecture (i.e., reduces redundant peer-to-peer communication), (b) supports easy access to the data for real-time and ex-post analysis, and (c) enhances scalability and extensibility of the entire system considering the possible integration of new sensing technologies in the future.

Figure 10 shows the overview of the relational database used in the proposed framework. In this database, data are categorized into 10 classes. At the top layer of the database, the pavement project is defined. The project includes such design data as the nature, location, and geometry of the project. Weather condition data (e.g., ambient temperature, humidity, wind direction/speed, and barometric pressure at the site location) is placed under the weather class. The client and contractor of the project are represented in terms of their name and ID. The asphalt layer is characterized by the details of the mix of design, and real-time and analytic data. In this class, in addition to the type and geometry of the asphalt layer, operational requirements of the compaction work (e.g., required compaction effort, effective compaction rate and surface cooling rate) are specified. Effective compaction rate is an indicator for the quality of the compaction process that represents the proportion of the asphalt layer that has been compacted enough within the appropriate temperature window (Makarov 2021). Cooling rate of the asphalt accounts for the rate at which the asphalt layer (i.e., both the core and surface) cools down. To facilitate data registration, analysis and visualization, the asphalt layer needs to be rasterized into a grid. This grid consists of many cells each of which hosts a suite of analytic data including the temperature, compaction pass, priority index, etc.

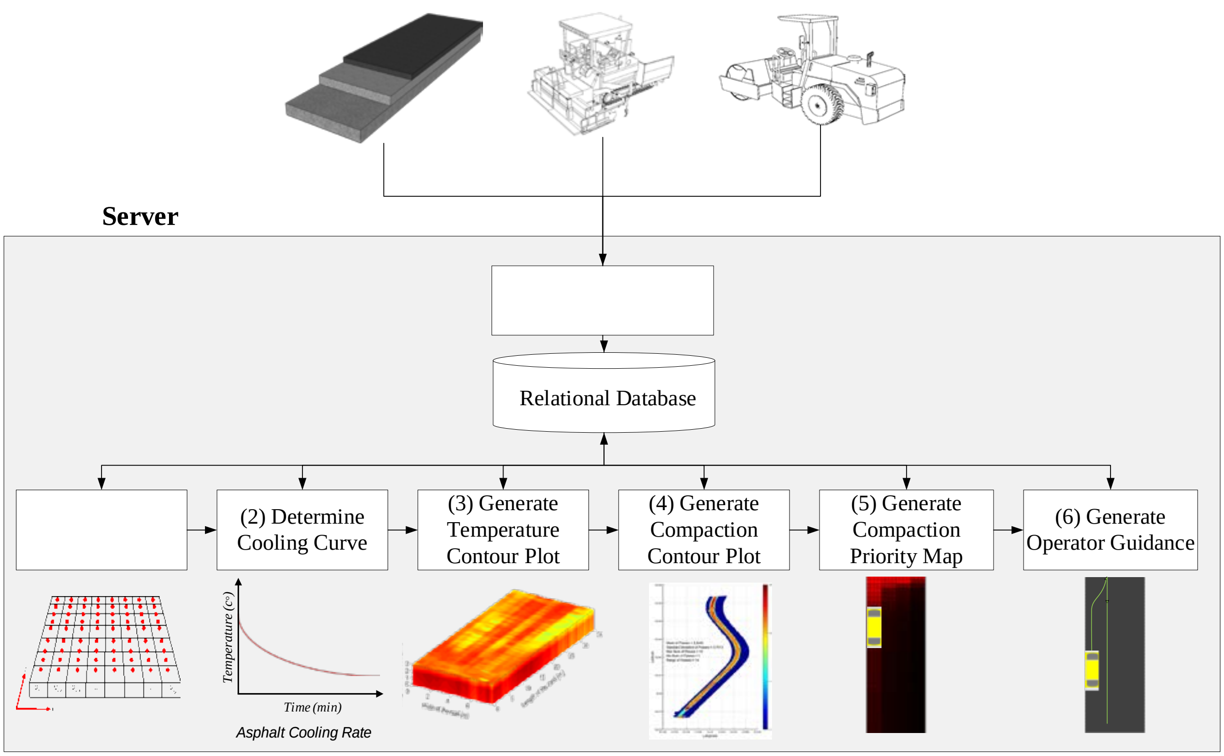


Figure 8: Server architecture

The access to the database and particular tables is restricted by the predefined permissions and it depends on the authorization level of the client. Within the proposed database model, each user (e.g., administrator, developer, contractor, site manager, representative of a road authority) has his/her own permission to work with the data in the database (e.g., real-time access to raw or pre-processed data).

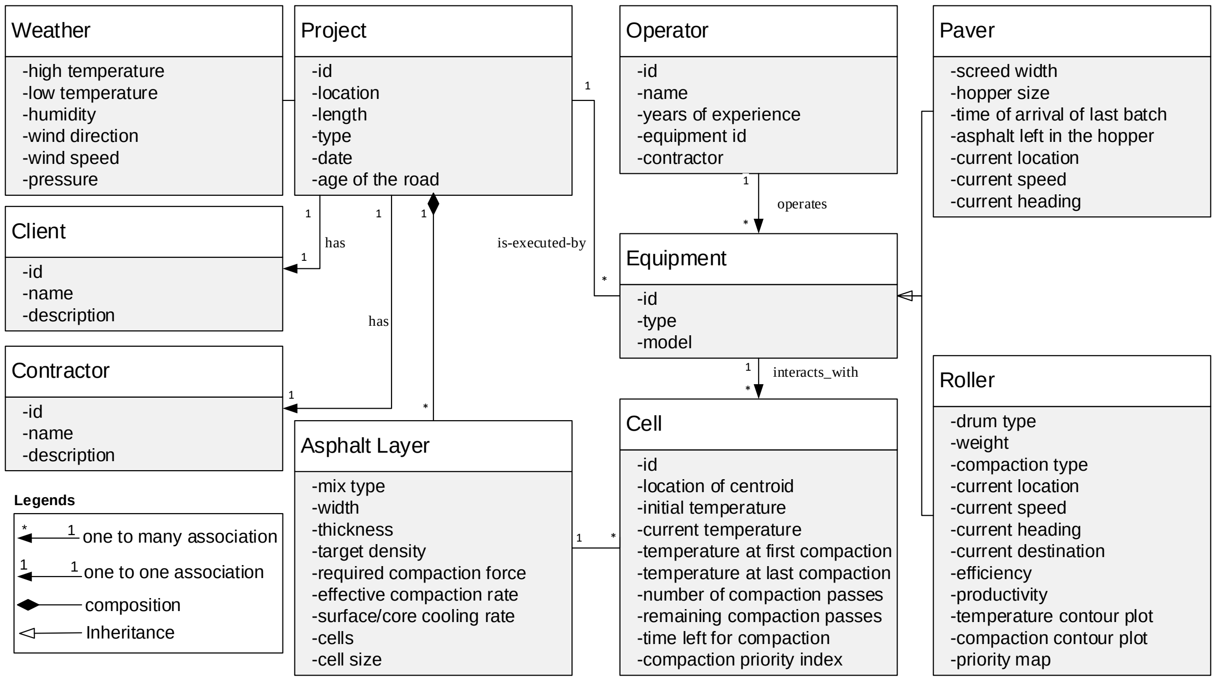


Figure 9: Database architecture (Makarov et al. 2021a)

4.2.2 Rasterization of Space

As mentioned before, for the consistent registration of data the asphalt layer needs to be rasterized into a grid. It is important to note that the resolution of the data analysis is determined to a great extent by the size of the cells in this grid. The cell size depends on, most saliently, the update rate and accuracy of location and sensors, available processing power, and the size of rollers. It can logically be argued that the cell width cannot be smaller than the resolution of the temperature sensor installed at the back of the paver. For instance, if the infrared camera is used to capture 20 points across the specified width, the cell size must be at least greater than 1/20 of the width. Also, it is important that the cell size is not bigger than half of the width of the rollers' drum to make sure the information provided to the operator is accurate and actionable. These constraints are represented in Equation 1. Based on these considerations, the system's operator can specify the cell size at the beginning of the operation.

$$\frac{W}{R} \leq w \leq \frac{W_r}{2} \qquad \text{Eq. 1}$$

Where:

W: Width of the road
R: Resolution of the infrared camera
w: Width of the cell
W_r: Width of the roller's drum

The grid begins as soon as the paving starts. As shown in Figure 11, each cell of the grid ($C_{i,j}$) is represented by such data as the initial temperature ($IT_{i,j}$), current temperature ($CT_{i,j}$), the temperature at the first pass ($TFP_{i,j}$), the temperature at the last compaction pass ($TLP_{i,j}$), number of compaction passes ($P_{i,j}$), remaining compaction passes ($RP_{i,j}$), time left for compaction ($TL_{i,j}$), and compaction priority of cell ($R_{i,j}$).

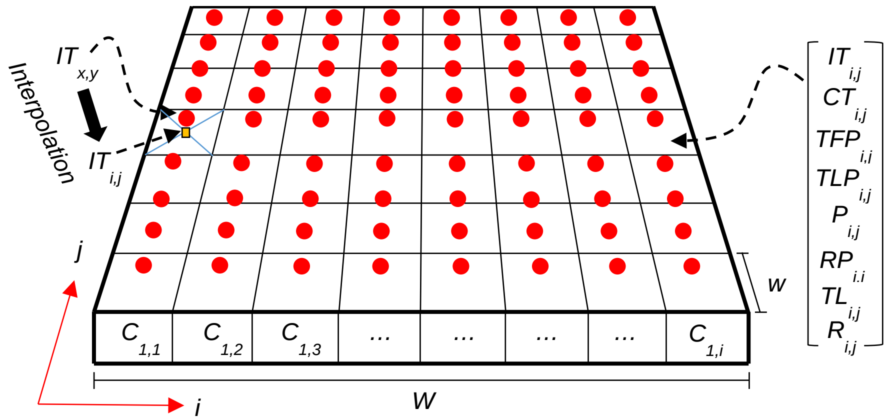


Figure 10: Cell formation on the asphalt layer (Makarov et al. 2021)

The first step in the rasterization of the data is to geo-referenced the surface temperature data, which can be achieved by determining the translation vector between the location and temperature sensors on the paver and then calculating the transformation matrix for the temperature data (considering the translation vector, installation configuration of the camera (height and tilt angle) and camera's intrinsic matrix).

Geo-referenced temperature data then need to be mapped into the grid. This is done by intersecting the geo-referenced temperature data ($IT_{x,y}$) with the grid. To calculate the temperature at the centroid of each cell, interpolation can be applied

4.2.3 Generation of the Cooling Curve

The decisions about when the compaction should start and how much time is left for compaction are made, mostly based on the cooling rate of the asphalt. In other words, the thermal behaviour of the asphalt layer during construction activities (e.g., paving, compaction) can be represented by the asphalt cooling rate. The methodology presented by (A. Vasenev, 2012) indicates how this cooling curve can be derived from the temperature measurements. Essentially two types of data are required:

1. **Surface Temperatures (T_{surf}):** asphalt surface temperatures that are gathered by the Asphalt node;
2. **Core Temperatures (T_{core}):** asphalt core temperatures that are gathered by the Asphalt node.

Figure 13 represents an example of the data collected by the Asphalt node and the cooling curve.

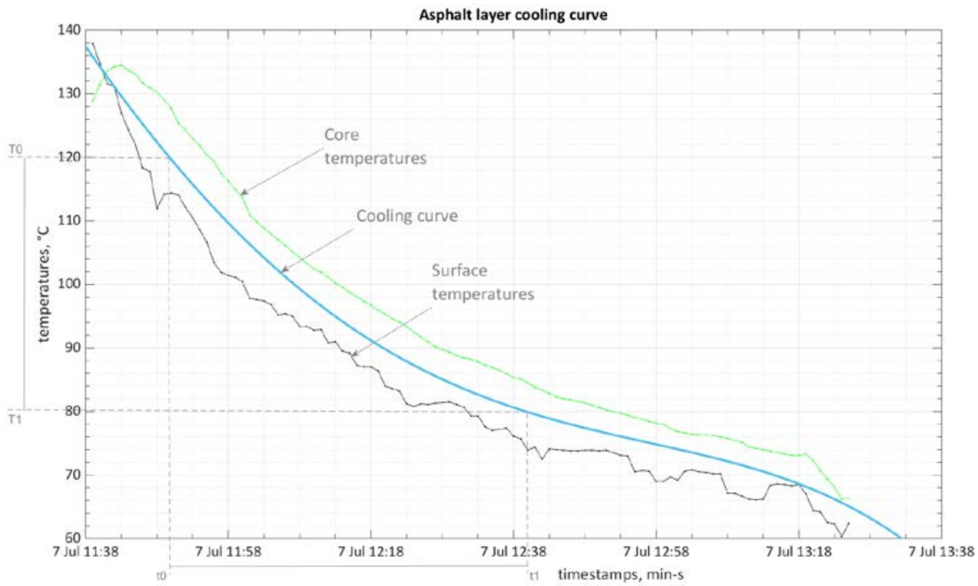


Figure 11: Generation of the cooling curve (Makarov et al. 2021a)

4.2.4 Generation of the Temperature Contour Plot

Temperature Contour Plots (TCPs) are the 2D top-view visuals that represent the time-dependent condition of the asphalt layer in terms of momentary surface/core temperature, as shown in Figure 13. This plot visualizes the surface temperature of the newly paved asphalt road and provides insight into the homogeneity of the asphalt mixture. The server uses the data collected by the Paver node (i.e., surface temperatures of the asphalt mat behind the screed of the paver and paver coordinates during the paving operation) to generate TCPs. After the paver operator manually determines the width and length of the field in the system, the collected surface temperatures of the paved asphalt layer are synchronized with paver coordinates. Geo-referenced surface temperature data are then intersected with the cell grid ($IT_{x,y}$ → $IT_{i,j}$) and saved as initial temperatures.

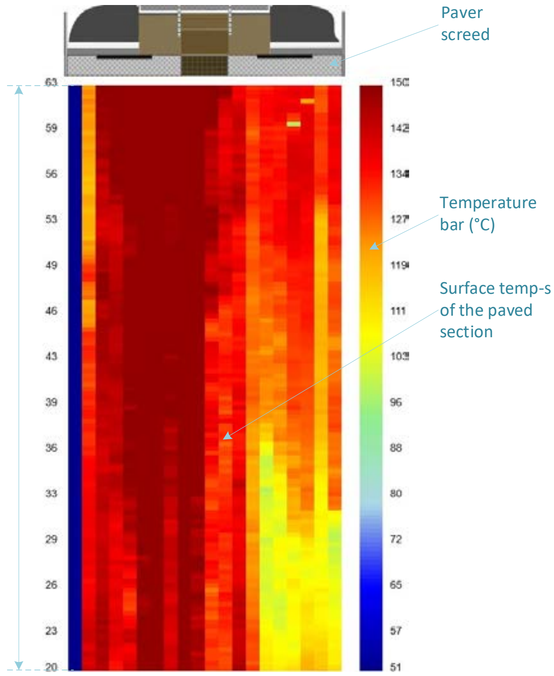


Figure 12: Temperature contour plot (Makarov et al. 2021a)

4.2.5 Generation of the Compaction Contour Plot

Compaction Contour Plots (CCPs) are generated for each roller considering the movement trajectory of the roller. The location data of rollers are sent to the server where they are first filtered (to remove outliers) and then analyzed to determine the number of compactions passes achieved. The cells created during the rasterization are used for the calculation of the compaction counts.

It is important to note that for the purpose of assessing process quality, it is important to record the temperature of each cell at the first ($TFP_{i,j}$) and last ($TLP_{i,j}$) roller passes. For further analysis of the compaction process, Equation 2 is used to calculate the number of remaining compaction passes ($RP_{i,j}$) for each cell.

$$RP_{i,j} = \left(TRP - P_{i,j}\right) \qquad \text{Eq. 2}$$

Where:

$RP_{i,j}$*: Number of remain compaction passes for a certain cell of a grid*
TRP: Target number of compaction (roller passes), defined by project (site) manager
$P_{i,j}$*: Number of compaction passes calculated for a certain cell of a grid*

Based on $P_{i,j}$ and $RP_{i,j}$, the compaction map is created (as shown in Figure 14). This map represents accomplished compaction with the predefined colour scheme.

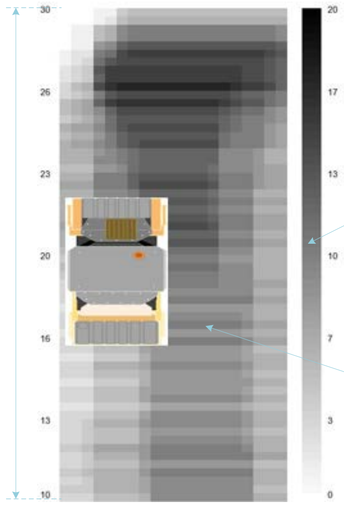


Figure 13: Compaction contour plots (Makarov et al. 2021a)

In ideal circumstances, the roller operator is expected to begin and end the compaction of each cell, after and before the upper and lower limits of a compaction temperature window, respectively (i.e., temperatures T_0°C, T_1°C at times t_0 and t_1, in Figure 12). The cooling curve and the current temperature of each cell (CTi,j) are used to calculate the time left for the compaction of each cell (TLi,j). The roller operator can develop or modify the compaction strategy (i.e. speed up or slow down the process, or focus on particular parts of the mat) based on the combination of $TL_{i,j}$, $CT_{i,j}$, and $RP_{i,j}$.

4.2.6 Generation of the Compaction Priority Map

As shown before, TCP's and CCP's merely indicate the state of the asphalt layer in terms of asphalt temperature and compaction. Interpretation of these data in real-time can be cognitively burdensome for the operators. Therefore, another type of data representation can be envisioned that combines the TCP and CCP into a priority index ($R_{i,j}$) that represents the compaction priority of different cells. The visualization that represents these priority indexes is referred to as a Compaction Priority Map (CPM). In order to generate the priority map, first the time left for the compaction of each cell ($TL_{i,j}$) is calculated based on the cooling rate and the current temperature, as shown in Figure 15. As shown in Equations 3~5, $TL_{i,j}$ is combined with the number of compaction passes achieved for each cell ($P_{i,j}$) to determine the priority index of each cell.

$$R_{i,j} = CP_{i,j} \times TP_{i,j} \qquad \text{Eq. 3}$$

$$CP_{i,j} = \begin{cases} \dfrac{PD - P_{i,j}}{PD} & PD \geq P_{i,j} \\ 0 & PD < P_{i,j} \end{cases} \qquad \text{Eq. 4}$$

$$TP_{i,j} = \begin{cases} 0 & TL_{i,j} > t_c \\ \frac{t_c - TL_{i,j}}{t_c} & 0 < TL_{i,j} \leq t_c \\ 0 & TL_{i,j} = 0 \end{cases} \qquad \text{Eq. 5}$$

Where:

$R_{i,j}$: Priority of cell i and j
$CP_{i,j}$: Compaction priority of cell i and j
$TP_{i,j}$: Temperature priority of cell i and j

PD: Desired number of compaction (number of roller passes)
$P_{i,j}$: Compaction achieved at cell i and j (number of roller passes)
TC: Maximum possible compaction time
$TL_{i,j}$: Time left for compaction of cell i and j

Figure 15 represents an example of the CPM. In the map, the cells highlighted in red has a high compaction priority while the black cells have a low priority because they have either already been properly compacted or are too cold to compact. Operators can use the CPM to detect the parts that need more compaction effort.

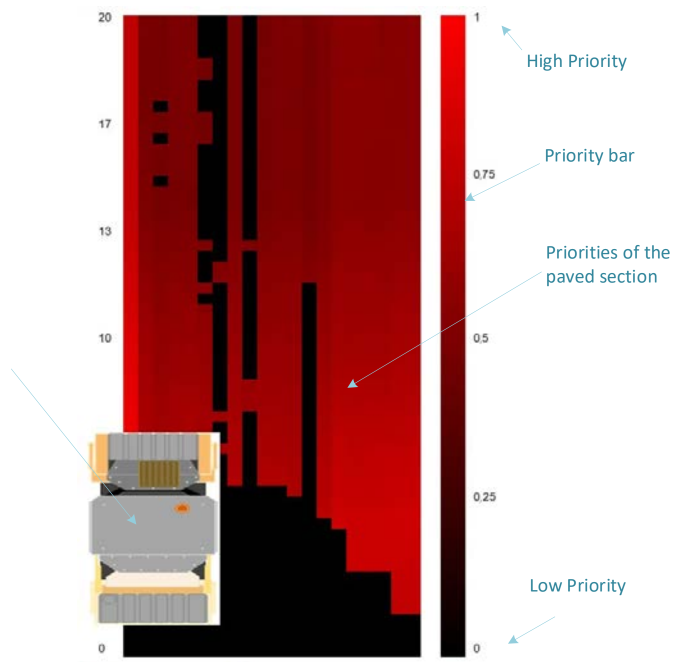


Figure 14: Compaction Priority Map (Makarov et al. 2021a)

4.2.7 Generation of Operator Guidance

Another level of support for the operator is guidance. At this level, the operator is guided through the provision of a compaction path, as shown in Figure 16. To generate operator guidance in terms of the stretch of road that needs to be compacted at any given time, the following pieces of data are considered: speed of compaction equipment (i.e., roller and paver), the cooling rate of the asphalt, the current temperature, the compaction temperature window, the width of the road, and the achieved and target compaction passes. The core logic of the operator guidance is that the next stretch of compaction needs to meet the following constraints: (1) the asphalt in the stretch remains within the compaction temperature window over the entire time required to compact the stretch, (2) after compaction of the stretch, at least a stretch with a similar length is available for compaction ahead of the roller (i.e., to maintain the continuity of compaction). Once the stretch is ascertained, a compaction path can be projected onto this stretch. The technical details of this method are out of scope of the present project and will be presented in the future work of the ASPARi research group.

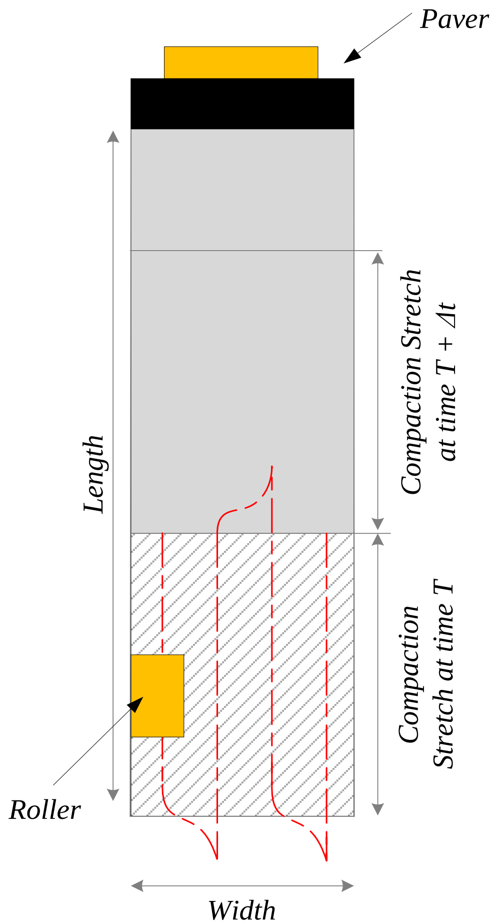


Figure 15: Schematic representation of the compaction guidance (Makarov et al. 2021b)

4.2.8 Additional features

In addition to the main functions presented above, a few additional features can be added to the system to make the use of the system more practical and user friendly. These features server visualizer and server status check.

Server visualizer**:** Given that a user may monitor the functioning of the entire system on the server side, it is important to present the system information on the server. This visualizer can show the location of different pieces of equipment and other debug information such as state of the sensors and the errors happening in them. Additionally, this interface can be used to define new projects, add new equipment to the system, apply different settings such as system update frequency, GPS offset positions and others into the project.

Server status checker: To make sure the PQi measurement is going well, there is a status server which shows the latest status of each sensor. This information can be accessed by phone or onsite PC.

4.3 Hardware alternatives

Based on the above architecture, there can be several technological alternatives for different components of both client and server sides.

4.3.1 Client Components

Table 1 depicts different alternatives for the client components, advantages and disadvantages of each technology, and a rough indication of the cost of each alternative. Different alternatives can be combined to create several solutions, or scenarios, for PQI measurements. Figure 17 presents 5 different scenarios that will be investigated in this study. Each one of these scenarios are explained further below.

Table 1: Sensor alternatives

Sensor/device application	Option	Advantages	Disadvantages	Rough estimated price (€)
Gathering equipment positioning on site (locations)	Drone with image processing solution	- Cheap to scale - Can cover a big area	- Capacity of drone's battery - Needs an operator - Site's preparation before drone's usage during construction activities	5.000
	Ublox	- Cheap	- Low accuracy	100
	BLE	- Cheap - Possible usage in tunnels and dense areas	- Low accuracy in distances more than 5 meters - Site's preparation before usage during construction activities	5
	Ultra-wide band	-	- Low accuracy - Complex synchronization between sensors	
	Lidar	- High accuracy	- Expensive	10.000
	DGPS	- High accuracy - Ready for application	- Expensive - Low accuracy in dense areas	5.000
	RTK	- High accuracy - Ready for application	- Expensive - Needs RTK server - Low accuracy in dense areas	7.000
Obtaining asphalt surface temperatures	Temperature line scanner (paver)	- High accuracy	- Expensive - Complex installation procedure	30.000
	IR thermal camera (paver)	- Cheap	- Can only measure initial temperatures due to restricted field of view (by the predefined mounting place on a paver)	7.000
	IR thermal camera (drone)	- Cheap - Can measure wider area of asphalt layer with changes of surface layer temperatures in real-time	- Capacity of drone battery - Needs an operator	2.000
Obtaining asphalt core temperatures	Thermocouple	- Cheap	- Laborious setup and re-setup during construction activities	50
	Fiber optic	- Can capture data after paving is finished	- Difficult to install - Expensive	10.000
Machine 2 machine communication	WIFI	- No extra payments for usage	- Small coverage area	200
	GSM	- Large coverage	- Monthly fees according to the data usage	500

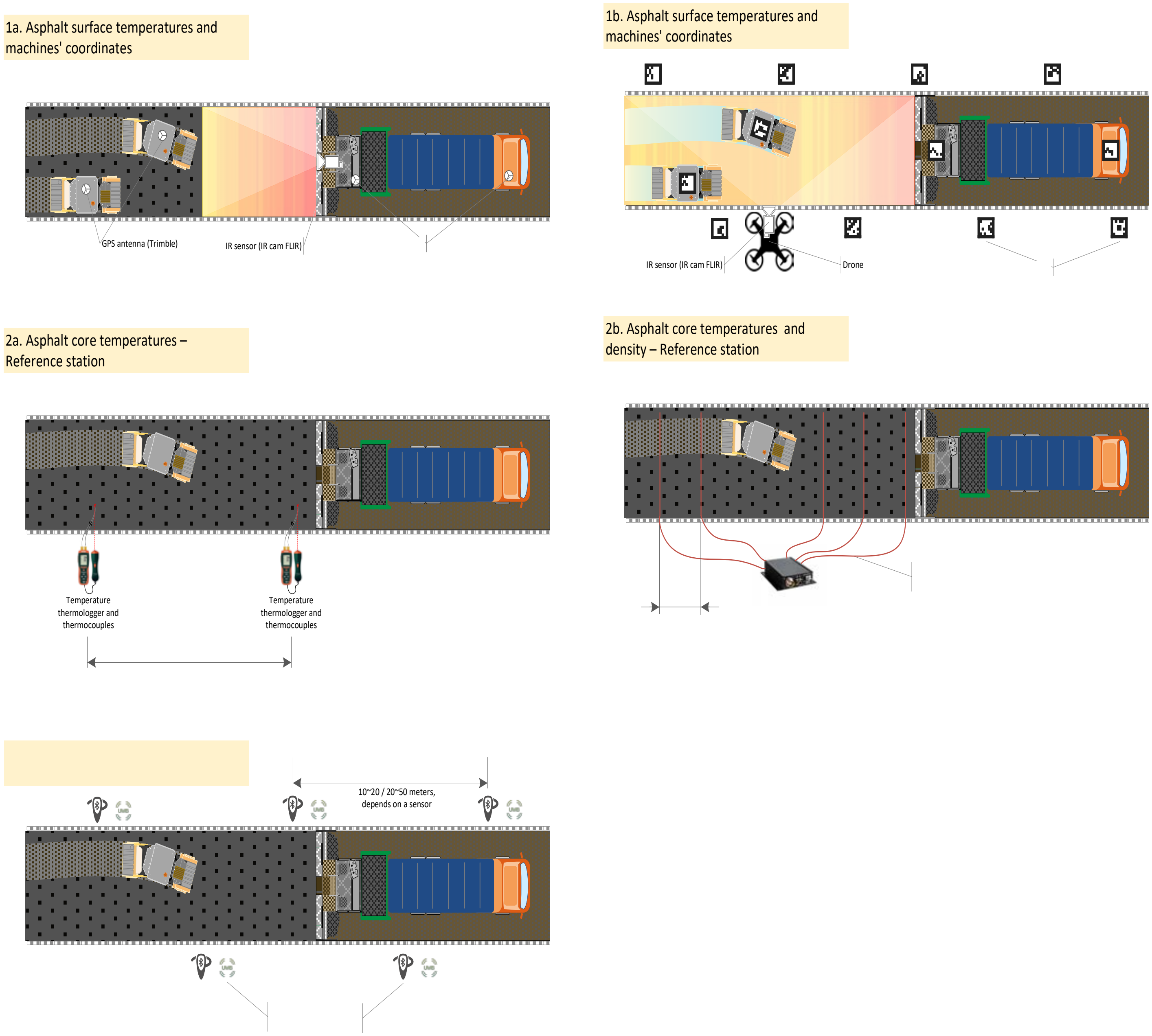


Figure 16: Different sensor alternatives

Scenario 1a:

In this scenario, equipment location can be collected with GPS sensors and the surface temperature is captured using an IR camera attached to the paver.

Scenario 1b:

In this scenario, a drone with a RGB camera and thermal camera is used. In this way, the location of the equipment can be found via image processing and the surface temperature can be measured via the thermal camera attached to the drone.

Scenario 2a:

For the core temperature, thermocouples are used to measure core temperature from different asphalt layers.

Scenario 2b:

In this scenario, fibre optics are used to measure the core temperature of different asphalt layers.

Scenario 3a:

Bluetooth low energy devices or Ultra-Wide Band devices are used to find the location of each equipment.

The selected Scenario:

Most of our tests were done by combining scenario 1a and 1b to satisfy the functional requirements. Although as a proof on concept, scenario 1b was also tested separately.

4.3.2 Server Alternatives

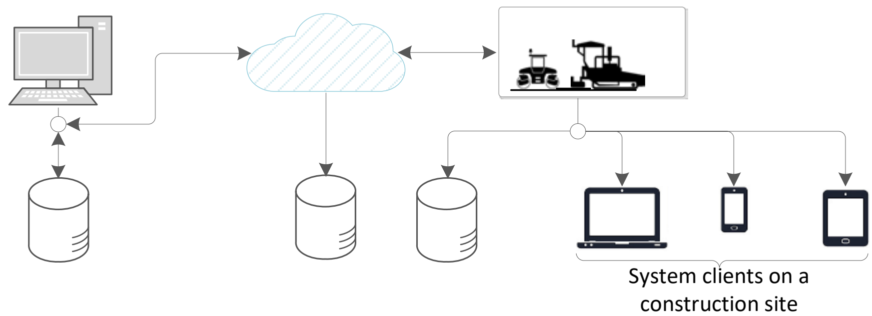
Depending on the medium and means of communication, the Clients and the server can be coupled in different configurations. Figure 18 schematically represents the three main configurations of the server.
In configuration 1, the server is placed at a remote location and the collected data is transmitted to the server through the internet. A local backup of the data will be maintained on the site to increase the robustness of the system and prevent the loss of data due to disruption of an internet connection. The local backup would guarantee that even if the real-time system is disrupted, data is available for post processing.
In configuration 2, the server is placed on the construction site. The connection between the sensors and server is established through a Wi-Fi network. In this configuration, data can be transmitted to a remote database for backup.
Finally, in configuration 3, the server is dissolved into a local processing unit in each equipment. In this sense, the architecture is based on decentralized computation, where each client has a processing unit that would receive/transit data from/to other Clients using machine-to-machine communication. The connection between the clients is established through a Wi-Fi network. Although the system is able to operate with this configuration, we only tested the system with the first configuration.
Table 2 represents the advantages and disentangles of each configuration.

Table 2. Comparison of different configuration of the system

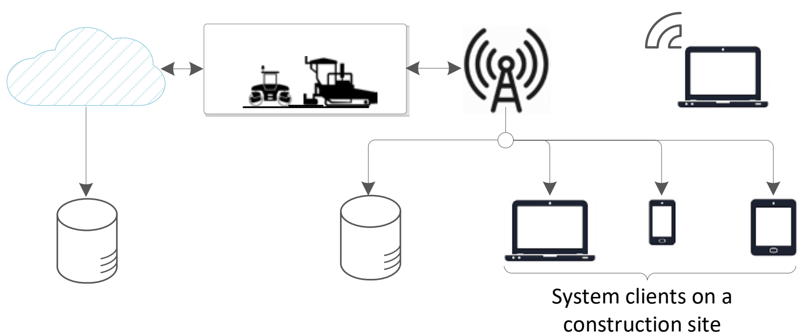
Configuration	**Advantages**	**Disadvantages**
Remote centralized server	- Store all the data in one place - Reduced cost - Reduced maintenance - More security in data	- Cost for data transfer over internet per project - Bandwidth problem over internet
Local centralized server	- No extra cost per usage - No bandwidth problem	- Less security (physical access to server) - Cost for buying a server per contractor - Need to transfer data for backup
Local de-centralized architecture	- Share the process power of PCs to increase utilization of resources	- Hard to implement - Network overhead

	- Increased availability	

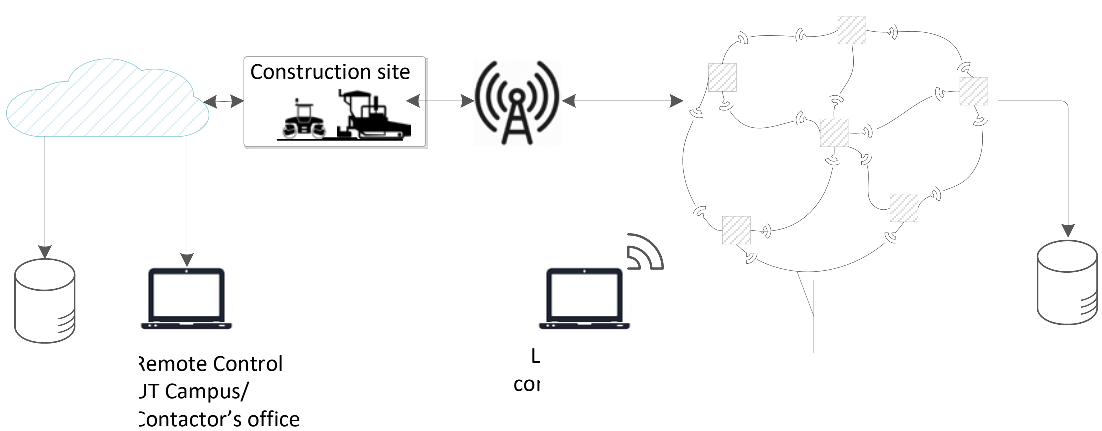
In any case, the structure needs to meet the real-time data processing requirement (i.e., update rate on a client side less than 1 second). Also, the system must be scalable for adding new sensors and equipment.



(a) Remote centralized server



(b) Local centralized server



(c) Local de-centralized computation

Figure 17: Server architecture alternatives

5 Implementation

The software of the ASPARi PQi methodology is a real-time system implemented as a thin client-server application (i.e., the processing at the client side is kept minimal to ensure faster communication with minimal latency) using a .Net framework. Criteria for choosing the programming language were (1) ease of development, (2) ease of maintenance, and (3) efficiency for real-time applications. High level programming languages such as Matlab were easy to develop and maintain but, not powerful enough for keeping the processes real-time. Low level programming languages such as C++ were harder to implement and very hard for civil engineers to maintain. C# and Java had a good balance for the situation and C# was chosen because the trainee was faster in it.

The thin client-server application is chosen because different nodes and sensors operate for a single goal from different locations. In addition, the clients are thin to reduce the power consumption and reduce the total price for scaling up. Moreover, the system is not distributed because a useful output can be generated once we have all the data merged together from different sensors.

The SQL server is chosen for storing data into a relational database as all the PQi data look the same in a project and are relational by nature (comparing to NoSQL). Moreover, to keep the application real-time, the database is not involved in the real-time processes. The database is only used for storing and retrieving data at start-up and the data required for the real-time processing are handled by the real-time application. Nevertheless, a database is needed because the amount of raw and process data can be huge, and databases can help to keep record of the data and to search the data. The SQL language is also easy enough for non-technical people to retrieve the data. Finally, EntityFramework is used as the interface with database for easier data storing/retrieval for the real-time application.

In terms of the communication between clients and the server, SignalR is used. SignalR helps to have real-time full duplex machine to machine communications which means clients and server can communicate with each other at the same time in both directions.

5.1 Implementation structure

The ASPARi PQi system consists of 7 different projects in 2 different categories. By separating the system into modules as illustrated in Figure 19, the system modularity can be secured, the codes can be shared between the modules as much as possible, and new sensors can be added easily and reused by modules.

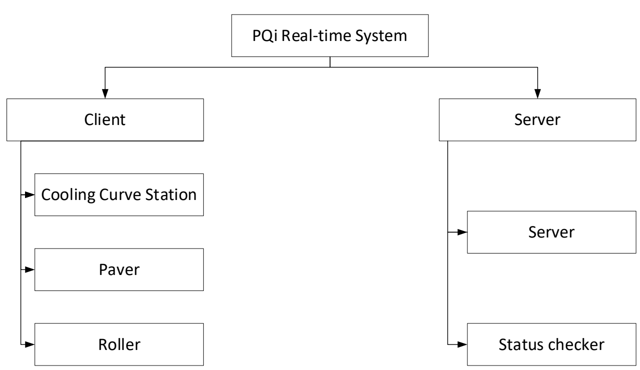


Figure 18: Implementation structure

5.2 Clients

The client nodes are developed as explained in Section 4.1. The full list of clients' features, how to interact with the applications and how to change the configurations can be found in the user manual.

5.2.1 Paver

The Paver application interface is shown in Figure 20. This interface shows how the user can specify the range for the data collection of the IR camera. Regarding the visualizations, the paver application will visualize the TCP behind the paver. The length of the visualizations and the temperature ranges can be set in the database.

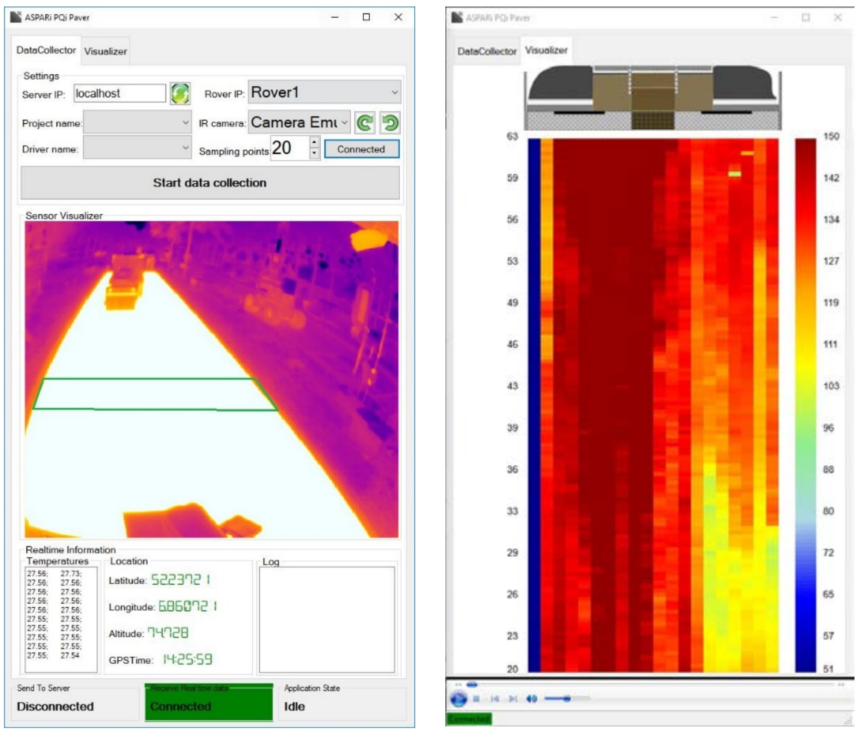


Figure 19: Paver application

5.2.2 Roller Node

Roller node interface is shown in Figure 21. A Roller only needs GPS data to function. For the visualization part, the operator can choose between assistant visualizations or guidance. Assistant visualization consists of TCP and CCP which is visualization of the raw data while the guidance shows where to focus on the road by merging and processing all the data. All the visualizations are based on previous and ongoing ASPARi research. For this project, the author only implemented the visualizations for real-time usage.

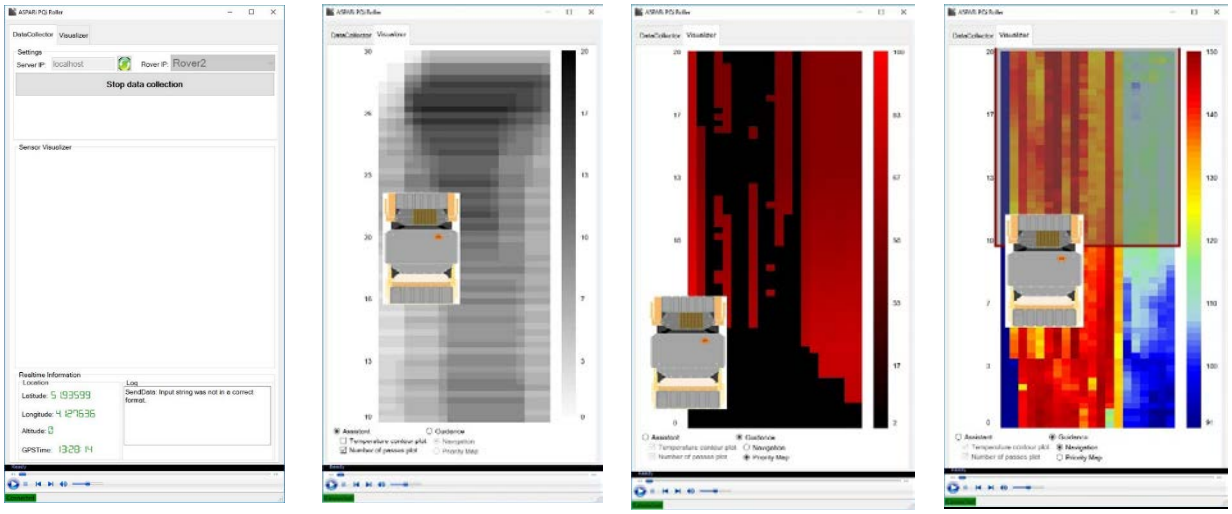


Figure 20: Roller application

5.2.3 Cooling Curve Node

The Cooling Curve Node is responsible for collecting asphalt core temperature and sending them to the server as well visualizing the cooling curve and the temperature window. Figure 22 shows the interface of the cooling curve application.

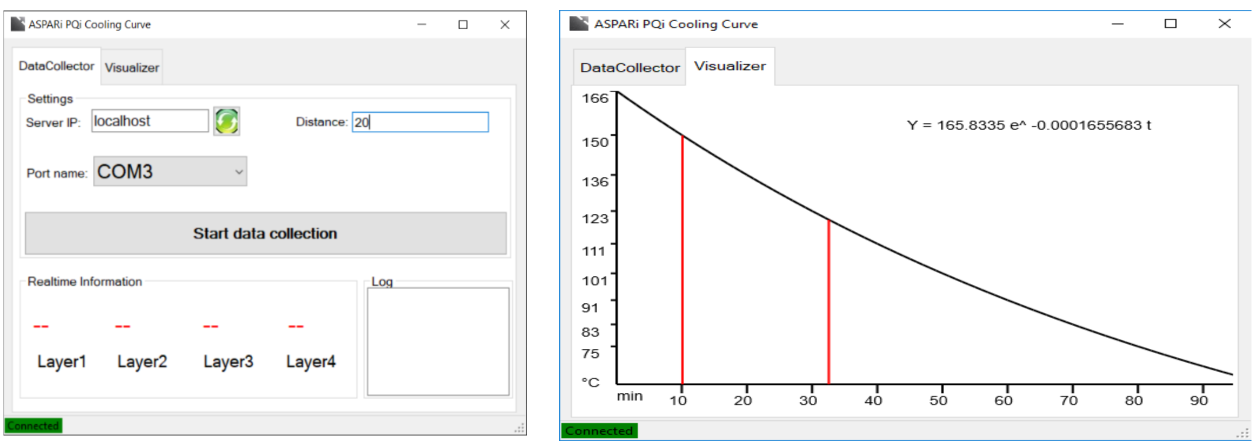


Figure 21: Cooling curve application

5.3 Server

The applications under server category are responsible to provide service to the clients/users which consists of two modules listed as follows:

5.3.1 Server

As mentioned in Section 4.2, the server is responsible for receiving data from the Clients, saving the data into the database, applying different filtering algorithms to the input data, and combining data from different clients. The pre-processed data is also stored into database which is shown in Figure 23.

Figure 23 presents the structure and setup of the tables in a database that will be used for storing the data from construction sites in the real-time application. These tables are defined based on a source of data, for instance, table 'Weather' contains all weather-related information obtained from the on-site weather station. Compared to the database model provided during the author's qualifier, this model is a bit simplified to be able to use it with the current real-time application, as the application needed to be optimized for the available hardware for the server. Moreover, the model is simplified to make sure a PQi can be done with the least effort to be able to focus more on the functionality of the developed system.

Interacting with the database will slow down the process as it usually involves hard disk which is much slower compared to RAM and CPU cache. To avoid this, the server will keep the state of the application and all the needed data into memory (RAM) to keep the application running in real-time. Each time a set of data arrives from clients, the raw, pre-processed and post-processed data will be stored into a buffer to be written into the database at a later stage. However, although the use of a buffer will increase the performance, it also increases the risk of losing data in case an error happens. To reduce this risk, the data will be written on disk each few seconds which, is defined in the settings file. As the data can be easily reprocessed, the intermediate data such as number of passes of each step, do not need to be stored into the database. Instead, these intermediate data are processed and kept in the server as a session. A session contains all the configuration data such as cell size and visualization distance, raw, intermediate and the output data needed for the application to operate. In other words, a session is simply all the data from the time the application starts. This method will help keep the application real-time for the lower-end hardware. Moreover, by having a state-full server and stateless clients, there is no need to physically access the clients to restart the application. Instead, only the session can be restarted from the server. This unified middle-step data structure will also help to reduce the calculations. For instance, by saving data into cells and matrices, instead of searching for 10 meters range in geo-location data, which requires the system to convert and process all the geo-location data each time, it is enough to find the start and end row and use the data in between, in the matrix.

To make easier use of the system, all the necessary settings are listed in the server application as it is usually easier to be physically reached on a construction site. This is so because if the server is physically located on the paver, there is no need to stop the paver to enter settings. This is not the same for the rollers. For instance, project definitions, GPS offset location, roller size, etc. are all set in the server.

In the implementation of the system, the server provides the following functionalities:

1. **Process paver inputs**: Each time an input arrives from a paver, the raw data will be queued to be saved into the database. Then, a filtering algorithm can be applied to the location data. Currently, the data is not filtered but it will be checked if the distance with the previous data and the direction of movement is within acceptable predefined thresholds in the settings. This can help quickly identify the outliers. In case the data is valid, and the paver has moved for at least a cell, the data will be stored in a pre-processed table. If the session is new, all the new points will be accepted to be able to determine the paver direction. Regarding the temperature data, based on cell size and road width, a specific number of temperature points are expected for each reading. In case of any differences, the data will be interpolated/extrapolated to fit the cell structure.

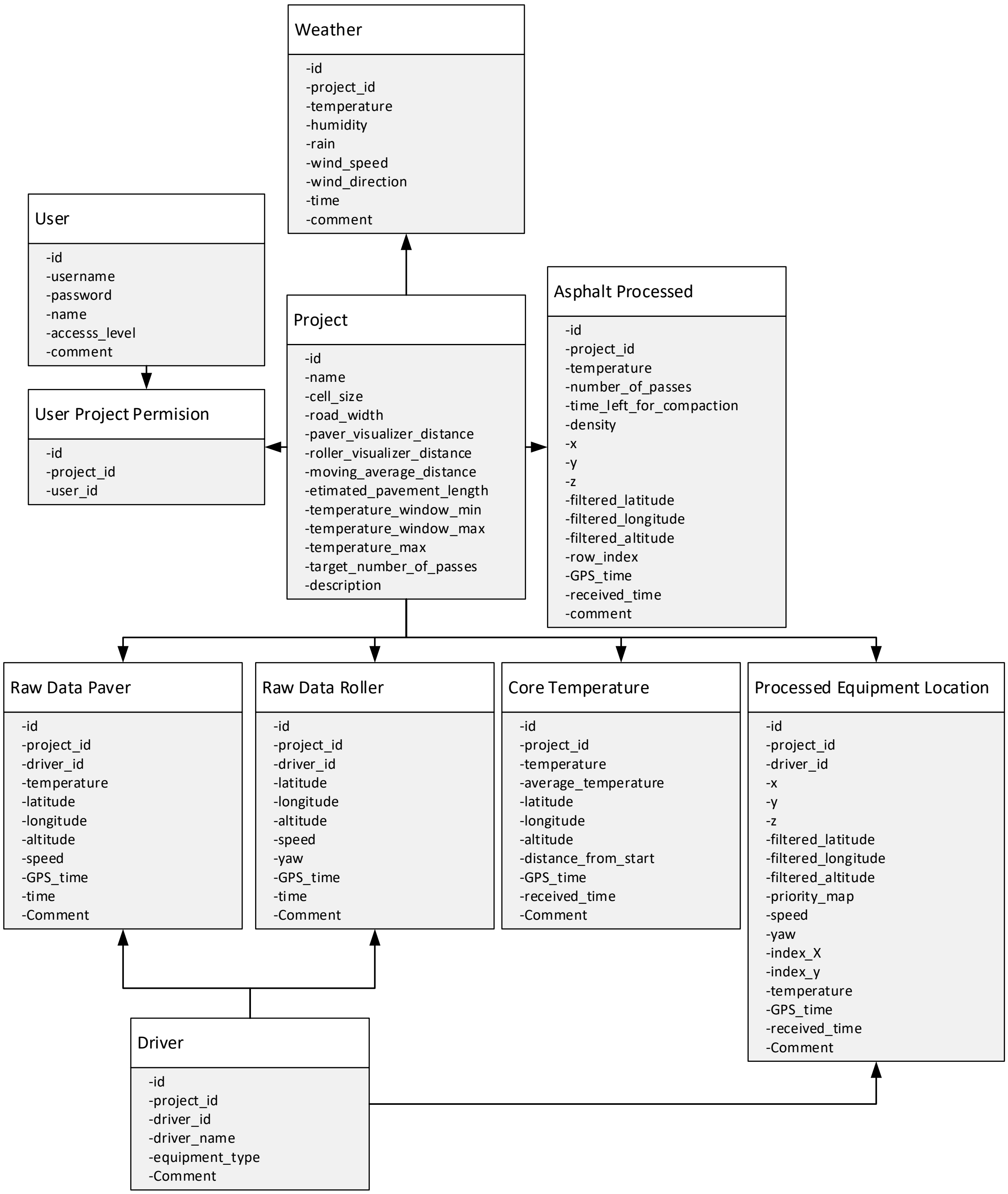


Figure 22: Database tables and relations

2. **Process roller inputs**: Each time an input arrives from a roller; the raw data will be queued to be saved into the database. Currently, no filtering algorithm is applied for rollers, so pre-processed data is the raw data plus UTM locations and cell position of the roller in the output matrix. Updating the number of passes and needed attributes for a priority map such as time left for compaction of each cell, will be updated here.

3. **Process cooling curve input**: Cooling curve data will be stored in the intermediate data structure as well as the buffer for the database.
4. **Update cooling curve**: As the cooling curve data doesn't need high frequency update rates, the cooling curve data will be updated in a separated thread every predefined second in the settings. A new curve will be fitted with the data collected so far. Therefore, all the affected attributes by the cooling curve should be updated, such as the current temperature of the cells and the guidance map.
5. **Generate TCP, CCP, CPM and Guidance**: the data collected from the rollers and paver are used to generate different levels of operator support discussed in Section 4.2.
6. **Send visualizations to clients:** The output data for each client will be prepared and sent to the clients based on the current settings such as visualizer distance. The frequency can be set in the configuration file. To minimize the data transfer over the network between clients and server, the binary serialized of the TCP, CCP, priority and guidance map with some configuration data are sent. As the visualization can be heavy to process, the visualizer module uses a SharpDX library which is the .Net wrapper of DirectX. This has a low overhead visualizer; we literally paint the needed pixels on screen. Based on the screen and visualization distance, the size of each cell will be calculated in pixels. Then based on the type of visualization, a colour will be assigned to the representative pixels for each cell. Finally, the overlays such as legends will be added. As an optimization, only needed pixels will be updated. This method is much harder to implement but much faster to process. This is one of the main reasons a cheap mini-PC for clients can also handle the processes in real-time.
7. **Saving data into database:** As previously mentioned, reading from the database and saving into database can have a significant impact on the performance. To utilize the resources more efficiently, there is a buffer for the data that should be saved into the database. The data will be stored into the database every second as defined in the settings.
8. **Preparing a post-project report:** The server can provide a post-project report that encompasses TCP and CCP map as well as efficiency statistics and detailed TCP and compaction map data for each session. To prepare post-project report for old projects, the data needs to be loaded into a session. Also, to have intermediate data, such as a priority and guidance map, the user can use a replaying data feature which is added to the system.
9. **Exporting data:** To have interoperability between different applications, the raw, pre-processed and output data can be exported in different formats. The exported data can be from a session or from a defined range from the database.
10. **Replaying data:** To have interoperability between different applications, the raw data from other sensors and applications can be imported into the application. It should be noted that because the ASPARi data collection process is different from other commercial products used by some of the contractors, when importing other formats, only raw data visualization is functional. Moreover, another useful use case of replaying data is the ability to see the middle step data such as TCP and CCP maps during the compaction, as well as priority and guidance map. Instead of having a predefined animation, the users can also interact with the map (zoom in, zoom out and pan) to have more detailed information. The user can set the replay speed in real-time and can pause the replay to discuss certain scenarios. As the replay data will be loaded into a session, all the session features such as visualizations on the Clients are available. Moreover, while pausing the replay, post-project reports can be also generated which can be very useful for detailed discussions based on the data. Having all the mentioned features working together in replay and making sure to have exactly the same results each time a set of data is replayed is only possible if they are considered in the system architecture because, in floating point calculations, x * y / y does not necessarily equal x and the result can be

different based on different software settings. Moreover, when time is considered in calculations, processing the inputs based on predefined exact timeframes cannot be guaranteed. Although the errors are very small in each calculation, the accumulative error after an 8-hour run (workday) can be significantly high.

In addition to the above functionalities, the server includes a map service. Map is responsible for visualizing the current state of the application. It uses Google map for the map tiles and shows the current location of paver, rollers and the paved asphalt with icons as well as their work paths displayed as lines.

With respect to extensibility of the system, the used architecture can easily accommodate new sensor alternatives as long as they follow certain protocols. For example, each module should have a creating, initialization, finalization, set configuration and read from the device in defined formats. As long as the format is followed, one can easily switch between the sensors without any problems. Currently, the application can read from different GPS modules such as Trimble SPS852, R12 and Ublox as well as different thermocouples such as HD500, HD200 and SDL200.

Figure 24 shows the server application interface. The right panel shows some of the settings that can be changed for the server applications (as will be explained below). There is also a configuration file where other less common settings such as different module update frequency and GPS offset can be changed. The full list of settings and their functionalities are listed in the user manual.

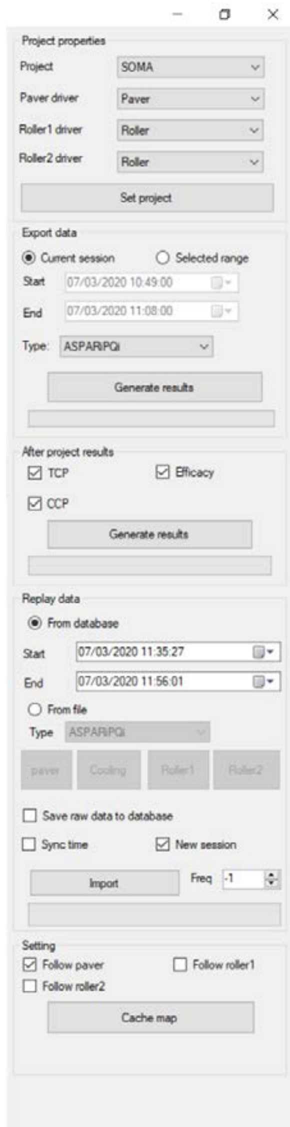


Figure 23: Server visualizer with one paver and one roller movement

5.3.2 Status server

During the development of the application, as the applications were not stable yet, it was important to check the status of all the applications in one place. Without the status server, someone had to interrupt the compaction process to check the application status on each machine, which obviously was not safe and reliable. So, the status server was developed to be able to check all the application status without interrupting all the devices including mobile phones. Later, it turned out that site managers are also interested to explore some of those data in real-time, especially the initial asphalt temperature without measuring the temperature themselves. Therefore, this was implemented in the status server. A sample output of status server is shown in Figure 25.

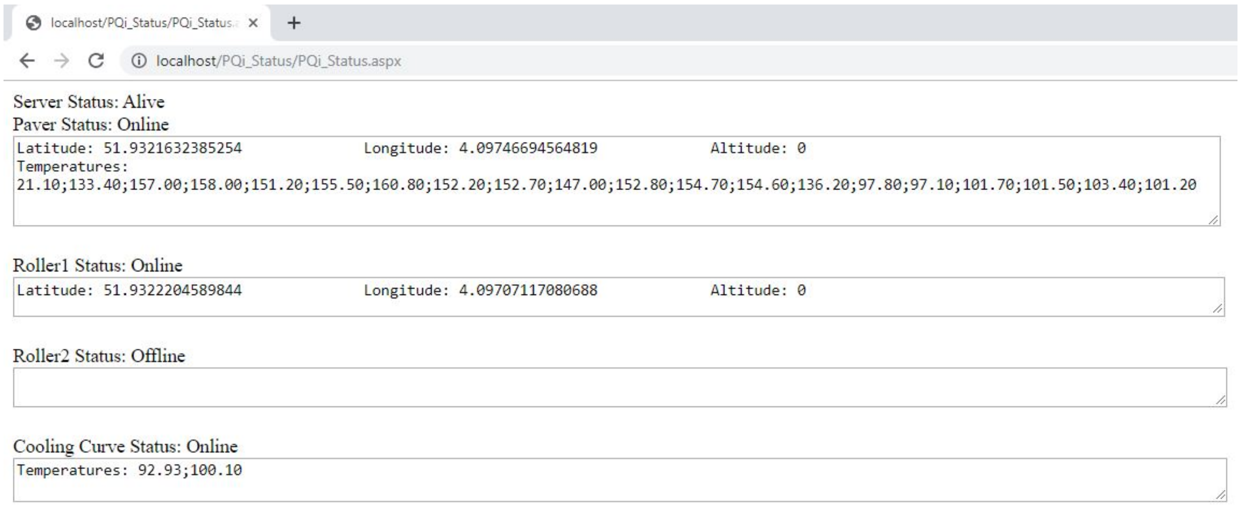


Figure 24: Status server

6 Case Study and Validation

Following the design methodology presented in Section 2.1, a system prototype was developed, tested and validated on multiple road construction projects in the Netherlands. A brief overview of these projects is presented in Table 3.

Table 3: Case studies of the developed system prototype

#	Construction project	Contractor	Conditions	Tested features
1	2019-07-05 N639, residential road	Boskalis	Real construction conditions*	-setup of system nodes -server-client connections -usage of gps-base station
2	2019-09-26 SOMA Colledge Opdrachtgevers workshop, training track	TWW	Semi-controllable environments**	-setup of system nodes -system wireless network setup -communication between system nodes
3	2019-10-05 Laan van Erica, Apeldoorn, residential road	KWS	Real construction conditions	-full system setup -communication between system nodes -testing of status server application
4	2019-10-18 Raapopseweg, Arnhem, residential road	Roelofs	Real construction conditions	-full system setup -communication between system nodes -testing status server application
5	2019-11-26 Kroezenhoek-West, Den Ham, service road	Roelofs	Real construction conditions	-full system setup -communication between system nodes -testing status server application
6	2019-12-19 Nieuw-Zeelandweg, Amsterdam, service road	Boskalis	Real construction conditions	-full system setup -communication between system nodes
7	2020-04-18 Kattenburgerstraat, Amsterdam, residential road	KWS	Real construction conditions	-full system setup -communication between system nodes
8	2020-05-13 SOMA Colledge, training track	SOMA	Controllable environments***	-full system setup -testing of machine's operator support modes
9	2020-07-03 SOMA Colledge, training track	SOMA	Controllable environments	-full system setup -testing of machine's operator support modes
10	2020-09-25 Laan van Erica, Apeldoorn, residential road	BAM	Real construction conditions	-full system setup -communication between system nodes
11	2020-10-04 Laan van Erica, Apeldoorn, residential road	BAM	Real construction conditions	-full system setup -communication between system nodes
12	2020-10-09 N765, Kampen, residential road	TWW	Real construction conditions	-full system setup -communication between system nodes

Table 4 and Figure 26 show all the instruments (hardware) used in the development of the prototype.

Hardware	 IR-camera	 WiFi router	 Battery	 Power converter	 GPS rover	 LCD screen	 Compact Mini-PC	 IR-camera	 Thermo-logger	 Laptop
Nodes	Paver Node	Paver Node	All Nodes	All Nodes	Paver, Roller Node	Paver, Roller Nodes	Paver, Roller Nodes	Asphalt Node	Asphalt Node	Asphalt Node

Figure 25: Prototype hardware

* Real construction conditions – system prototype was tested under real conditions of asphalt construction site (developers were not allowed to interfere in the construction process);
** Semi-controllable environments – environments where developers could partially interfere in the process and define the characteristics of the machines' movements;
*** Controllable environments – developers had full control during the tests, could interfere in the process and define the characteristics of the machines' movements.

Table 4: Hardware used in the prototype system

Node	Hardware
Paver node	• a wide range FLIR IR camera (collected surface temperatures of the asphalt mat behind screed of the paver) • Netgear broadband wireless router (facilitated communication activities between system nodes) • 10.1inch HDMI LCD screen (displayed the user interface) • Ultra HD Compact Mini-PC (collected, stored and pre-processed the data) • CSB power supply battery (12V-30Ah) • Power converter (12V-230V) • Trimble SPS 852 RTK GPS rover (used for localization of the paver on site)
Roller node	• a wide range FLIR IR camera (collected surface temperatures of the asphalt mat behind screed of the paver) • Netgear broadband wireless router (facilitated communication activities between system nodes) • 10.1inch HDMI LCD screen (displayed the user interface) • Ultra HD Compact Mini-PC (collected, stored and pre-processed the data) • CSB power supply battery (12V-30Ah) • power converter (12V-230V) • Trimble SPS 852 RTK GPS rover (used for localization of the paver on site)
Asphalt node	• voltcraft IR-1600 CAM infrared thermometer (measured surface temperatures of the asphalt mat) • Extech HD500 thermologger and thermocouples (collected core temperatures of the asphalt mat) • Laptop Dell Latitude 5480 (collected, stored and pre-processed the data) • CSB power supply battery (12V-30Ah) • power converter (12V-230V)

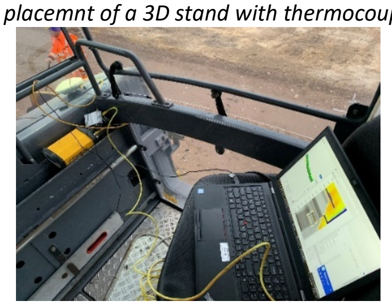
For the sake of brevity, one case study (2019-11-26 Kroezenhoek-West, Den Ham, service road) will be explained in full as an example.

In Den Ham, as shown in Figure 27, several stretches (which collectively amounted to approximately ~350 meters) were paved and then compacted. During this case study, one paver and one roller were equipped with corresponding system nodes (Paver and Roller Nodes, Figure 28 (b,d)). The Asphalt node was repeatedly deployed each ~50 meters along the paved road (Figure 28(a)). After the setup of all sensors and devices of the system prototype, the system was run through the entire project, facilitating paving and compaction operations during the ~4-hour-long project.

Figure 26: Testing site


(a) placemnt of a 3D stand with thermocouples


(b) IR camera on the paver


(c) paver operator display


(d) roller operator display

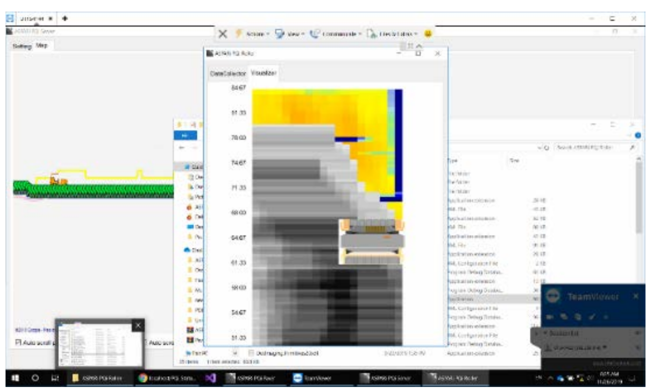

(e) site manager display

Figure 27: System prototype nodes' setup on site

Through the use of the system, the paver and roller operators had the ability to follow the current state of the asphalt mat and the performed compaction effort. Based on the paver locations, which was gathered by GPS sensor, and temperature from the paver IR-camera, the visualizations for the paver operator were created (asphalt mat temperature profile, Figure 28 (c)). The roller operator was supported in real-time with compaction visuals based on the temperature data from the Paver and Asphalt Nodes in combination with GPS coordinates of the compactor (Figure 28 (d)). Also, the site manager had a remote overview of the construction process through the corresponding interface that included (1) the map of the construction project with current locations of the involved machinery (paver and roller), and (2) generated visuals for the paver and roller operators (Figure 28 (e)).

The results of the case study showcased the functionality of the proposed system in a real project. The operators of the construction machines, during short feedback sessions, expressed their willingness and interest to work and test the system prototype more frequently during different working situations. The operators emphasized that the use of the compaction and asphalt temperature profile visuals provide insights into the current state of the project and help to adjust their working patterns.

The above case study focused only on the presentation of the TCP and CCP. To test the prototype for the priority mapping, another case study was conducted. In this case study, the premise of the SOMA college was used to simulate an actual paving operation using actual equipment. The SOMA college is the largest construction equipment vocational training school in the Netherlands and different pieces of construction equipment are used for training the young students. Figure 29 shows the setup of the case study. The hardware setup was similar to the previous case study. Figure 30 shows the presentation of the data in different modes. Again, this case study also indicated the successful performance of the system in generating priority maps to the operators.

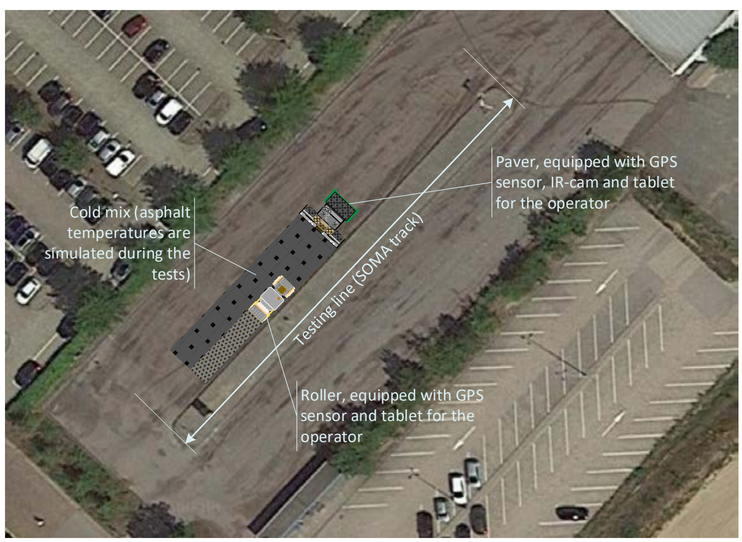
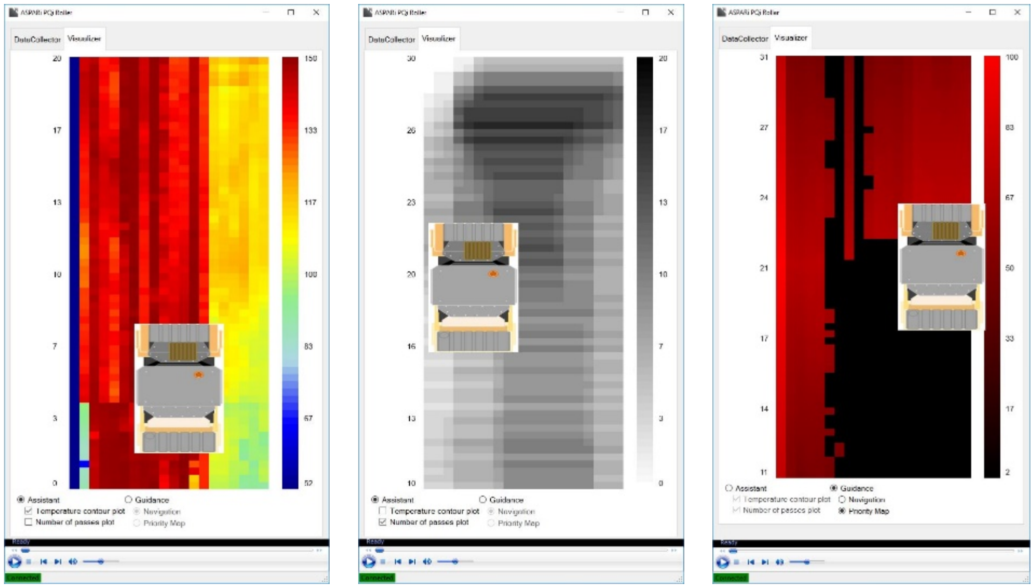


Figure 28: The paving site in SOMA College (Makarov et al. 2021)



(a) TCP (b) CCP (c) CPM

Figure 30: The visualization of the results to the operators (Makarov et al. 2021)

7 Exploration of Alternative Solutions

In addition to the development of the core system presented in the previous chapters, the researcher has ventured into developing and experimenting with several alternatives for different software/hardware components of the system. This was done in an effort to provide a roadmap towards possible directions in which this system can be further improved in the future. Two alternatives were explored: (1) sensors: different types of sensors were explored to make the system more practical and accurate, and (2) database: an alternative structure of the database was explored to enhance the extensibility and scalability of the system.

7.1 Sensors

7.1.1 Mini PC

To be able to use different sensors in real-time, a PC is needed to read the data and send them to the server and receive the outputs for visualization. Conventionally, this was done by a bulky laptop that would render the system impractical and exposed to different types of accidental damages. So, the first step in modernizing process control was adding a mini-PC to each equipment and develop application for each one to be able to interact with the sensors. Synchronization of data from different sources with separated time sources is not only cumbersome, but also introduce latency to the system. Having a mini-PC will help to synchronize the data from different sensors easier and with higher accuracy. In this way, in addition to the synchronization of the data based on the time tags of GPS and server, the mini-PC can also be used for an additional layer of synchronization. The mini-PC and its size comparison is showed in Figure 31.

Figure 31: Mini-PC size comparing to a laptop

7.1.2 GPS

After connecting the available GPS to PC, it was discovered that the real-time GPS data of the paver are too noisy which is shown in Figure 32. Offline corrections which can be accessed after 24 hours can fix most of the errors, but then it is only useful for post-processing. Filtering algorithms such as Kalman filter and moving average was implemented but still the data was not usable for the real-time application.

Figure 29: Noisy data on paver (blue) comparing to roller (pink)

To fix the issue, two separate avenues were explored: (1) Exploring alternative GPS and (2) finding the optimum location for the installation of the GPS rover.

7.1.2.1 GPS alternatives

For GPS alternatives, two extreme cases were investigated. A very high-end and a very low-end GPS system. These two were compared with the current GPS system in a dedicated case study.

- **Trimble SPS852 (the current GPS)**: This is a middle range GPS which is designed for heavy duty use. This model needs an external device for receiving the real-time corrections.
- **Trimble R12 (high-end GPS)**: This is the highest-end GPS currently available in the market from Trimble. It can receive the corrections standalone[4].
- **Ublox M8 (low-end GPS)**: This GPS is one of the cheapest GPS modules available in the market. Ublox accuracy has been measured in the previous research (Marius Bredesen, 2019).

7.1.2.2 GPS antenna location

As the data from rollers was acceptable in most of the cases, it was conjectured that placement of the GPS can be a contributor to the low quality of GPS data. So, the antenna was placed in different locations to check the noise level. In the following images, R12 data is shown in red, SPS is blue and Ublox is illustrated in yellow. SPS and R12 are receiving RTK correction data in real-time while Ublox is not receiving the RTK corrections.

First, all three antennas were placed inside the paver as shown in Figure 33. The R12 data is stable enough for real-time usage while SPS data is not usable at all. The Ublox data can be usable when RTK is applied and filtering algorithms are used.

[4] *This GPS was borrowed from Geometius (https://www.geometius.nl/) for testing purposes.*

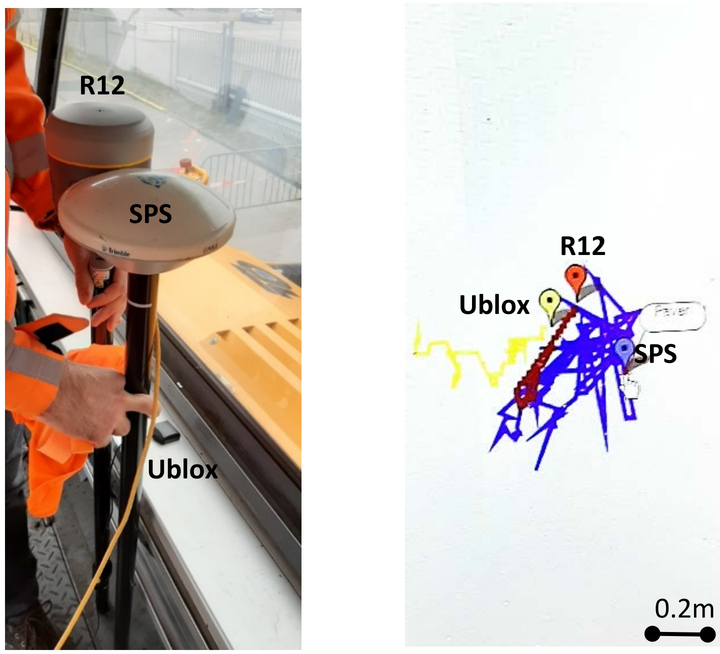


Figure 30: All the antennas in the paver

As shown in Figure 34, by placing the SPS on the roof, the data is much more stable and can be used with filtering algorithms while placing R12 on the hood will make the data noisy. Ublox is still relatively stable but with data drifts which is due lack of RTK data.

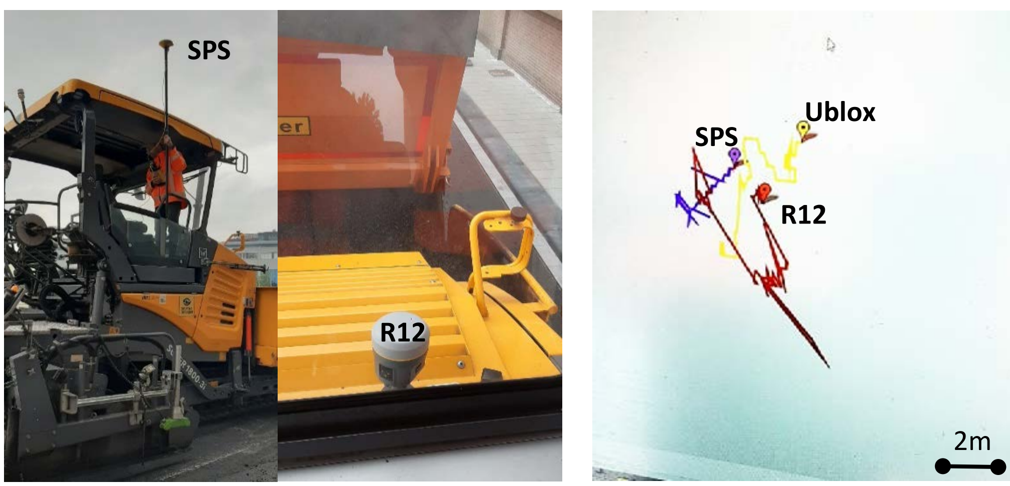


Figure 31: SPS on the roof, R12 on the hood, Ublox in the paver

Placing Ublox on the hood does not add noticeable noises while placing R12 on the paver stairs will reduce the noises but still there are big jumps in the data as illustrated in Figure 35.

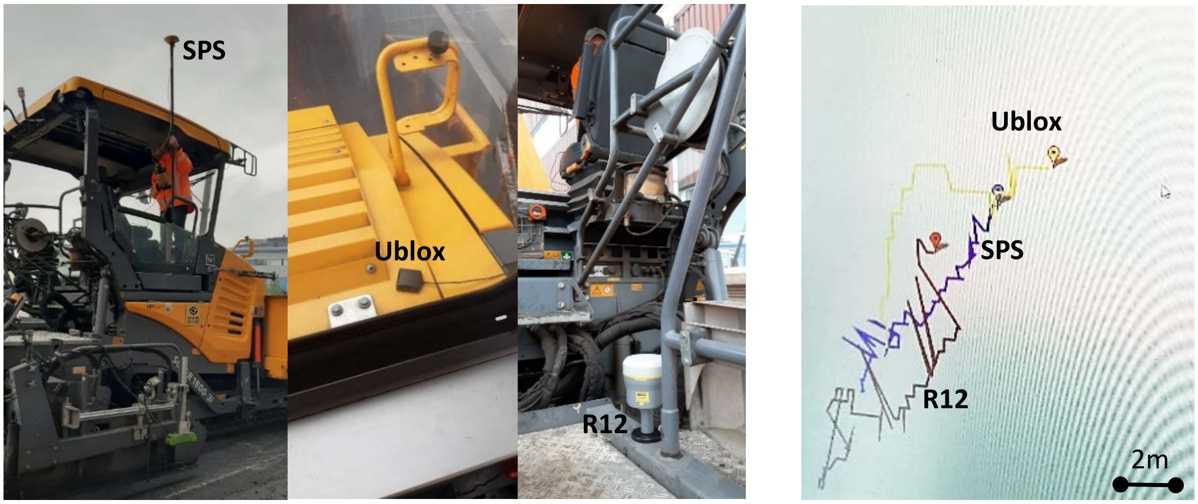


Figure 32: SPS on the roof, Ublox on the hood and R12 on the stairs

Changing the positions as shown in Figure 36 will make all the received data relatively stable. Filtering algorithms can be used for this level of noise for SPS and Ublox.

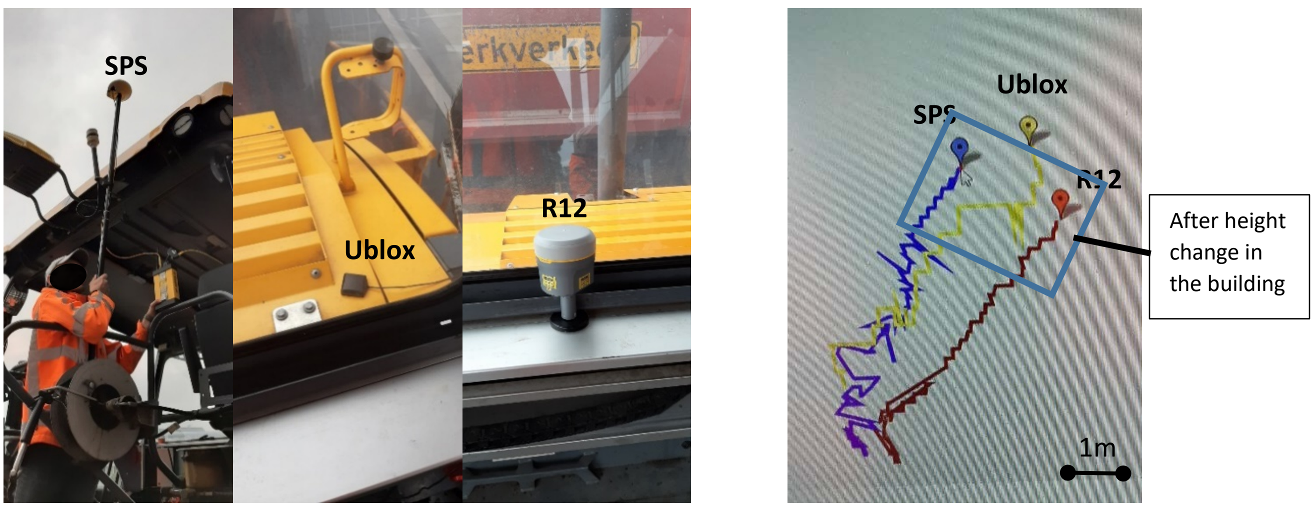


Figure 36: SPS on the roof, Ublox on the hood and R12 in the paver

In Figure 36, it is clear that the data becomes more reliable compared to Figure 32. By observing the environment as shown in Figure 37 it can be observed that as the height of the building decreases, the accuracy of the GPS data increase.

Figure 33: Area around the pavement

When the project reached a relatively less dense area, the SPS data became much more stable. Thus, the application was able to filter the small noises and did not get stuck in an invalid input state due to the big jumps. The Ublox data is also reliable considering the lack of RTK data. The R12 is reliable in most of the cases but still big jumps appear in the data as shown in Figure 38. In the analysis of the accuracy of GPS data, the stability of the GPS antenna needs to be taken into account. Even in a completely open area without any obstacles, GPS data can be noisy because of the high vibration of the equipment or tilts of more than 15 degrees.

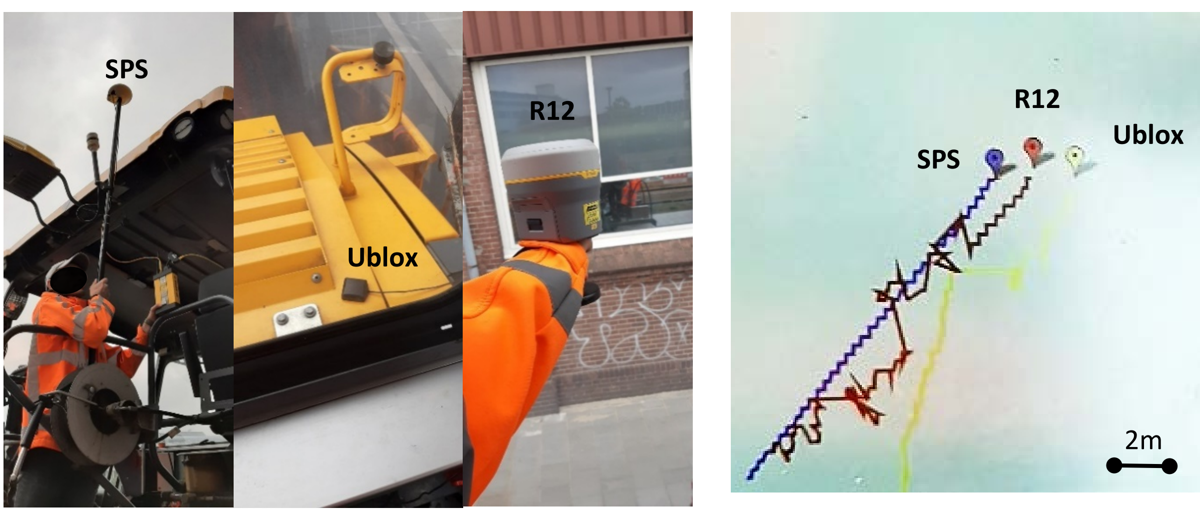


Figure 34: SPS on the roof, Ublox on the hood and R12 outside of the paver but unstable

As shown in Figure 39, in the case of the Ublox antenna, the position does not have a significant influence on the data quality as there is no RTK data altogether.

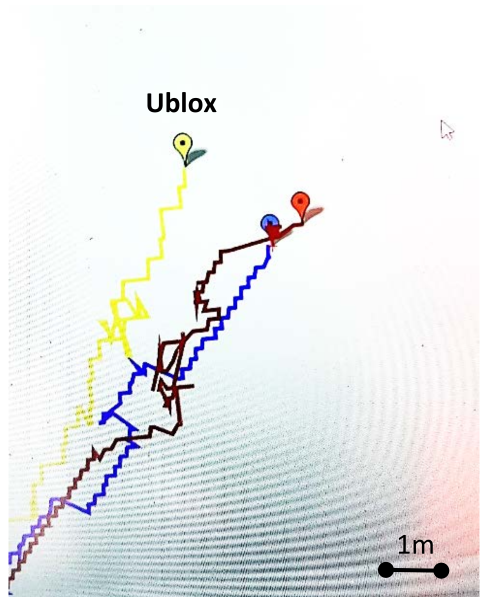
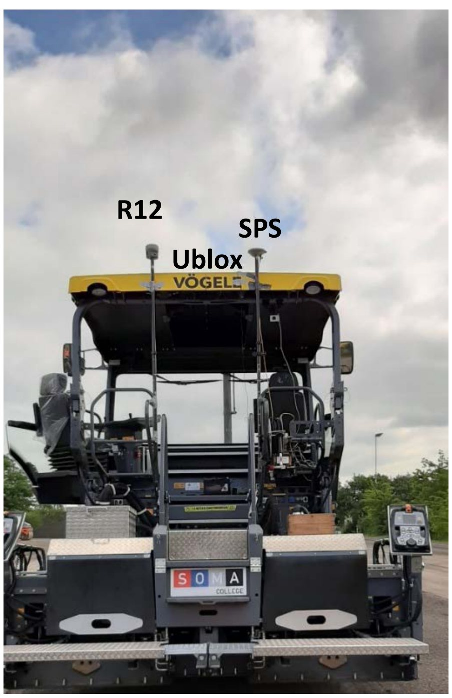


Figure 35: SPS and R12 on the roof but unstable, Ublox on the stair rails

Finally, all the three antennas were placed on the paver roof as illustrated in Figure 40 at SOMA college. The SOMA test track is in an open environment. So, as expected, by placing the sensors on the paver roof, all the GPS data became stable enough to be used in a real-time application. Since there was no ground truth for this test to measure the actual errors, only visual inspection can be used for the subjective assessment of the accuracy. In this way, the R12 data was found to be more reliable. There were small jumps in the SPS data and a larger ones in the Ublox data but, still acceptable for the real-time application.



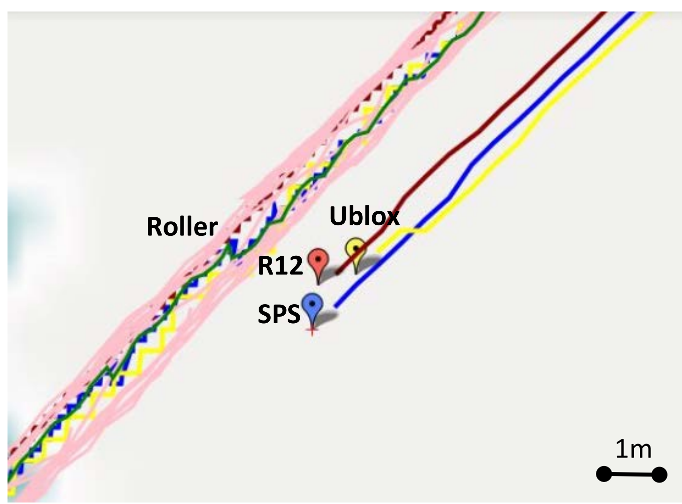


Figure 40: All the sensors on the roof

To sum up, the GPS antenna position and stability makes a significant difference in the performance of the system. The best place for GPS antenna is on the roof of the equipment, which is far away from hot asphalt and other noise sources in the equipment. Surprisingly, although Ublox antenna is the smallest and the cheapest, it operates acceptably in less dense areas. Therefore, it is conceivable to use the lower-end GPS systems when the project area is less dense. Finally, while lower-end GPSs are reliable in less dense area, some noise filtering may still be needed.

7.1.3 IR camera

For modernizing the IR camera, the possible ways to stream the IR images to PC were explored. There are different libraries and mediums to read from IR camera. We chose an ethernet cable because of the higher bandwidth and C# low level libraries for achieving higher speed. Also, the possibility of adding the software features for selecting the right part of image which is being paved. The previous application only read the centre horizontal line of the image, while in the new application one is able to choose the focus area. The software features are described in Paver client section 5.2.1.

In terms of hardware interaction, the current implementation can read from all the IR cameras that can connect to PC via Ethernet or Wi-Fi. To test the system, a previous Flir AX5 model was successfully tested. Also, IR images from a drone (as will be discussed in Section 7.1.5) were successfully connected to the system.

7.1.4 Cooling curve

Determining the asphalt cooling curve is one of the main characteristics of ASPARi's PQi method. By knowing the asphalt temperature behaviour, the cooling pattern can be predicted, and more meaningful information can be provided to the operators. To emphasise the importance of the cooling curve and to ease the installation of cooling curves stations, a battery, Mini PC with LCD, cheap GPS sensor, thermologger and thermometers were packed into a case. Having all the sensors in a small box helps collect the core temperatures easily and send the data to the server. By using a GPS sensor, there is no need to measure the exact location of the cores (after construction) as they will be measured by the system itself.

8 Discussions and recommendations

In this chapter several discussion points are raised. The focus is to highlight some of the technical issues and challenges that are still pertinent to the development of the real-time operator support system. These discussion points will be followed by some recommendations regarding the future development of the system.

8.1 Linear vs Top view visualization

Finding the best match for operator needs can be tricky and needs more investigation. Linear visualization means straightening the curves of the road, while top view refers to the way we see the digital maps from the top.

Linear visualizations are easier to understand in most cases and make better use of the screen area, as the whole screen can be filled with the output. Having said that, intersections and roundabouts cannot be visualized properly in the linear visualization. Visualizing two separate paved lanes next to each other is also not possible with this strategy.

Top view visualizations work best for more complex road shapes such as intersections and roundabouts, but they can waste the screen area for curvy roads. A User or the application can rotate the visualization to fit the screen, but it will add another level of cognitive overload as the screen needs to be rotated all the time. The current system uses linear visualization. However, since compaction of intersections is complex, the current system is very likely to fail in these complex scenarios.

Both methods can be used in a single application and the operators can switch between them. However, it needs more developmental work.

8.2 Cooling curve real-time accuracy and prediction

As mentioned earlier, the Cooling curve is one of the main features of the ASPARi PQi methodology. Unlike all the other available systems, which can only visualize the raw data collected on site, this methodology can predict future asphalt mat temperature by knowing the temperature behaviour of the asphalt mat. So, not only can the current raw data visualization be more accurate, but also operators can be supported to make more informed decisions when considering the mat temperature. Currently, these decisions are based on operator experience. The accuracy of cooling curve measurements is of utmost importance because it will affect all operational decisions and operator efficiencies. Based on previous experience, a new cooling curve is needed every 50-100 meters otherwise the error can rise to 20 degrees.

The current cooling curve fits a curve to the available data set which, is good for offline calculations since it needs all the future data to fit a curve to be able to process each cell. Installing a cooling curve station is cumbersome and it needs an operator on site. Additionally, it is usually not possible to install the cooling station right after the project starts (usually everyone is busy when the project starts, and they won't accept any disturbances). More importantly, at least 20 minutes of data are required to fit an accurate cooling curve. This means the priority map, guidance and even raw data visualization are not very reliable, at least during the first 20 minutes after the first cooling curve installation. This means that while waiting for an accurate cooling curve, 50-100 meters has been already paved and then the cooling curve station need to be repositioned.

To overcome this problem, the calculation strategies need to be changed. Currently, the ASPARi Cool Tool helps predict a cooling curve based on historical data, current mixture properties and weather data.

However, to train a model in this tool, a large set of training data is required. In addition, the accuracy of ASPARi Cool Tool needs to be measured more systematically and rigorously. Finally, given that the tactical crew on the site has little information about the details of the asphalt being used in the project, it would be difficult to know about the exact mixture for ASPARi Cool Tool.

As a side note, while using cooling curve data, the data of roller and paver need to be merged together. Therefore, the impact of the localization error is amplified.

8.3 Types of after-project reports and statistics

There are no standards or requirements for post-project reports and statistics. The answer to "What types of reports do you need?" is always "What can you provide?". Currently the provided system can report the following:

- TCP map
- CCP map
- TCP detailed data (initial temperature of each temperature cell and its location)
- Efficacy report (number of cells compacted correctly, over compacted and under compacted, cells compacted within, above or below compaction window, efficacy of the operator and number of passes and their temperatures of each cell)
- Replay interactive animation of the whole project
- All the row, pre-processed and post processed data

Although the above outputs seem to be enough for the stakeholders, more investigation is needed to know what stakeholders need when using the real-time system.

8.4 Merging data from different systems

Since different types of systems are used in the PQi method, interoperability is needed between the systems. The system proposed in this research can import and export data of the other systems. While this can be a very useful feature, more discussion is needed regarding the practicality of data collection of other systems such as HCQ which, are different from ASPARi. In other systems, the current cell temperature is measured by roller compactors, while ASPARi measures the current cell temperature with cooling curve stations at one location and calculates (predicts) asphalt temperature in other locations. This means that the data exported from ASPARi can be used in other systems but, not the other way around.

To solve this, either the cooling curve can be predicted based on the mixture type and weather data, i.e., using machine learning, or it can be calculated based on the temperatures collected from rollers. The accuracy of the system should be measured with both strategies to see whether they are viable.

8.5 Local or central server

Strategically and logistically, it is preferred to have the sever at a safe location, especially when the system further develops, and a more powerful server is needed. Although having a centralized server is safer, as it needs internet for data communications, it will add complexity to the software development. The bandwidth and data loss will become much more important. In addition, having a remote server will increase ongoing costs of the system for the internet access of each client. The visibilities of the systems over internet, data safety and data encryptions should also be considered.

Although the current system can be used with local and remote servers, it is optimized for the application on a local server, and it lacks optimized strategies for data compaction, encryption and data loss managements.

8.6 Toward plug and play system

A true plug and play system will reduce the costs of the contractors because they can buy a few sets of sensors and use them on different projects rather than buy a set of sensors for all their equipment. Defining the target installation time and setups can help make decisions about the sensor architectures. With the current technology, the system cannot be considered as plug and play because lots of configurations and calibrations are needed to achieve higher accuracy. For instance, the location of GPS antenna and the rotation angle of IR camera affect the system accuracy and needs to be reconfigured per installation. Also, many wires are needed for the system to work. A good solution can be to have hardware and industrial designers design stands and connectors for the sensors. Stands can be used to enable easy installation of sensors on equipment. As the stands are fixed, the configuration on the machine would not change. In addition, hardware designers can design more compact hardware with a designed connector that contains all the needed wires.

8.7 Considering weather data

Although weather data is collected on site, none of the algorithms actually use the weather data. Being clear about weather data and the required accuracy is therefore essential. Maybe weather data from weather stations is enough for cooling curves.

8.8 Appropriate cell size, sensor accuracy and data frequency

Cell size should be the minimum resolution of the system. For instance, if it is required to know the temperature data at least every 10 cm, then the cell size should be 10 cm. To make the cell size fully configurable, it is necessary to define the minimum acceptable accuracy and frequency of the data. As accurate sensors are very expensive and data frequency will also affect computational power, the cell size, sensory accuracy and data frequency becomes an optimization problem that can be solved based on users' requirements and resources.

To optimize cell size, other similar systems can be reviewed to find the minimum cell size currently offered in the commercial solutions. By using a standard and consistent cell-size (i.e., across all the available systems in the market), the interoperability and compatibility of the system can be further improved. The preliminary review done in this project suggests that currently the minimum cell size offered in other system is around 20 cm. Alternatively, the drilled asphalt core sizes can be used as the potential target for the cell size. The average/minimum crack size of the asphalt over the guarantee period can also be a good indication of the minimum cell size. In addition, the minimum manoeuvre area of a roller can also be considered as a reference. Usually, roller operators divide a lane into 2 to 3 sections, i.e., based on their roller width, and focus on each section at a time. In this case, the cell size can be around 1 meter.

To achieve 20 cm cell size accuracy (the current minimum cell size offered in the market), 5 cm accuracy is needed for the GPS data and other mathematical calculations such as IR image unwarping, IR camera tilt correction, etc. This is because in the proposed system the roller and paver data are merged for the final output. However, a 5 cm GPS accuracy in real-time will cost more. Note that an offline RTK correction is completely different to a real-time RTK correction. If a GPS has 5 cm offline accuracy, its real-time accuracy could be more than 30 cm which means 1.2 m cell size.

Update frequency has the least priority as update frequency of the sensors are high enough for the use cases of the current system. Nevertheless, an optimized update frequency can reduce the data transfer rate as well as power consumption. Currently, update frequency of different modules of the system is dynamic and can be set in the configuration files. Sensors will update once every second and the visualizations update twice per second to make sure everything is shown smoothly. Other system such as HCQ update every 5 seconds.

8.9 Which sensors to use

When deciding on appropriate sensors, accuracy is not and should not be the only concern. Cost and usability in different conditions is equally important.

- One-time cost / Recurring cost: While talking about cost, both one-time cost and recurring costs need to be considered. For instance, using centralized server will increase the recurring costs of internet.
- Usability in different conditions: Asphalt construction locations are widespread, on highways, in tunnels, city centres and even restricted areas. A cheap GPS sensor can be used in wide highways while they would not be as effective in denser areas, even if high accuracy is not needed. Also, while drones can be cheaper to simultaneously monitor several machines , it might not be usable due to the low flight time and flight restriction zones.

8.10 GPS antenna locations

GPS antenna location can play an important role in terms of system accuracy. Previously, while using the line scanner, the GPS antenna was attached to the line scanner frame. After replacing the line scanner with a smaller IR camera, as it was hard to place the antenna on the paver roof, the antenna was placed close to the paver hood. When the paver starts paving, the asphalt heat, engine vibrations, antenna tilt and signal blocks by the asphalt truck, introduced a considerable level of noise to the real-time data. After testing different antenna locations, the best place to locate it appeared to be the paver roof.

8.11 Considering compaction strategy

Current implementation only focuses on achieving the total number of passes within the temperature window. In terms of the proposed guidance and priority maps, the current solution can only provide information based on the assumption that there is only one compaction stage and there is one roller per lane of compaction. In practice, there are at least a break down and a finishing phase for the rollers. There might also be more than 2 rollers focusing on a single part of the road. The strategy also might differ on different asphalt layers and intersections. Sometimes the role of rollers might also change during compaction. For instance, roller number 1 starts with the break down phase and roller 2 does the finishing phase. However, the roles may quickly reverse when roller 2 goes closer to the paver and undertakes the breakdown phase, while roller 1 compacts behind roller 2 and does the finishing part. In addition, in the future, there may be different optimization factors in terms of guiding the operator. One might need to minimize the compaction time while others may want to minimize CO_2 emission rate or fuel consumption.

To overcome these challenges, there should be a fleet of rollers with a predefined structure. Alternatively, the system can set rules for each roller operator to attain a certain goal or a set of goals. Examples are achieving the highest compaction rate, reducing fuel consumption, reducing emission rates, etc.

8.12 Actual temperature window and target number of passes

Knowing the actual temperature window and target number of passes is much harder than it seems since it needs to be determined in the laboratory. The system contains some general rules about different asphalt mixtures such as predefined temperature windows for different mixture types. However, these could add inaccuracies to the system, since all the outputs rely on a pre-determined temperature window and the target number of passes for which laboratory measurements are needed.

8.13 Considering roller roles, weight, speed, vibration, water spray, sand spray, etc

The compaction process is basically about compacting certain number of passes within a temperature window. Currently the system does not consider if the project is in the breakdown phase or the finishing phase. Also, roller speed, weight, vibration, and other factors influencing compaction are not considered in the calculations. These attributes should be considered, and it is important to study their influence on the overall process quality. For instance, if the target number of passes is 10, it is important to know what proportion should be done in the break down phase and what proportion in the finishing phase. In addition, if the break down roller compacted a cell 3 times without vibration and once with vibration and very slowly, should the finishing roller compact that cell 5 or 6 times? If a roller stands still, how many numbers of passes should be added each minute? In more complex scenarios, if a few cells are not compacted within a fully compacted area and there is still time left for the compaction, should the system suggest compacting those cells while it will over-compact the others?

8.14 University role

The roles of universities should be defined clearly. Generally, the university role is to prototype new ideas and try to steer industry towards optimized performances. At the same time, the university is providing PQi measurement as a service to ASPARi members, which requires a fully working product. Also, some of the project definitions assumes there is a fully working real-time system. To be clearer, if the university role is to provide insight on different sensors/algorithms/features, the tests should be done in the laboratory environment to be able to have control over different parameters. As this level, testing on the construction site adds orders of magnitude to the complexity of the problem, especially when certain contractually-binding outputs are expected from the PQi measurements by the contractors.

In addition, company supervisors and ASPARi members in general, are looking for an industrial output which can compete with current products in the market. For instance, certain companies want to decide which set of applications should be used for their future projects and compare the current ASPARi system with the available solutions in the market. If this project is defined to show some possible ways of modernizing PQi measurement, accuracy of the system and sensor cost could have lower priority while the system architecture, ontology, system interoperability, different types of options in the application, etc. could be more accentuated.

8.15 Real-time density measurement

The ultimate goal of a compaction operator support system is to help operators achieve the desired level of density accurately and consistently. However, the current system in the market, in general, and the ASPARi system does not measure density in real-time. Although real-time density measurement is complex, maybe focusing on one final parameter is a good idea. This would allow simplification of most of the attributes in different projects without knowing their actual effects on the final objective (i.e., achieved density).

8.16 Building trust with operators and companies

One of the biggest barriers in current tests is the lack of operator trust. In some extreme cases, operators disconnect the system module and throw them away or they simply ignore the visualizations and work on their own. In most of the cases, they are afraid of reports showing they are not doing a decent job. Others, usually trust their experiences and their gut feeling more than the system outputs.

Moreover, none of the ASPARi contractors will take the risk to follow the system instructions in an actual project as they have to pay penalties or redo that work in case of any failure during the guarantee period. Although it makes sense from the business perspective, the whole trust issue holds the research back.

To build trust, it is important to have a fully working product with proven effectiveness. At the same time, to build a fully working product, it is important to have the operators' trust to test and measure the errors on actual construction site without any consequences. Defining a clear road map, research goals and long-term strategies seems very essential here.

8.17 Use cases

There are hundreds of attributes affecting the final quality of a pavement project. As the overall project context is complex, it is nearly impossible to develop an efficient product without considering all the possible use cases. For instance, when calculating the number of passes, how many passes should be considered if the roller stands still at a given location? What should one do in case of jumps and outliers in the GPS data? A real-time product can be much harder to develop as all the questions have to be answered with the current available data. For instance, filtering noise is way harder in real-time. Listing all the possible use cases on a construction site and the expected application outputs in different scenarios is the key to have a fully working product in all the scenarios.

Also, for each use case, there should be a general condition that cover all the affected attributes and generated outputs. In case the behaviour is supposed to be different for different attribute ranges, the behaviour of each range should be clear and boundary conditions and cases should be fully considered. For instance, when talking about a temperature window, all cases where the temperature window has an impact should have defined scenarios for above, within and below the temperature window. In a more complex scenario, when talking about the guidance map, the affected attributes are the target number of passes (below, equal, above), temperature window (below, equal, above), time left for compaction, time to compact a cell, number of compacting rollers, stage of the compaction (break down, finishing), etc. So, there should be a clearly defined output for the combination of each scenario. In addition, as this system is real-time, the issue of data loss should also be taken into account. Unfortunately, most of the available algorithms in the ASPARi system are based on one ideal situation.

8.18 Project strategies

Introducing new technology to people usually frustrates them in many cases. First, providing information about a whole new process can be cognitively overloading. Second, people might get frustrated by new technologies because they cannot trust them immediately. Finally, operators of asphalt equipment (paver, roller and truck) might feel uncomfortable with new technology as they think it might be a threat to their jobs.

To overcome these problems, a virtual reality simulator that is based on real data can be considered. Through the use of this simulator, it is possible to first check and then test the new ideas with the end users. The simulator removes the need for prototype development to a great extent. By involving end users in the development process, it also becomes possible to obtain better results that can help reduce the fear of end users about the new technology and help develop the custom-made solutions. Training centres such as the SOMA college can also benefit from this simulator as a new tool for training operators. The University of Twente as another external stakeholder, also can use the simulator for presenting its prototypes and can analyze the construction process by reviewing the whole process and suggest new methods.

9 Conclusions and future work

This project focused on the modernization of the PQI system that has been used within the ASPARi network for more than a decade. The focus was mainly on the development of a robust module to enable the real-time application of the PQI system on the site following a systematic design methodology. Previously, the main focus of the PQI data collection method was ex post analysis. With the development done in this research and through the design/implementation of a new system architecture (including IoT nodes, relational database, loose coupling between modules, etc.), the new PQI system can now be used for real-time applications, albeit with caution because the short time of the project did not allow the optimization of the architecture for all possible scenarios. Additionally, this research made the first foray into investigating new alternatives that can improve the PQI system from the perspective of accuracy, practicality, interoperability, and reliability. Some of the investigated alternatives were developed further into the system while the remaining were only explored at the feasibility testing level.

In general, the following conclusions can be made:

1. While the current real-time system can function under certain ideal conditions, much more research and development are required to further develop the system into a market-ready product. Many aspects of the system need to be considered and strategies need to be devised. This falls well beyond what can be done by a single researcher and requires an industry-wide investment.
2. The ontological approach (and the implementation manifestation of that in form of relational database) proved to be a right direction in securing interoperability not only between different operator support systems (horizontal interoperability) but also between various systems used in different stages of the asphalt lifecycle (vertical interoperability). This research was only the first step in indicating its potential. Again, an industry-wide awareness and commitment is required to further hone the ontology proposed in this research based on the analysis of the lifecycle of asphalt.
3. The placement of GPS antennae proved to be of great significance. This research highlighted some of these aspects and proposed developing optimized strategies for them. However, more work is needed to develop a fully operational plug-and-play system.
4. Different modes of operator support (TCP, CCP, Priority maps, and guidance) were developed in this research. While the feasibility of them were fully demonstrated in this research, it is important to (a) further optimize the methods to consider a wider range of use cases, and (b) investigate the suitability of different modes with respect to usefulness (i.e., the contribution to improving process quality) and usability (i.e., user acceptance and satisfaction).
5. This project built on the research of Makarov et al. (2021) to define the process quality indicators in a more concrete way. However, it is of cardinal importance for the further development of the PQI system (or any other operator support system), to have a process quality indicator that is clearly and explicitly linked to the conventional product quality indicators (e.g., achieved density and layer thickness).

9.1 Project impact

By introducing and further implementing a data model that unites information from asphalt construction activities, this project aimed to improve the management and quality of the asphalt construction process.

Moreover, the data model provides the ability for each and every contractor to choose his/her own hardware for the installation on site. This provides system independence from sensors and devices that would be chosen for the prototype design, allowing for more flexibility to adjust and upgrade sensors/devices. The data model forms a new standard that can integrate all types of data from construction sites and a medium between municipalities/clients, contractors, and researchers for data exchange and sharing projects' results.

This project can be considered as a good example/reference for other construction fields, where the same approach for process related data collection can be applied. Further, the interoperability of data models from different construction projects can help build interconnected and interdependent construction map on a specific city or country level/scale.

Additionally, most of the previous ASPARi algorithms and assumptions were based on offline processes. Additional, full cycles of real-time application on actual construction sites will help future research investigate more practical problems leading towards higher quality asphalt.

9.2 Future Work

Improvements can be made both for clients and server nodes. The list below represents the future tasks of this project:

1. Server
    a. Fine tune the ontology and database to make sure it covers all the future needs
    b. Real-time filtering algorithms
    c. Apply the access level for different users
    d. More test cases
    e. Real-time alarm system (asphalt temperature dropped, paver/roller stops, strategy changed, …)
    f. Back up data, compress, index, optimize data base
    g. Extend importing and exporting protocols
    h. Cover all road shapes and pavement strategies
2. Client
    a. Implement new sensors
    b. Implement different visualizations
    c. More test cases
    d. Casing for mini-PC
    e. Rewrite applications with C++ for raspberry pi to reduce client cost/power usage
    f. Auto update system for new version

10 References

25010, I. (n.d.). Retrieved from https://iso25000.com/index.php/en/iso-25000-standards/iso-25010

A. Vasenev, F. B. (2012). A real-time system for prediction cooling within the asphalt layer to support rolling operations. *Euroasphalt and Eurobitume Congress*.

Acton, M. (2017). *Solving the Right Problems for Engine Programmers*. Retrieved from https://www.youtube.com/watch?v=4B00hV3wmMY

Acton, M. (2019). *Everyone Watching This Is Fired: Tips for Game Industry Programmers*. Retrieved from https://www.youtube.com/watch?v=cV5HArLYajE

Arbeider, C. G. (2017). Planning the asphalt construction process: Towards more consistent paving and compaction operations. *17th AAPA International Flexible Pavements Conference Exhibition.*

B., F. (2008). SIMD (Single Instruction Multiple Data Processing). Springer, Boston, MA.

Belvoir, F. (2001). *Systems Engineering Fundementals.* VIRGINIA: DEFENSE ACQUISITION UNIVERSITY PRESS.

Bijleveld, F. (2015). *Professionalising the asphalt construction process: aligning information technologies, operators' knowledge and laboratory practices.* Enschede: University of Twente.

Bijleveld, F. e. (2015). Aligning laboratory and field compaction practices for asphalt – the influence of compaction temperature on mechanical properties. *International Journal of Pavement Engineering*, (pp. 1-14).

Dorée, A. J. (2003). Co-operation and competition in the construction industry in The Netherlands.

Kniberg, H. (2018). Retrieved from https://blog.crisp.se/wp-content/uploads/2018/10/20181017-leanforum-agile-everywhere.pdf

Langroodi, A. K. (2020). *Design of a Feedback Support System for the Training of Construction Equipment Operators.* Enschede.

Makarov, D. (2017). *Developing a real-time process control system for asphalt paving and compaction.* Enschede: University of Twente.

Makarov, D., Vahdatikhaki, F., Miller, S., Jamshidi, A., & Dorée, A. (2021). A framework for real-time compaction guidance system based on compaction priority mapping. *Automation in Construction*.

Marius Bredesen, E. H. (2019). *Analysis of the positional accuracy of a low-cost dual frequency GNSS module.* Norway: Norwegian University of Science and Technology.

Miller, S. R. (2010). *Hot mix asphalt construction, towards a more professional approach.* Enschede, the Netherlands: University of Twente.

Miller, S. R. (21-23 May 2008). Paving the way forward: A case study in innovation and process control. In E. Beuving (Ed.), Asphalt - roads for life. *4th Eurasphalt and Eurobitume Congress.*

Peter Denning, P. A. (December 2009). The profession of IT Computing's paradigm. *Communications of the ACM.* ACM.

Sun, S. (2019). *A new method for monitoring machinery movement using an Unmaned Arial Vehicle (UAV) system.* Enschede: University of Twente.

ter Huerne, H. a. (2006). Professionalisering asfalt wegenbouw sector. *Wegbouwkundige werkdagen.* The Netherlands: CROW.

Vasenev, A. (2015). *Visualizing asphalt roller trajectories in context: acquiring, processing, and representing sensor readings.* Enschede: University of Twente. doi:https://doi.org/10.3990/1.9789036538602