INTEGRATION OF SWIFT NAVIGATION PIKSI MULTI SOLUTION IN THE AUTONOMOUS DTU DYNAMO

Project leads:
Henning Si Høj
Arkadiusz Bogdanowski
Alessandra Bellina

Advisors:
Jesper Schramm
Ole Ravn
Jens Christian Andersen
Nils Axel Andersen

DTU Roadrunners
Department of Mechanical Engineering – Section of Thermal Energy
Department of Electrical Engineering – Automation and Control
Energivej Building 413 Port 5
Technical University of Denmark
2800 Kgs. Lyngby
Denmark
Tel: 45 25 41 79

contact@dtucar.com

Project period:	30 January 2017 – 30 June 2017
Education:	MSc Electrical Engineering
Class:	Public
Remarks:	This report documents the application of the GNSS solution in the autonomous DTU Dynamo project for the Shell Eco-Marathon 2018 in the Autonomous UrbanConcept category.
Copyrights:	©2017 DTU Roadrunners

DTU Mechanical Engineering
Department of Mechanical Engineering

DTU Electrical Engineering
Department of Electrical Engineering

DTU

Abstract

In the coming years, autonomous vehicles will bring a paradigm change in the way automobiles are operated. With the propagation of these self-driving vehicles, new challenges arise for which innovative solutions must be found.

The Shell Eco-Marathon will launch a competition for fuel-efficient Autonomous UrbanConcept vehicles in 2018. As part of the student-driven DTU Roadrunners project, a platform of hardware and software has been integrated into the DTU Dynamo car. Sensors and actuators have been designed, implemented and tested, augmenting the car with autonomous capabilities. LIDARs, GNSS, IMUs and wheel sensors form the basis for the car's ability to sense its surroundings. The performance of the various sensors has been proven through data collected at the Shell Eco-Marathon Europe 2017. This report presents the application of Swift Navigation's Piksi Multi Real Time Kinematics (RTK) GNSS Receiver hardware on the DTU Dynamo.

Contents

1 Background

1.1 Shell Eco-Marathon

The Shell Eco-Marathon is an annual competition about fuel efficiency held at three separate events in Asia, America and Europe. In 2017, Shell Eco-Marathon Europe took place May 25-28 at the Olympic Park in London with over 200 teams from European schools and universities participating. The goal of the competition is to drive a certain number of laps on the race track while consuming as little energy as possible. Teams can choose to compete in three different energy classes: electric, hydrogen and internal combustion (including Gasoline, Diesel, CNG, and Ethanol). Two car classes are competing, the UrbanConcept class and the Prototype class. The UrbanConcept class should resemble a small city car with four wheels, where the driver needs to sit upright in an adequately sized cabin. On the other hand, the Prototype class does not have as many requirements, often resulting in a smaller three-wheeled vehicle which barely fits a driver laying down. Figure 1 shows all the participants at the event in 2016 with cars from the different classes represented in the front.

Figure 1: Group photo at the Shell Eco-Marathon 2016

Figure 2: DTU Dynamo 13.0 at the Shell Eco-Marathon 2017

1.2 DTU Roadrunners

DTU Roadrunners (Figure 4 and Figure 3) is a student-driven team at the Technical University of Denmark, whose goal it is to construct fuel-efficient ecocars and participate in the Shell Eco-Marathon each year. The team consists mostly of engineering students from the mechanical and electrical departments. Since 2005, the team has participated with DTU Dynamo in the UrbanConcept class and has won its category 10 times. The current version of the car can be seen in Figure 2. It has a custom carbon fiber monocoque and runs on a specially designed ethanol powered combustion engine.

Figure 3: DTU Dynamo 13.0 and the DTU Roadrunners team at the Shell Eco-Marathon 2017

Figure 4: The DTU Roadrunners team after receiving the prize for runner-up in Internal Combustion UrbanConcept at the Shell Eco-Marathon 2017

1.3 The Autonomous UrbanConcept

With fully autonomous vehicles expected to be generally available from major manufacturers before 2020, it is a field that is currently in rapid expansion. For the organizers of the Shell Eco-Marathon, the event is an opportunity to showcase the innovation that is possible in today's personal mobility. Starting in 2018, they plan to introduce a new category to the competition: the autonomous UrbanConcept. Even though exact details of this category are still unknown, it is expected that all technologies will be allowed. The primary challenge will be to manage to drive one or more laps on the track autonomously without any other cars or obstacles on the track. In subsequent competitions, more challenges will be added to showcase the vehicle's ability to handle various real-world situations.

2 The GNSS Hardware

2.1 Basics and Selection

GNSS (Global Navigation Satellite System) is widely used for global positioning in vehicles, ships, and planes. The system is based on receiving broadcasts from at least four satellites containing data about their orbital location (ephemeris) and the precise clock. With the known speed of propagation (the speed of light), the position can be triangulated. The most widely used variant, GPS, can usually deliver accuracies in the range of 1 to 5 m, depending on the number of satellite signals that can be received and the quality of these signals. At these accuracies, combined with precision in the same range of 1 to 5 m, GPS would only be usable to obtain a rough global position estimate.

The main sources of this error in position are the transmission delays due to ionospheric and tropospheric interference, as well as errors in the satellites' position data and internal clocks. Both the accuracy and precision can be dramatically improved by using a base station with a fixed position to send correctional data to the moving unit (often called the rover). A simple implementation of this concept is differential GPS (DGPS), which broadcasts the difference between the known fixed position and the position obtained by the satellite system. DPGS can improve the accuracy to 0.5 to 1 m, which is still insufficient for localization.

A more advanced technique is using Real Time Kinematics (RTK). The GPS implementation, called Carrier-Phase Enhancement GPS, measures the phase of the GPS signal's carrier wave. A fixed base station with known position broadcasts real-time corrections based on that measurement. The base station needs to be in the vicinity of the rover, specifically within a distance of about 10 km, so that it can see the same satellites as the rover and measure the same atmospheric disturbances. Using this technique, accuracies within a few centimeters can be obtained.

In addition to this, the accuracy can be further improved by using a multiband receiver which can receive signals from the L2 band in addition to the traditional, most commonly used L1 band. This adds redundancy to the measurements and can help compensate for signal interference and ionospheric error, resulting in sub-centimeter accuracy. Furthermore, the robustness can be improved by tracking additional GNSS constellations such as GLONASS and Galileo simultaneously in addition to GPS.

There exist public networks of base stations that broadcast their correction data over the internet, notably the Continuously Operating Reference Station (CORS) network focused on the United States and the EUREF Permanent GNSS Network focused on Europe. The base stations are however coarsely distributed, and not all stations transmit real-time data. For example, the nearest public station to the London race track is located at a distance of 50 km. This means a base station would have to be erected to ensure relevant and compatible correction data.

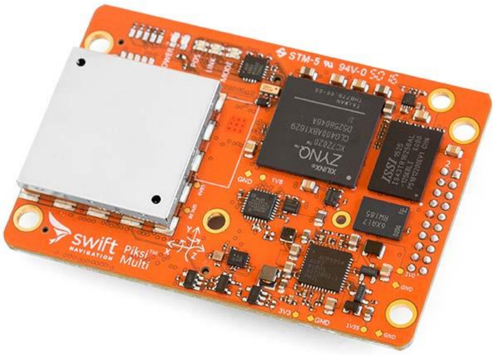


Figure 5: The SwiftNav Piksi Multi RTK GNSS module

Several products were considered for the GPS solution of which the Swift Navigation Piksi Multi (see Figure 5) was chosen. The Piksi Multi has the best accuracy and convergence time of the selection, being the only one capable of multi-band as well as multi-constellation operation. The convergence time determines the time between initialization until the receiver can provide RTK based location. According to its specifications, it is accurate to 2 cm horizontally and 6 cm vertically on short baselines with a good sky view. Its accuracy degrades at a rate of 1 mm horizontal and 3 mm vertical for each kilometer between the base station and the rover. The time until convergence of the RTK fix is less than a minute, and the update rate is 10 Hz. After communicating with Swift Navigation, the company graciously agreed to sponsor the Piksi Multi evaluation kit, including two modules, radios, and high-gain antennas. The standard implementation uses 2.4 GHz radios to transmit the RTK correction data. Since these radios have a limited range, communication over the internet using the 3G or 4G GSM networks was implemented.

2.2 Setup

For the RTK base station, a Piksi Multi module is installed on a tripod with an antenna and a 12V battery. The correction data is sent to an Android phone using the included RS232 to USB adapter and a USB OTG adapter. On the rover, the data is received by a GSM modem and sent through the PC over USB to the RS232 port of the on-board Piksi Multi (see Figure 6). A 3D printed enclosure was produced to shield the Piksi Multi and its evaluation board.

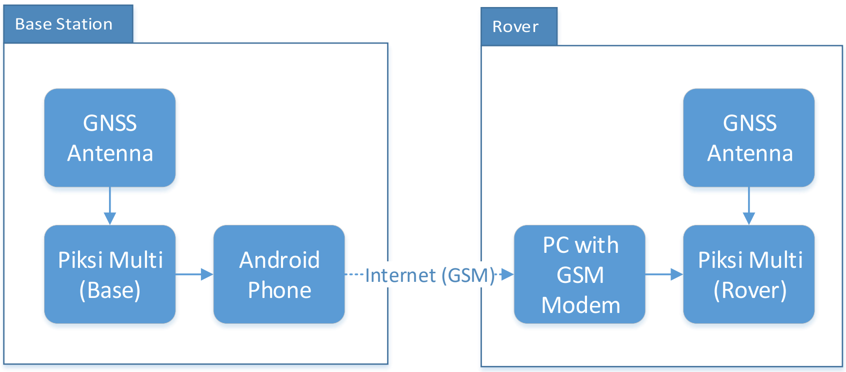


Figure 6: Setup for the Piksi Multi modules

2.3 Mounting the Hardware

The GPS antenna was mounted externally to ensure an unobstructed view of the sky (Figure 7).

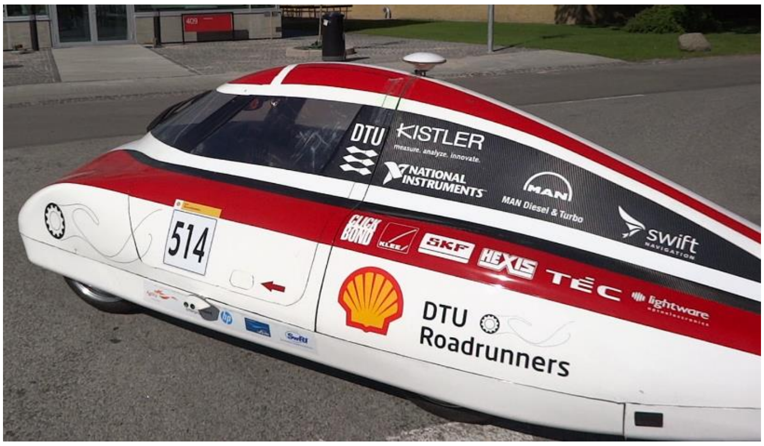


Figure 7: Side view of the car with visible GPS antenna on top

On the left side of Figure 8, the RTK base station can be seen, mounted to a tripod with the antenna, the Piksi Multi, a battery and a phone.

Figure 8: RTK base station and DTU Dynamo without top shell

3 Software

3.1 General

Our self-made software written in C++ runs on the software platform ROS (Robot Operating System) on a low power Ubuntu computer. We decided to use ROS because it is free, open-source and has a lot of useful features, e.g. for visualization and messaging. The hardware interface is realized by a custom PCB with an ARM Cortex M4 microprocessor programmed in C++, tying in to the vehicle's CAN bus. The interaction between the various components is illustrated in Figure 9.

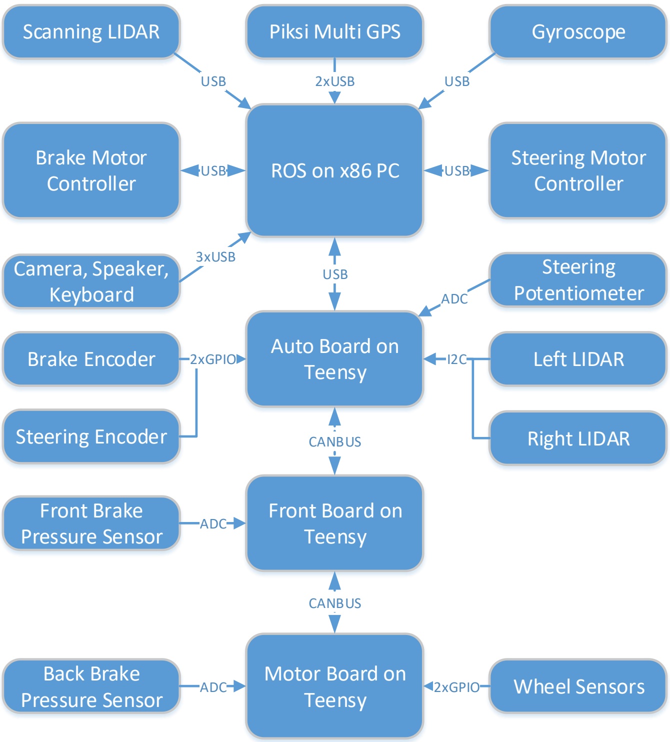
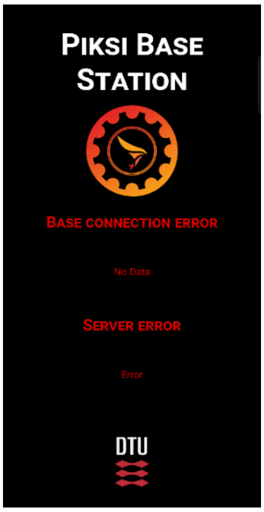


Figure 9: Block diagram of devices in the vehicle relevant for the autonomous system

The Piksi node connects to the Piksi Multi board through its USB serial port and its RS232 RTK port using a USB to RS232 adapter. The libsbp library handles the parsing of the data stream from the Swift Binary Protocol to GPS and IMU data, which are published from callbacks as soon as the data of interest is received. Communication to the Piksi Multi base station is established through a peer-to-peer socket over the internet. The TCP protocol is used to receive data from the base station at 10 Hz and transmit the received bytes to the RTK port of the onboard Piksi Multi with minimal delay.

3.2 Android Base Station App



PIKSI BASE STATION

BASE CONNECTION ERROR

No Data

SERVER ERROR

Error

DTU

Figure 10: The Android Piksi Base Station app

The Piksi base station Android app (Figure 10) is written in Java and connects to the server set up by the Piksi node on the computer. It receives the RTK correction data from the Piksi Multi through an RS232 to USB adapter plugged into the smartphone's USB port. This data is then sent directly to the socket of the computer. The smartphone activity displays the status of the USB and the socket connection.

4 Testing and Results

4.1 GPS performance

The Piksi Multi running firmware 1.0.11 lived up to its specifications. When an RTK fix was achieved, its best accuracy was measured slightly better than the specified 2 cm horizontal and 6 cm vertical. In the test illustrated in Figure 11, the RTK fix mode was maintained throughout the lap while 9 satellites were in view. This resulted in a perfect trace of the courtyard with the worst accuracy peaking at 6.2 cm. No post processing has been done on the data. In the 5 m x 1 m excerpt of Figure 11, the car was running at 4.2 km/h, and the update rate of 10 Hz was maintained resulting in the plotted data.

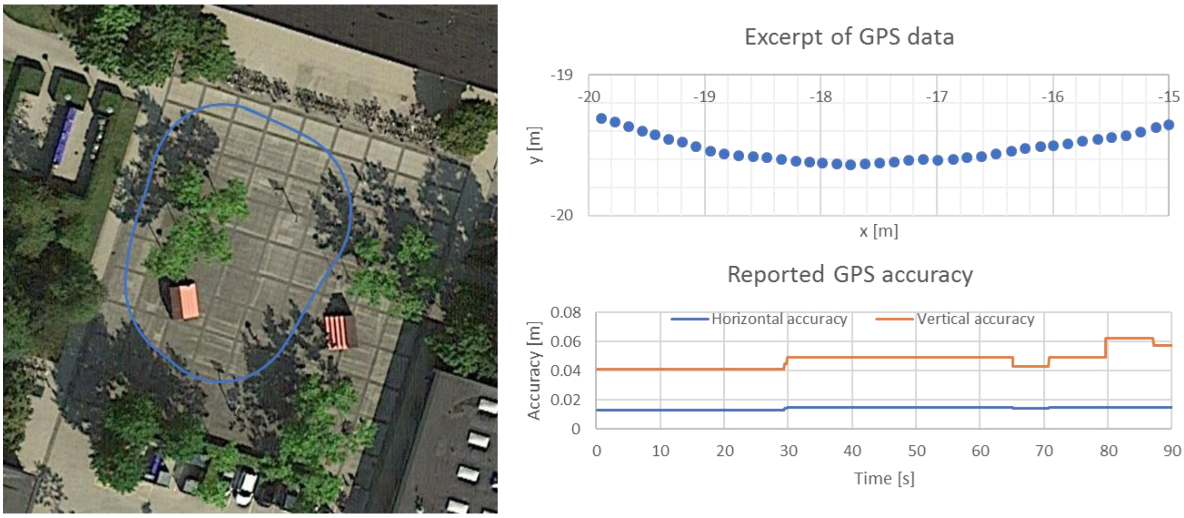


Figure 11: GPS data from a lap on Produktionstorvet in Kongens Lyngby with a 5 m x 1 m excerpt and the reported accuracy over time

In Figure 12, a recording of GPS data from five laps at the competition in London can be seen. The data is colored depending on the status reported by the GPS. The highest accuracy is achieved in RTK Fixed mode, where all data points lie perfectly on the track with a single exception. Shortly after initialization, for a period of 3.7 seconds, the Piksi reported a position 2.9 km away from its actual position and 6500 m below the surface of the earth. All other points are consistent and provide the most accurate data in the system. RTK Float mode still provides reliable data with a reported average accuracy of around 20 cm. In DGPS mode, the nominal accuracy drops to 1-2 m. In the figure, it can be seen that DGPS can be very inaccurate, reporting locations far outside of the track. It falls back to this mode, or the even more inaccurate SPP (single point precision) mode in locations where the GPS reception is poor, such as under the bridge or close to buildings. Interestingly, the location provided by the Android smartphone continues to update even when the car is under the bridge. This is the result of the Android system combining data from the GPS, the IMU and the GSM network to provide the location, something that could also be implemented on the vehicle's system, given that access to this sensor data is already provided and knowledge about the GPS precision is reliable.

At the time of the competition, the firmware of the Piksi Multi did not have the ability to provide heading or bearing information. It has since been updated to provide this functionality. Figure 13 shows bearings calculated post-event based on the latitude and longitude data provided by the GPS using the "Haversine" formula. The figure is used to compare the gyroscope data with bearing data obtained from the GPS data. The gyroscope data is clearly coherent and mostly matches the GPS bearing. After 4 laps, there is a divergence of 13 degrees, which is probably caused by gyroscope drift, since the GPS bearing seems to be coherent with the initial bearing.

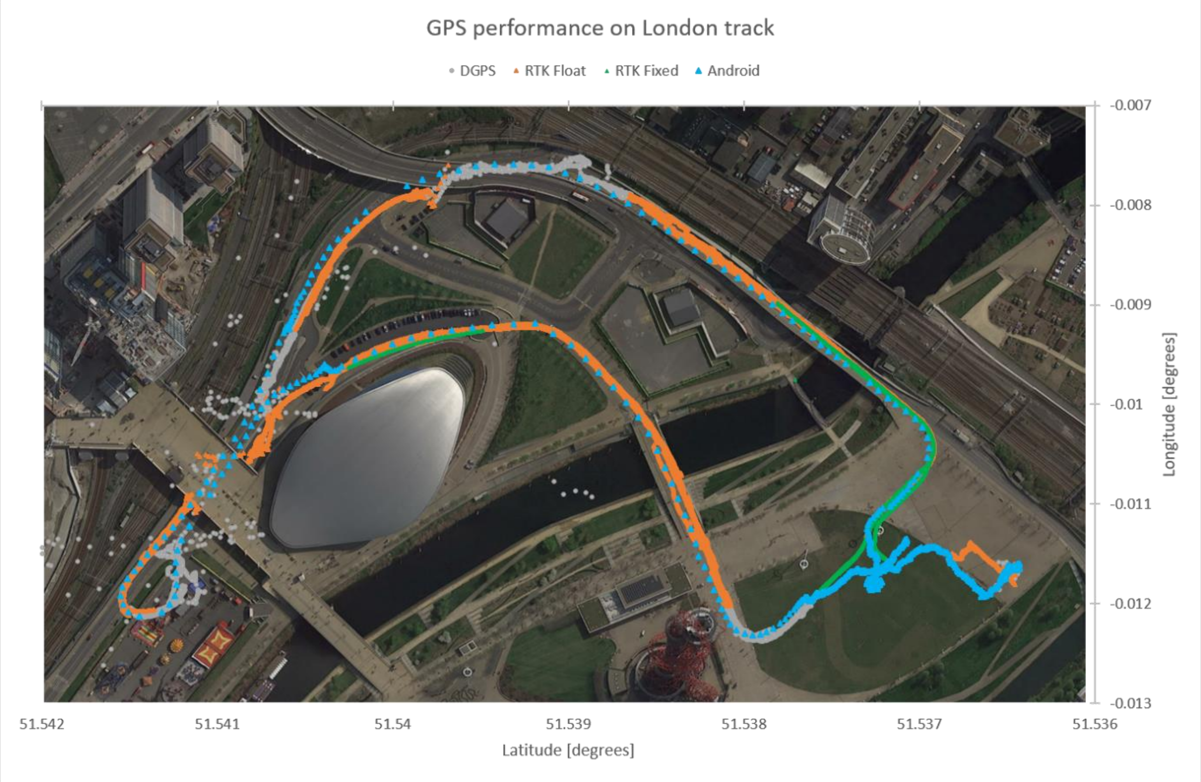


Figure 12: GPS performance of the Piksi Multi and an Android smartphone on the London track

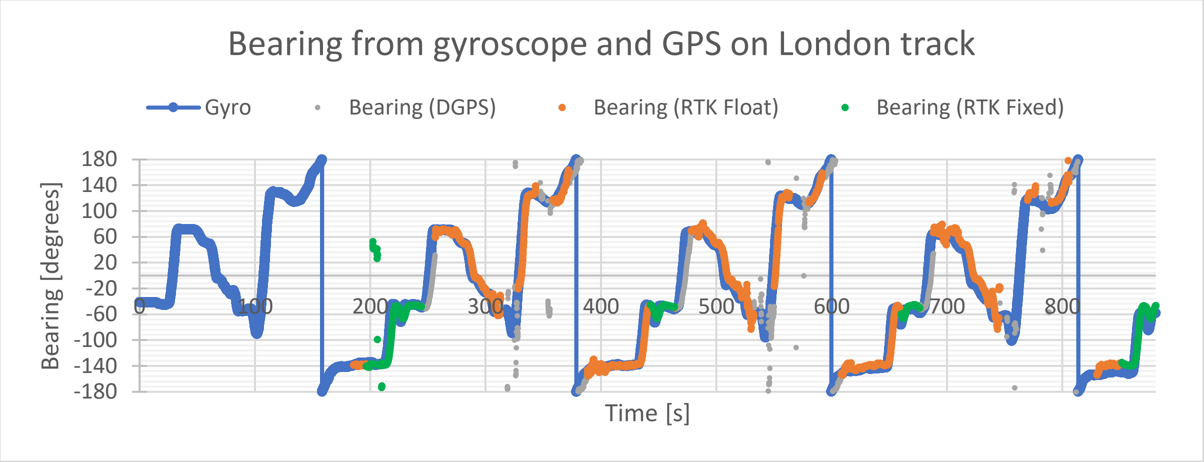


Figure 13: Gyroscope and GPS bearing over 4 laps on the London track

4.2 Power

The power measurements were carried out with an amperemeter connected in series with the voltage supply to the system. The total consumption of the Piksi Multi board when active averaged at a very reasonable 2.7 W.

5 Conclusion

Accuracy and robustness are two key metrics when implementing a sensor for autonomous driving. As seen in the data presented here, the Swift Navigation Piksi Multi navigation solution achieves both. Unless the antenna’s view to the sky is obstructed, the system provides reliable and extremely precise position and heading data. This can be used to accurately determine the location and bearing of the autonomous vehicle in a global frame. In the autonomous systems of the future, a high quality GNSS solution such as the Piksi Multi will be essential for its accurate navigation.

6 References

[1] H.S. Høj, “*Platform Integration for Autonomous Self-Driving Vehicles*”, Master’s thesis, DTU Electrical Engineering - Automation and Control, 2017

[2] A. Bogdanowski, “*Fitting Shell Eco-Marathon car with autonomous functions*”, Synthesis report, DTU Electrical Engineering - Automation and Control, 2017

[3] A. Bellina, “*Fitting Shell Eco-Marathon car with autonomous functions*”, Synthesis report, DTU Electrical Engineering - Automation and Control, 2017