Performance Evaluation of ROS Local Trajectory Planning Algorithms to Social Navigation

Fagner de Assis Moura Pimentel
Dpto. de pós Graduação em Engenharia Elétrica
Centro Universitário FEI
São Bernardo do Campo - SP, Brazil
fpimentel@fei.edu.br

Plinio Thomaz Aquino-Jr
Dpto. de pós Graduação em Engenharia Elétrica
Centro Universitário FEI
São Bernardo do Campo - SP, Brazil
plinio.aquino@fei.edu.br

***Abstract*—Accuracy and safety are necessary characteristics in social navigation and still constitute a challenge. The ROS Navigation Stack (RNS) allows the variation of local path planning methods through plugins for navigation. This paper brings you the comparison of those methods, which are directly connected with the safety and naturalness of the robot. Therefore, four different methods were compared by varying the sensors and the simulated environment. A thousand experiments were performed for each combination using the standard parameters of each method in a total of 24000 experiments. This paper concluded that the Elastic Band (EBand) method presents more safety and accuracy than the Dynamic window approach (DWA), method commonly used in several robots that participate in RoboCup@home, so it is more suitable for social navigation - reaching 90% accuracy in some cases and collision rate below 5%.**

***Keywords*-ROS; path planning; Performance evaluation;**

I. Introduction

A mobile robot should be able to navigate freely in its environment. It should address common issues regarding autonomous navigation such as world model, localization, motion planning, and motion control. However, the coexistence of robots and humans in the same environment adds some news dimensions to mobility, like comfort and sociability. For ordinary mobile robots, mobility can generally be summarized as the calculation of motions that are both safe and optimal, such as the distance traveled that should preferably be as small as possible. People are not common obstacles, because there is a set of social and cultural rules that dictate how people should move, such as always approaching a person in the front or making a move on the left. According to Kruse et al. [7], the challenges of social navigation involve naturalness, comfort and sociability. Naturalness is related to similarities between robots and humans in the pattern of low-level behavior; comfort refers to suppression of annoyance or stress to the human in interactions with the robot; while sociability deals with the robot's suitability to high-level socio-cultural patterns.

In this paper, we will focus on the criteria of accuracy and safety as components to naturalness of social navigation as part of the development and implementation of a social navigation model in a service robot. In future work, we will focus on criteria of comfort and sociability. This work has motivation by the difficulties encountered to treat the navigation of an autonomous service robot in a safe and natural way for humans that interact in the same context of use. People have context-driven behavior that includes their experience over time, as well as characteristics of the physical, social and cultural environment [8]. As the robot is recognized as an element of the environment that has autonomous behavior, it is expected to present adherent behavior with the local context [11].

This paper aims to increase human receptivity by selecting a optimal planner in a set of those used in the ROS Navigation Stack (RNS) to make the robot navigation more secure and natural for the human being, since the naturalness in this behavior of the robot is one of the pillars for social navigation according to Kruse et al. [7]. To do this, some methods of local planning were tested, also varying sensors and environments.

In the next section, the following topics will be presented: In section II the review on navigation and how it is used in the main test environment of this research: RoboCup@home. In section III will be presented the materials and methods used in this study, such as testing environment, robots, software tools and applied methodology. In section IV we will shown how and what tests were performed. In section V the results are presented. Finally, in section VI presents conclusions and future works.

II. Background

In order to verify the current scenario of navigation in mobile robots within the context of the RoboCup, a review of literature was carried out. in order to answer the following questions: (Q1) Which direction systems are most used, (Q2) which sensors are most used, (Q3) which local planning algorithms are most used, (Q4) in which environments the robots were tested, (Q5) what is the test methodology used and (Q6) how the main research groups address the question

978-1-7281-4268-5/19/$31.00 ©2019 IEEE
DOI 10.1109/LARS-SBR-WRE48964.2019.00035

Table I: Information about navigation of the participating teams on RoboCup@home

TEAM	DRIVE	SENSORS	LOCAL PATH PLANNING
aisl-tut	OMNI	L, IMU	randomizer path planner [2]
alle	DIFF	L, S, IMU	Trajectory Rollout [5]
aupair	OMNI	L, S, G	Trajectory Rollout [5]
cmpepperbot	OMNI	L, S, G	not informed
duckers	OMNI	L, S, IMU	DWA [4]
erasers	OMNI	L, S, IMU	not informed
gentlebots	OMNI	L, S, G	Trajectory Rollout [5]
golem	DIFF	L, S, B	Trajectory Rollout [5]
happy	DIFF	L	DWA [4]
Hibikino	OMNI	L, IMU	Trajectory Rollout [5]
homer	DIFF	L	zelinskys path transform [16]
jskathome	OMNI	L, IMU	jsk_maps [6]
kamerider	OMNI	L, S, G	Trajectory Rollout [5]
liu	OMNI	L, S, G	octomap e cartographer
lyontech	OMNI	L, S, G	not informed
oit_trial	DIFF	L	not informed
pumas	OMNI	L	not informed
robofei	OMNI	L	Trajectory Rollout [5]
rt_lions	DIFF	L	miracenter e cognidriver
socrob	OMNI	L, X	DWA [4]
spqrel	OMNI	L, S, G	SPQReL [12] e Strands [13]
northeastern	OMNI	L, IMU	Ros karto [9]
techunited	OMNI	L, IMU	Modified DWA [15]
tinker	OMNI	L	Trajectory Rollout [5]
tobi	OMNI	L, S, G	Trajectory Rollout [5]
tritons	OMNI	L, IMU	Omni mapper [14]
uchile	OMNI	L, S, G	Trajectory Rollout [5]
unsw	OMNI	L, IMU	not informed
ut_austin_villa	OMNI	L, IMU	not informed
uts_unleashed	OMNI	L, S, G	not informed
uva	OMNI	L, S, G	coastal navigation mobile [10]
walkingmachine	OMNI	L	DWA [4]
wrighteagle	DIFF	L	VFH* [3]

L = Laser, S = Sonar, G = Gyrosensor, X = 3D depth camera
B = Bumper, IMU = Inertial measurement unit

about naturalness of navigation in robots for the human being today.

The inclusion criteria were works that deal with navigation and/or navigation naturalness. The exclusion criteria were works that did not use the ROS framework. The researcher was conducted in TDPs submitted to RoboCup@home in the last 2 years (2017-2018). The RoboCup is an international initiative aimed at developing the state-of-the-art of intelligent mobile robots, while @home it's a RoboCup league that aims to develop service robots and assistive devices applied to home environments.

We reviewed 47 articles from 33 teams available on the RoboCup@home [1] wiki. The teams, steering systems of the robots, sensors used in navigation and navigation methods used, can be observed in the table I.

From these articles, it was possible to observe that the navigation method most used was the ROS Navigation Stack with the local DWA planner. Most of the robots use omni directional steering system and laser sensor as main source for localization, and encoders in the wheels.

[1] https://github.com/RoboCupAtHome/AtHomeCommunityWiki/wiki/Team-Description-Papers

III. Materials and Methods

This session presents the materials and methods used in this study. The sub-section III-A presents the infrastructure of the RoboFEI@Home laboratory on FEI University Center, where this research was carried out. In the sub-section III-B the main software tools used are presented. Finally, in the sub-section III-C the methodology applied in this work is verified.

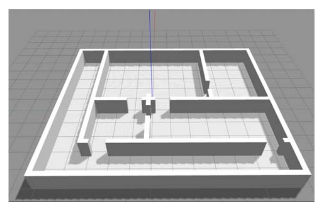
A. Infrastructure

The infrastructure of the project presents the materials used for the simulations, as robots and URDF (Unified Robot Description Format) models.

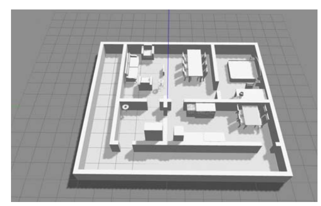
The simulated version of the robot youbot robot [2] was used in the tests of this work. It has 4 omni-directional wheels capable of moving the robot in any direction on a 2D plane and sensors used for 2D and 3D mapping.

For simulation and testing, a desktop computer was used with 8 Intel® Core™ i7-3770 CPU, 12GB RAM, a GeForce GT640 graphics card and Ubuntu 16.04 LTS 64bits operating system.

During the simulator tests, the environments [3] (Figure 1) were modeled in URDF by the RoboFEI lab replicating an actual competition environment used in one of the RoboCup@home editions [1].



(a) Environment 1



(b) Environment 2

Figure 1: The simulated environment with and without furniture. Modeled on RoboFEI lab.

B. Tools

1) ROS: The main tool used in this project is the ROS platform (Robotic Operating System). ROS is a framework that assists in writing software for robots. It has a collection of tools, libraries and conventions to simplify the creation of programs for complex and robust tasks in robots, independent of the platform.

2) Gazebo: The next tool presented is the Gazebo [4]. It is a 3D dynamic simulator with the ability to simulate efficiently and correctly robot populations in complex internal and external environments. Similar to game engines, Gazebo offers physics simulation with a high degree of fidelity, a set of sensors and interfaces for users and programs.

[2] https://github.com/youbot/youbot_description
[3] https://gitlab.com/robofei/at-home/worlds
[4] http://gazebosim.org/

3) Navigation packages: The following packages are part of the ROS navigation stack [5]. They are used for mapping, location and navigation.

Gmapping is a package capable of providing simultaneous mapping and location based on a laser sensor.

The map_server is a ROS package that provides information about the map generated by gmapping and can be used by the robot for navigation.

The amcl is a probabilistic location system for a mobile robot in a 2D environment. It implements the Monte Carlo adaptive localization method that uses a particle filter to track the robot's positions on a known map.

provides the implementation of an action that given a position in the world (goal), the robot will try to achieve this position with a mobile base. The move_base node combines global and local planning to accomplish the navigation task.

Planners:

- **Base Local Planner and DWA Local Planner:** These planners provide control to a mobile base. Using a map, the planner creates kinematic trajectories for a robot to move from a starting point to an end point. The function of the controller is to determine the speeds that will be applied to the mobile base. The DWA differs from Base by the way it controls discretized space. While Base checks all future simulation states, the DWA checks only the spaces immediately after the current state.
- **EBand and TEB:** EBand (Elastic Band) calculates an elastic band within the local costmap and tries to follow the generated path connecting the central points of the band using several heuristics. The Timed Elastic Band (TEB) uses the same EBand principle, however it focuses on temporal optimization. TEB also works with cost function minimization rather than force application.

4) Others Packages: Other packages used are either util_ros [6] and ira_laser_tools [7]. The ros_util has two nodes. The first is the util_move_base, it is responsible for verifying that the path generated by move_base goes over some costmap. The second is util_gazebo_collision, it is responsible for checking if the robot has collided with some obstacle in the environment. The ira_laser_tools also has two nodes. The first is laserscan_virtualizer, responsible for transforming a cloud of dots from a 3D sensor into a series of 2D sensors. The second is laserscan_multi_merger, which combines two or more 2D sensors into a single laser sensor. Finally, the experiments package executes a script that follows the test methodology of this work: (1) defines initial and final positions randomly and within the area of movement of the robot; (2) take the global plane of the move_base and validate it using the util_move_base; (3) restart the environment and the robot; (4) clean costmaps; (5) initiates experiment by sending a new command to the move_base; (6) Save the experiment data when the robot arrives at the destination or is interrupted.

[5]http://wiki.ros.org/navigation

[6]https://gitlab.com/fagnerpimentel/util_ros

[7]https://github.com/iralabdisco/ira_laser_tools

C. Methodology

The development methodology of this work aims to evaluate safety and accuracy criteria in local navigation planning methods using different combinations of sensors and in different environments. The job here is to select local navigation as close as possible to the optimal planned navigation and to be as natural as possible for the human being. With this it is possible to approach a more efficient, effective, safe and comfortable navigation.

The following variables are analyzed in this study in an objective way.

- **Success rate:** Determines the number of hits, where the navigation was successfully completed. It is given by the formula:

$$Sr = s/ex\_max * 100 \tag{1}$$

  where s is the number of experiments completed successfully and ex_max is the maximum number of experiments performed.
- **Spatial coefficient:** determines the distance traveled from the planned navigation space. Given by formula:

$$SpC = 1 - (s_e - s_{min})/(s_{max} - s_{min}) \tag{2}$$

  where s_e is the space traveled by the robot, s_{min} is the minimum space between the initial and the final position, s_{max} is the maximum space the robot can navigate in this experiment, being defined as five times the s_{min}.
- **Temporal coefficient:** Determines how close the execution time is to the estimated time to perform the navigation. Given by formula:

$$TeC = 1 - (t_e - t_{min})/(t_{max} - t_{min}) \tag{3}$$

  where t_e is the time used by the robot in the experiment, t_{min} is the minimum time needed by the robot to travel from the starting point to the end in a straight line, given the maximum speed of the robot, t_{max} is the maximum time that the robot can navigate in this experiment, being defined as five times the t_{min}.
- **Smooth coefficient:** Determines how smooth the trajectory performed by the local planner is. It is used with a measure to evaluate the naturalness of the robot. which is given by the average of the differences of the angles of each line that create the trajectory.

$$SmC = 1 - \frac{\sum_{i=1}^{n}(arctan(y_i - y_{i-1}, x_i - x_{i-1})/\pi)}{n-1} \tag{4}$$

  where n is the number of points present in the executed route and x and y are the coordinates of the points.

IV. TESTS

Some scenarios in simulated environments were prepared to carry out the tests of this project. The following presents the variations made on sensors, planning methods and scenarios.

The sensors used were a front Laser, a back laser and a front 3D depth camera, they were added progressively in the evaluations. The planners were the Base Planner Local, the DWA Local Planner, the EBand Local Planner, and the TEB Local Planner. The scenarios were:

- **Scenario 1:** Simulated environment without objects - It aims to evaluate navigation in a static environment without objects.
- **Scenario 2:** Simulated Environment with Static Objects - It aims to evaluate navigation with 3D objects that are out of reach of the laser.

1000 experiments were performed for each configuration combination in simulated environments. Each battery of tests took between 0.6 and 1.3 days to be performed, one battery per day, the fastest being performed in scenario 1 (less than one day to complete each battery) and slower in scenario 2 (more than one day to complete each battery). It took around 1 month for the completion of these tests. In total were performed 24000 simulation experiments with random initial and final positions. The main objective of these experiments is to evaluate the robot navigation behavior using the move_base package.

V. RESULTS

From the table II we can see the results of the experiments performed in scenario 1 and 2. It can be observed that the EBand planner obtained the best success rates in scenario 1, being 91.60% using the frontal laser, 87.50% using front and back laser and 90.10% using the 2 lasers and 3D camera, while the TEB had the best success rates in scenario 2, being 38.30% using the frontal laser, 36.50% using front and rear laser and 35.70% using the 2 lasers and 3D camera. The worst results in both scenarios were observed with the DWA planner, with 49.50% using one laser, 27.10 % using two lasers and 29.60% using 2 lasers and the 3D camera for scenario 1 and 0.30% using one laser, 0.40% using two lasers and 0.70% using 2 lasers and 3D camera for scenario 2.

For scenario 1, the EBand had the lowest error rates such as TIME_EXCEEDED, below 4% and COLLISION, below 5.1%. The TEB and Base methods had the lowest ABORTION rates, reaching 0.0% while the worst cases are with the DWA method, reaching 6.7%. The TEB method had the worst result in terms of collision when only the front laser was used (42.0%), however, this rate falls by half when using the back laser and 3D camera and in these cases, the DWA method has the worst cases, reaching 54.4% of collisions. No method presented problems due to the excess of space traveled (SPACE_EXCEEDED). For scenario 2, all methods have problems with the TIME_EXCEEDED rate. This rate error reaches 90% in some cases. Among the success cases,the TEB planner had the best result with values around 36% and the DWA had the worst result, below 1%.

In terms of spatial efficiency, the methods do not present great differences, the best was the DWA and Base planners with a spatial coefficient greater than or equal to 1.0 and the worst was the TEB planner with coefficient around 0.93% for scenario 1 and 0.96 for Scenario 2. These values are explained due to the TEB's characteristic of optimizing the robot's movement to become more similar to the car behavior giving less focus to spatial optimization and smoothing its movement. For this coefficient, it is possible to observe that some values exceed 1.0, which means that the robot walked less than the minimum space necessary to reach the destination. These values can be explained by the tolerance used in the move_base to reach the destination, the 'xy_goal_tolerance' parameter that causes the robot to stop shortly before arriving at the destination.

In terms of time efficiency, the methods presented considerable differences. Given that the DWA has the worst result at an average of 0.15% while the others are closer to each other. The best value observed was the base planner with coefficient around 0.63%

In terms of smooth coefficient, all methods presented close values, around 0.60 and 0.65 for scenario 1, except for the Base method below 0.60, with the best values found in the method around 0.65. For scenario 2, the values fall around 0.1 points, this is due to the presence of more obstacles in the scenario, which forces the planner to make more curves to be able to deviate. In this scenario, the EBand and TEB methods with values around 0.53 continue to have an advantage over the DWA and Base with values around 0.50. In Figure 2 we can see examples of the trajectories traversed by the robot in two experiments, the first with the DWA method in an environment without furniture where it is possible to perceive a less smooth path and second using the EBand method in an environment with where it is possible to perceive a smoother path in the obstacles avoidance.

VI. CONCLUSION

This paper presented a comparative study of local trajectory planning methods using the ROS navigation stack. The comparative study is one of the objectives of implementing social navigation in a service robot, which aims to improve the naturalness behavior in the movement of the robot independently of the sensors or environment in which it is acting. Therefore, variations of methods of local trajectory planning, quantity and types of sensors used and environments were made.

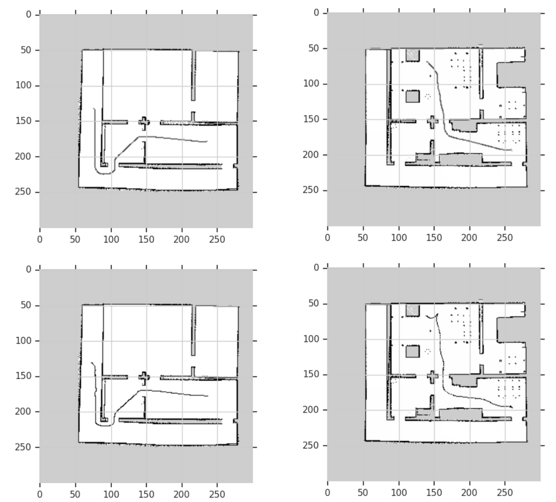
With the results of this work it is possible to conclude that the DWA planner and the Base planner, despite being the most commonly used in the ROS navigation stack and

Table II: Results of experiments varying planners, sensors and scenarios

	Front laser							
	Scenario 1				Scenario 2			
	dwa	eband	teb	base	dwa	eband	teb	base
Space coefficient	1.00	0.99	0.93	1.00	1.10	1.00	0.95	1.01
Time coefficient	0.14	0.53	0.63	0.63	0.14	0.14	0.35	0.26
Smooth coefficient	0.64	0.65	0.61	0.58	0.51	0.54	0.53	0.50
SUCCESS	49.50%	91.60%	41.80%	56.80%	0.30%	28.60%	38.30%	16.90%
SPACE_EXCEEDED	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
TIME_EXCEEDED	20.00%	3.50%	16.00%	14.60%	90.10%	69.80%	61.60%	58.80%
ABORTION	6.70%	1.10%	0.20%	0.50%	9.60%	1.60%	0.10%	0.60%
COLLISION	23.80%	3.80%	42.00%	28.10%	0.00%	0.00%	0.00%	23.70%
	Front and back lasers							
	Scenario 1				Scenario 2			
	dwa	eband	teb	base	dwa	eband	teb	base
Space coefficient	1.00	0.99	0.92	1.00	1.13	1.00	0.96	1.01
Time coefficient	0.15	0.58	0.59	0.62	0.29	0.08	0.28	0.35
Smooth coefficient	0.62	0.65	0.62	0.57	0.49	0.52	0.52	0.51
SUCCESS	27.10%	87.50%	61.10%	47.70%	0.40%	20.70%	36.50%	25.90%
SPACE_EXCEEDED	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
TIME_EXCEEDED	14.80%	4.00%	19.70%	12.20%	91.60%	66.40%	54.10%	72.40%
ABORTION	3.70%	3.40%	0.00%	0.20%	8.00%	5.60%	0.30%	1.70%
COLLISION	54.40%	5.10%	19.20%	39.90%	0.00%	7.30%	9.10%	0.00%
	Front and back lasers and 3D camera							
	Scenario 1				Scenario 2			
	dwa	eband	teb	base	dwa	eband	teb	base
Space coefficient	1.00	1.00	0.93	1.00	1.16	1.00	0.96	1.00
Time coefficient	0.17	0.59	0.60	0.63	0.44	0.07	0.27	0.32
Smooth coefficient	0.63	0.66	0.61	0.57	0.49	0.53	0.53	0.51
SUCCESS	29.60%	90.10%	58.20%	46.50%	0.70%	19.10%	35.70%	24.20%
SPACE_EXCEEDED	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
TIME_EXCEEDED	13.60%	2.80%	20.70%	12.40%	90.70%	62.50%	54.40%	74.80%
ABORTION	5.80%	2.90%	0.00%	0.00%	8.60%	8.40%	0.20%	1.00%
COLLISION	51.00%	4.20%	21.10%	41.10%	0.00%	10.00%	9.70%	0.00%

by several RoboCup@Home participating teams, presented the worst results both in terms of security (collision), as well as natural (smooth coefficient). Although we have not done tests with people until this moment, these two indices suggest that the EBand method is more suitable for social navigation. The best results of the DWA method were in relation to the spatial coefficient index, however all other results had indices below the other methods. Considering that the objective of this work is to detect methods that are closer to a natural and safe navigation behavior, the spatial coefficient indices are not always the most relevant since the smallest path may not be the most natural and comfortable for the human being. It should be noted that the DWA method was not optimized, as none of the methods used in this work were optimized. All methods were used with their standard parameters presented in their respective documentation.

At the end of this work it was possible to identify the EBand method as the most qualified for social navigation among the methods compared in this work. This method was the one that presented more safety characteristics according to the collision index and naturalness according to the coefficient of softness used, being therefore the most suitable for application in this project. It was also possible



(a) Environment 1 with DWA planner

(b) Environment 2 with EBand planner

Figure 2: Example of trajectories calculated by the global planner (top) and executed by the local planners (bottom) in scenarios 1 and 2.

to conclude that EBand methods present good independent results from the sensors used.

As future work, it is expected to test other local planning methods found in the bibliographic review as randomizer path planner and VFH*. It is also expected to optimize the parameters of the local planning methods using reinforcement learning in simulated environment and to implement and optimize the human comfort and sociability of the robot, thus giving continuity to the implementation of a complete social navigation model. We also intend to conduct experiments in simulated environment with specific situations (navigation in environments with obstacles out of laser range, interaction with people interacting with other people or objects and follow situations) Finally, we expect to realize these experiments with people in a real environment, continuing the development of a framework for social navigation.

ACKNOWLEDGMENT

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001

REFERENCES

[1] Plinio T. Aquino-Jr et al. "HERA: Home Environment Robot Assistant". In: *Proceedings of the II Brazilian Humanoid Robot Workshop (BRAHUR) and III Brazilian Workshop on Service Robotics (BRASERO)*. IEEE South Brazil Robotics & Automation Society Chapter. 2019, pp. 68–73. ISBN: 978-85-98123-14-1.

[2] Igi Ardiyanto and Jun Miura. "Real-time navigation using randomized kinodynamic planning with arrival time field". In: *Robotics and Autonomous Systems* 60.12 (2012), pp. 1579–1591.

[3] Johann Borenstein and Yoram Koren. "The vector field histogram-fast obstacle avoidance for mobile robots". In: *IEEE transactions on robotics and automation* 7.3 (1991), pp. 278–288.

[4] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. "The dynamic window approach to collision avoidance". In: *IEEE Robotics & Automation Magazine* 4.1 (1997), pp. 23–33.

[5] Brian P Gerkey and Kurt Konolige. "Planning and control in unstructured terrain". In: *ICRA Workshop on Path Planning on Costmaps*. 2008.

[6] jks. *JKS Maps*. URL: https://github.com/jsk-ros-pkg/jsk_demos/tree/master/jsk_maps.

[7] Thibault Kruse et al. "Human-aware robot navigation: A survey". In: *Robotics and Autonomous Systems* 61.12 (2013), pp. 1726–1743.

[8] Denise Marczal and Plinio Thomaz Aquino Junior. "Behavioural Variables Analysis in Mobile Environments". In: *Design, User Experience, and Usability: Users and Interactions*. Ed. by Aaron Marcus. Cham: Springer International Publishing, 2015, pp. 118–130. ISBN: 978-3-319-20898-5.

[9] ROS. *ROS Karto*. URL: http://wiki.ros.org/karto.

[10] Nicholas Roy et al. "Coastal navigation-mobile robot navigation with uncertainty in dynamic environments". In: *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*. Vol. 1. IEEE. 1999, pp. 35–40.

[11] Thiago Freitas dos Santos et al. "Behavioral persona for human-robot interaction: a study based on pet robot". In: *International Conference on Human-Computer Interaction*. Springer. 2014, pp. 687–696.

[12] spqrel. *SPQReL Navigation*. URL: https://github.com/LCAS/spqrel_navigation/wiki.

[13] strands. *Strands Navigation*. URL: https://github.com/strands-project/strands_navigation.

[14] Alexander JB Trevor, John G Rogers, and Henrik I Christensen. "Omnimapper: A modular multimodal mapping framework". In: *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE. 2014, pp. 1983–1990.

[15] TUE. *cb_base_navigation*. URL: https://github.com/tue-robotics/cb_base_navigation.

[16] Alexander Zelinsky. "Environment exploration and path planning algorithms for a mobile robot using sonar". In: *PhD Thesis, Wollongong University* (1991).