TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Vladimir Šulžik 204701IASM

Optimization of Sensory Deprived SLAM for Robotic Miner

Master's thesis

Supervisor: Asko Ristolainen
PhD
Co-supervisor: Roza Gkliva
PhD

Tallinn 2023

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Vladimir Šulžik 204701IASM

Sensoorsete häiretega SLAM-i optimeerimine kaevandusrobotil

Magistritöö

Juhendaja: Asko Ristolainen
PhD
Kaasjuhendaja: Roza Gkliva
PhD

Tallinn 2023

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Vladimir Šulžik

03.05.23

Abstract

Robotic mining is an innovation in the field of mining. For navigation in tunnels and mines, lidar and visual sensor data is not always available or trustworthy due to poor conditions. This thesis describes the creation of the SLAM system that uses tactile sensors and is compatible with ROS2.

This thesis presents various configurations of such sensors. It also describes an evaluation method for finding the best configuration for a miner's robot. Moreover, an analysis of which configurations are best suited for certain situations is presented. The results of this analysis were applied to a real robot. This thesis proves the possibility of using tactile sensors on a mining robot.

This thesis is written in English and is 42 pages long, including 5 chapters, 28 figures and 6 tables.

Annotatsioon

Sensoorsete häiretega SLAM-i optimeerimine kaevandusrobotil

Kaevandusrobotid on uudsed kaevandamise valdkonnas. Tunnelites ja kaevandustes navigeerimiseks ei saa robotid alati kasutada lidarit ega visuaalseid andureid kehva nähtavuse tõttu. Selles lõputöös kirjeldatakse taktiilsete andurite abil töötavat ROS2 ühilduvat SLAM süsteemi arendust.

See lõputöö keskendub taktiilsete andurite erinevate konfiguratsioonide uurimisele. Töö käigus pakuti hindamismeetod kaevandusroboti jaoks parima taktiilsete sensorite konfiguratsiooni leidmiseks. See hindamine toimub SLAM-i loodud teekonna ja odomeetria tee võrdlemisel tegeliku roboti poolt läbitud teega. Lisaks esitatakse analüüs selle kohta, millised konfiguratsioonid sobivad teatud olukordades kõige paremini ja parima konfiguratsiooni võimaluste hindamine pikemate marsruutidega töötamiseks.

Selle analüüsi tulemusi rakendati ka reaalsel robotil, et veenduda, kas SLAM-i on võimalik pärismaailmas kasutada. See lõputöö näitab puutetundlike andurite kasutamise võimalust kaevandusrobotil.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 42 leheküljel, 5 peatükki, 28 joonist, 6 tabelit.

List of abbreviations and terms

SLAM	Simultaneous Localization and Mapping
ROS	Robot Operating System
RMSE	Root Mean Square Error
Lidar	Light detection and ranging

Table of contents

List of figures

List of tables

1 Introduction

The thesis is a part of ROBOMINERS (EU Horizon 2020 (grant agreement n°820971) project. ROBOMINERS is working on the development of a mining robot. This robot should work on small and hard-to-reach mineral deposits with the opportunity to mine in various conditions both on the ground and underground. The primary objective of ROBOMINERS is to develop an autonomous robot whose autonomy will depend on its ability to collect information about the environment and its current state. For these purposes, the project is investigating methods of perception and navigation for these challenging environments. As localization and mapping are important for the robot to know where it is, one of the methods that are investigated is SLAM (Simultaneous Localization and Mapping).

SLAM is a problem of simultaneous localization of a robot and mapping the environment. SLAM is a crucial method for achieving autonomous navigation of mobile robots, particularly in environments where high levels of accuracy are required. However, it becomes more challenging in complicated, unknown and uneven underground environments where robot's odometry is inaccurate due to slippery, uneven surfaces and varying soils. Whereas ground robots traditionally use GPS for their navigation tasks, its utility in mines is limited due to its narrow tunnels and lack of the signal. Moreover, muddy and dusty environments worsen the operation conditions of the visual sensors, making them less effective in such harsh conditions. Additionally, environments with repetitive appearances, such as tunnels and mines, can lead to false loop closures, which can significantly distort the entire map [1]. To tackle these problems, the thesis focuses on investigating the usage of tactile sensing with whiskers of the robot to navigate and create an accurate map of its surroundings using SLAM.

In the previous work at the Centre for Biorobotics at TalTech, tactile SLAM was developed in ROS 1 along with 'whisker'-sensors [2]. Since the robot uses ROS2 [3] as its main framework, this thesis aims to implement a new SLAM system compatible with ROS2. In this way, it is possible to avoid the use of additional software for the integration

of ROS1 and ROS2, such as the ROS bridge, as well as ensure the ideal performance of the robot without burdens on additional functions.

A big problem is also the number of used whiskers. If it is too small, then SLAM will not receive enough data about the robot environment. On the other hand, if the number of whiskers is too large, there might be computational and communications restrictions, since a lot of data shall be processed. This is of particular importance when running SLAM for a long time, as the whiskers data point cloud can become too large and require a lot of resources to process. Moreover, the robot must work in narrow and irregular mines, it should not be very bulky. Otherwise, there is a chance that the robot will get stuck somewhere or not get through in an impassable place. Thus, it is not possible to have a high number of whisker sensors all around the robot, minimum possible configuration should be found. For this purpose, along with the software development, the thesis sets out to evaluate how the different configurations of whiskers affect the navigation software performance and find the optimal number and placement of whiskers on the robot.

The contributions of this thesis are the following:

- Development of a software for whisker-SLAM in ROS2
- Development of evaluation framework in simulated environment.
- Evaluation of whiskers configurations performance for the SLAM system
- Evaluation of the SLAM system with a physical prototype

2 Background

The following section introduces the background of the ROBOMINERS project. It also includes a review of the latest SLAM frameworks in the literature, with a particular emphasis on those that are relevant to the thesis topic.

2.1 ROBOMINERS project

The ROBOMINERS' project is trying to find a new innovative approach to creating a mining ecosystem with novel ideas from other sectors, in particular with the inclusion of disruptive concepts from robotics. Its goal is to develop a bio-inspired, modular, and reconfigurable robot that can explore and develop mines. The use of the robot miner will especially be relevant for mineral deposits that are small or difficult to access. This covers both abandoned, nowadays flooded mines, that are not accessible anymore for conventional mining techniques, or places that have formerly been explored but whose exploitation was considered as uneconomic due to the small size of the deposits or the difficulty to access them. [4]

The ROBOMINERS project develops various scaled prototypes that address different aspects of the project. In this work, ROBOMINERS prototype RM3 (Figure 1) was used that was developed in TalTech for investigation of locomotion and sensing in mining environments. The robot moves with the help of Archimedean screws. The Archimedean screws, which are essentially helical threads, allow the robot to move smoothly over rough terrain and surfaces, including steep inclines and rocky surfaces. This makes the robot particularly useful in mining operations, where it can navigate difficult areas and access hard-to-reach locations for tasks such as drilling, excavation, and inspection.

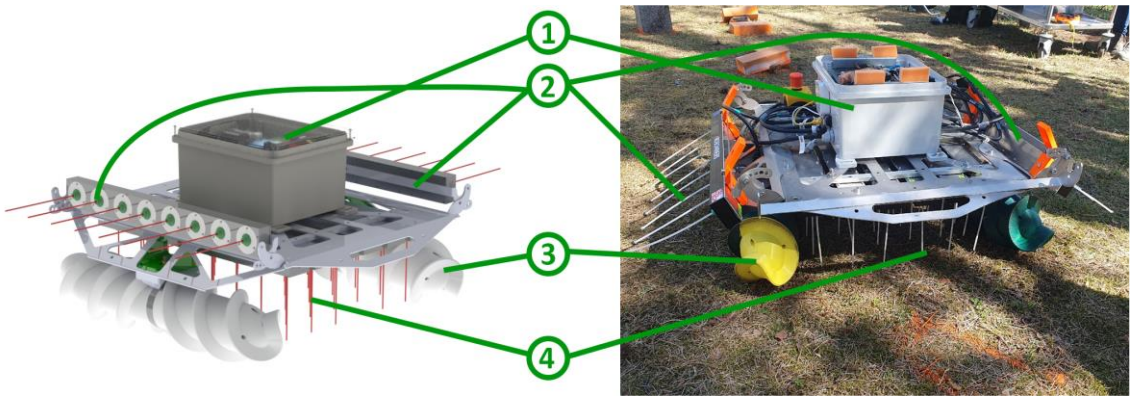

Figure 1. Robot with whiskers. CAD model is on the left picture and physical prototype is on the right. 1: control unit, 2: side whiskers, 3: Archimedean screws, 4: bottom whiskers.

2.1.1 Tactile sensors

The robot uses tactile sensors or whiskers for exploring the environment as lidar. Their concept was based on the flow measurement device called hydromast [5]. The whisker sensor unit consists of a 3D Hall sensor and a mast with a magnet attached to the end close to the sensor (Figure 2). The mast is held in place by a compliant joint made of silicone rubber. Any deflection of the mast leads to a deflection of the magnet above the sensor, which senses the strength of the magnetic field in 3 dimensions. The housing is partially 3D-printed and partially cast from silicone. The silicone provides the masts the required ability to deflect and bounce back to the resting position.

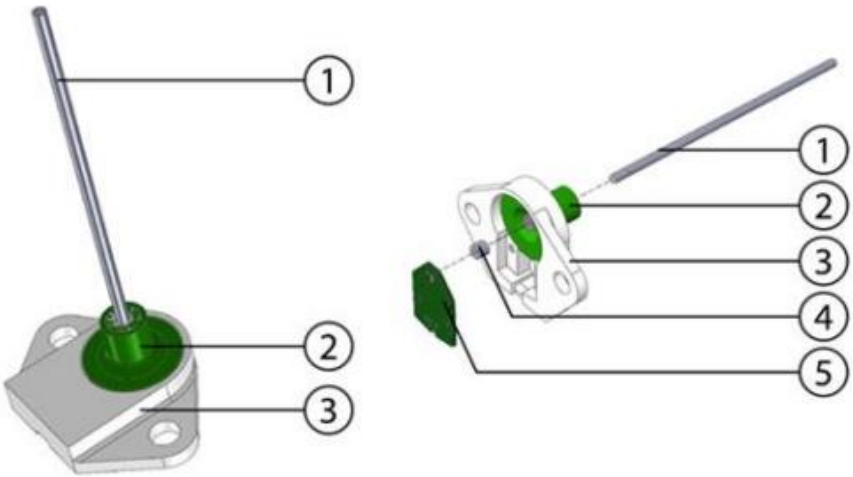

Figure 2. Whisker unit physical design. 1: mast (polyamide), 2: compliant joint (silicone rubber, Elite Double 22, Zhermack), 3: housing (photopolymer), 4: neodymium magnet, 5: PCB holding the sensor chip TLV493D-A1B6 by Infineon Technologies.

Each whisker is mounted individually on a special array of eight each. There are eight arrays located on the robot platform resulting in a total of 64 whiskers. Six of these arrays are located on its bottom (Figure 1), which enables it to have a better understanding of the environment by detecting surface irregularities and objects that may occur under the robot. This is particularly useful to explore the changes in an environment below the robot, and the whiskers provide the robot with a means to sense and respond to these changes with greater accuracy. Two arrays are mounted on the robot's sides.

2.2 SLAM

Before describing the SLAM, some definitions should be provided for a better understanding the problem:

- Pose – the position and the orientation of an object in space.
- Scan frame - set of poses for one moment of scanning.
- Loop closure - the problem of determining whether the robot has visited a given area before or not.
- Odometry - estimation of the robot's position and orientation obtained using data of the motion of the robot.

SLAM is a problem, in which mobile autonomous robots build a map in an unknown space or update a map in a previously known space while simultaneously monitoring the current location and the distance travelled. The SLAM problem appears when the robot does not have access to both an initial map of the area and its own coordinates. The urgency of the problem stems from the fact that the maps commonly used for navigating agents basically reflect the view of the space fixed at the time of their construction, and it is not at all necessary that the view of the space will be the same at the time the maps are used. SLAM is more complicated than localization because the map is unknown and must be estimated along the way [6]. Popular methods for the approximate solution of the SLAM problem include the filter-based SLAM and the graph optimization-based SLAM.

Filtering approaches are some of the first and simplest SLAM approaches. Filter-based SLAM is usually a feature-based algorithm. It determines the position of the robot and

updates the map, estimating the probability of matching sensor data when available. Filtering approaches work with the state of the current position of the robot and the map in real time [7]. In such approaches, only the latest data is used, after finishing work with a particular frame, all previous robot poses are discarded. These approaches typically include Extended Kalman filter, Unscented Kalman Filter, and particle filter methods [8].

The main idea of the SLAM method based on graph optimization is to store all previously observed information about the environment for use in estimating the robot's trajectory and creating a terrain map. This method converts map elements into pose constraints and then evaluates their sequence. As the robot moves, poses are constantly being adjusted to satisfy the constraints. One of the important aspects of graph optimization is the problem of loop closure. Its task is to determine whether the robot is in the already visited region or not. Thus, the robot understands the topology of the environment better [9].

In conclusion, the filter-based SLAM does not consider map records up to the previous moment. Thus, the cumulative error caused by sensor noise uncertainty will eventually lead to map inconsistency as the travelled distance increases. Moreover, loop closure concept is not in used in some filter-based SLAM or its capabilities are restricted because these approaches use only the latest data. In this case if at some stage the SLAM makes a mistake, graph optimization-based SLAM has more probability to correct the results when the robot returns to a previously visited region. Considering these facts, it was decided to use the graph optimization-based SLAM. This method will help to get the best results on terrains where there are no clear landmarks.

2.2.1 SLAM with underground robots

This section presents an analysis of previous work on the topic of underground robots using SLAM. Most of the SLAM methods in underground settings use lidar (light detection and ranging) sensors or distance sensors. This approach proves to be highly efficient in scenarios with spacious environments that possess sufficiently prominent landmarks to ensure that they are not obscured by the background when employing SLAM. However, in narrow tunnels without significant landmarks, where the ROBOMINERS robot must work, a different concept is needed, as visual sensors don't always work properly. Therefore, the decision was made to analyze whiskers in this environment, as in underground environments some difficult conditions such as dust, smoke, mud etc can interfere with visual sensors. In contrast to infrared or sonar sensors

that only capture distance information from a single point, whisker sensors retrieve details about the surface's local orientation through individual contacts [10]. Three interesting SLAM solutions for underground robots can be found below.

The first solution is reported in [1]. The author describes a lidar-based SLAM on a multi-robot system. The paper presents a back-end system that utilizes an Incremental Consistent Measurement Set Maximization method to reject any outlying loop closures.

The second solution is a 2D LIDAR SLAM [11] based on point-line feature extraction using a curvature detector. While this approach is suitable for straight tunnels, since it is designed for 2D mapping and uses a filter-based technique rather than graph optimization, it is not well suited for a specific mining robot that is designed to operate in narrow and uneven tunnels.

The last solution is described in this paper [12]. The author presents a SLAM approach using a Rao-Blackwellized Particle Filter (RBPF) to process the data from a Ground Penetrating Radar (GPR) system. RBPF generates possible trajectories, each of which is associated with a unique 3D occupancy grid map.

Lidar, visual and other sensors solutions have been made for large tunnels and mines [13] [14]. But in small tunnels, where the performance of sensors can be difficult or completely limited due to difficult conditions, such as dust, smoke or mud a different approach is needed. The GPR described in the last solution is also not suitable, since it is unknown whether there will be something in the ground that the radar can respond to or not. Given the limitations of the systems described above, the ROBOMINERS project has decided to investigate the use of whisker sensors for SLAM for its target environments.

2.2.2 Open-sourced SLAM frameworks

This section provides an overview of various open-sourced SLAM software with their benefits and disadvantages. The first approach SLAM Toolbox [15] offers graph optimization-based mapping, including both synchronous and asynchronous options. One of the main benefits is that SLAM Toolbox has been integrated into ROS2 navigation stack [16], replacing GMapping. However, SLAM Toolbox is more designed for 2D maps. It does not have a detailed image of three-dimensional objects. Moreover, it is designed to work with lidar and after it was modified to work with tactile sensing, its

results were not so satisfactory. Despite that, continued work with SLAM Toolbox may be a further field of study.

The second approach is the RTAB-Map (Real-Time Appearance Based Mapping) [17]. It is an appearance-based SLAM. RTAB supports both lidar and visual sensors, such as cameras. The system is designed to provide users with a variety of 3D and 2D mapping solutions. Additionally, using a bag-of-words approach. This approach extracts visual data from an RGB image and transforms it into visual words, forming a so-called visual dictionary. If the images are generic enough, the same vocabulary can be used across different environments with good performance. Thus, the RTAB-Map can work with large loop closures even when there is a significant difference between the estimated and true positions. Although the software performs well in ROS, the author met several issues with installing and did not manage to run this package.

The ORB-SLAM2 [18] is an approach developed for various types of cameras, including monocular, stereo, and RGB-D cameras. It can generate a sparse 3D reconstruction and compute the camera's trajectory. However, it is only compatible with ROS, which is not very suitable for this project, where the robot’s software is written in ROS2. Furthermore, it is more difficult to adapt a camera-based system like ORB-SLAM2 for whiskers than a system developed for lidar.

The last approach, Google's Cartographer [19] is specifically designed for indoor mapping. Cartographer does not use any filtering and instead relies on pose optimization. However, since the software matches data with recent submaps, it can accumulate the estimated position error. Moreover, it creates only 2D maps, which is not suitable for tunnel exploration.

2.2.3 SLAM with robots using whisker sensors

This section provides examples of robots with whisker sensors using SLAM. The first solution is reported by [20]. There authors developed a robotic arm with an active tactile whisker array along with the RatSLAM algorithm. Since in this solution a robotic arm was used as a moving mechanism, which moved along a surface with obstacles, it remains unclear how a self-moving robot will behave with such a system.

The second paper [21] introduces an approach combining the use of both vision and tactile data for SLAM purposes. As a benefit, this solution can be used in scenarios when one of the sensors cannot properly identify false loop closures. However, this paper does not examine the consequences of situations where one of the sensors, due to difficult conditions, cannot receive any data. For example, in a mine, where there is not enough light for the visual sensor, its effectiveness will be so low that there will be a need to rely only on tactile sensors. Despite that, this approach can be used on a miner robot, for example, if the visual sensor is replaced with a lidar.

The work in [22] presents a robot using tactile sensors in front of it for orientation in the environment. This solution is not very suitable for working in mines since it uses the filter-based SLAM. Additionally, robot in this work is in the environment made of straight edges, which is not very suitable for a mining robot. Moreover, the robot uses prior knowledge about the world's structure, which cannot be provided in the mine.

Although whiskers and tactile sensors have already been used on robots for SLAM navigation, most of these robots were small-scale prototypes and were used on flat and smooth surfaces. Moreover, fixed configuration was used on these robots and no analysis of the efficacy of different whisker configurations was provided.

3 Methodology

The whisker placement problem pertains to determining the optimal number and position of whiskers for the purpose of implementing SLAM on a miner robot. The goal is to improve the accuracy of the robot's path estimation and ultimately, enhance its navigation capabilities in challenging underground environments. For that, a set of experiments were planned to test various configurations of whiskers and robot trajectories to evaluate the impact of different combinations on SLAM accuracy. The findings from this study can provide valuable insights into the optimal whisker placement for SLAM in miner robots.

To estimate the accuracy of SLAM, the root mean square error (RMSE) between the robot's path estimated by SLAM and the true path was calculated, as well as the error between the robot's odometry and the true path. The comparison of these errors helped to determine whether there was an impact from SLAM and how significant it was in terms of the robot's overall performance. Further, SLAM path error and odometry path error were compared with each other to determine whether there was an impact from SLAM and how significant it was in terms of the robot's overall performance.

The experimental phase of this study consisted of 16 different configurations of whiskers and 4 different trajectories. The experiments were divided into two parts. In the first part, each whisker configuration was tested once on each trajectory, resulting in a total of 16 experiments for each trajectory. Based on the initial test results, the top 4 configurations were selected and used for additional experiments. In this phase each trajectory was repeated in 5 experiments for each of the 4 chosen configurations. This experimental process allowed for a comprehensive analysis of the optimal whisker configuration for use in SLAM on miner robots.

3.1 Simulator

In order to facilitate experimentation and optimize the process, a previously developed simulator in Gazebo environment was used in this thesis. The simulator allows for the manipulation of a variety of variables including terrain, trajectory, the number and position of whiskers. By utilizing the simulator for the majority of experiments, especially at the early stages of this investigation, the process was made more efficient and

streamlined, as it enabled the ability to quickly and easily modify robot parameters and run multiple trials.

3.2 Experiments

3.2.1 Tracks

For conducting the experiments, a total of four tracks were designed, namely linear, circular, U-shape, and 8-shape. In each experiment, the robot is required to complete three loops on the chosen track. During the movement of the robot, the whiskers register various obstacles on the track, as well as ground irregularities. This data is then used in the developed SLAM. Each track is simulated once, then the simulation data is recorded and played back for each whisker configuration. Thus, each experiment has the same input data. This contributes to the accuracy of the results since deviations cannot be created due to the fact that one whisker configuration has more favorable input data than another.

- Line track (Figure 3) – in this scenario, the robot follows a linear trajectory consisting of three stages. Initially, it moves in a forward direction for a certain distance from its initial position. Subsequently, the robot proceeds in reverse, returning to its original position and then moving back for a certain distance. Finally, the robot completes its trajectory by returning to its original position, having completed one loop. The robot is shown in the initial position in the figures.

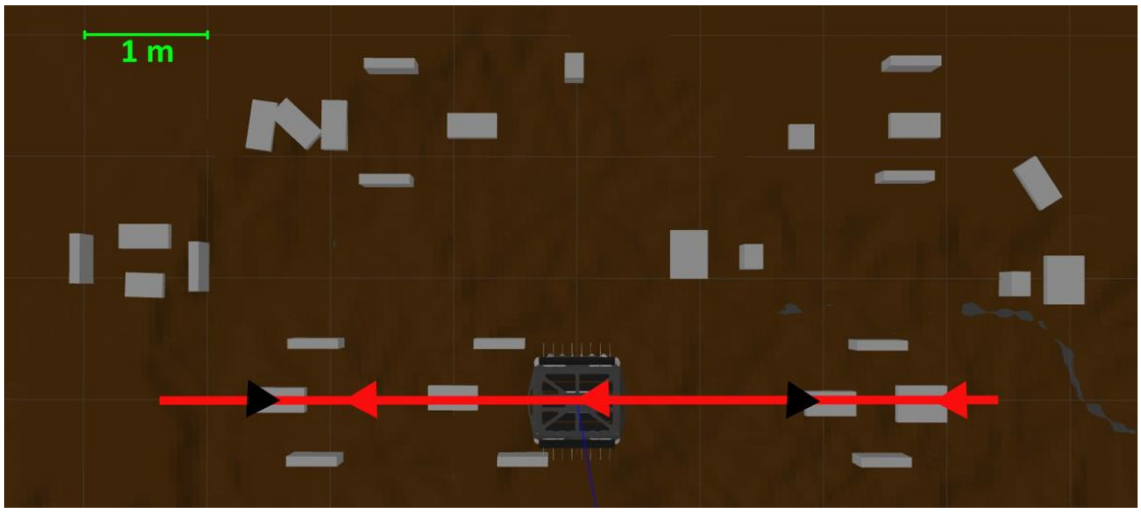

Figure 3. The line track in the simulator. Red arrows show forward moving, black arrows show reversed moving.

- Circle track (Figure 4) - in this scenario, the robot navigates along a circular trajectory. At the outset, it moves in a forward direction, encountering and surpassing two obstacles. At that point, the robot takes the first turn to the right, where it passes yet another obstacle before making another turn. Then it passes past several additional obstacles before making its third right-hand turn. Eventually, the robot passes the last obstacle and makes its last turn, ultimately returning to its initial position and completing one loop.

Figure 4. The circle track in simulator.

- U-shape track (Figure 5) – this scenario resembles the circular track described earlier, with one key difference: instead of making the third right turn, the robot retraces its path in reverse, effectively moving backwards along the same

trajectory it has already travelled. Upon reaching its original starting position, the robot completes a full loop.

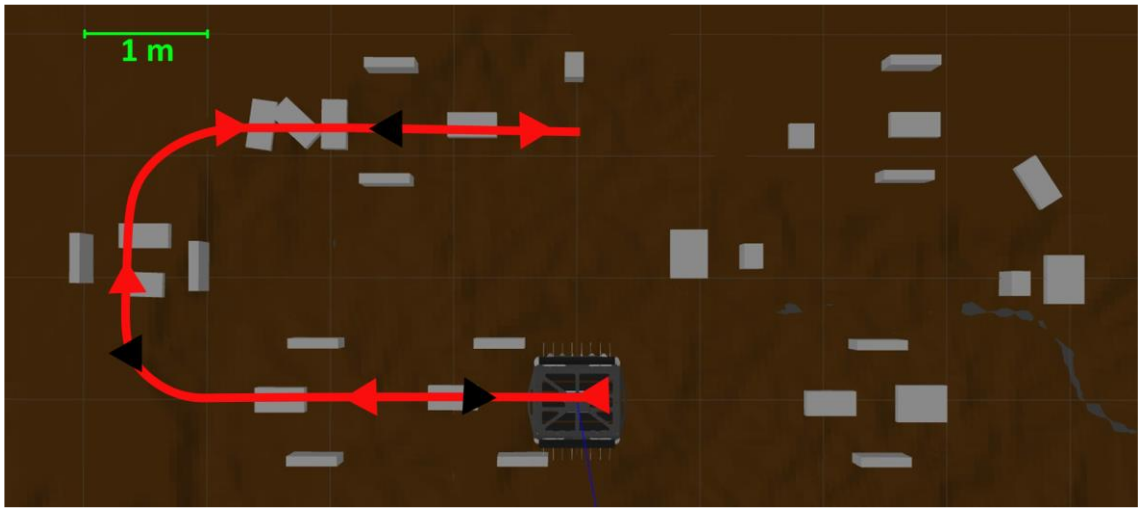

Figure 5. The U-shape track in the simulator. Red arrows show forward moving, black arrows show reversed moving.

- 8-shape track (Figure 6) – this scenario also resembles the circular track. However, in this case, the robot goes through the first three turns, but on the fourth turn, it turns left instead of right to make another small circle. After completing this smaller circle, the robot returns to its original starting position, having completed a full loop.

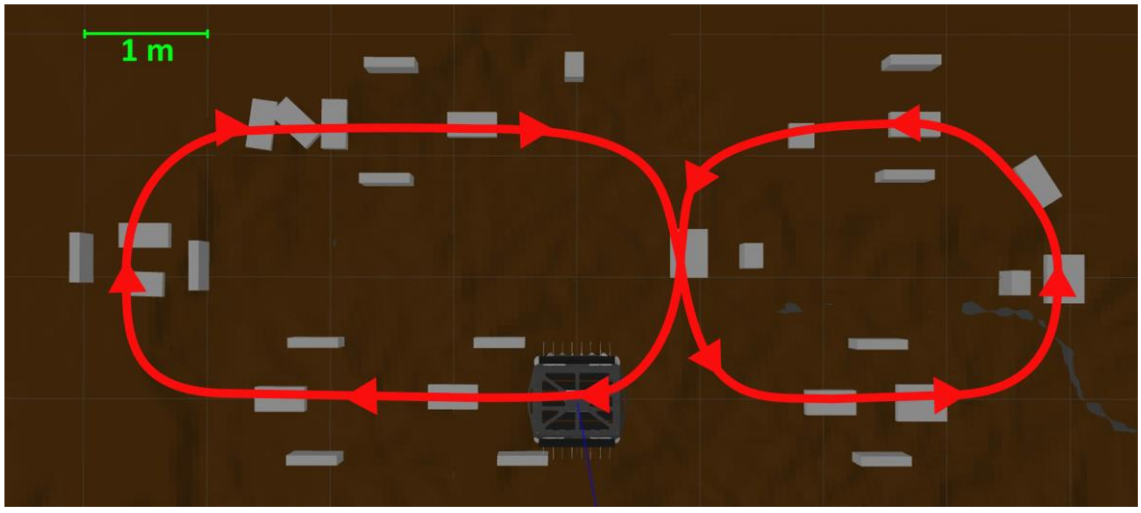

Figure 6. The 8-shape track in simulator.

On these tracks there 4 types of boxes (Figure 7) are used on the ground and for sensing with bottom whiskers. Additionally, there are 2 types of walls for the side whiskers:

1. Box with dimensions x = 0.2m, y = 0.2m and height 0.08m.

2. Box with dimensions x = 0.4m, y = 0.2m and height 0.1m.

3. Box with dimensions x = 0.3m, y = 0.2m and height 0.1m.

4. Box with dimensions x = 0.3m, y = 0.4m and height 0.08m.

5. Wall with dimensions x = 0.4m, y = 0.08m and height = 0.3m.

6. Wall with dimensions x = 0.15m, y = 0.2m and height = 0.4m.

Figure 7. Objects in simulator environment.

3.2.2 Whisker configuration

There are 64 whiskers on the robot that are divided into 8 arrays with 8 whiskers in each. 6 arrays are located under the robot and two on the sides. This configuration of whiskers (Figure 8) on the robot is the maximum possible arrangement. However, to explore the possibility of using configurations with fewer whiskers, various configurations were tested by turning on and off the whiskers in different combinations. Through these experiments, it was analysed whether the performance of the robot can be maintained with a smaller number of whiskers and a different arrangement of the existing whiskers.

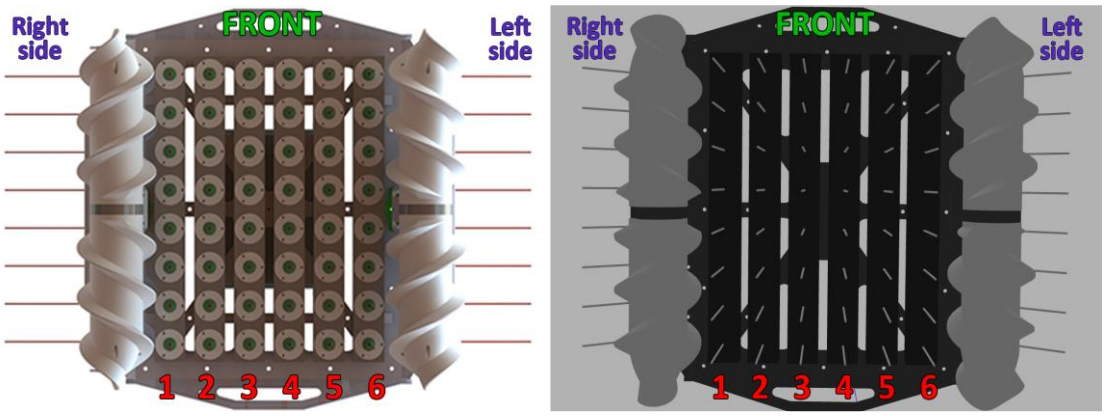

Figure 8. Whisker arrangement view from the bottom of robot. CAD model is on the left picture and simulator prototype is on the right.

During the experiments 16 different whisker configurations were evaluated. Eight of them had side whiskers and eight configurations did not have:

- All bottom whiskers configuration (N1) – it is the standard configuration, where all bottom whiskers are enabled.
- 1, 3, 4, 6 arrays configuration (N2) – there are 4 enabled bottom whisker arrays.
- 2, 5 arrays configuration (N3) – in this configuration only 2 bottom arrays are enabled.

These 3 configurations are shown in Figure 9.

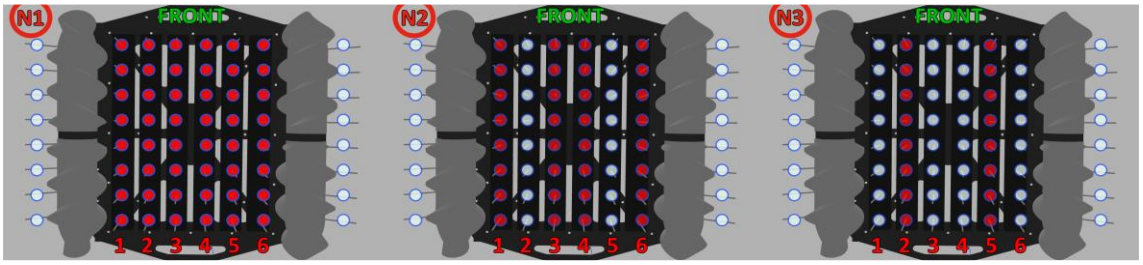

Figure 9. Whisker configurations N1, N2 and N3.

- Configurations with disabled even whiskers (N4-N6) - these configurations look like the previous ones, with the only difference that every second whisker in the array is deactivated (Figure 10). This is made in order to investigate the impact of having a smaller number of whiskers in an array on the performance of the robot's SLAM.

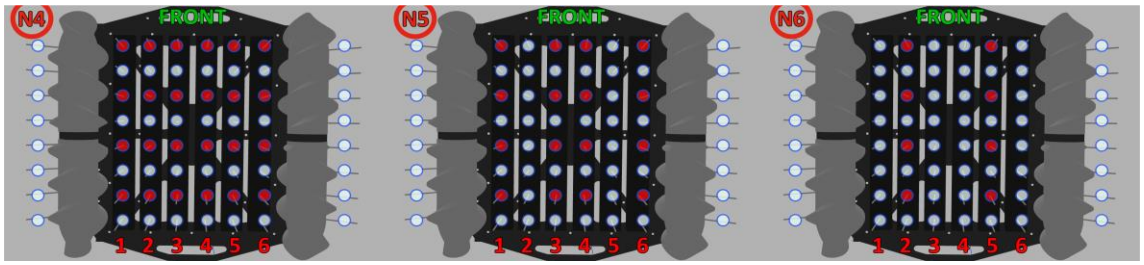

Figure 10. Configurations with disabled even whiskers.

- Chessboard configurations (Figure 11) - the last two configurations resemble a chessboard pattern, where the state of each vertically or horizontally adjacent whisker is inverted. The first configuration has the top right whisker enabled (N7), while the second configuration has the top left whisker enabled (N8).

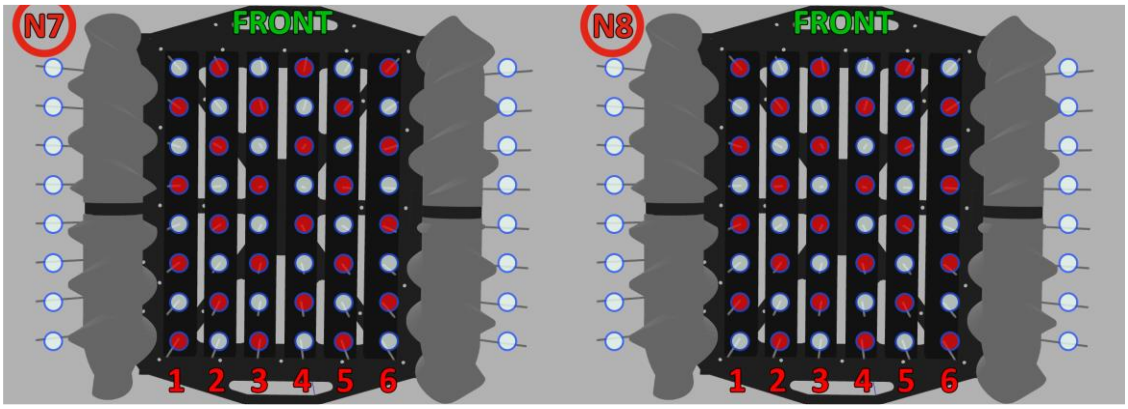

Figure 11. Chessboard whisker configurations.

- Configuration with side whiskers (S1-S8) - The eight mentioned earlier configurations were also tested with enabled side whiskers (Figure 12).

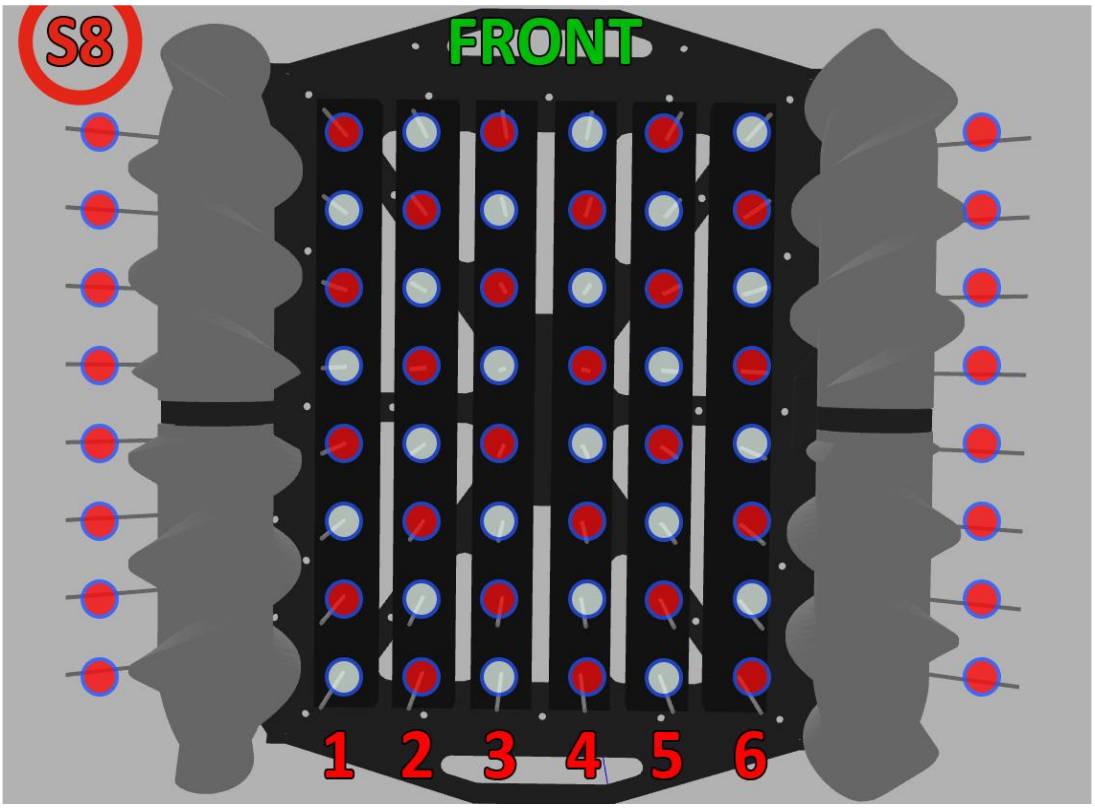

Figure 12. Side whiskers with chessboard configuration.

3.2.3 Experiments with the physical robot

The experiments were conducted using the physical robot as well. For this purpose, the environment that was used in the simulator was approximately recreated in real life (Figure 13). On an area measuring 11.80m by 7.20m, bricks with length = 0.25m, width = 0.12m and height = 0.065m were laid out for the bottom whiskers. Four foam walls with length = 1.12m (one of these walls had length = 1.04m), width = 0.10m and height = 0.4m and four large blocks with length = 0.195m, width = 0.185m and height = 0.485m were placed there for the side whiskers. All 4 tracks (circle, line, U-shape, 8-shape) were repeated in real environment.

The ground truth position of the robot was obtained by using a vision-based tracking system developed by Centre for Biorobotics. An ArUco tag was mounted on the robot and two tags on the field. Footage from a Realsense D455 camera overlooking the entire field was processed in ROS2 to obtain the robot's pose. For each experiment the robot's initial position is considered the world's inertial frame.

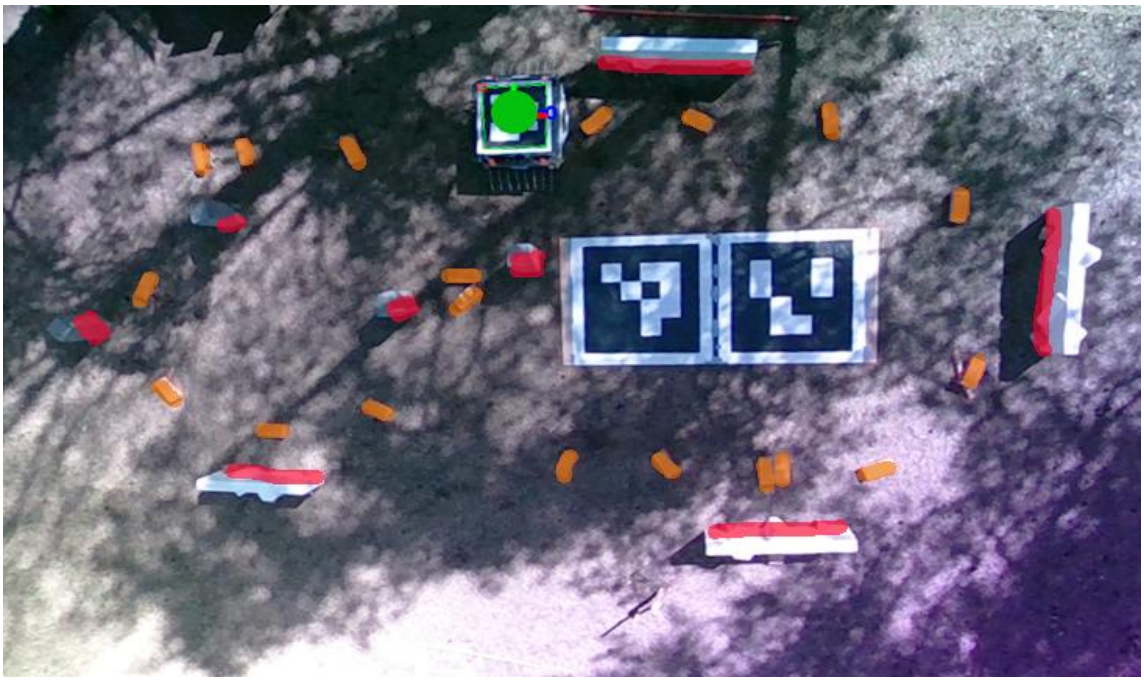

Figure 13. Top view of experiment in real environment. Orange – bricks, red – foam walls and large blocks, green starting position.

3.3 Error calculations

During conducting all the experiments, the points that constituted the odometry path, the ground truth path, and the path generated by SLAM were obtained. The ground truth path was obtained in simulator environment by using robot position data. In the real environment it was obtained by using a camera. The odometry data of the robot was obtained using a dynamic model of the robot's locomotion system developed by Centre for Biorobotics [23]. The model uses IMU data and the angular velocities of the screw actuators to output an odometry estimation. For each point from the odometry and SLAM paths, the root mean square errors were calculated in comparison to the corresponding points in time from the ground truth path. This approach allows to determine the accuracy of SLAM in relation to the ground truth path of the robot and to assess its ability to improve the accuracy of the odometry path.

The RMSE is calculated by finding the difference between x and y coordinates of two points. Afterwards, the difference between x coordinates and y coordinates is squared and summed. This is done for each point of the path. Then the arithmetic mean for these results is found and the root is calculated from it. The RMSE is calculated with this equation:

$$RMSE = \sqrt{\frac{\sum_{i=0}^{n}(x_1 - x_2)^2 + (y_1 - y_2)^2}{n}}, \quad (1)$$

where x_1 and y_1 is coordinates of the positions in the ground truth path, x_2 and y_2 is coordinates of the positions in the SLAM or the odometry paths, and n is number of positions in path.

Such calculations are carried out for each whisker configuration. After, the results are compared in order to determine which configurations performed better in different conditions. This comprehensive analysis provides valuable insights into the optimal configuration of whiskers for SLAM implementation miner robots. Thus, it will be possible to find the most optimal variant of the number and position of the whiskers for the ROBOMINERS robot.

Moreover, in order to analyze which path best reproduces the shape of the ground truth path, each path was centralized in the global frame relative to its centers so that the centers of all the paths coincided at a single point (Figure 14). After the paths were transformed, the deviations of both the odometry and SLAM paths from the ground truth path were calculated (1) to analyze how SLAM helps to preserve the true shape of the robot's path and consequently create an accurate map of the observed world. This approach was taken due to the fact that SLAM operates in its own frame, which may not be aligned with the global frame. As a result, it can accurately capture the shape of the explored environment but may be shifted relative to the ground truth path. By using this approach, the accuracy of recreating the form of the ground truth path can be evaluated relative to the path's own frame. This approach is used as an additional parameter to analyze which configuration is better in case several configurations will show the same result.

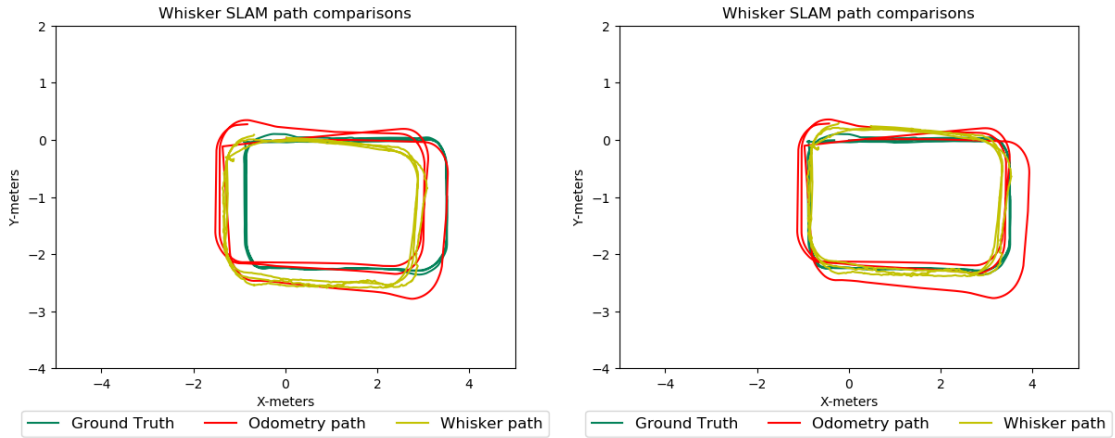

Figure 14. Comparing general paths (left) and centralized paths (right).

3.4 Whisker-SLAM

During the work on this thesis, the ROS2 compatible SLAM system was developed for whisker sensors. For this purpose, an open-source package lidar SLAM [24] was used. This package is graph optimization-based SLAM algorithm for 3D lidar using NDT/GICP registration and pose-optimization. Since it is designed for the 3D lidar sensor, it is convenient to set it up for use with whiskers.

This package has been modified to receive whiskers data as input, regardless of whether this data comes from a real robot or a simulator. The package creates the point cloud of the environment from lidar data. Additional changes were made to reformat the whisker tips' data to the point cloud. Knowing the length of the whisker mast, and the deviations in azimuth and inclination, the software finds the xyz coordinates in robot frame for every whisker tip. After that package reformats xyz coordinates into the point cloud data type and then use this point cloud in the SLAM. This package has been also modified to work with smaller amount of data. If previously the amount of data in one frame was about 10 000 poses, now it is maximum 64. By reducing the input data size, the frequency of software has been increased from 0.67 Hz to 100 Hz. Moreover, reducing the input data size allows the SLAM package to check for loop closure in the shorter distances. The minimum and maximum distance between frames to find the loop close is changed to 0.5 and 5 meters respectively. The minimum distance is the travelled distance from the frame, after which SLAM begins to combine this frame with other frames, while the maximum travelled distance indicates when SLAM stops using this frame.

In addition to SLAM itself, additional software for SLAM work visualization, testing and error evaluation has been developed. This software takes robot odometry as input of its hypothetical position on the map. Then, by analysing the environment and the received whisker data, it decides whether this position is true or not. It is important to consider that the odometry data, which SLAM uses, can also experience significant accuracy degradation with each loop. Thus, an increase in odometry path error can lead to an increase in SLAM path error. Additionally, software for converting the robot's true position and odometry coordinates into paths of ground truth and odometry has been designed. It helps to collect paths data and visualize them in real time.

Moreover, for testing and error evaluation software has been developed that calculates the error and the saves the paths images. Each path consists of travelled positions. This software takes SLAM, odometry and ground truth paths and divides them into travelled positions. The software takes the SLAM path and searches positions with similar time stamps in odometry and ground truth path for synchronization. After that the software calculates the RMSE error between the SLAM path positions and the ground truth path positions as well as between the odometry path positions and the ground truth path positions. Communication between all nodes in the software is shown in Figure 15.

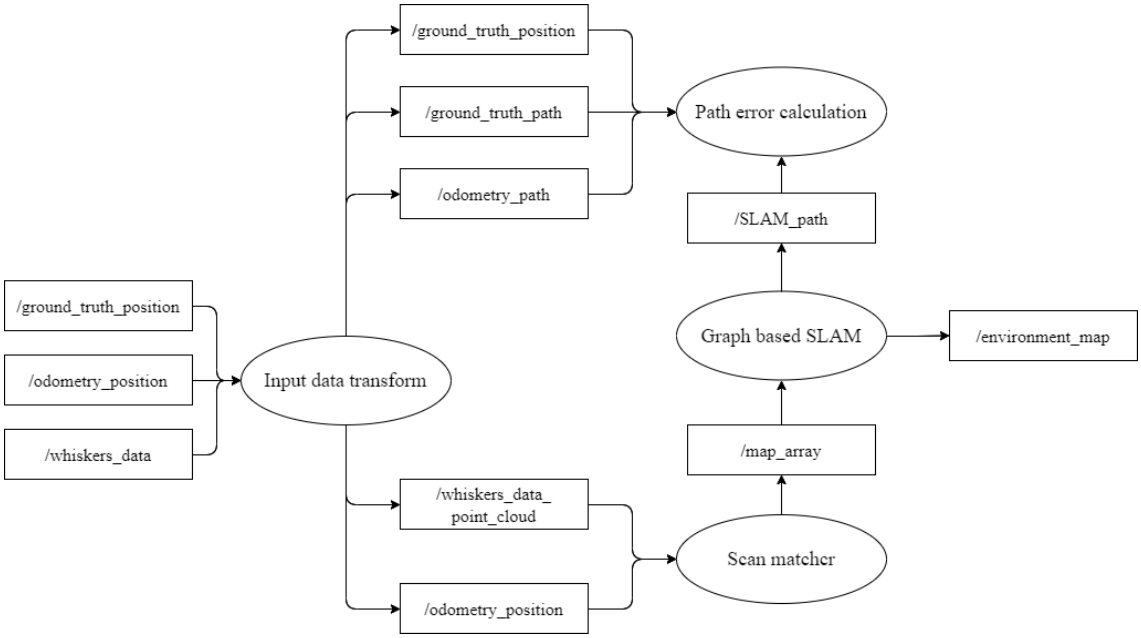

Figure 15. Graph of ROS2 nodes and topics. Rectangular cells – topics, elliptical cells – nodes.

3.5 Used hardware

A Lenovo Y50-70 laptop was used for simulator experiments and SLAM implementation during real experiments It has Intel Core i5-4210H processor, 12 GB RAM and NVIDIA GeForce GTX 960M Graphics processing unit. Ubuntu 20.04 compatible with ROS2 Foxy was used.

The RM3 robot has two Olimex A20 System-on-Chip modules with a multicore ARM processor for communicating with whiskers, other sensors and actuators, and for running the robot's control software. The computers of RM3 use Ubuntu 20.04 with ROS2 Foxy.

4 Results

This chapter presents an analysis of the results of the experiments that were run in simulator and in the real environment with the physical prototype. A discussion of the results and the future outlook of this work are also offered.

4.1 Simulator experiments results

The experiments were divided into two phases. For the first phase, the 16 whisker configurations described in Section 3.2.2 were compared. The best four of them were chosen to be evaluated in the second phase to find the best configuration.

4.1.1 First phase

During the first phase of the experiments, each whisker configuration passed each track (Section 3.2.1) once. The track consisted of 3 loops. The deviation of the SLAM path and odometry path from the true path as well as the error of centralized paths for each loop in the track are shown in the Appendices 2-9. In these tables, it can be seen that the error decreases throughout the loops. There are significant changes in the error between the first and second loops, whereas minor adjustments are observed after the second and all subsequent loops. Thus 3 loops are enough for the robot to assess the environment.

As can be seen from the Table 1, on most tracks the error improves on the last loop compared to the first one. Positive values in this table mean that the SLAM path accuracy has improved, while negative values means that accuracy has worsened. The only exception is the U-shape track, where the results worsen. This is due to the special shape of the track. In it, the robot makes two turns and comes back. It does not close in a circle, making it difficult for SLAM to determine that the correct angle of rotation is 90 degrees, so with each new loop this angle becomes larger for some whiskers configurations. The graph of the changes in errors during the passage of tracks is shown in Figure 16.

Table 1. Error difference [m] between 1 and 3 loops for all tracks and whisker configurations.

Whiskers configuration	**8-shape**	**U-shape**	**Circle**	**Line**
N1	0.2380	-0.2034	0.1350	0.0702
N2	0.0973	-0.3977	-0.3614	0.2939
N3	0.1015	0.0447	0.0169	-0.0508
N4	0.0711	0.2010	0.2359	0.1763
N5	0.0213	-0.0221	-0.0084	0.1535
N6	0.0284	-0.0731	-0.0028	-0.0597
N7	0.2044	0.4856	-0.0040	0.0241
N8	0.4113	-0.1948	0.2060	0.1551
S1	0.3413	-0.3144	0.2130	0.1584
S2	0.2531	-0.1595	0.3523	0.1692
S3	0.3751	-0.0740	-0.2070	0.0797
S4	1.9114	-0.0691	-0.0425	0.0782
S5	0.2037	-0.2189	-0.1051	0.0860
S6	0.1897	-0.0508	-0.1109	-0.0040
S7	-0.0286	-0.3558	-0.1561	0.0737
S8	1.9719	-0.2932	0.3278	0.0766
Average result	0.3994	-0.1060	0.0305	0.0925

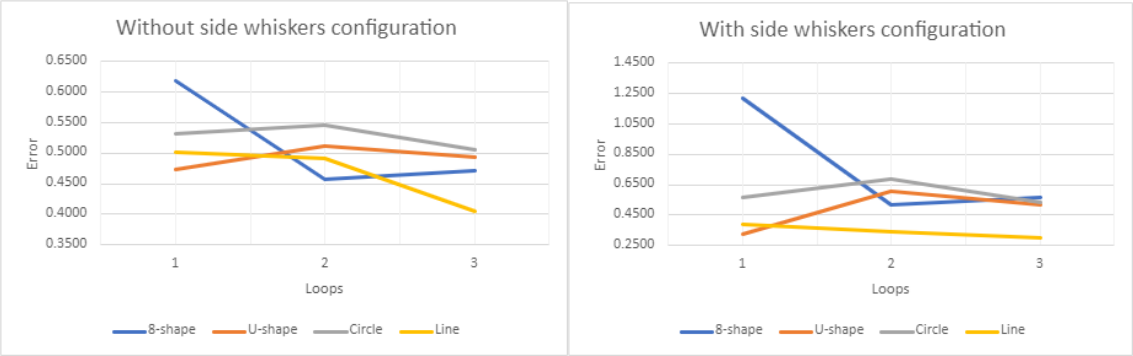

Figure 16. Graphs of error change. Graphs with average error of configurations without side whiskers N1-N8 (left graph) and with side whiskers S1-S8 (right graph).

The whisker configurations S4 and S8 in the 8-shape track stand out particularly. On the first loop, the SLAM path error from the ground truth path is 3.0033m and 2.5198m, respectively, although by the last loop they decrease to 1.0919m and 0.5478m. In these configurations, a decrease in the error by about 3 and 5 times can be seen, respectively. Of course, this is caused by the incorrect definition of the robot's path during the first loop, but it shows how SLAM can correct errors during the loop closures while passing through the same points.

The paths errors in the end of the track for each whisker configuration are shown in the Appendices 10 - 11. The centralized paths are represented in Appendices 12 – 13. Tables provided in these Appendices show the percentage of improvement or deterioration of the SLAM path error from the true path compared to the error of the path odometry from the true path.

$$Difference\ in\ \% = \frac{Odometry\ Error - SLAM\ error}{Odometry\ Error} * 100\%$$

This is done for each track and then the average result is calculated. For each track, the whisker configurations are ranked from 1 to 16 based on their ability to improve the odometry path using SLAM compared to other configurations. These rankings are also averaged.

Looking at these tables, it can be concluded that the configurations with side whiskers show slightly better results than the configurations without them. This can be seen in Appendices 10 and 12 tables. There, the configurations that perform better with the side whiskers are marked in red, and the configurations that were better without them are highlighted green. In 18 cases out of 32, the configurations with the side whiskers show better results than the configurations without the side whiskers in terms of path errors.

Similarly, when evaluating the centralized paths, the configurations with the side whiskers perform better in 19 out of the 32 cases. Moreover, the average results (Table 2 and Table 3) from all the tracks show that the performance of the configurations with the side whiskers is better. Based on the results of the evaluation, it can be concluded that the use of the combinations with side whiskers is more effective for the implementation of SLAM on a miner robot.

Table 2. Error difference in % between odometry and SLAM paths.

Whiskers configuration	**8-shape**	**U-shape**	**Circle**	**Line**	**Average result**
Configurations without side whiskers	32.30%	-70.58%	-4.65%	-80.80%	-30.93%
Configurations with side whiskers	18.26%	-78.01%	-10.75%	-34.76%	-26.32%

Table 3. Error difference in % between odometry and SLAM centralized paths.

Whiskers configuration	**8-shape**	**U-shape**	**Circle**	**Line**	**Average result**
Configurations without side whiskers	26.39%	-4.55%	-38.24%	-39.56%	-13.99%
Configurations with side whiskers	25.42%	-9.82%	-25.29%	41.07%	7.84%

As can be seen from the tables in Appendices 10-13, the configurations with a higher number of whiskers perform better on the circle and 8-shape trajectories. While on the line and the U-shape track, the configurations with a small number of whiskers give better results. For example, on the 8-shape track the five best configurations are N7, N8, S1, S2 and S5 configurations. On the circle track, the best performing configurations are N1, N4, N7, S1 and S2. The configurations with the minimum number of whiskers, which are N4, N7, and N8, have a total of 24 whiskers. On the other hand, the five best configurations on U-shape track are N3, N5, N6, S3 and S6. There are only three configurations, namely

N3, N6, and S6, which demonstrate positive results on the line track. S3 configuration has the maximum number of whiskers. It has 32 whiskers. All the other configurations have no more than 24 whiskers.

This happens due to the fact that on the more complicated tracks, as circle track and 8-shape track, more whiskers are needed to create more poses. Thanks to this, the robot has more knowledge of the environment, and it is easier for it to correct errors combining the right poses. However, on the simple tracks, such as the line or U-shape track, the most significant issue is the generation of incorrect robot movement angles due to the correlation of the wrong points. Therefore, creating more poses increases the chance of a significant error.

Based on the data obtained, it was decided to select the two best configurations from the circle and 8-shape tracks and two configurations from the line and U-shape tracks, and then compare them in the second phase. For these purposes, all the data were divided into two tables – one for the circle and 8-shape tracks data (Appendix 14), and the another for the line and U-shape data (Appendix 15). Based on the table of the circle and 8-shape tracks result, configurations S1 and S2 have the best averaged results. Moreover, they are in the top four best configurations in the 8-shape rack as well as in the circle track. Configurations N4, N7 and S5 also deserve special mention, they show excellent results on one of the tracks, but on the other tracks, their results are not as impressive. In the line and U-shape tracks group, configurations N3 and S6 show the best results. They are the only ones to show a positive average result on both tracks. On each of the tracks, they also show the best results. Closest to them is the N6 configuration. Thus, the decision is to carry out tests with configurations S1, S2, N3 and S6 in the next phase.

4.1.2 Second phase

During the second phase of the experiments four configurations of whiskers were evaluated. For these purposes, for every track a new data file was recorded for the robot including odometry and whiskers information. During this phase, every track consisted of 3 loops. Five replicates were performed for each track and each whiskers configuration. Thus, a total of 80 experiments were conducted. The results of these experiments can be found in Appendices 16 – 19. These tables present data from each test for each configuration, as well as the average results.

As can be seen from the table of the average results of all the tests (Appendix 20), the performance of all the configurations is quite similar in the range of 5%. U-shape is still the most challenging track for all the configurations. Here, no configuration fails to show a positive result. On the line track, configuration N3 performs best, it is the only one that demonstrates a positive result. Conversely, it is the only configuration that shows a negative result on the circle track. On the 8-shape track, all configurations show a positive result.

The two best configurations are S2 and N3 with the overall results on all the tracks of 4.63% and 3.11% respectively. Configuration N3 shows such a result because it overperformed on the line track. The paths of configurations S1, S2 and S6 on this track have moved aside from the ground truth path in the global frame. Based on the centralized values (Appendix 21) on the line track, it can be observed that the other whisker configurations show better results than configuration N3 and their overall results are also better. Due to all these facts, the conclusion I made is that the whisker configuration S2 is the best for the use on the robot miner. In this configuration, the robot has enough whiskers to assess its environment, but at the same time not so many for the robot to combine incorrect poses and make mistakes. Additionally, this configuration has side whiskers that provide improvement in performance.

4.1.3 Final results

For the best whiskers configuration (S2) additional tests were conducted. Using this configuration, the robot passed all 4 tracks. This time robot did 5 loops on each track. This section provides the results of these tests. Table 4 shows that SLAM made some improvements for the odometry path. This configuration showed the positive result for every track on the final loop. The first loop results were not so good, for example on the circle and 8-shape tracks the difference between the error of the SLAM and odometry paths was negative. On the U-shape track this difference was less than 10%. However, during the next loops this result was improved. Furthermore, the data in Table 4 reveals that the most significant changes in SLAM occur between the first and second loops, while subsequent loops only involve minor adjustments. The best changes between the first and second loops were on the circle and 8-shape tracks, the errors were improved by 0.26756 and 0.3268 times respectively.

Table 4. The 5 loops test results for S2 configuration. The odometry is denoted in this and all the following tables as odom.

		Circle	**U-shape**	**8-shape**	**Line**
	Odom error [m]	0.43022	0.66144	0.84513	0.48330
	SLAM error [m]	0.63352	0.60216	0.97944	0.30266
1. loop	Difference in %	-47.26%	8.96%	-15.89%	37.38%
	Odom error [m]	0.44912	0.72636	0.87430	0.58046
	SLAM error [m]	0.36596	0.47686	0.65264	0.25748
2. loop	Difference in %	18.52%	34.35%	25.35%	55.64%
	Odom error [m]	0.49501	0.94317	0.97881	0.53904
	SLAM error [m]	0.42740	0.48132	0.80369	0.27361
3. loop	Difference in %	13.66%	48.97%	17.89%	49.24%
	Odom error [m]	0.58748	1.09076	0.93807	0.50092
	SLAM error [m]	0.49946	0.56672	0.79251	0.34180
4. loop	Difference in %	14.98%	48.04%	15.52%	31.77%
	Odom error [m]	0.63415	1.34086	0.93846	0.49449
	SLAM error [m]	0.54875	0.71031	0.80326	0.37244
Final loop	Difference in %	13.47%	47.03%	14.41%	24.68%

The worst result was shown on the circle track. It was 13.47%. The reason for this is that the loops do not converge perfectly on each other (Figure 17). Some of them are slightly higher, and some are lower, resulting in duplicated images of the bricks (Figure 18). Although, on the straight line after the second turn, the figures are quite accurate.

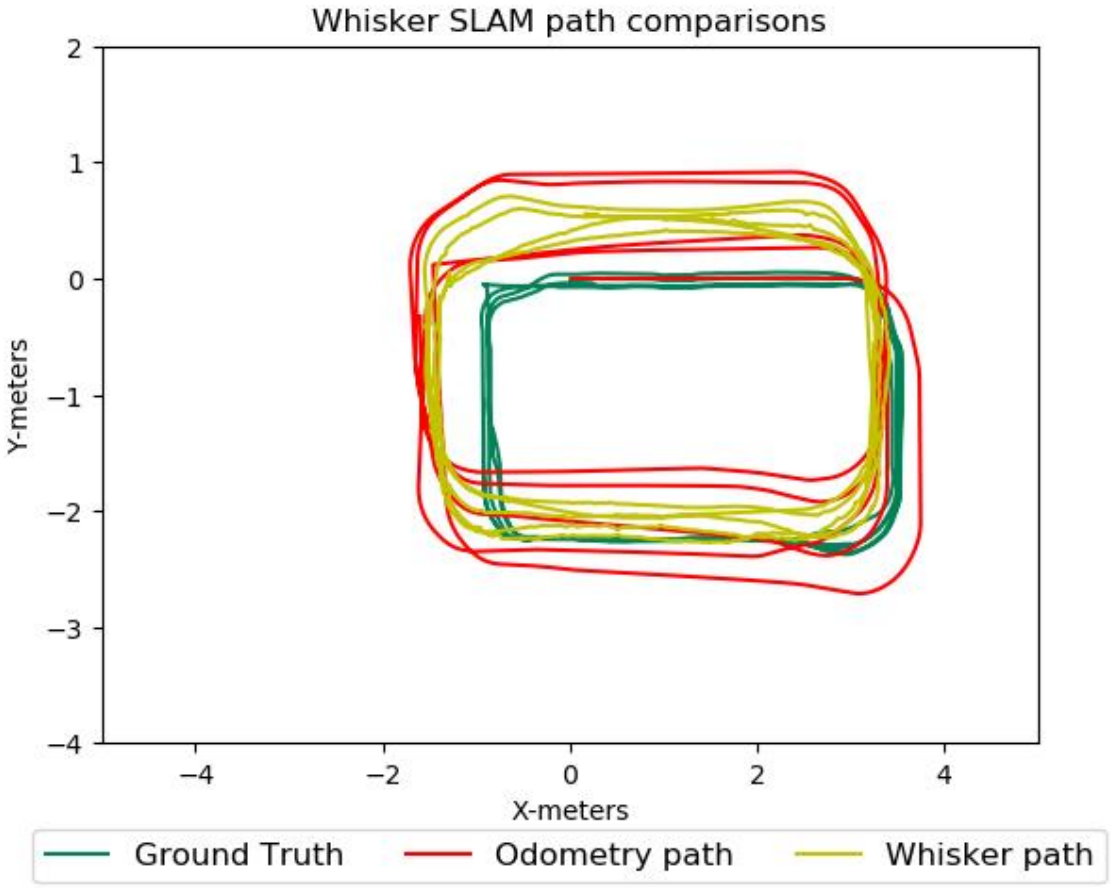

Figure 17. Final paths from the circle track test.

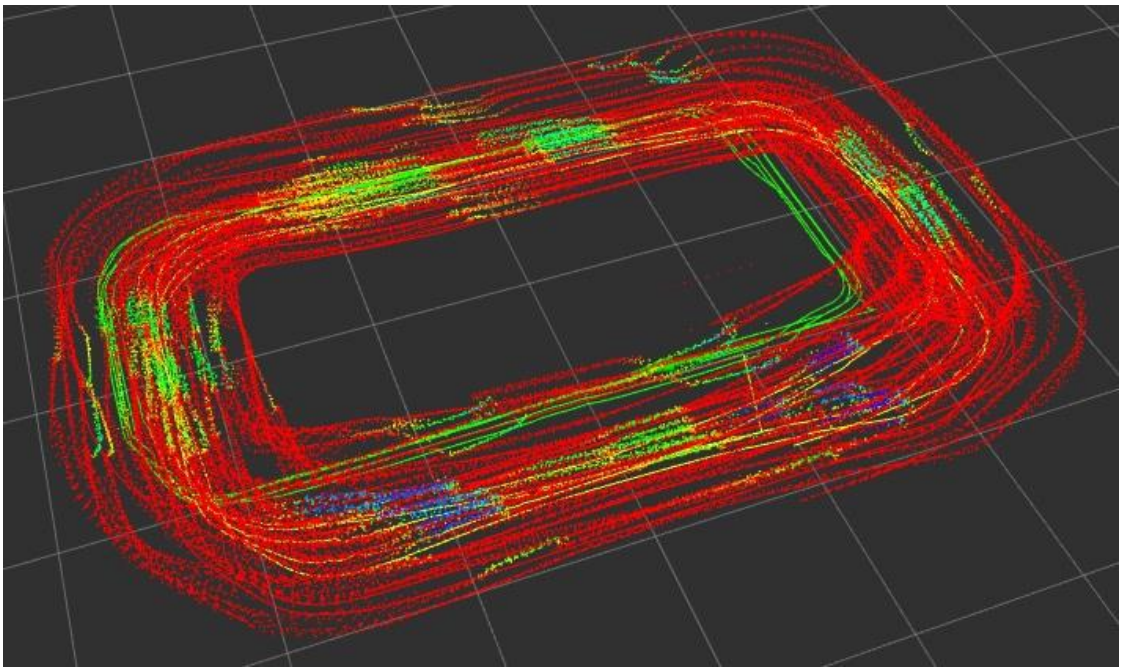

Figure 18. A simulated map of the circle track with paths. Orange/red - lowest points, blue/green - highest points.

SLAM performs especially well on the U-shape track, although such performance is more likely due to bad odometry data. As can be seen from Figure 19 and Figure 20, after the first turn of the fourth loop the SLAM path is devided into two separate lines. This is the reason that the error increases rapidly on both the fourth and final loops.

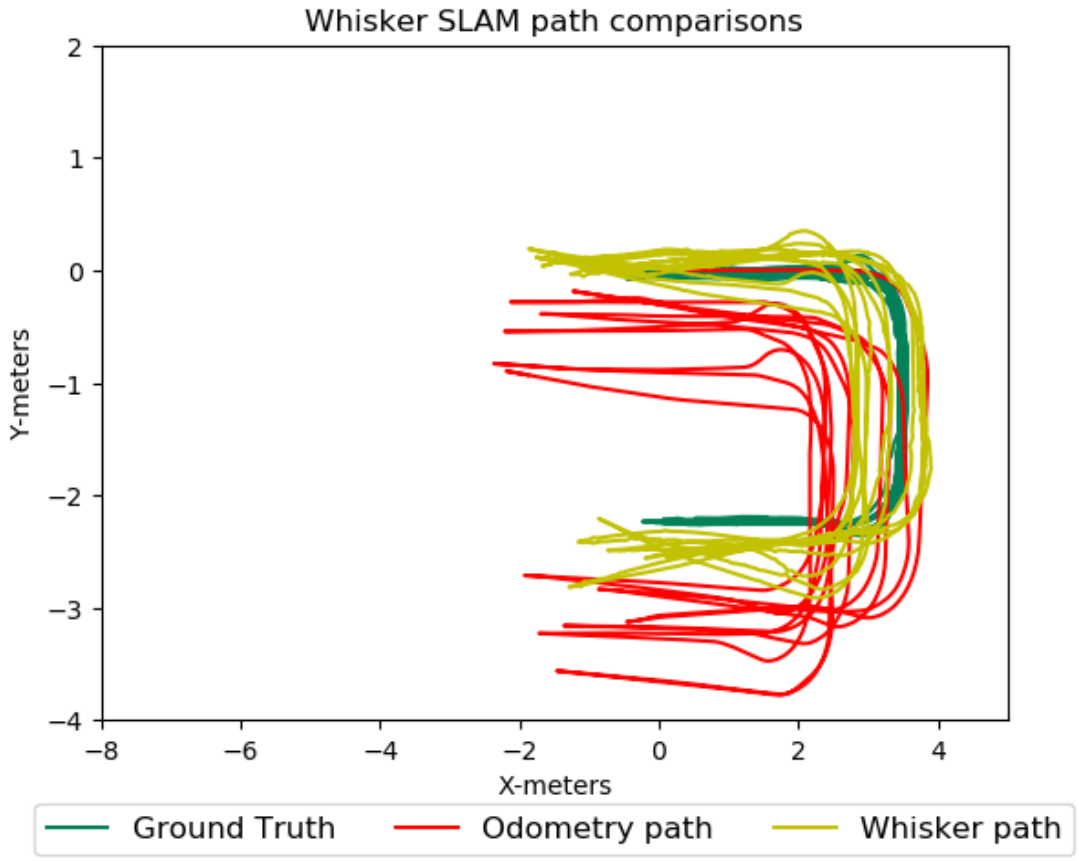

Figure 19. Final paths from the U-shape track test.

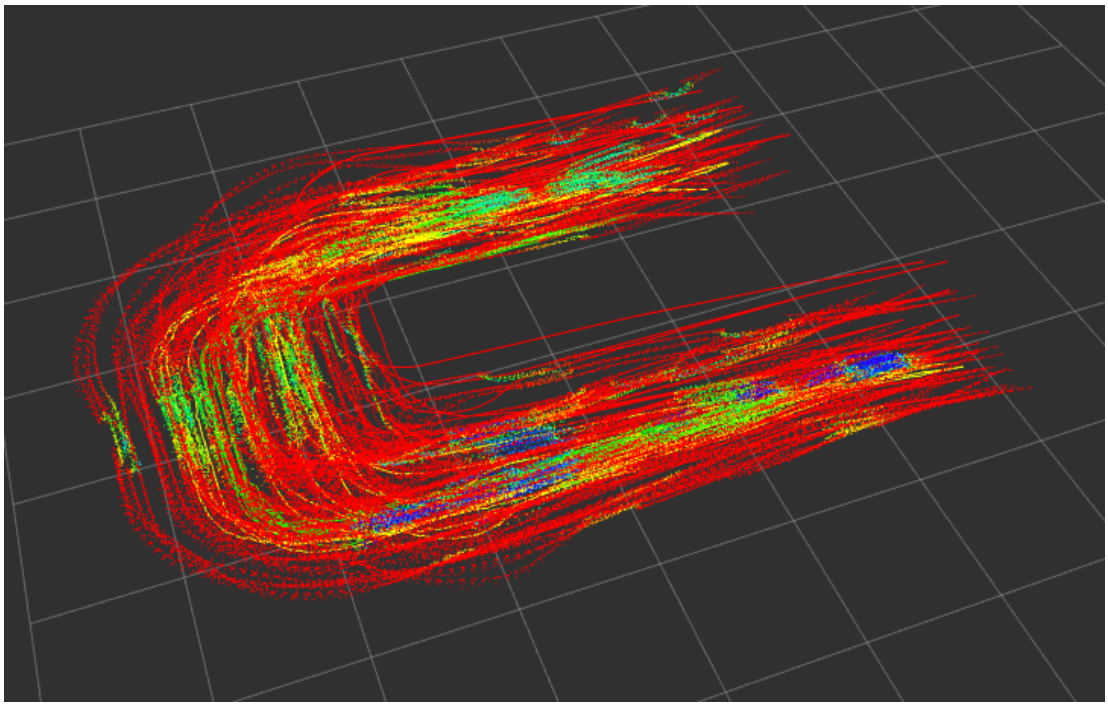

Figure 20. A simulated map of the U-shape track with paths. Orange/red - lowest points, blue/green - highest points.

8-shape is the most challenging track. It is the longest track and has the greatest number of turning points. However, SLAM shows good results here. Every brick shape is visible quite well there, despite that some shape point clouds are distorted by overlapping each on the other (Figure 22). It can save the track shape, although the SLAM path moves aside to the odometry path (Figure 21).

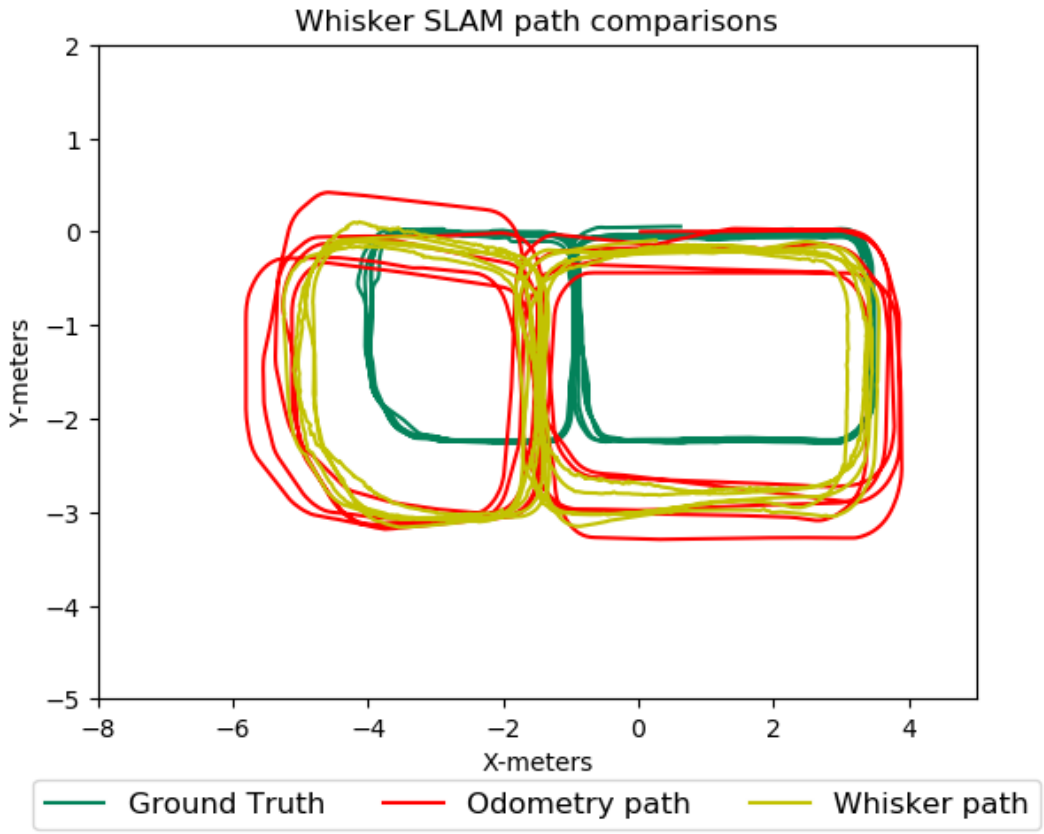

Figure 21. Final paths from the 8-shape track test.

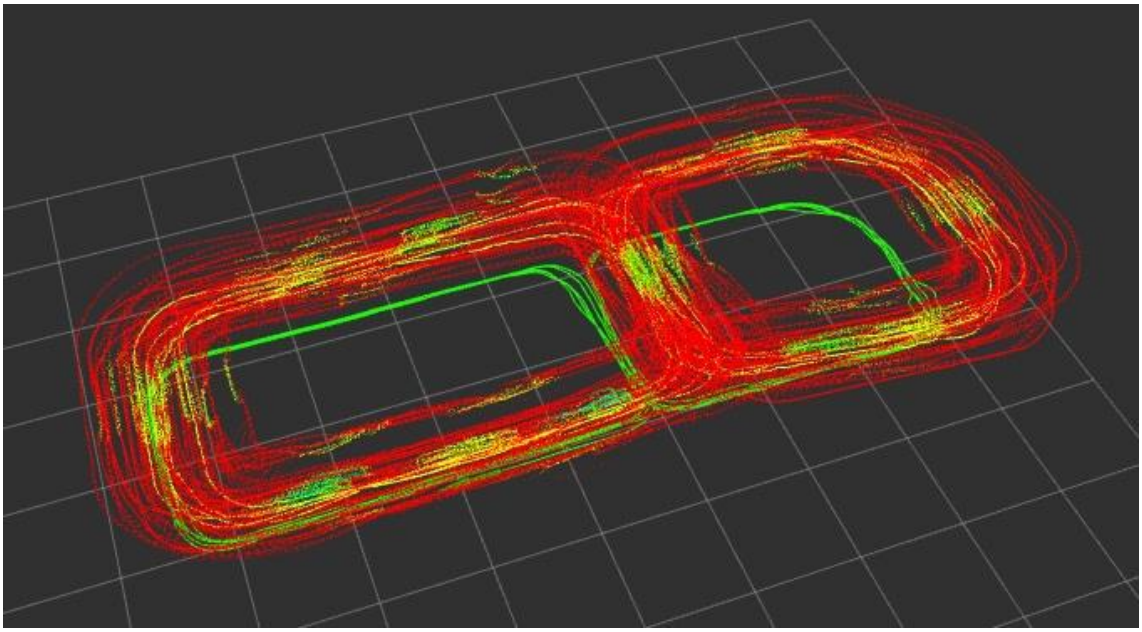

Figure 22. A simulated map of the 8-shape track with paths. Orange/red - lowest points, blue/green - highest points.

On the line track, the shapes of every brick on the ground are visible well (Figure 24). During this test SLAM helped the robot keep its position in one line (Figure 23). However, the SLAM path moves a bit aside from the ground truth path.

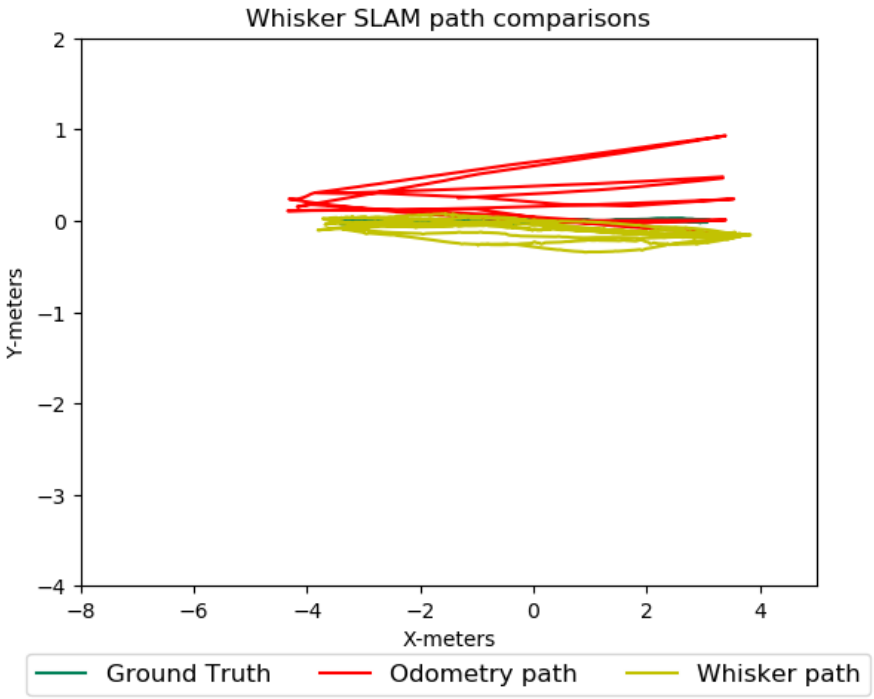

Figure 23. Final paths from the line track test.

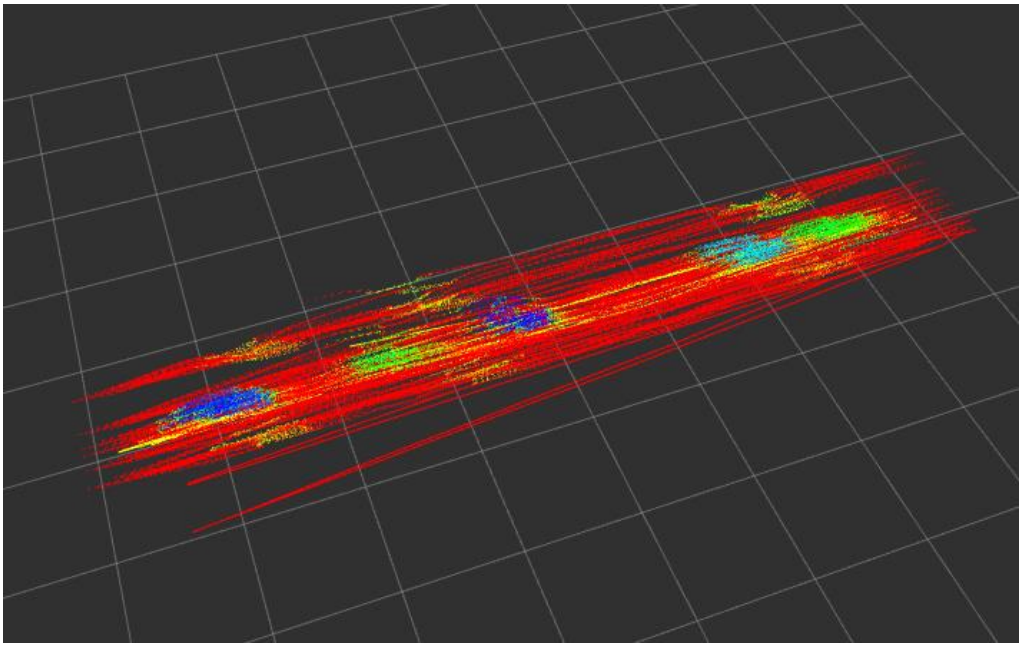

Figure 24. A simulated map of the line track with paths. Orange/red - lowest points, blue/green - highest points.

4.2 Real experiments results

The best whiskers configuration S2 was used to implement SLAM on real robot. The whiskers data acquisition and motor control were implemented on two Olimex modules onboard the robot, while SLAM and odometry calculations were done on a laptop. Thus,

it was possible to observe the work of SLAM along the robot's movement. Additionally, laptop was used for recording ROS2 bag files with robot motion data and whiskers data. These bag files were later replayed to repeat the experiments three times and evaluate the SLAM performance.

Table 5 shows that whisker configuration S2 on the line and circle tracks showed positive results. This happens since the odometry data was very accurate on these tracks, so the SLAM data does not perform as well in comparison. Although, as can be seen in Figure 25, the SLAM path does not stray far from the ground truth path. This is confirmed by the data from Table 5. Not a single SLAM error is higher than 1, even on the most difficult 8-shape track. On the line and the circle tracks, the odometry strays far from the ground truth path, but SLAM is able to correct this mistake. These findings prove the possibility of using SLAM on a real robot.

Table 5. Real tests results.

		Circle	**U-shape**	**8-shape**	**Line**
	Odom error [m]	0.81319	0.26334	0.48347	0.96479
	SLAM error [m]	0.68201	0.42786	0.65434	0.54433
1. test	Difference in %	16.13%	-62.47%	-35.34%	43.58%
	Odom error [m]	0.79444	0.29131	0.44849	0.97493
	SLAM error [m]	0.61651	0.66364	0.66637	0.55332
2. test	Difference in %	22.40%	-127.81%	-48.58%	43.25%
	Odom error [m]	0.81520	0.25728	0.46438	0.96586
	SLAM error [m]	0.62041	0.44389	0.90233	0.58286
3. test	Difference in %	23.89%	-72.53%	-94.31%	39.65%
Average results	Average difference in %	20.81%	-87.61%	-59.41%	42.16%

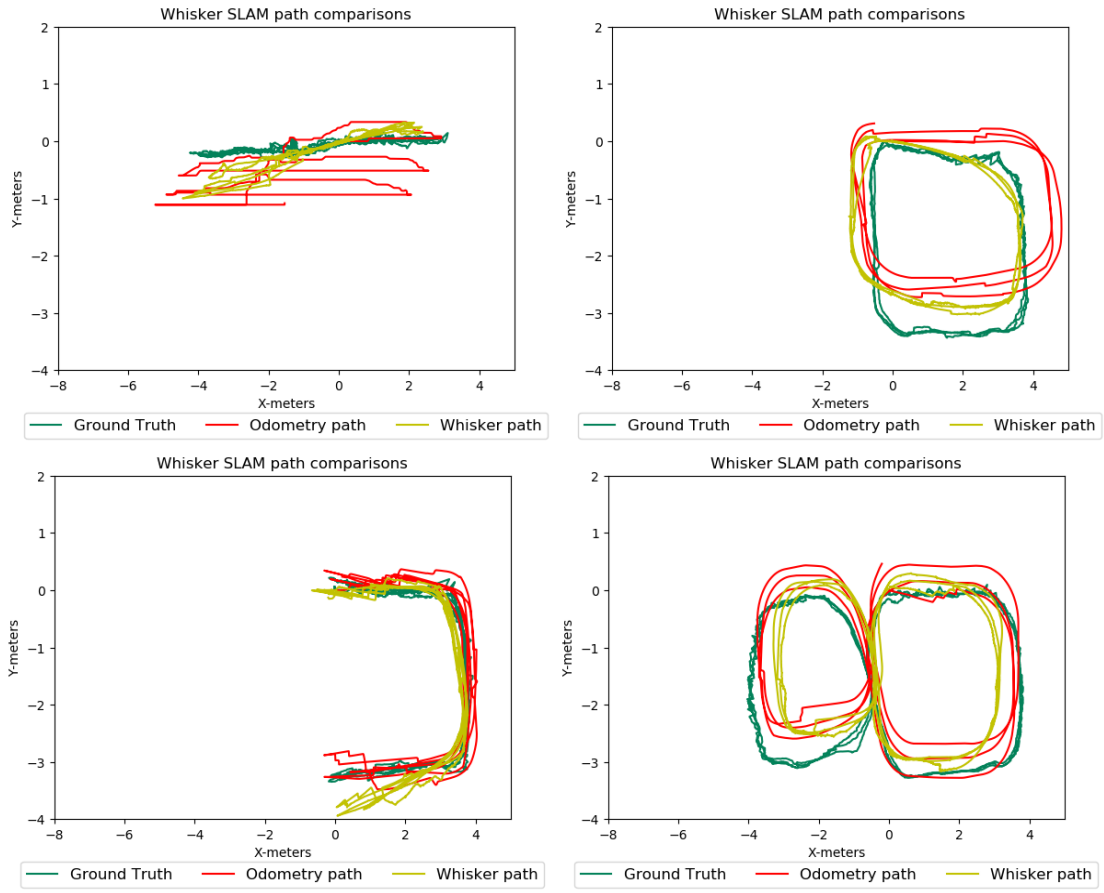

Figure 25. Paths of the first tests on real robot.

Moreover, an experiment on real robot using random trajectory was conducted. During this experiment the robot combined elements of various tracks, and also added new ones. The Figure 26 shows the path that was followed. Firstly (1), it made a small circle, after that (2 and 3) was two big circles with following (4) reversed 8-shape trajectory. Finally (5), it passed through the middle of the right circle, made a line and (6) finished with U-shape moving backwards.

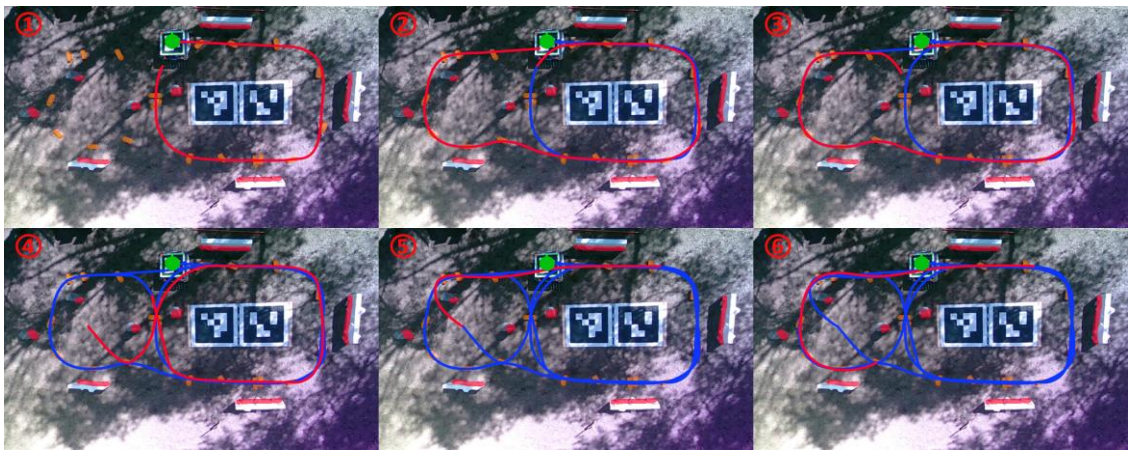

Figure 26. Trajectory of the test on the random track. Red path shows actual segment for current picture, blue paths show the old segments.

The SLAM copied well the first circle, but when robot performed the big circles the SLAM made a mistake with map orientation (Figure 27 and Figure 28), that which led to worsening the error (Table 6). Although, it was still able to find the previously visited areas and correlate its path relative to them.

Table 6. Random track results. Error [m].

	Random track
Odom error [m]	0.89148
SLAM error [m]	1.34633
Difference in %	-51.02%

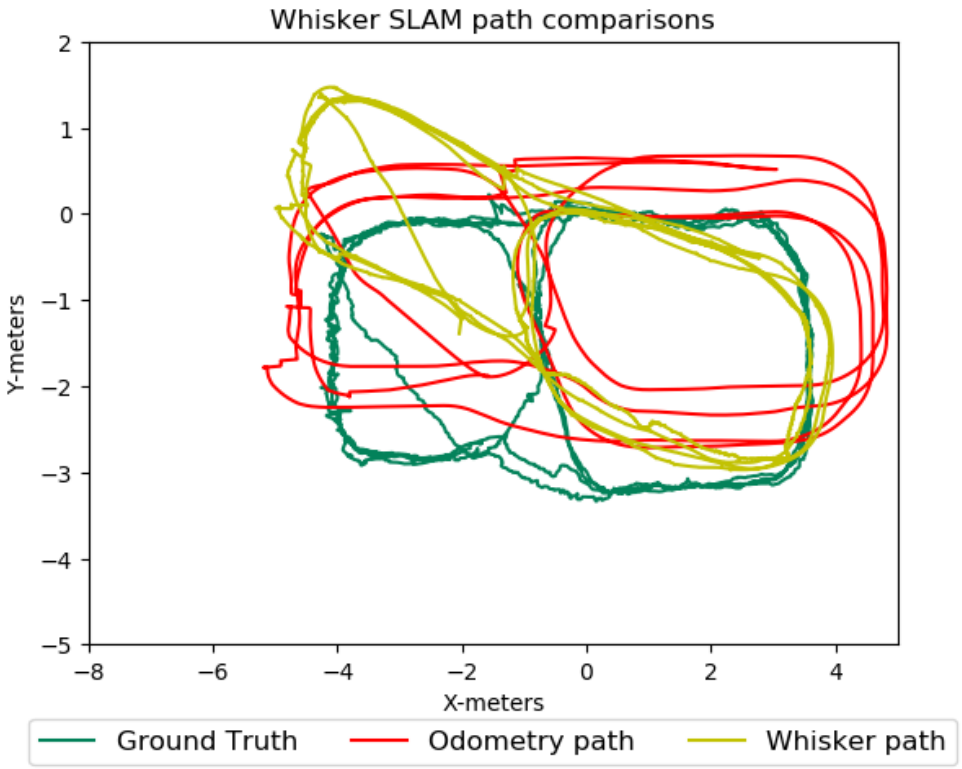

Figure 27. Paths of the test on the random track.

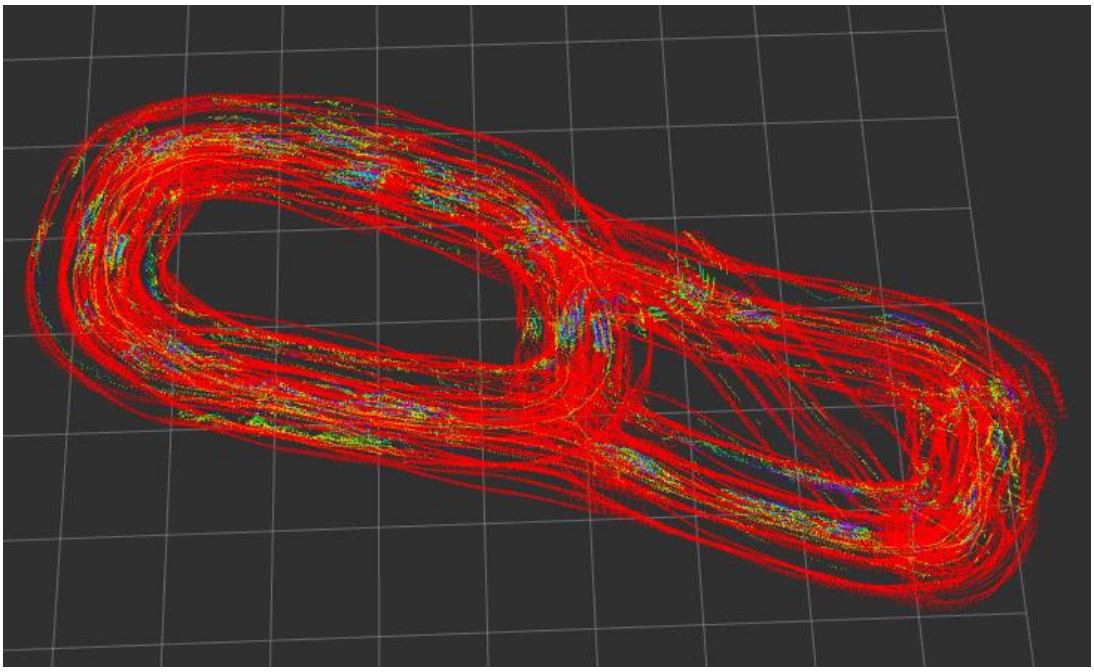

Figure 28. Map of the environment of the test on the random track. Orange/red - lowest points, blue/green - highest points.

4.3 Discussion

During the thesis, 16 configurations on 4 tracks were investigated. This research made it clear that the largest possible number of whiskers on a robot is not always the best solution. Of course, more whiskers provide more information about the environment. However, this also increases the chances that the frames will not match correctly, especially if the whiskers are fairly close together. For example, in situations where, in a loop closure, the frame from the previous loop does not coincide with its identical frame, but with the next one that has similar features. Moreover, the optimal whiskers configuration (S2), which has 48 whiskers, showed better results than the configuration with maximum whiskers number (64 whiskers). It proves that reasonable reduction of the whiskers does not reduce the performance and makes the system simpler.

During tests in the real environment, it was revealed that SLAM did not improve but was worse than the result of the odometry on the U-shape and the 8-shape tracks. The same was observed during the first phase of experiments on the line and the U-shape tracks, the odometry path error on these tracks was less than 0.3 meters. This is due to the fact that SLAM estimates the probability of the robot passing certain points. Models based on probabilities are not as accurate as models based on mathematical equations. Moreover, if SLAM sees the same scan frames at two different points, then it tries to find an average

position for these two frames. Because of this, SLAM can make a mistake that will move some sections of the SLAM path away from the ground truth path. Of course, later such errors are levelled during subsequent passes through these places, which will give SLAM more information and help improve the error. However, in cases where the result of the odometry path is close to ideal, such errors will make the result of the SLAM path worse than that of the odometry. In cases of bad odometry, SLAM corrects odometry errors. It sees the visited areas, and when it passes close to them again, it can correct the position of the robot. In this case, the errors caused by inaccurate calculations of the SLAM do not play such a big role, although they are still existed and negatively affect the operation of the SLAM.

For the same reason, SLAM has problems with orientation. This can be seen in the last experiment with a random trajectory. SLAM performed well with the small circle, but when it started the big circle, it couldn't match the frames correctly. Because of this, an error occurred with the orientation of the robot in space, which eventually turned slightly clockwise the entire map and SLAM path. In the future, this problem can be solved by using a system of weights that will give priority to earlier frames. Moreover, an additional sensor (lidar or visual) can be used for better orientation of the robot in a global environment.

4.4 Future work

For the purpose of this thesis, the whiskers-SLAM system compatible with ROS2 was created. Moreover, the optimal whiskers configuration was found and implemented on the real robot. However, this area is not sufficiently researched, and there are still many things to be explored.

First of all, the effect of other parameters on SLAM performance can also be investigated. For example, during experiments the robot moved along the tracks in the same environment. The influence of the testing environment on the work of SLAM can be also explored. It can be studied how the changes in minimum and maximum distances between 2 frames affect SLAM performance in the different environments.

Secondly, SLAM software was based on one existing framework [24]. However, other frameworks may show better results. There is a potential for further exploration into how

whiskers can be integrated in various frameworks and for investigation of whether there is one software that is optimal for all whisker configurations, or different configurations work better with different frameworks.

Moreover, further research can focus on the work with real robots rather than a simulator. Even if during this thesis SLAM software was integrated to the real robot, there is a possibility to evaluate how onboard SLAM with different whiskers configurations affect the real robot. Since the robot computing resources are restricted, it can be studied how much resources (for example memory) the robot spends on SLAM, whether SLAM affects other functions of the robot, and how fast onboard SLAM works with different configurations.

Finally, since Artificial Intelligence topics are becoming more and more popular these days, it can be integrated with the SLAM system. For example, the SLAM system based on reinforcement learning can be developed. Such systems after some runs can improve SLAM performance. If we use these systems on a real robot, which works every time in the same mine, SLAM results will improve with every new run of the robot. As can be seen from all these ideas, SLAM systems based on whiskers have a lot of potential for future research.

5 Summary

This thesis was set out to create the whisker-SLAM system compatible with ROS2 and evaluate different whiskers configurations performance. For that purposes the SLAM algorithm based on lidar was modified to work with whiskers sensors.

After that, the number of whiskers and their location were evaluated through several phases of experiments. For that purpose, 16 whiskers configurations were designed, eight of them were with side and bottom whiskers and eight were with only bottom whiskers. For testing purposes four simulated tracks were designed – circle, line, 8-shape and U-shape. The evaluation was carried out by comparing the RMSE error obtained from the SLAM path and the odometry path relative to the ground truth path. For this purpose, an assessment software was developed that allowed to evaluate the error at the end of each loop of a track.

During the first phase of the experiments all 16 configurations passed all four tracks once with three loops. As a result, the four best configurations were identified. Moreover, it became clear that the configurations with side whiskers showed better results than the configurations without them. Additionally, during the research it was found out that on the complex circular tracks, such as the circle and 8-shape tracks, better results were demonstrated by configurations with greater number of whiskers as they could obtain more information from the environment. However, on the line and U-shape track configurations with lower number of whiskers results were better, because on these simple tracks the biggest problem issue was the generation of incorrect robot movement angles due to the correlation of the wrong poses.

During the second phase the four best configurations from the first phase were evaluated. They passed each track five times. Based on the average results it was found out that the best configuration for robot miner is S2 – the configuration with four arrays on the bottom of the robot and with side whiskers. Moreover, using this configuration additional tests were conducted where the robot did 5 loops for every track. By this test it became clear that the biggest improvements in SLAM appear between the first and the second loops.

Finally, tests on real robot were conducted. All previously mentioned four tracks as well as one track with random trajectory were tested. The results of these tests prove that the SLAM can be used on real robot miner and help to map the environment and to improve robot position in case odometry makes a mistake.

This thesis showed the possibilities of using tactile sensors SLAM on the robot miner, as well as through experiments the best configuration of whisker number and location was found here.

References

[1] K. Ebadi, Y. Chang, M. Palieri, A. Stephens, A. Hatteland, E. Heiden, A. Thakur, N. Funabiki, B. Morrell, S. Wood, L. Carlone and A.-a. Agha-mohammadi, "LAMP: Large-Scale Autonomous Mapping and Positioning for Exploration of Perceptually-Degraded Subterranean Environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, 2020.

[2] A. Nagel, “Blind Mapping and Localisation for Small-Scale Mining Robots”, Master's thesis, Tallinn: Tallinn University of Technology, 2021.

[3] S. Macenski, T. Foote, B. Gerkey, C. Lalancette and W. Woodall, "Robot Operating System 2: Design, architecture, and uses in the wild," *Science Robotics,* vol. 7, no. 66, 2022.

[4] "Robominers," [Online]. Available: http://robominers.eu. [Accessed 01 04 2023].

[5] A. Ristolainen, J. A. Tuhtan and M. Kruusmaa, "Continuous, Near-Bed Current Velocity Estimation Using Pressure and Inertial Sensing," *IEEE Sensors Journal,* vol. 19, no. 24, pp. 12398-12406, 15 Dec 2019.

[6] S. Thrun, W. Burgard and D. Fox, Probabilistic Robotics. Cambridge, The MIT Press, 2005.

[7] G. Grisetti, R. Kümmerle, C. Stachniss and W. Burgard, "A Tutorial on Graph-Based SLAM," in *IEEE Intelligent Transportation Systems Magazine, vol. 2, no. 4*, 2010.

[8] T.-j. Lee, B.-m. Jang and D.-i. D. Cho, "A non-iterative pose-graph optimization algorithm for fast 2D SLAM," in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, Bali, Indonesia, 2014.

[9] J. Soares and M. Meggiolaro, "Keyframe-Based RGB-D SLAM for Mobile Robots with Visual Odometry in Indoor Environments Using Graph Optimization," in *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*, João Pessoa, Brazil, 2018.

[10] M. H. Evans, C. W. Fox, M. J. Pearson and T. J. Prescott, "Object location, orientation, and velocity extraction from artificial vibrissal signals," in *Society for Neuroscience Abstracts. Society for Neuroscience,* Chicago, IL, 2009.

[11] D. Wu, Y. Meng, K. Zhan and F. Ma, "A LIDAR SLAM Based on Point-Line Features for Underground Mining Vehicle," in *2018 Chinese Automation Congress (CAC)*, Xi'an, China, 2018.

[12] E. Menendez, S. Martínez De La Casa, M. Marín and C. Balaguer, "uSLAM Implementation for Autonomous Underground Robot," in *019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Leicester, UK, 2019.

[13] M. Tranzatto, T. Miki, M. Dharmadhikari, L. Bernreiter, M. Kulkarni, F. Mascarich, O. Andersson, S. Khattak, M. Hutter, R. Siegwart and K. Alexis, "CERBERUS in the DARPA Subterranean Challenge," *Science Robotics*, vol. 7, no. 66, 2022.

[14] N. Hudson, F. Talbot, M. Cox, J. Williams, T. Hines, A. Pitt, B. Wood, D. Frousheger, K. L. Surdo, T. Molnar, R. Steindl, M. Wildie, I. Sa, N. Kottege, K. Stepanas, E. Hernandez and G. Catt, "Heterogeneous Ground and Air Platforms, Homogeneous Sensing: Team CSIRO Data61's Approach to the DARPA Subterranean Challenge," *Field Robotics*, vol. 2, no. 1, pp. 595--636, 2022.

[15] S. Macenski and I. Jambrecic, "SLAM Toolbox: SLAM for the dynamic world," *Journal of Open Source Software*, vol. 6, no. 61, p. 2783, 2021.

[16] S. Macenski, F. Martín, R. White and J. G. Clavero, "The Marathon 2: A Navigation System," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA, 2020.

[17] M. Labbe and F. Michaud, "RTAB-Map as an Open-Source Lidar and Visual SLAM Library for Large-Scale and Long-Term Online Operation," *Journal of Field Robotics*, vol. 36, no. 2, p. 416–446, 2019.

[18] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255-1262, 2017.

[19] W. Hess, D. Kohler, H. Rapp and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden, 2016.

[20] M. Salman and M. J. Pearson, "Advancing whisker based navigation through the implementation of Bio-Inspired whisking strategies," in *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Qingdao, China, 2016.

[21] O. Struckmeier, K. Tiwari, M. Salman, M. J. Pearson and V. Kyrki, "ViTa-SLAM: A Bio-inspired Visuo-Tactile SLAM for Navigation while Interacting with Aliased Environments," in *2019 IEEE International Conference on Cyborg and Bionic Systems (CBS)*, Munich, Germany, 2019.

[22] C. Fox, M. Evans, M. Pearson and T. Prescott, "Tactile SLAM with a biomimetic whiskered robot," in *2012 IEEE International Conference on Robotics and Automation*, Saint Paul, MN, USA, 2012.

[23] W. Remmas, R. Gkliva and A. Ristolainen, "Dynamic modelling of a screw actuator for improved locomotion control on various terrains," in *EGU General Assembly 2022*, Vienna, Austria, 2022.

[24] R. Sasaki, "rsasaki0109/lidarslam_ros2," 30 April 2020. [Online]. Available: https://github.com/rsasaki0109/lidarslam_ros2. [Accessed 04 March 2023].

[25] M. Korkmaz, Y. Nihat and D. Akif, "Comparison of the SLAM algorithms: Hangar experiments," in *MATEC Web of Conferences*, 2016.

[26] Z. Chen, Y. Zhou, F. Zhang, M. Xu, X. Liu and Z. Li, "Graph Optimization based Visual SLAM fusing KeyPoints and Markers," in *2020 39th Chinese Control Conference (CCC)*, Shenyang, China, 2020.

Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis[1]

I Vladimir Šulžik

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “Optimization for Sensory Deprived SLAM for Robotic Miner”, supervised by Asko Ristolainen.
   1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.
   1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.
2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.
3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

03.05.2023

1 The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.

Appendix 2 – First test results for line track and configurations without side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.0994	0.5078	0.0760	0.2326	0.2238	0.4930	0.1880	0.2594	0.2227	0.4376	0.1954	0.3229
S2	0.0990	1.2445	0.0757	1.1502	0.2272	1.2826	0.1897	1.1807	0.2249	0.9506	0.1964	0.8558
S3	0.0978	0.0942	0.0745	1.0798	0.2273	**0.1193**	0.1904	**0.0757**	0.2250	**0.1450**	0.1967	**0.0929**
S4	0.0998	0.4865	0.0765	0.1310	0.2244	0.4440	0.1883	0.1030	0.2233	0.3102	0.1957	0.0935
S5	0.0977	0.5977	0.0745	0.2274	0.2264	0.5378	0.1890	0.1436	0.2248	0.4442	0.1963	0.2148
S6	0.0995	**0.0888**	0.0760	**0.0757**	0.2281	0.1350	0.1908	0.0953	0.2258	0.1485	0.1971	0.0993
S7	0.0978	0.5017	0.0747	0.2657	0.2261	0.4777	0.1890	0.2663	0.2240	0.4776	0.1960	0.2968
S8	0.0993	0.4849	0.0760	0.1492	0.2251	0.4406	0.1887	0.1342	0.2238	0.3298	0.1961	0.2142

The best result is highlighted with bold font. The odometry is denoted in this and all the following tables as odom.

Appendix 3 – First test results for line track and configurations with side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.0986	0.4894	0.0754	0.1764	0.2264	0.4365	0.1886	0.1166	0.2239	0.3310	0.1960	0.1094
S2	0.0993	0.4502	0.0760	0.1608	0.2251	0.3948	0.1888	0.1019	0.2237	0.2810	0.1960	0.1291
S3	0.0959	0.4101	0.0747	0.1601	0.2249	0.3567	0.1884	0.1152	0.2142	0.3304	0.1955	0.1119
S4	0.0995	0.4552	0.0762	0.1891	0.2266	0.3987	0.1877	0.1448	0.2243	0.3771	0.1959	0.1403
S5	0.0965	0.3897	0.0753	0.1660	0.2220	0.3390	0.1869	0.1036	0.2217	0.3037	0.1948	0.1088
S6	0.0971	**0.1298**	0.0758	**0.1279**	0.2223	**0.1128**	0.1881	**0.0997**	0.2220	**0.1338**	0.1949	0.1124
S7	0.0977	0.3896	0.0758	0.1616	0.2251	0.3421	0.1897	0.1081	0.2244	0.3159	0.1966	**0.0969**
S8	0.0969	0.3963	0.0756	0.1708	0.2212	0.3507	0.1862	0.1152	0.2216	0.3198	0.1949	0.1132

The best result is highlighted with bold font.

Appendix 4 – First test results for U-shape track and configurations without side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.2530	0.2929	0.1584	0.2166	0.3277	0.6140	0.2028	**0.2157**	0.2903	0.4963	0.2557	0.2272
S2	0.2523	**0.2703**	0.1598	**0.1868**	0.3273	0.7200	0.2038	0.2793	0.2902	0.6679	0.2541	0.2829
S3	0.2518	0.3649	0.1618	0.2330	0.3270	**0.4081**	0.2021	0.2675	0.2896	**0.3202**	0.2534	0.2485
S4	0.2516	0.7573	0.1612	0.2795	0.3266	0.6329	0.2027	0.2206	0.2891	0.5562	0.2524	**0.2168**
S5	0.2518	0.3829	0.1609	0.2439	0.3265	0.4204	0.2030	0.2903	0.2894	0.4050	0.2528	0.3035
S6	0.2520	0.3191	0.1605	0.2532	0.3275	0.4573	0.2045	0.3239	0.2902	0.3922	0.2543	0.3112
S7	0.2516	1.0989	0.1616	0.5037	0.3266	0.6789	0.2017	0.2523	0.2892	0.6133	0.2531	0.2770
S8	0.2520	0.3083	0.1605	0.2394	0.3269	0.5781	0.2051	0.2554	0.2902	0.5031	0.2535	0.2543

The best result is highlighted with bold font.

Appendix 5 – First test results for U-shape track and configurations with side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.2522	0.2527	0.1607	**0.1615**	0.3257	0.7065	0.2023	0.3131	0.2880	0.5670	0.2528	0.3010
S2	0.2515	0.3930	0.1613	0.2692	0.3268	0.6253	0.2051	0.3480	0.2904	0.5525	0.2550	0.3208
S3	0.2510	0.3109	0.1614	0.2392	0.3260	0.4725	0.2064	**0.2577**	0.2888	0.3850	0.2518	**0.2390**
S4	0.2514	0.4549	0.1613	0.3434	0.3262	0.6685	0.2030	0.3411	0.2895	0.5240	0.2539	0.2715
S5	0.2509	0.2926	0.1612	0.2206	0.3248	0.6788	0.4078	0.6168	0.2890	0.5115	0.2543	0.2746
S6	0.2511	0.2834	0.1612	0.2080	0.3257	**0.4354**	0.2020	0.2922	0.2886	**0.3342**	0.2520	0.2397
S7	0.2516	**0.2425**	0.4305	0.3851	0.3266	0.6345	0.2049	0.2620	0.2894	0.5982	0.2531	0.2582
S8	0.2505	0.3510	0.1616	0.2602	0.3264	0.6578	0.2017	0.2869	0.2888	0.6443	0.2526	0.3201

The best result is highlighted with bold font.

Appendix 6 – First test results for circle track and configurations without side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.3935	0.5483	0.2520	0.4576	0.4600	0.3738	0.2816	0.3494	0.4831	0.4133	0.2612	0.3880
S2	0.3945	0.4669	0.2520	0.3548	0.4599	0.9195	0.2809	0.6747	0.4831	0.8283	0.2606	0.5868
S3	0.3950	0.5431	0.2512	0.4495	0.4574	0.5916	0.2790	0.3086	0.4825	0.5262	0.2599	0.2409
S4	0.3929	0.5190	0.2519	0.4224	0.4565	**0.3649**	0.2790	**0.2600**	0.4815	**0.2831**	0.2599	**0.2069**
S5	0.3959	0.4822	0.2515	0.3729	0.4605	0.4429	0.2805	0.4220	0.4846	0.4906	0.2604	0.4555
S6	0.3947	0.6164	0.2513	0.5360	0.4584	0.7015	0.2786	0.3711	0.4832	0.6192	0.2595	0.2951
S7	0.3945	**0.4344**	0.2522	**0.3109**	0.4594	0.3967	0.2810	0.3728	0.4835	0.4384	0.2610	0.4363
S8	0.3945	0.6507	0.2524	0.5758	0.4578	0.5772	0.2800	0.3695	0.4820	0.4446	0.2602	0.2709

The best result is highlighted with bold font.

Appendix 7 – First test results for circle track and configurations with side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.3934	0.6130	0.2520	0.5334	0.4609	0.5104	0.2821	0.3611	0.4844	0.4000	0.2617	0.2769
S2	0.3941	0.6586	0.2521	0.5848	0.4582	**0.5068**	0.2807	**0.3326**	0.4826	**0.3064**	0.2610	0.2672
S3	0.3858	0.4363	0.2567	0.3256	0.4558	0.8471	0.2864	0.5543	0.4800	0.6433	0.2664	0.3923
S4	0.3947	**0.4218**	0.2520	**0.2926**	0.4588	0.5787	0.2807	0.3783	0.4834	0.4642	0.2610	0.2910
S5	0.3865	0.4765	0.2569	0.3772	0.4545	0.7535	0.2856	0.4921	0.4803	0.5816	0.7060	1.0216
S6	0.3865	0.4494	0.2540	0.3422	0.4527	0.7181	0.2819	0.3625	0.4776	0.5604	0.2630	**0.2496**
S7	0.3897	0.4997	0.2537	0.4027	0.4558	0.7925	0.2826	0.4820	0.4803	0.6558	0.2629	0.3692
S8	0.3866	0.9743	0.2539	0.9297	0.4529	0.8127	0.2819	0.5421	0.4793	0.6465	0.2634	0.4093

The best result is highlighted with bold font.

Appendix 8 – First test results for 8-shape track and configurations without side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.6296	0.6916	0.5116	0.5863	0.6136	0.4738	0.4630	0.3576	0.6980	0.4536	0.5661	0.3811
S2	0.6310	0.5490	0.5116	0.4062	0.6135	0.4662	0.4580	0.3701	0.7046	0.4517	0.5180	0.4110
S3	0.6163	0.5684	0.4867	0.4245	0.6056	0.5375	0.4456	0.5195	0.6956	0.4669	0.5065	0.4658
S4	0.6123	0.6448	0.4860	0.5264	0.6050	0.5126	0.4483	0.3125	0.6954	0.5737	0.5083	0.3223
S5	0.6184	**0.4927**	0.4886	**0.3148**	0.6066	0.4480	0.4457	0.4359	0.6960	0.4714	0.5064	0.4374
S6	0.6154	0.5891	0.4871	0.4535	0.6065	0.4872	0.4473	0.2638	0.6958	0.5607	0.5068	**0.2950**
S7	0.6316	0.5753	0.5128	0.4416	0.6152	**0.2883**	0.4623	**0.2514**	0.7033	**0.3709**	0.5166	0.3646
S8	0.6177	0.8422	0.4884	0.7525	0.6073	0.4480	0.4490	0.3791	0.6965	0.4309	0.5089	0.3646

The best result is highlighted with bold font.

Appendix 9 – First test results for 8-shape track and configurations with side whiskers

	1. loop				**2. loop**				**3. loop**			
Whiskers configuration	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]	Odom error [m]	SLAM error [m]	Centralized odom error [m]	Centralized SLAM error [m]
S1	0.6328	0.7012	0.5128	0.5952	0.6139	0.4829	0.4633	0.3891	0.7009	**0.3599**	0.5171	0.3069
S2	0.6306	0.6851	0.5124	0.5782	0.6105	0.3823	0.4594	0.3605	0.7022	0.4320	0.5197	0.3695
S3	0.6099	0.8226	0.4862	0.7355	0.6025	**0.3602**	0.4480	0.3524	0.6920	0.4475	0.5070	0.3827
S4	0.6083	3.0033	1.1897	5.5450	0.6019	0.9588	0.4474	0.3589	0.6930	1.0919	0.5085	0.3875
S5	0.6091	**0.6238**	0.4858	**0.5040**	0.6020	0.3823	0.4487	0.3410	0.6911	0.4201	0.5102	0.3314
S6	0.6043	0.7374	0.4831	0.6419	0.6003	0.4825	0.4474	**0.2696**	0.6869	0.5477	0.5069	**0.2917**
S7	0.6074	0.6573	0.4855	0.5467	0.6011	0.5861	0.4477	0.5336	0.6901	0.6858	0.5077	0.5923
S8	0.6185	2.5198	1.1903	4.7275	0.6067	0.5139	0.4586	0.4409	0.6942	0.5478	0.5148	0.3882

The best result is highlighted with bold font.

Appendix 10 – Paths difference in first test for configurations without side whiskers

	8-shape		**U-shape**		**Circle**		**Line**		**Results**	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
N1	35.01%	8	-70.94%	6	**14.44%**	4	-96.48%	13	-29.49%	7.75
N2	35.90%	6	-130.17%	16	-71.45%	16	-322.59%	16	-122.08%	13.5
N3	32.88%	9	**-10.56%**	**1**	-9.06%	9	**35.56%**	2	**12.20%**	**5.25**
N4	17.50%	14	-92.41%	11	**41.21%**	**1**	-38.94%	6	**-18.16%**	8
N5	32.26%	10	**-39.98%**	5	-1.24%	8	-97.57%	14	-26.63%	9.25
N6	19.42%	13	**-35.12%**	4	-28.15%	12	**34.22%**	3	**-2.41%**	8
N7	**47.26%**	2	-112.08%	14	**9.33%**	5	-113.23%	15	-42.18%	9
N8	**38.14%**	5	-73.37%	7	7.76%	6	-47.38%	9	-18.72%	**6.75**

The best result among all 16 whiskers configurations is highlighted with bold font. Red squares mean that the same configuration with side whiskers showed itself better in this test, green means that better was configuration without side whiskers.

Appendix 11 – Paths difference in first test for configurations with side whiskers

	8-shape		**U-shape**		**Circle**		**Line**		**Results**	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
S1	**48.65%**	**1**	-96.91%	12	**17.43%**	3	-47.79%	10	-19.65%	**6.5**
S2	**38.47%**	4	-90.24%	10	**36.52%**	2	**-25.61%**	4	**-10.22%**	**5**
S3	35.34%	7	**-33.31%**	3	-34.03%	13	-54.22%	11	-21.55%	8.5
S4	-57.56%	16	-81.00%	9	3.96%	7	-68.12%	12	-50.68%	11
S5	**39.21%**	3	-77.01%	8	-21.10%	11	**-37.00%**	5	-23.98%	**6.75**
S6	20.26%	12	**-15.81%**	2	-17.33%	10	**39.71%**	**1**	**6.71%**	**6.25**
S7	0.62%	15	-106.74%	13	-36.54%	15	-40.80%	7	-45.87%	12.5
S8	21.08%	11	-123.09%	15	-34.88%	14	-44.29%	8	-45.29%	12

The best result among all 16 whiskers configurations is highlighted with bold font.

Appendix 12 – Centralized paths difference in first test for configurations without side whiskers

	8-shape		**U-shape**		**Circle**		**Line**		**Results**	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
N1	32.67%	6	**11.11%**	2	-48.57%	12	-65.18%	15	-17.49%	8.75
N2	20.65%	13	-11.35%	11	-125.16%	16	-335.84%	16	-112.92%	14
N3	8.04%	15	**1.94%**	5	**7.31%**	2	**52.77%**	**1**	**17.51%**	**5.75**
N4	**36.58%**	4	**14.09%**	**1**	**20.40%**	**1**	**52.23%**	2	**30.83%**	**2**
N5	13.64%	14	-20.06%	13	-74.91%	15	-9.42%	13	-22.69%	13.75
N6	**41.79%**	2	-22.38%	14	-13.75%	8	**49.64%**	4	**13.82%**	**7**
N7	29.43%	7	-9.44%	10	-67.16%	14	-51.38%	14	-24.64%	11.25
N8	28.36%	9	-0.30%	6	**-4.09%**	5	-9.24%	12	3.68%	8

The best result among all 16 whiskers configurations is highlighted with bold font. Red squares mean that the same configuration with side whiskers showed itself better in this test, green means that better was configuration without side whiskers.

Appendix 13 – Centralized paths difference in first test for configurations with side whiskers

	8-shape		**U-shape**		**Circle**		**Line**		**Results**	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
S1	**40.65%**	3	-19.07%	12	-5.82%	6	**44.18%**	5	**14.99%**	**6.5**
S2	28.91%	8	-25.83%	15	**-2.39%**	4	34.12%	10	8.70%	9.25
S3	24.53%	11	**5.12%**	3	-47.23%	11	42.77%	7	6.29%	8
S4	23.79%	12	-6.92%	8	-11.50%	7	28.40%	11	8.44%	9.5
S5	**35.04%**	5	-8.02%	9	-44.70%	10	44.14%	6	6.62%	7.5
S6	**42.46%**	**1**	**4.89%**	4	**5.09%**	3	42.31%	8	**23.69%**	**4**
S7	-16.66%	16	-2.02%	7	-40.40%	9	**50.73%**	3	-2.09%	8.75
S8	24.60%	10	-26.70%	16	-55.39%	13	41.93%	9	-3.89%	12

The best result among all 16 whiskers configurations is highlighted with bold font.

Appendix 14 – First test results for circle and 8-shape track

	8-shape		**Cirle**		**Results**	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
N1	35.01%	8	**14.44%**	4	**24.73%**	6
N2	35.90%	6	-71.45%	16	-17.78%	11
N3	32.88%	9	-9.06%	9	11.91%	9
N4	17.50%	14	**41.21%**	**1**	**29.36%**	7.5
N5	32.26%	10	-1.24%	8	15.51%	9
N6	19.42%	13	-28.15%	12	-4.36%	12.5
N7	**47.26%**	2	**9.33%**	5	**28.29%**	3.5
N8	**38.14%**	5	7.76%	6	22.95%	5.5
S1	**48.65%**	**1**	**17.43%**	3	**33.04%**	**2**
S2	**38.47%**	4	**36.52%**	2	**37.49%**	**3**
S3	35.34%	7	-34.03%	13	0.66%	10
S4	-57.56%	16	3.96%	7	-26.80%	11.5
S5	**39.21%**	3	-21.10%	11	9.06%	7
S6	20.26%	12	-17.33%	10	1.47%	11
S7	0.62%	15	-36.54%	15	-17.96%	15
S8	21.08%	11	-34.88%	14	-6.90%	12.5

The best result among all 16 whiskers configurations is highlighted with bold font.

Appendix 15 – First test results for line and U-shape track

	U-shape		**Line**		**Results**	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
N1	-70.94%	6	-96.48%	13	-83.71%	9.5
N2	-130.17%	16	-322.59%	16	-226.38%	16
N3	**-10.56%**	**1**	**35.56%**	2	**12.50%**	**1.5**
N4	-92.41%	11	-38.94%	6	-65.67%	8.5
N5	**-39.98%**	5	-97.57%	14	-68.77%	9.5
N6	**-35.12%**	4	**34.22%**	3	**-0.45%**	3.5
N7	-112.08%	14	-113.23%	15	-112.65%	14.5
N8	-73.37%	7	-47.38%	9	-60.38%	8
S1	-96.91%	12	-47.79%	10	-72.35%	11
S2	-90.24%	10	**-25.61%**	4	-57.93%	7
S3	**-33.31%**	3	-54.22%	11	**-43.76%**	7
S4	-81.00%	9	-68.12%	12	-74.56%	10.5
S5	-77.01%	8	**-37.00%**	5	-57.01%	6.5
S6	**-15.81%**	2	**39.71%**	**1**	**11.95%**	**1.5**
S7	-106.74%	13	-40.80%	7	-73.77%	10
S8	-123.09%	15	-44.29%	8	-83.69%	11.5

The best result among all 16 whiskers configurations is highlighted with bold font.

Appendix 16 – Second test results for U-shape track

	Whiskers configuration	**All bottom whiskers with side whiskers (S1)**	**Array (0,2,3,5) whiskers with side whiskers (S2)**	**Array (1,4) whiskers (N3)**	**Array (1,4) whiskers with disabled odd column with side whiskers (S6)**
1. test	Odom error [m]	0.47294	0.48661	0.48819	0.48964
	SLAM error [m]	0.55453	0.50245	0.60512	0.59513
	Difference in %	-17.25%	-3.26%	-23.95%	-21.55%
	Centralized odom error [m]	0.33246	0.34164	0.34263	0.34384
	Centralized SLAM error [m]	0.29284	0.24144	0.32889	0.31749
	Difference in %	11.92%	29.33%	4.01%	7.66%
2. test	Odom error [m]	0.48942	0.48981	0.48962	0.48906
	SLAM error [m]	0.62289	0.59063	0.56909	0.67566
	Difference in %	-27.27%	-20.58%	-16.23%	-38.16%
	Centralized odom error [m]	0.34379	0.34406	0.34367	0.34316
	Centralized SLAM error [m]	0.34324	0.31249	0.30779	0.31482
	Difference in %	0.16%	9.18%	10.44%	8.26%
3. test	Odom error [m]	0.48808	0.48533	0.49011	0.48838
	SLAM error [m]	0.61438	0.60441	0.47782	0.52351
	Difference in %	-25.88%	-24.54%	2.51%	-7.19%
	Centralized odom error [m]	0.34351	0.34168	0.34312	0.34345
	Centralized SLAM error [m]	0.30276	0.28887	0.28630	0.23975
	Difference in %	11.86%	15.46%	16.56%	30.19%

	Whiskers configuration	All bottom whiskers with side whiskers (S1)	Array (0,2,3,5) whiskers with side whiskers (S2)	Array (1,4) whiskers (N3)	Array (1,4) whiskers with disabled odd column with side whiskers (S6)
4. test	Odom error [m]	0.48694	0.48865	0.48843	0.48916
	SLAM error [m]	0.63799	0.59604	0.61053	0.53994
	Difference in %	-31.02%	-21.98%	-25.00%	-10.38%
	Centralized odom error [m]	0.34202	0.34366	0.34350	0.34430
	Centralized SLAM error [m]	0.33771	0.29792	0.33898	0.31372
	Difference in %	1.26%	13.31%	1.32%	8.88%
5. test	Odom error [m]	0.48697	0.48843	0.48858	0.48902
	SLAM error [m]	0.54683	0.52344	0.55546	0.62588
	Difference in %	-12.29%	-7.17%	-13.69%	-27.99%
	Centralized odom error [m]	0.34141	0.34349	0.34372	0.34394
	Centralized SLAM error [m]	0.29181	0.24107	0.31808	0.31867
	Difference in %	14.53%	29.82%	7.46%	7.35%
Results	Average difference in %	-22.74%	-15.50%	-15.27%	-21.05%
	Average difference in % for centralized paths	7.95%	19.42%	7.96%	12.47%

Appendix 17 – Second test results for circle track

	Whiskers configuration	**All bottom whiskers with side whiskers (S1)**	**Array (0,2,3,5) whiskers with side whiskers (S2)**	**Array (1,4) whiskers (N3)**	**Array (1,4) whiskers with disabled odd column with side whiskers (S6)**
1. test	Odom error [m]	0.58423	0.58483	0.58573	0.58505
	SLAM error [m]	0.41765	0.48480	0.66545	0.46645
	Difference in %	28.51%	17.10%	-13.61%	20.27%
	Centralized odom error [m]	0.38028	0.38071	0.37959	0.37977
	Centralized SLAM error [m]	0.33096	0.36341	0.43018	0.34537
	Difference in %	12.97%	4.55%	-13.33%	9.06%
2. test	Odom error [m]	0.57900	0.58342	0.58494	0.58072
	SLAM error [m]	0.40765	0.40367	0.68298	0.45675
	Difference in %	29.59%	30.81%	-16.76%	21.35%
	Centralized odom error [m]	0.37877	0.37971	0.37951	0.37821
	Centralized SLAM error [m]	0.30328	0.31300	0.47363	0.33943
	Difference in %	19.93%	17.57%	-24.80%	10.25%
3. test	Odom error [m]	0.58583	0.58249	0.58458	0.58783
	SLAM error [m]	0.38258	0.47478	0.64001	0.45865
	Difference in %	34.69%	18.49%	-9.48%	21.98%
	Centralized odom error [m]	0.38176	0.37987	0.37862	0.38117
	Centralized SLAM error [m]	0.27400	0.36694	0.44358	0.32558
	Difference in %	28.23%	3.40%	-17.16%	14.58%

	Whiskers configuration	**All bottom whiskers with side whiskers (S1)**	**Array (0,2,3,5) whiskers with side whiskers (S2)**	**Array (1,4) whiskers (N3)**	**Array (1,4) whiskers with disabled odd column with side whiskers (S6)**
	Odom error [m]	0.58327	0.58283	0.58672	0.58485
	SLAM error [m]	0.43892	0.42911	0.62739	0.47059
	Difference in %	24.75%	26.38%	-6.93%	19.54%
	Centralized odom error [m]	0.38009	0.38016	0.37941	0.37935
	Centralized SLAM error [m]	0.33710	0.31085	0.43498	0.34489
4. test	Difference in %	11.31%	18.23%	-14.64%	9.08%
	Odom error [m]	0.58272	0.58127	0.58568	0.58557
	SLAM error [m]	0.45391	0.45843	0.60644	0.54939
	Difference in %	22.10%	21.13%	-3.54%	6.18%
	Centralized odom error [m]	0.38022	0.38086	0.37925	0.37980
	Centralized SLAM error [m]	0.32976	0.34858	0.41481	0.42112
5. test	Difference in %	13.27%	8.47%	-9.37%	-10.88%
	Average difference in %	27.93%	22.78%	-10.07%	17.86%
Results	Average difference in % for centralized paths	17.14%	10.44%	-15.86%	6.42%

Appendix 18 – Second test results for 8-shape track

	Whiskers configuration	All bottom whiskers with side whiskers (S1)	Array (0,2,3,5) whiskers with side whiskers (S2)	Array (1,4) whiskers (N3)	Array (1,4) whiskers with disabled odd column with side whiskers (S6)
	Odom error [m]	0.74181	0.74265	0.74006	0.74409
	SLAM error [m]	0.37291	0.54599	0.59461	0.47595
	Difference in %	49.73%	26.48%	19.65%	36.04%
	Centralized odom error [m]	0.56299	0.56347	0.56116	0.56507
	Centralized SLAM error [m]	0.28523	0.50773	0.39220	0.45701
1. test	Difference in %	49.34%	9.89%	30.11%	19.12%
	Odom error [m]	0.74253	0.74074	0.74270	0.74164
	SLAM error [m]	0.43833	0.40445	0.64836	0.40909
	Difference in %	40.97%	45.40%	12.70%	44.84%
	Centralized odom error [m]	0.56336	0.56234	0.56483	0.56345
	Centralized SLAM error [m]	0.34416	0.30952	0.64581	0.34510
2. test	Difference in %	38.91%	44.96%	-14.34%	38.75%
	Odom error [m]	0.74353	0.74063	0.74055	0.74317
	SLAM error [m]	0.41968	0.43785	0.68037	0.45640
	Difference in %	43.56%	40.88%	8.13%	38.59%
	Centralized odom error [m]	0.56459	0.56152	0.56379	0.56410
	Centralized SLAM error [m]	0.35016	0.34341	0.51294	0.38778
3. test	Difference in %	37.98%	38.84%	9.02%	31.26%

	Whiskers configuration	**All bottom whiskers with side whiskers (S1)**	**Array (0,2,3,5) whiskers with side whiskers (S2)**	**Array (1,4) whiskers (N3)**	**Array (1,4) whiskers with disabled odd column with side whiskers (S6)**
4. test	Odom error [m]	0.74426	0.74361	0.74207	0.75173
	SLAM error [m]	0.41400	0.46761	0.74071	0.43433
	Difference in %	44.37%	37.12%	0.18%	42.22%
	Centralized odom error [m]	0.56563	0.56488	0.56367	0.56676
	Centralized SLAM error [m]	0.30625	0.33294	0.51785	0.41874
	Difference in %	45.86%	41.06%	8.13%	26.12%
5. test	Odom error [m]	0.74365	0.74327	0.73914	0.74170
	SLAM error [m]	0.53428	0.38379	0.67091	0.42024
	Difference in %	28.15%	48.36%	9.23%	43.34%
	Centralized odom error [m]	0.56499	0.56434	0.56349	0.56258
	Centralized SLAM error [m]	0.43045	0.31752	0.44847	0.38247
	Difference in %	23.81%	43.74%	20.41%	32.02%
Results	Average difference in %	41.36%	39.65%	9.98%	41.01%
	Average difference in % for centralized paths	39.18%	35.70%	10.67%	29.45%

Appendix 19 – Second test results for line track

	Whiskers configuration	**All bottom whiskers with side whiskers (S1)**	**Array (0,2,3,5) whiskers with side whiskers (S2)**	**Array (1,4) whiskers (N3)**	**Array (1,4) whiskers with disabled odd column with side whiskers (S6)**
1. test	Odom error [m]	0.37090	0.36586	0.36816	0.36647
	SLAM error [m]	0.59014	0.50038	0.28372	0.49621
	Difference in %	-59.11%	-36.77%	22.94%	-35.40%
	Centralized odom error [m]	0.35088	0.34778	0.34918	0.34862
	Centralized SLAM error [m]	0.16785	0.16633	0.26635	0.18337
	Difference in %	52.16%	52.17%	23.72%	47.40%
2. test	Odom error [m]	0.36995	0.36734	0.36928	0.37116
	SLAM error [m]	0.49568	0.52792	0.26206	0.48849
	Difference in %	-33.99%	-43.71%	29.03%	-31.61%
	Centralized odom error [m]	0.35036	0.34881	0.35005	0.35165
	Centralized SLAM error [m]	0.17176	0.21236	0.24577	0.18375
	Difference in %	50.98%	39.12%	29.79%	47.75%
3. test	Odom error [m]	0.36578	0.36969	0.36856	0.36404
	SLAM error [m]	0.52716	0.43154	0.26660	0.54069
	Difference in %	-44.12%	-16.73%	27.66%	-48.52%
	Centralized odom error [m]	0.34717	0.34984	0.34915	0.34700
	Centralized SLAM error [m]	0.21981	0.18125	0.25325	0.19584
	Difference in %	36.69%	48.19%	27.47%	43.56%

	Whiskers configuration	**All bottom whiskers with side whiskers (S1)**	**Array (0,2,3,5) whiskers with side whiskers (S2)**	**Array (1,4) whiskers (N3)**	**Array (1,4) whiskers with disabled odd column with side whiskers (S6)**
	Odom error [m]	0.36836	0.36996	0.36986	0.36914
	SLAM error [m]	0.55696	0.42537	0.27898	0.47624
	Difference in %	-51.20%	-14.98%	24.57%	-29.01%
	Centralized odom error [m]	0.34883	0.35071	0.35026	0.34942
	Centralized SLAM error [m]	0.16802	0.19230	0.26241	0.21978
4. test	Difference in %	51.83%	45.17%	25.08%	37.10%
	Odom error [m]	0.36785	0.36812	0.36872	0.36810
	SLAM error [m]	0.56075	0.47783	0.24012	0.61170
	Difference in %	-52.44%	-29.80%	34.88%	-66.18%
	Centralized odom error [m]	0.34855	0.34891	0.34981	0.34893
	Centralized SLAM error [m]	0.16210	0.16677	0.16837	0.18558
5. test	Difference in %	53.49%	52.20%	51.87%	46.82%
	Average difference in %	-48.17%	-28.40%	27.82%	-42.15%
Results	Average difference in % for centralized paths	49.03%	47.37%	31.59%	44.53%

Appendix 20 – Paths difference in second tests

	8-shape		U-shape		Circle		Line		Results	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
All bottom whiskers with side whiskers (S1)	41.36%	1	-22.74%	4	27.93%	1	-48.17%	4	-0.41%	2.5
Array (0,2,3,5) whiskers with side whiskers (S2)	39.65%	3	-15.50%	2	22.78%	2	-28.40%	2	4.63%	2.25
Array (1,4) whiskers (N3)	9.98%	4	-15.27%	1	-10.07%	4	27.82%	1	3.11%	2.5
Array (1,4) whiskers with disabled odd column with side whiskers (S6)	41.01%	2	-21.05%	3	17.86%	3	-42.15%	3	-1.08%	2.75

Appendix 21 – Centralized paths difference in second tests

	8-shape		U-shape		Circle		Line		Results	
Whiskers configuration	Difference in %	Place	Difference in %	Place	Difference in %	Place	Difference in %	Place	Average difference in %	Average place
All bottom whiskers with side whiskers (S1)	39.18%	1	7.95%	4	17.14%	1	49.03%	1	28.32%	1.75
Array (0,2,3,5) whiskers with side whiskers (S2)	35.70%	2	19.42%	1	10.44%	2	47.37%	2	28.23%	1.75
Array (1,4) whiskers (N3)	10.67%	4	7.96%	3	-15.86%	4	31.59%	4	8.59%	3.75
Array (1,4) whiskers with disabled odd column with side whiskers (S6)	29.45%	3	12.47%	2	6.42%	3	44.53%	3	23.22%	2.75