SLAM in Simulation and Test Environment

Károly Széll
Alba Regia Technical Faculty
Óbuda University
Székesfehérvár, Hungary
szell.karoly@amk.uni-obuda.hu

***Abstract*— Smart manufacturing encompasses a wide range of specializations. One of the most important areas is intralogistics, which besides being part of the supply chain, is a safety critical field as well. Industry has an emerging need for automated solutions thus there is a trend to use more and more automated guided vehicles (AGV). This study introduces a simulation and test environment for simultaneous localization and mapping (SLAM) methods.**

Keywords—SLAM, autonomous robotics, mobile robotics, localization, mapping, ROS

I. Introduction

The SLAM problem is simple, but very important for an autonomous mobile robot [1]–[4]; starting from an unknown starting position in an unknown environment, creating a map of the place (mapping) incrementally, and using this map to determine its current position (localization) [5]–[7].

The task was defined by the research community back in 1986, and they began experimenting with probabilistic methods in robotics. The work started successfully; it has been shown that landmarks are highly correlated, and this correlation increases with increasing number of observations. The first attempts to apply the Kalman filter were also made that time. However, later solutions split the problem of positioning and mapping. This was a consequence of the generally accepted opinion that the estimated errors of the map do not converge, but rather resemble a random motion within an infinitely increasing margin of error. The researchers focused on one or the other sub-problem, trying to provide approximate solutions. This involved minimizing or neglecting the correlation between landmarks.

The breakthrough came in 1995 when they realized that the combined mapping and positioning problem was, in fact, convergent in terms of a single estimator, and also recognized that landmark correlation, which many researchers had sought to minimize, was critical to solving the problem; the stronger the correlation the better the solution. Since then, interest in the area has increased; they primarily seek to reduce the computational complexity of the algorithms, and further refine data association and loop closure.

II. Problem Description

Fig. 1 shows a typical SLAM task. The mobile robot travels through an unknown area of which he has no prior knowledge. As the robot moves, it uses a built-in sensor (relative to its current position) to make relative observations of landmarks along the way.

As the robot odometry responsible for navigation often proves to be inaccurate, we cannot rely solely on odometry information. We can use the results of the built-in sensor to take measurements of the environment for more accurate data. We can do this by identifying certain landmarks of the environment and observing them again and again as we move. When the odometry changes as a result of robot movement, the uncertainty about the new position of the robot changes (odometric update). The landmarks are then identified in the new position of the robot. The robot attempts to connect the newly discovered landmarks to previous measurements (re-observation). The re-observed landmarks serve to update the robot's position.

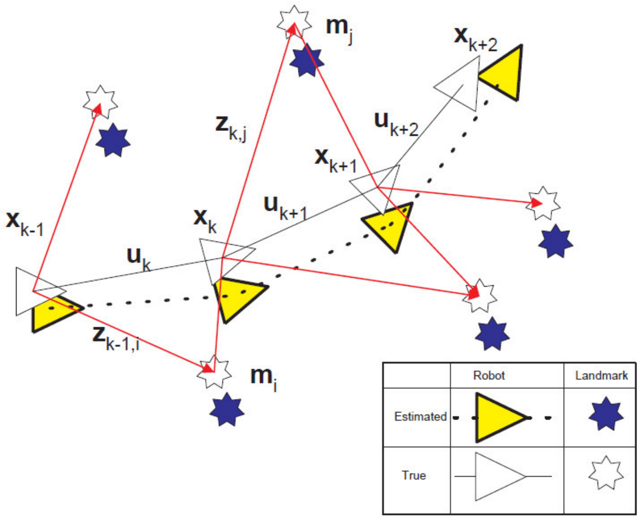

Fig. 1: SLAM problem [8], [9]

The most commonly used methods for solving a SLAM problem is the extended Kalman filter (EKF), the Fast Slam [10] and the Graph SLAM [11]. There are newer solutions and improved versions of these basic methods for which we can find comparison in the literature [12]–[14]. A list of the most common SLAM methods:

- Lidar SLAM methods
  - GMapping [15]
  - Hector SLAM [15]
  - Cartographer
- Monocular SLAM methods [16]
  - Parallel Tracking and Mapping (PTAM)
  - Semi-direct Visual Odometry (SVO)
  - Dense Piecewise Parallel Tracking and Mapping (DPPTAM)
  - Large Scale Direct monocular SLAM (LSD SLAM) [17]
  - ORB SLAM (mono) [17]
  - Direct Sparse Odometry (DSO)
- Stereo SLAM methods [16]
  - ZEDfu
  - Real-Time Appearance-Based Mapping (RTAB map)
  - ORB SLAM (stereo) [17]
  - Stereo Parallel Tracking and Mapping (S-PTAM)

The focus of this paper is the lidar based SLAM methods as these are the most common solutions in AGV navigation. 2D lidar sensors provide range and bearing data of the obstacles around the robot (see Fig. 2). The SLAM methods are based on these measurements and the odometry of the mobile platform [18], [19].

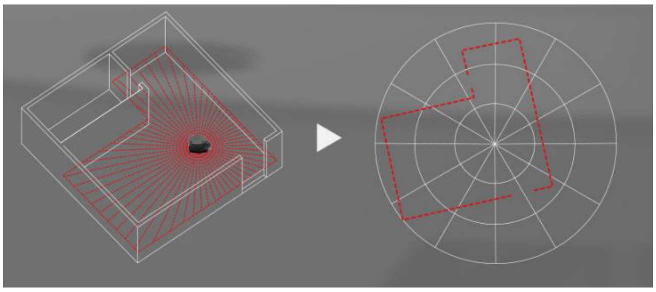

Fig. 2: Lidar measurement [20]

EKF is usually used for state estimation assuming an ideal map is provided. Thus, the procedure is missing the map update required by the SLAM algorithm. In contrast to the EKF used for state estimation, the matrices used in the SLAM-EKF change. Most of the SLAM-EKF procedure follows the classic EKF after the matrices are correctly formed.

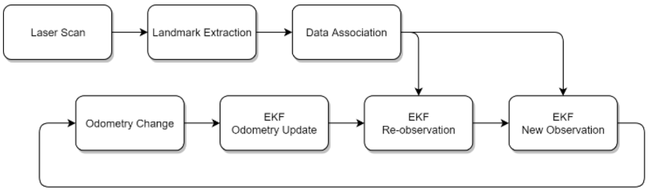

Fig. 3: EKF SLAM method

The Kalman filter and the EKF represent the probability distributions with a parameterized model. Particle filters [21], on the other hand, represent distributions with a finite set of sample states called particles. High probability regions have denser particles, while lower probability regions have few or no particles. For a sufficient number of samples, this non-parametric representation can be used to estimate arbitrarily complex multidimensional distributions. If the number of samples goes to infinity, the true distribution can be accurately reconstructed. Fast SLAM, based on the particle filters, takes advantage of the conditional independence that results from the structure of the SLAM problem.

The third common approach to the SLAM problem is Graph SLAM. Graph SLAM divides the problem into two parts: front end and back end, which are connected by the pose graph of Fig. 4.

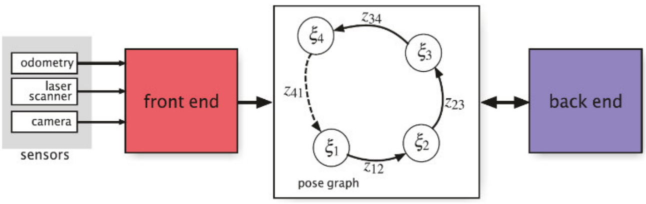

Fig. 4: Graph SLAM method [22]

The path of the robot is considered as a series of discrete positions, and the task is to estimate these positions. The constraints between unknown positions are based on measurements of various sensors, including odometry, laser scanners and cameras. The problem is formulated as a directed graph.

III. Simulation Environment

As simulation environment V-REP software (see Fig. 5) was chosen. V-REP is capable of performing most robotics-related simulation tasks, and what is important in our study it is compatible with Robot Operating System (ROS). ROS is a flexible robotic software framework that incorporates a number of toolkits to integrate some of the aforementioned SLAM algorithms and also facilitates building of industry 4.0 systems [23]–[27].

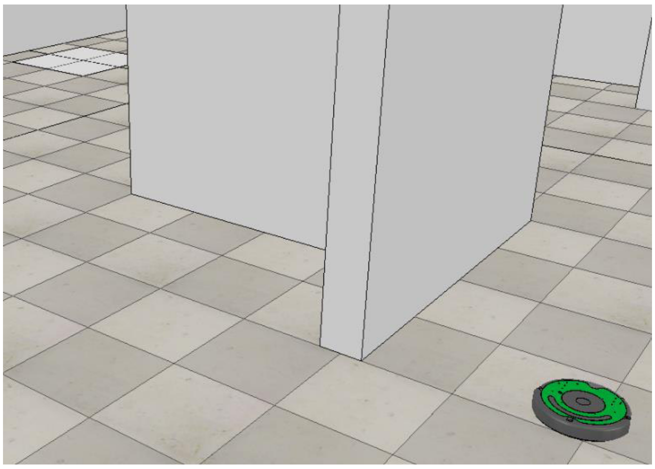

Fig. 5: V-REP simulation environment

In addition to the 3D rendering, V-REP also has various physical motors that allow real-time motion modeling. Sensors can also be simulated, allowing the simulation of real Lidar measurements. Fig. 5 shows the mobile robot platform moving towards the white target in the upper left corner.

IV. Experimental Setup

The real environment is two adjacent halls of a robotics center between which mobile robots are traveling (see Fig. 6 and Fig. 7).

Fig. 6: ABB-KUKA robot laboratory

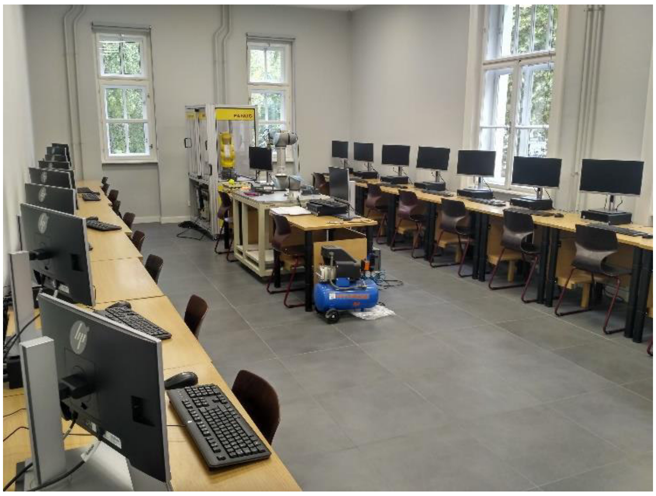

Fig. 7: Fanuc-UR robot laboratory

The robotic center is equipped with Fanuc, Universal Robots, ABB and KUKA robot arms. From the viewpoint of the SLAM environment the most important features of the laboratories are the tiled floor and the several legs of the tables and chairs. The previous one might deteriorate the measurements of the mobile robot platform by influencing the orientation. Several legs might be hard to detect and associate during the SLAM algorithm.

The mobile robot platforms are based on the iRobot Roomba 605, which is commercially available, cost effective, easy to repair and has an open communication protocol for its controller that provides access to its actuators and sensors, and its battery accessible. We built three versions on this platform:

- Lidar (see Fig. 8), which can be used to test laser scanning based algorithms.
- Kinect (see Fig. 9), which can be used to test image-based algorithms.
- Mobile manipulator, where the mobile platform is equipped with a small robot arm (not relevant for this paper)

Fig. 8: Lidar mobile robot platform

Fig. 9: Kinect mobile robot platform

The connection between the components was solved with Raspberry Pi (see Fig. 10). The Raspberry Pi and the sensors are powered by the mobile robot's battery, where a highly reliable circuit is needed [28]–[30]. As the battery voltage of the mobile robot platform is too high for the elements built up on it, a buck converter was needed to transform the DC voltage to the suitable level. Communication with the mobile robot, such as moving the wheels, is done by Raspberry Pi. The raw data of the sensors (IMU, Lidar, Kinect) is received by Raspberry Pi and then transmitted to an external PC into ROS environment, which performs data processing requiring high computing capacity. This PC is also responsible for performing tasks of industry 4.0 like ERP system connectivity [31] or Big Data [32].

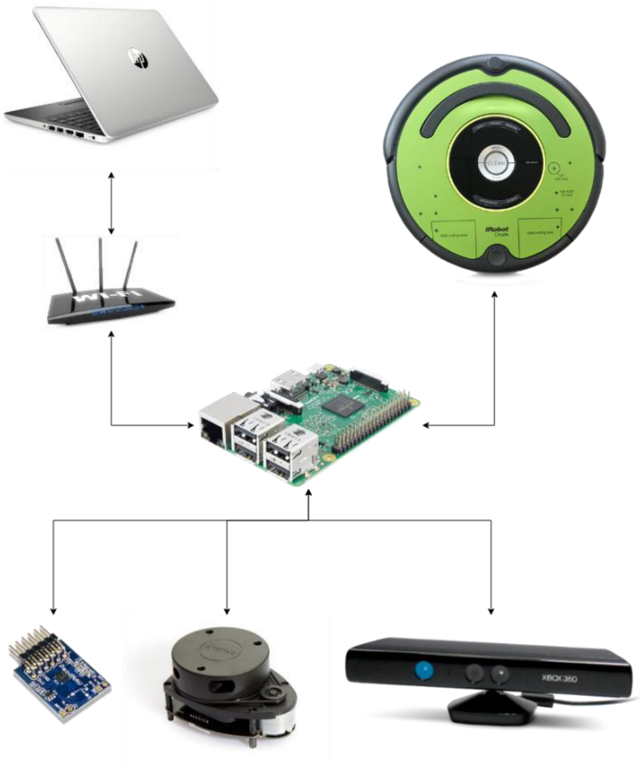

Fig. 10: Hardware architecture

V. Conclusion

This article introduces a simulation and test environment for testing SLAM algorithms. The presented environment is built up of cost-effective, easily accessible elements, but it also allows testing most SLAM methods in both software and hardware environments. The equipment has also proven to be a good basis for research, development and education purposes of SLAM and industry 4.0 components.

VI. Acknowledgement

The authors wish to thank the support to the Arconic Foundation.

VII. References

[1] P. Baranyi, I. Nagy, B. Korondi, and H. Hashimoto, "General guiding model for mobile robots and its complexity reduced neuro-fuzzy approximation," in *Ninth IEEE International Conference on Fuzzy Systems. FUZZ- IEEE 2000 (Cat. No.00CH37063)*, 2000, vol. 2, pp. 1029–1032 vol.2.

[2] L. Raj and A. Czmerk, "Modelling and simulation of the drivetrain of an omnidirectional mobile robot," *Automatika*, vol. 58, pp. 232–243, 2017.

[3] B. Kovács, G. Szayer, F. Tajti, M. Burdelis, and P. Korondi, "A novel potential field method for path planning of mobile robots by adapting animal motion attributes," *Robot. Auton. Syst.*, vol. 82, pp. 24–34, 0 2016.

[4] F. Tajti, G. Szayer, B. Kovacs, P. Barna, and P. Korondi, "Optical flow based odometry for mobile robots supported by multiple sensors and sensor fusion," *Automatika*, vol. 57, 0 2016.

[5] A. Singandhupe and H. La, "A Review of SLAM Techniques and Security in Autonomous Driving," in *2019 Third IEEE International Conference on Robotic Computing (IRC)*, 2019, pp. 602–607.

[6] C. Cadena *et al.*, "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, 0 2016.

[7] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE Trans. Intell. Veh.*, vol. 2, no. 3, pp. 194–220, Sep. 2017.

[8] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.

[9] T. Bailey and H. Durrant-Whyte, "Simultaneous Localisation and Mapping (SLAM): Part II State of the Art," *State Art IEEE Robot. Autom Mag*, vol. 13, 2006.

[10] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," *ResearchGate*. [Online]. Available: https://www.researchgate.net/publication/224773156_FastSLAM_A_Factored_Solution_to_the_Simultaneous_Localization_and_Mapping_Problem. [Accessed: 29-Sep-2019].

[11] W. Burgard *et al.*, "A comparison of SLAM algorithms based on a graph of relations," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 2089–2095.

[12] M. Kühn, R. Koch, and S. May, "Benchmarking the pose accuracy of different SLAM approaches for rescue robotics," 2015.

[13] L. Nardi *et al.*, "Introducing SLAMBench, a performance and accuracy benchmarking methodology for SLAM," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5783–5790.

[14] R. Yagfarov, M. Ivanou, and I. Afanasyev, "Map Comparison of Lidar-based 2D SLAM Algorithms Using Precise Ground Truth," 2018.

[15] M. Rojas-Fernández, D. Mújica-Vargas, M. Matuz-Cruz, and D. López-Borreguero, "Performance comparison of 2D SLAM techniques available in ROS using a differential drive robot," in *2018 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, 2018, pp. 50–58.

[16] T. Taketomi, H. Uchiyama, and S. Ikeda, "Visual SLAM algorithms: a survey from 2010 to 2016," *IPSJ Trans. Comput. Vis. Appl.*, vol. 9, no. 1, p. 16, Jun. 2017.

[17] A. Huletski, D. Kartashov, and K. Krinkin, "Evaluation of the modern visual SLAM methods," in *2015 Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference (AINL-ISMW FRUCT)*, 2015, pp. 19–25.

[18] S. Riisgaard and M. R. Blas, *SLAM for Dummies A Tutorial Approach to Simultaneous Localization and Mapping By the 'dummies.'* .

[19] M. Naminski, "An Analysis of Simultaneous Localization and Mapping (SLAM) Algorithms," *Math. Stat. Comput. Sci. Honors Proj.*, May 2013.

[20] "RPLIDAR-A2 Laser Range Scanner_ Solid Laser Range Scanner|SLAMTEC." [Online]. Available: https://www.slamtec.com/en/Lidar/A2. [Accessed: 04-Nov-2019].

[21] G. Grisetti, C. Stachniss, and W. Burgard, "Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters," *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 34–46, 0 2007.

[22] P. Corke, *Robotics, Vision and Control : Fundamental algorithms in MATLAB*, 1st ed. 2011. Berlin: Springer-Verlag Berlin and Heidelberg GmbH & Co. K, 2011.

[23] J. M. Santos, D. Portugal, and R. P. Rocha, "An evaluation of 2D SLAM techniques available in Robot Operating System," in *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2013, pp. 1–6.

[24] I. Z. Ibragimov and I. M. Afanasyev, "Comparison of ROS-based visual SLAM methods in homogeneous indoor environment," in *2017 14th Workshop on Positioning, Navigation and Communications (WPNC)*, 2017, pp. 1–6.

[25] M. Filipenko and I. Afanasyev, "Comparison of Various SLAM Systems for Mobile Robot in an Indoor Environment," in *2018 International Conference on Intelligent Systems (IS)*, 2018, pp. 400–407.

[26] X. S. Le, L. Fabresse, N. Bouraqadi, and G. Lozenguez, "Evaluation of Out-of-the-Box ROS 2D SLAMs for Autonomous Exploration of Unknown Indoor Environments," in *Intelligent Robotics and Applications*, 2018, pp. 283–296.

[27] M. Kuzmin, “Review. Classification and Comparison of the Existing SLAM Methods for Groups of Robots,” in *2018 22nd Conference of Open Innovations Association (FRUCT)*, 2018, pp. 115–120.

[28] G. Györök and B. Beszédes, “Adaptive Optocoupler Degradation Compensation in Isolated Feedback Loops,” *2018 IEEE 12th Int. Symp. Appl. Comput. Intell. Inform. SACI*, pp. 167–172, 2018.

[29] G. Györök and B. Beszédes, “Fault tolerant power supply systems,” in *11th International Symposium on Applied Informatics and Related Areas (AIS 2016)*, 2018, pp. 68–73.

[30] G. Györök and B. Beszédes, “Highly reliable data logging in embedded systems,” in *2018 IEEE 16th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2018, pp. 49–54.

[31] A. Selmeci and T. Orosz, “Usage of SOA and BPM changes the roles and the way of thinking in development,” in *2012 IEEE 10th Jubilee International Symposium on Intelligent Systems and Informatics*, 2012, pp. 265–271.

[32] É. Hajnal, “Big Data Overview and Connected Research at Óbuda University Alba Regia Technical Faculty,” in *AIS 2018 - 13th International Symposium on Applied Informatics and Related Area*, 2018, pp. 1–4.