Russian mobile robot Servosila Engineer: designing an optimal integration of an extra laser range finder for SLAM purposes

Neil Alishev, Roman Lavrenov, Yuriy Gerasimov
Intelligent Robotic Systems Laboratory, Higher Institute for Information Technology and Information Systems,
Kazan Federal University, 35 Kremlyovskaya street,
Kazan, 420008, Russian Federation
E-mail: alishev.neil@gmail.com. lavrenov@it.kfu.ru, yurger2009@gmail.com
http://kpfu.ru/robolab.html

Abstract

In our current research we determine an optimal design for integration of Hokuyo UTM-30LX-EW LRF into control system of Russian mobile robot Servosila Engineer. We designed and constructed a special static stand with an option to select an inclination of a scanning beam toward the surface of an environment. RBPF SLAM algorithm was tested to perform localization and mapping. Experiments were conducted in order to determine the best configuration of the LRF position and RBPF SLAM algorithm parameters.

Keywords: Robotics, algorithm, ground mobile robot, SLAM, LRF, ROS, RViz, experiments.

1. Introduction

Sensory information is crucial for operation of any mobile robot. In mobile robotics, sensors are typically used for such tasks as localization, mapping, obstacle avoiding and more. In this research, we tackle the challenge of simultaneous localization and mapping (SLAM) of a Russian crawler-type mobile robot Servosila Engineer[1]. SLAM could be performed for UGVs[2] and UAVs[3] in many ways using different sensors:

- Using single or multiple cameras (visual based SLAM). This type of SLAM algorithm uses monocular or stereo camera as a main source of information about environment.
- Using laser range finder (LRF-based SLAM). LRF SLAM algorithms use reflected laser beam to get the information about environment.
- Using sonar. These SLAM algorithms use reflected sound to get information about environment.

Different SLAM approaches have their own benefits and limitations. Visual-based SLAM does not require special expensive equipment and can be performed using standard monocular or stereo cameras but sometimes lack accuracy, especially in indoor environments with low number of distinctive features[2]. Laser-based SLAM on the other hand is known for its' accuracy and stability but requires special equipment. Sonar-based SLAM can be beneficial in some situations where both previous SLAM approaches fail (e.g. detecting doors and walls made of glass).

In this research LRF-SLAM approach was selected for building a map of the environment and localizing robot within this map. As Russian crawler-type mobile robot Servosila Engineer is not equipped with LRF in its original configuration, we added a special platform on the top of the robot head with Hokuyo UTM-30LX-EW LRF, integrated this LRF into robot control system and verified its performance within LRF-SLAM.

The rest of this paper is structured as follows: Section 2 describes system setup. Section 3 describes the process of laser integration and SLAM. Section 4 summarizes everything and hints on a future work.

2. Robot system setup

Servosila Engineer[4] (Fig. 1) is a Russian crawler-type mobile robot designed for various areas of use including search and rescue operations[5], operations within dangerous or inaccessible for a human environment and many other areas. The robot in its original configuration

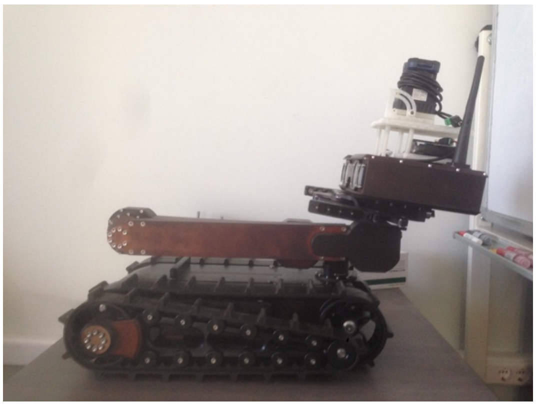

Fig. 1. Servosila Engineer crawler-type robot with a Hokuyo UTM-30LX-EW LRF mounted on its top.

is equipped with an optical zoom camera and a pair of stereo vision cameras. The robot is operated with the original interface in a teleoperation mode only and an operator controls speeds of the servos and positions of the robot's parts.

For SLAM purposes, Hokuyo UTM-30LX-EW range finder was used. We designed and constructed with a 3D printer a special static stand for the LRF with an option to select an inclination of a scanning beam toward the surface of an environment. Due to mounting with adjustable angle it is possible to use the LIDAR in overcoming obstacles and solving the problem of robot balance while moving through 3D debris[6]. The stand was attached to the top of the robot head (Fig. 2).

In addition to the original server and software, which were provided by the maker, we installed Robot Operating System (ROS) Indigo in order to run ROS nodes, which are required for the laser range finder output data streaming and LRF SLAM algorithms. Even though modern versions of ROS, e.g., Kinetic Kame, are already available, we were restricted to use ROS Indigo version because Sevosila Engineer comes with a special version of Ubuntu 14.04, which is tailored to the robot's internals so we can't install another version of Linux Ubuntu to the robot. Being restricted with the OS version, we decided that the best option of ROS would be Indigo.

3. Laser range finder integration and LRF-based SLAM

First step in establishing SLAM for the robot was to get data stream from the laser range finder. For these

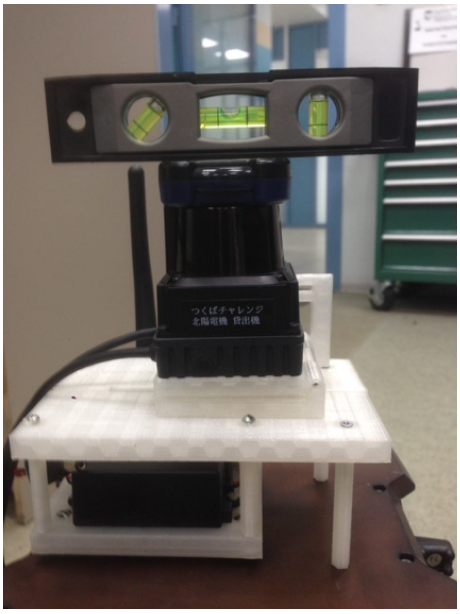

Fig. 2. Hokuyo laser on a stand. Laser is strictly parallel to the surface of the floor.

purposes urg_node ROS package was used. This package allows to run ROS-node, which reads data from the laser range finder and publishes LaserScan messages to the /scan topic[7]. Next, laser_scan_matcher ROS package was used to get odometry from the laser scans[8]. This package compares consecutive LaserScan messages to estimate position of the LRF in space. Laser-based odometry turned out to be quite accurate but in Section 4 we describe our future plans in establishing odometry from the crawler encoders in order to further improve odometry accuracy through the use of multiple data sources and combining this information together. After getting odometry, we aimed to perform mapping and localization for our robot.

Prior to that, experiments were conducted in a simulation in order to identify the most suitable for our purposes LRF-SLAM algorithm. We had tested gmapping[8,9], Google Cartographer[10] and Hector SLAM[11] algorithms on prerecorded rosbag files with laser range data and concluded that gmapping together with laser_scan_matcher significantly outperformed other SLAM algorithms. Thus, we opted for gmapping as the package for SLAM[8].

Gmapping SLAM package is implemented using Rao - Blackwellized particle filter. Each particle in this

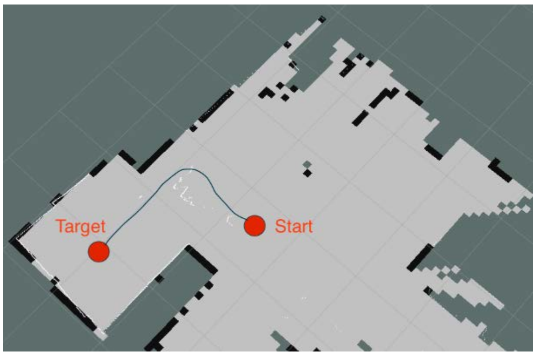

Fig. 3. RBPF-SLAM with default parameters results: virtual experiment in RViz.

algorithm is a separate Dynamic Bayesian Network that stores its own version of a map. Rao-Blackwellized particle filter is applied to these particles in order to pick out the most plausible information about the environment[12]. Also, the process called marginalization is used to reduce the number of particles. This process solves the main challenge when using the particle filter. It groups similar particles into one particle in some area

Fig. 4. The same experiment with the new RBPF-SLAM parameters: virtual experiment in RViz.

R. This reduces the number of particles and allows faster execution of the algorithm and lower memory costs[13]. After establishing gmapping for our robot with default parameters, real-world experiments were conducted in order to determine the quality of mapping and localization. During the experiment, the robot was teleoperated from the start point to the target point while performing LRF-SLAM and the generated map was stored. Figure 3 demonstrates that the obtained map is not detailed enough. Thus, further experiments were conducted in order to improve mapping results.

Figure 4 demonstrates the results of the same experiment, but this time the map, which provides significantly more details about the environment, was obtained by manipulating the RBPF-SLAM configuration parameters. For example, linear and angular update parameters were both set to 0.1. These parameters depend on laser frequency and range, odometry information source and level of sensor shaking when the robot moves. Therefore, these parameters must be finely tuned for each particular case in order to obtain the best mapping and localization quality.

After fine-tuning of RBPF-SLAM parameters further

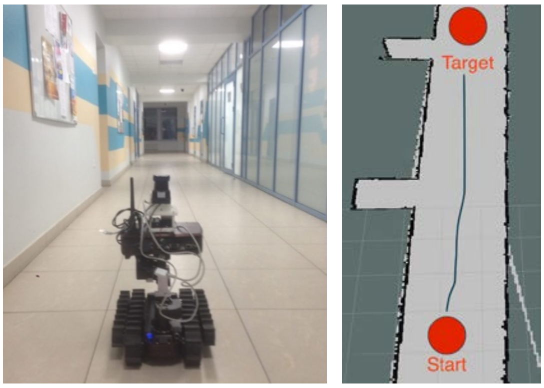

Fig. 5. RBPF-SLAM experiment in the corridor (left) and virtual experiment in RViz (right).

experiments were conducted in another location. A long corridor was selected as the next experiment location because such locations are the most challenging ones when the odometry is coming from the laser range finder. In the experiment, the robot was teleoperated along the corridor from the start point to the target point, and the generated map was stored. Figure 5 (on the right) demonstrates that although the corridor itself does not contain a large number of feature points, laser_scan_matcher algorithm succeeded to provide quite accurate odometry information, which in turn provided the map without any significant errors.

4. Conclusions and future work

Accurate mapping and localization is important for the majority of tasks in the mobile robotics. In our research we used laser range finder to perform simultaneous localization and mapping. We used laser range finder both for SLAM and for getting odometry information. We demonstrated that this configuration is sufficient for getting quite accurate mapping and localization results. Fine tuning is required and it strongly depends on a particular laser range finder and mobile base characteristics. Unfortunately, under certain circumstances (e.g., in feature-poor environments) laser_scan_matcher may fail and start to accumulate odometry error. Therefore, as a part of our future work we plan to obtain odometry information from crawler encoders of the robot and to perform sensor fusion of laser range finder odometry and encoders odometry. This should result in better odometry accuracy, which is crucial for mapping and localization of a robot.

After completing SLAM part, we plan to apply it to the problem of robot autonomous return to the starting point. This means that on the outward way the robot will record its' position and the map of the environment, and next, on the return way, it will autonomously navigate itself to the starting point similarly to[14,15].

Acknowledgements

This work was partially supported by the Russian Foundation for Basic Research (RFBR) and Ministry of Science Technology & Space State of Israel (joint project ID 15-57-06010). Part of the work was performed according to the Russian Government Program of Competitive Growth of Kazan Federal University.

References

1. I. Afanasyev, A. Sagitov and E. Magid, ROS-based SLAM for a Gazebo-simulated mobile robot in image-based 3D model of indoor environment, *Int. Conf. on Advanced Concepts for Intelligent Vision Systems. Springer, Cham* (2015), pp. 273-283.
2. A. Buyval, I. Afanasyev and E. Magid, Comparative analysis of ROS-based Monocular SLAM methods for indoor navigation, *9th Int. Conf. on Machine Vision. – International Society for Optics and Photonics* (2017), pp. 103411K-103411K-6.
3. V. Khithov, A. Petrov, I. Tishchenko and K. Yakovlev, Toward autonomous UAV landing based on infrared beacons and particle filtering, *Proc. 4th Int. Conf. on Robot Intelligence Technology and Applications,* (Springer International Publishing 2016), pp. 529-537.
4. M. Sokolov, R. Lavrenov, A. Gabdullin, I. Afanasyev and E. Magid, 3D modelling and simulation of a crawler robot in ROS/Gazebo, *4th Int. Conf. on Control, Mechatronics and Automation* (2016), pp. 61-65.
5. E. Magid and T. Tsubouchi, Static Balance for Rescue Robot Navigation-Translation Motion Discretization Issue within Random Step Environment, *Proc. Int. Conf. on Informatics in Control, Automation and Robotics* (2010), pp. 415-422.
6. E. Magid and T. Tsubouchi, Static balance for rescue robot navigation: Discretizing rotational motion within random step environment, *Int. Conf. on Simulation, Modeling, and Programming for Autonomous Robots* (Springer, Berlin, Heidelberg, 2010), pp. 423-435.
7. Laser scan matcher, ROS package: http://wiki.ros.org/laser_scan_matcher
8. Gmapping, ROS package: http://wiki.ros.org/gmapping
9. M. Quigley, D. Stavens, A. Coates and S. Thrun. Sub-meter indoor localization in unmodified environments with inexpensive sensors, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems,* (2010), pp. 2039-2046.
10. Google Cartographer, repository: https://github.com/googlecartographer
11. Hector SLAM, ROS package: http://wiki.ros.org/hector_slam
12. M. He, E. Takeuchi, Y. Ninomiya and S. Kat, Precise and efficient model-based vehicle tracking method using Rao-Blackwellized and scaling series particle filters, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* (2016), pp.117-124.
13. A. Doucet, N. De Freitas, K. Murphy and S. Russell, Rao-Blackwellised particle filtering for dynamic Bayesian networks, *Proc. 16th Conf. on Uncertainty in artificial intelligence* (Morgan Kaufmann Publishers Inc. 2000, June), pp. 176-183.
14. N. Tsuda, S. Harimoto, T. Saitoh and R. Konishi, Mobile robot with following and returning mode, *18th IEEE Int. Symposium on Robot and Human Interactive Communication* (2009), pp. 933-938.
15. M. Awai, T. Shimizu, T. Kaneko, A. Yamashita and H. Asama, HOG-based person following and autonomous returning using generated map by mobile robot equipped with camera and laser range finder, *Intelligent Autonomous Systems 12* (2013), pp. 51-60.