

# Review, Classification and Comparison of the Existing SLAM Methods for Groups of Robots

Maxim Kuzmin

Saint-Petersburg Electrotechnical University “LETI”

St. Petersburg, Russia

max\_kuzmin@outlook.com

**Abstract**—Nowadays the promising line of research is an application of groups of mobile robots to various tasks. An effective SLAM algorithm is one of their main success factors. Due to the increasing popularity of the open-source robots framework, ROS, the best methods should be implemented on this platform. The development should be based on the theoretical research of the subject area. So, the paper is justified by this fact. Multi-robot SLAM methods have been classified according to their key features. Their advantages and disadvantages have been identified. The methods have also been compared according to the available experimental data. The methods most suitable for implementation have been selected.

## I. INTRODUCTION

The task of Simultaneous Localization and Mapping (SLAM) can be described as step-by-step continuous estimation and adjustment of the map based on data from a mobile robot and positioning of the robot on this map at the same time. An effective and precise SLAM algorithm is an important element in navigation and other high-level tasks [2]. In this context we consider any mobile robotic platform equipped with its own computing unit and sensors for getting information from the environment such as LIDAR, video and depth cameras, odometer, inertial sensor, etc.

Now there are a lot of effective and applied in practice single-robot SLAM algorithms. However, the research on this problem for a group of robots becomes more relevant because of the expansion of their usage in different spheres, rapid development of hardware and evolution of software [13]. The group usually consists of independent autonomous robots which are connected into a single network and cooperate for the common goal achievement.

As practice shows, there is still no widely accepted, proven and reliable navigation algorithm for groups of robots. Meanwhile, the scientific community is interested in the topic and this results in new inventions. The majority of the efforts doesn't become popular either because the authors don't publish the implementation details of their approaches and source code or because the very algorithms are not elaborated enough to be a universal solution. These difficulties along with the lack of communication between the research groups and a single formalized theoretical and code base slow down the progress significantly [3].

This paper is meant to contribute to the research on this problem. And its goal is the review, analysis, classification of the existing multi-robot SLAM methods, as well as the identification of their advantages and disadvantages and comparative evaluation. The next step should be the implementation of the promising methods on the available open-source platform for robot development, Robot Operating System (ROS), and the estimation of their functioning in different conditions based on real experiments.

## II. SLAM PRINCIPLES

The minimum requirement to a robot while solving the SLAM problem is its mobility and presence of a device retrieving information about the environment. The general scheme of single-robot SLAM can be represented in three steps [1] defined in probabilistic terms [2]:

1. The control signal moves the robot to new coordinates. But noise in the data from motion sensors, wheel slips and round-off errors lead to the uncertainty of the robot's location. This process is described by the motion model which depends on the type of odometer, platform and the way the robot moves. It is a nonlinear function:

$$s_t = f(s_{t-1}, u_t) + N(0, \Lambda_u)$$

It means that the new robot's location  $s_t$  depends on the previous one  $s_{t-1}$ , the control signal  $u_t$  and the Gaussian noise  $N$  with certain mean value and variance.

2. The robot receives new information about the environment. This stage is described by the inverse observation model which depends on the type of a sensor and its characteristics, and the algorithm of finding landmarks based on the sensor data. This is a nonlinear function:

$$z_{t,i} = h(s_t, \theta_{c_{t,i}}) + N(0, \Lambda_z)$$

In other words, the sensor measurement  $z_{t,i}$  is a function of the robot's location  $s_t$ , the landmark location  $\theta_{c_{t,i}}$  (where  $c_{t,i}$  is a landmark with index  $i$  at the step  $t$ ) and the Gaussian noise  $N$ .

3. The robot updates its location and the map. This is a key stage and it depends on a certain algorithm of integrating sensor data into the map and eliminating the noise. In general,

the estimation of the map and a robot's location can be described by the recursive expression:

$$p(s_t, m | z_{1:t}, u_{1:t}) = \\ = p(z_t | s_t, m) \int p(s_t | s_{t-1}, u_t) p(s_{t-1}, m | z_{1:t-1}, u_{1:t-1}) ds_{t-1}$$

where  $m = \sum_{i=0}^n c_i$  is a map represented by landmarks or cells of grid.

The single-robot SLAM methods can be adapted for the groups of robots. Such modifications are aimed at solving the number of problems appearing in a new context [3]:

- What data communication channels should the robots use?
- What network topology should we use for connecting the robots?
- What information about the map and the location should the robots exchange?
- What way should the integration of other robots' data be done?
- How should the robots estimate relative locations?

In case of solving these problems, we can get several advantages:

- Faster map construction compared to a single robot due to the work of several computing units and sensors;
- Higher algorithm precision due to the presence of several independent sources of data and map estimations;
- Algorithm robustness against the influence of negative environment factors on individual robots within the group and robot breakdowns.

Probabilistic expression of the multi-robot SLAM task appears as follows [13]:

$$p(s_t^{1:k}, m | z_{1:t}^{1:k}, u_{1:t}^{1:k}) = \\ = \prod_{i=1}^k \int p(s_t^i | s_{t-1}^i, u_t^i) p(s_{t-1}^i, m | z_{1:t-1}^{1:k}, u_{1:t-1}^{1:k}) ds_{t-1}^i$$

where  $k$  is a number of robots in the group.

### III. CHOICE OF RESEARCH METHOD

The research on the multi-robot SLAM methods was conducted in four steps:

Firstly, we have classified the existing methods according to the most significant criteria.

Secondly, we have compared them based on the advantages and disadvantages, indicative for certain classes of the methods. The results are shown in Table I.

Thirdly, we have compared the methods based on the experimental data presented in the corresponding articles. The data contains scenarios of simulated and real experiments, the

test environment characteristics, the number of robots and their parameters, images of the constructed maps. The results are shown in Table II.

Fourthly, we have made conclusions about the efficiency and practicality of the methods. They are presented in the corresponding section.

### IV. METHODS CLASSIFICATION

Despite the diversity of SLAM methods, they are all based on several typical approaches with different adaptations. And this allows to divide them into classes.

#### A. Software architecture

According to software architecture there are frontend and backend methods [19].

Backend methods perform construction of the map and localization having the whole data set collected by both the motion and observation models. This helps to manage bad quality data easily and to recover after serious errors in the estimation of the map and robots' location. Such methods have a complex probabilistic model and are usually based on the filtering algorithms. They can be used with or without frontend methods.

a) Filtering methods, historically, were the first SLAM algorithms which were developed during the time of the backlog of technical means (inaccurate and expensive sensors). Such methods are based on the probabilistic apparatus of the Extended Kalman filter (EKF) [5], [12], Sparse Extended Information Filter (SEIF) [9] or Rao-Blackwellized Particle Filter [6], [8], [15]. Experience has proven that EKF has become outdated and is scarcely used due to significant limitations of the map size and bad compatibility with high-precision laser rangefinders. At the same time particle filter is widely used and big variety of methods based on it are characterized by high estimate precision and robustness.

b) Pose-graph optimization methods process and store all the data collected during a robot's motion in the form of graph vertices. That is why the map cannot always be estimated in real time. However, it makes these methods the most precise and, in contrast to the other classes of methods, allows to get the estimation of the whole path, i.e. solution of the full SLAM problem [11], [19].

c) Methods based on machine learning and neural networks [17] are getting popular nowadays but already seem far-reaching. Their main benefit and disadvantage is a necessity for preliminary model training.

Frontend methods are mostly smoothing, i.e. based on error minimization algorithms. They are aimed at processing the data received according to the observation model. These methods can be used both as preliminary for backend ones or individually with high-precision sensors. It is these sensors that allow to dispense with complex mathematical apparatus without the loss of the quality of the result.

a) Methods that take sequences of scans from a laser rangefinder (LIDAR) or similar device (Kinect) as an input. They are mostly algorithms of scan-matching (ICP, IDC) [16], [18].

b) Methods that extract environment landmarks from a sequence of camera images and then match clouds of features. Such algorithms as SURF and SIFT are used [13], [14].

### B. Runtime

According to the runtime there are online and offline classes of methods [2].

Offline methods (full SLAM) are executed with a pre-collected data set and construct a robot's trajectory and the map only after the phase of the robot's activity is finished [11], [16]. Their main benefit is an opportunity of generating very accurate map and robots' path due to the lack of time and computational power limitations. But these methods are not applicable in most real tasks which require a robot's decision based on the current map in real time.

Online methods work along with robots' motion and data collection. They are applicable to the tasks in which the main requirement is a robot's autonomy and the ability to make independent decisions. The majority of the analyzed methods are online.

### C. Map representation

According to the inner map representation which directly depends of the type of sensor there are three classes.

The grid map uses an array of cells in which it stores the probability of finding an obstacle in a given space point [8], [10], [16], [17]. This representation is used for robots equipped with a laser rangefinder. The category includes the majority of the popular methods.

The map represented by the set of landmarks such as markers of special shape, object corners, unique textures [5], [6], [15]. Markers are extracted from images received from the camera based on computer vision algorithms. However, it is also possible to extract landmarks from laser scans which will represent positions of obstacles on the 2D surface.

The map represented by a pose-graph [10], [11], [16] where edges are translation vectors and vertices are sensor observations from these space points. It is possible to use any sensor because after optimization the data in the vertices is integrated and converted into representations 1 and 2.

The next classification criteria are specific only for multi-robot SLAM methods and usually characterize robots' interaction.

### D. Network topology

According to the network topology which connects robots there are centralized and distributed methods.

In a centralized network the key role belongs to a central node which performs main computations of a SLAM algorithm. And robots play roles of mobile sensors reading out

information about the environment and passing it to the central node [4]. This scheme is easy to implement, allows to arrange better coordination between robots and requires less resources of robots-sensors. However, among its drawbacks there are high requirements to reliability and capacity of data exchange channels, the central node power and the necessity of identifying initial robots' location. Moreover, the system becomes difficult to implement and impractical given a big explored area or a big number of robots.

Distributed scheme means that each robot is fully autonomous, constructs its own copy of the map and, if possible, exchanges it with the other robots while exploring the area. This group comprises almost all the reviewed methods. The scheme allows the construction of big maps by big groups of robots with high robustness to noise and parallel task execution. But these systems undergo the problem of mutual detection of robots and estimation of their relative location and also the difficulty of coordination.

### E. Data exchange method

According to the data exchange way there are algorithms with direct and indirect connections.

Direct data exchange is more popular and effective. It means that robots communicate with each other through their own receivers and transmitters. This is most consistent with the SLAM task requirements.

Indirect data exchange takes place in case of an algorithm transmitting all the data through the central node [4] or buffers which can be special transmitting trackers positioned all over the area [15].

### F. Type of data for exchange

According to the type of the environment data for exchange there are methods that transmit raw data from a sensor or parts of a map.

If the raw data is exchanged, it is possible to get more precise map estimation. However, the load on the transmit channel increases. Each robot uses this data just like it uses its own sensor data. That is why an error in map estimation made by each robot doesn't influence the overall result [8], [10].

The exchange of parts of a local map is less resource intensive because there is no need to store and transmit raw laser scans or point clouds. But while integrating this data estimation errors made by robots are accumulated [5], [6], [14], [17].

## V. METHODS ADVANTAGES AND DISADVANTAGES

Advantages and disadvantages of the reviewed methods are shown in Table I. The "Method" column contains the links to the corresponding articles (see references). The other two columns are the indices of the advantages and disadvantages which are listed below.

### A. Advantages

There are the following advantages:

- 1) The map is integrated by the group of robots online.

- 2) The robots don't have to be in direct contact to identify their relative locations.
- 3) The method solves the problem of association between the data from the sensor and the data already integrated into the map. This doesn't include the loop closure problem.
- 4) It is possible to combine the method with other algorithms which solve particular subtasks or to modify the method for getting better results.
- 5) The robots are fully autonomous, the network is decentralized.
- 6) The method effectiveness is proven by not only simulated experiments but also real ones.
- 7) The method does not require the assumption that initial mutual locations of robots are known.
- 8) The corresponding paper introduces a resource effective algorithm for data exchange between the robots.
- e) The method has the association problem of the data from a sensor and the data already integrated into the map, i.e. the problem of re-discovering the landmarks or rangefinder scan-matching or pose-graph edges mapping, etc.
- f) The authors make an assumption of initial conditions while implementing or testing the algorithm. Usually this means that robots' relative location or sensor data associations are known a priori.
- g) The method requires installing special markers or landmarks on the area.
- h) The algorithm can't work online because of high resource requirements.
- i) The method requires complex and/or not effective scheme of data exchange between robots.
- j) The method requires large amounts of RAM due to the use of the particle filter with the grid map.
- k) The method is not robust enough in some scenarios, for example, in case of limited contacts between robots, a lot of noise in data, small or too big number of map features, etc.

**B. Disadvantages**

There are the following disadvantages and problems:

- a) The area size is limited because of using the EKF filter and feature-based map.
- b) The method requires a central node or server which limits robots' autonomy.
- c) The robots have to be in direct contact to identify their relative location. Besides, it requires special markers on their bodies.
- d) The implementation of some parts of the algorithm is omitted, i.e. it does not implement the full solution of SLAM problem.
- l) The algorithm description in the article is not detailed enough for its implementation. Some important aspects, for instance, the problems of data association and data exchange, are not considered.
- m) The experimental data is ambiguous, i.e. incomplete, inaccurate or not demonstrating the method effectiveness.
- n) The method has a problem of loop closure in robots' paths.

TABLE I. SLAM METHODS ADVANTAGES AND DISADVANTAGES

Method / Paper	Advantages	Disadvantages and problems
Extending SLAM to Multiple Robots [4]	1	a, b, d, e, f, m
Multi-robot SLAM with Unknown Initial Correspondence: The robot rendezvous case [5]	1, 5, 6, 7	a, c, e
Cooperative Multi-Robot Map Merging Using Fast-SLAM [6]	1, 5, 7	c, e, g, m
Multi-robot Simultaneous Localization and Mapping using Particle Filters [7], Rao-Blackwellized Particle Filters Multi Robot SLAM with Unknown Initial Correspondences and Limited Communication [8]	1, 3, 5, 7	c, j, i, n
Multi-Robot SLAM with Sparse Extended Information Filers [9]	1, 2, 3, 5, 7	k
Multi-Robot SLAM using Condensed Measurements [10]	1, 3, 5, 6, 7, 8	c
Multi-Robot SLAM with Topological/Metric Maps [11]	5, 6, 7	c, h, l
Decentralized Cooperative SLAM for Sparsely-Communicating Robot Networks: A Centralized-Equivalent Approach [12]	1, 5, 7	a, c, g, l
Multi-robot visual SLAM using a Rao-Blackwellized Particle Filter [13]	1, 3, 4, 5, 6	c, f, l
Multi-Robot Marginal-SLAM [14]	1, 2, 5, 7	f, k, m
Multi-Robot Range-Only SLAM by Active Sensor Nodes for Urban Search and Rescue [15]	1, 2, 3	f, g, i, l
Multi-Robot Pose Graph Localization and Data Association from Unknown Initial Relative Poses via Expectation Maximization [16]	2, 3, 5, 7	f, h, n
A Neural Network-based Multiple Robot Simultaneous Localization and Mapping [17]	1, 2, 3, 5, 7	k
Multi-Robot Localization and Mapping based on Signed Distance Functions [18]	1, 2, 3, 6	f, k, l

VI. METHODS COMPARISON

The results of comparison are shown in Table II. The column “Based on” contains the mechanism of estimating the map and robots’ locations. “Size of map” column shows the size of the explored area, estimated using the photos of maps or indicated by the authors.

The column “Estimated precision” contains the rating of the functioning of methods. It considers the following factors:

- The presence of real experiments data.
- The size of the explored area. The bigger - the better.
- The number of robots participating in the experiment. The bigger - the better.
- The quality of the method elaboration and description in the corresponding paper, including the algorithm aspects defined and the assumptions made.
- The underlying SLAM algorithm for a single robot.

"Excellent" means that the method has the best accuracy of map estimating among those considered. "Good" means that the accuracy is sufficient for real-world applications. "Satisfactory" means that the method is interesting only for limited experiments. Since the articles do not contain metric parameters of environments and maps, it is impossible to give more complete precision estimate without carrying out our own experiments.

VII. COMPARISON RESULTS

In the result of the analysis and comparison of multi-robot SLAM methods the following conclusions have been made.

Despite the variety of multi-robot SLAM methods many of them have the disadvantages that make them non-universal. Besides, the majority of them is provided with little

information about the experiments in real conditions and/or with big groups of robots.

The filtering methods have been developing for a long time and are well studied. There is a number of proven single-robot SLAM algorithms with high robustness and precision. And there are also many theoretical efforts to adapt them for a multi-robot SLAM task.

Rao–Blackwellized particle filter algorithms [8], [13] have the advantages of precision, running speed and universality over the outdated extended Kalman filter. They are able to solve a multi-robot SLAM problem to the full extent.

Pose-graph optimization algorithms show the highest precision of map construction [10], [11]. The combination of these methods with filters and scan-matching algorithms allows to overcome the widespread difficulties. However, their execution is not always possible in online mode.

Due to the increased sensor accuracy smoothing frontend methods are getting more popular. They are fast and effective [18]. Hybrid methods combining the best features of frontend and backend algorithms appear [10], [16].

The methods based on machine learning and neural networks [17] are also promising despite their poor elaboration now. The possibility and necessity of preliminary model training are their strength and weakness at the same time.

The experiments show that the methods which don’t use map features are preferable because the corresponding computer vision algorithms are not perfect yet. Their big problems are data association and loop closure in robots’ paths.

It is obvious that the methods which don’t require direct robot contacts are better because the conditions of their robust functioning expand [16], [17], [18].

TABLE II. THE RESULTS OF THE METHODS COMPARISON

Method / Paper	Based on	Map type	Size of map, m x m	Estimated precision	Number of robots	Real experiment
[4]	EKF	Landmarks	-	Satisfactory	2	No
[5]	EKF	Landmarks	60×80 / 4800	Good	2	Yes
[6]	Particle Filter	Landmarks	5×5 / 25	Satisfactory	2	Yes
[7]	Particle Filter	Grid	2500	Good	4	No
[8]	Particle Filter	Grid	200	Good	2	Yes
[9]	SEIF	Landmarks	350×350 / 120000	Good	8	No
[10]	Scan Matching + Graph Optimization	Pose-graph + Grid	13×38 / 500	Excellent	8	Yes
[11]	Particle Filter + Graph Optimization	Pose-graph + Grid	77×36 / 2800	Good+	3	Yes
[12]	EKF	Landmarks	15×8 / 120	Good	9	Yes
[13]	Particle Filter	Landmarks	30×30 / 900	Good	3	Yes
[14]	Particle Filter	Landmarks	40×40 / 320	Good	2	Yes
[15]	Particle Filter	Landmarks + Grid	17×29 / 500	Good	4	Yes
[16]	Scan Matching + Graph Optimization	Pose-graph + Grid	15×15 / 225	Good	3	Yes
[17]	EKF + Neural Network	Grid	10×17 / 170	Good+	2	Yes
[18]	Scan Matching	Grid	28×14 / 392	Good	2	Yes

The most universal approaches are the methods with distributed network, particularly, those of them which don't need initial mutual locations [8], [11], [16], [17]. Here the robots are fully autonomous and SLAM tasks are solved simultaneously.

The above described methods are recommended for the implementation on the ROS platform due to their objective advantages identified in the result of the comparison.

### VIII. CONCLUSION

In this paper we have reviewed the principles of SLAM methods for a group of robots. The classification of these methods is developed according to key features: SLAM software architecture, algorithm runtime, inner map representation, robots network topology, data exchange channel and data type. The advantages and disadvantages of the methods have been identified. Their effectiveness and precision have been evaluated. The methods have also been compared according to the results of the experiments published by its authors.

The best results belong to the methods with decentralized network topology in which the map is represented by the grid and/or the pose graph. These methods are based on various combinations of such approaches to map assessment as the Rao-Blackwellized Particle Filter, pose-graph optimization and scan-matching algorithms.

The results of this paper will be used for implementing the selected SLAM methods for groups of robots on the open-source ROS platform.

### ACKNOWLEDGMENT

We would like to show our gratitude to all the authors of the reviewed SLAM methods for their contribution to the development of robotics. We also thank Anna Kupriyanova from ITMO University for great assistance with the translation of the paper.

### REFERENCES

- [1] J. Sola, "Simultaneous localization and mapping with the extended Kalman filter", 2014, Web: [http://www.iri.upc.edu/people/jsola/JoanSola/objectes/curs\\_SLAM/SLAM2D/SLAM\\_course.pdf](http://www.iri.upc.edu/people/jsola/JoanSola/objectes/curs_SLAM/SLAM2D/SLAM_course.pdf).
- [2] Y. Chen, "Algorithms for Simultaneous Localization and Mapping", 2013, Web:[http://cseweb.ucsd.edu/~yuc007/documents/re\\_report.pdf](http://cseweb.ucsd.edu/~yuc007/documents/re_report.pdf).
- [3] R.P.P. Rocha, "Building Volumetric Maps with Cooperative Mobile Robots and Useful Information Sharing - A Distributed Control Approach based on Entropy", 2006, chapter 1, Web: <https://repositorio-aberto.up.pt/bitstream/10216/11795/2/Texto%20integral.pdf>.
- [4] E. Howe, J. Novosad, "Extending SLAM to Multiple Robots", 2005, Web: [http://web.mit.edu/16.412j/www/html/Final%20Projects/FIN\\_Novosad\\_Howe.pdf](http://web.mit.edu/16.412j/www/html/Final%20Projects/FIN_Novosad_Howe.pdf)
- [5] X.S. Zhou, S.I. Roumeliotis, "Multi-robot SLAM with Unknown Initial Correspondence: The robot rendezvous case", in *Proc. IROS Conf.*, Oct. 2006, pp. 1785-1792.
- [6] N.E. Ozkucur, H.L. Akin, "Cooperative Multi-Robot Map Merging Using Fast-SLAM", in *Proc. RoboCup International Symposium*, Jun. 2009, pp. 449-460.
- [7] A. Howard, "Multi-robot Simultaneous Localization and Mapping using Particle Filters", *International Journal of Robotics Research*, vol. 25, Dec. 2006, pp. 1243-1256.
- [8] L. Carlone, M. K. Ng, J. Du, B. Bona, M. Indri, "Rao-Blackwellized Particle Filters Multi Robot SLAM with Unknown Initial Correspondences and Limited Communication", in *Proc. ICRA Conf.*, May 2010, pp. 243-249.
- [9] Thrun, Y. Liu, "Multi-Robot SLAM with Sparse Extended Information Filers", *International Journal of Robotics Research*, vol. 23, Aug. 2004, pp. 693-716.
- [10] M.T. Lazaro, L.M. Paz, P. Pinies, J.A. Castellanos, G. Grisetti, "Multi-Robot SLAM using Condensed Measurements", in *Proc. IROS Conf.*, Nov. 2013, pp. 1069-1076.
- [11] H. J. Chang, C. S. G. Lee, Y. C. Hu, Y.H. Lu, "Multi-Robot SLAM with Topological/Metric Maps", in *Proc. IROS Conf.*, Nov. 2007, pp. 1467-1472.
- [12] K. Y. K. Leung, T. D. Barfoot, H. H. T. Liu, "Decentralized Cooperative SLAM for Sparsely-Communicating Robot Networks: A Centralized-Equivalent Approach", in *Proc. IROS Conf.*, Oct. 2010, pp. 3554-3561.
- [13] A. Gil, O. Reinoso, M. Ballesta, M. Julia, "Multi-robot visual SLAM using a Rao-Blackwellized Particle Filter", *Robotics and Autonomous Systems*, vol. 58, Jan. 2010, pp. 68-80.
- [14] R. Martinez-Cantin, J. A. Castellanos, N. Freitas, "Multi-Robot Marginal-SLAM", in *Proc. IJCAI Conf.*, 2007.
- [15] D. Sun, A. Kleiner and T. M. Wendt, "Multi-Robot Range-Only SLAM by Active Sensor Nodes for Urban Search and Rescue", *RoboCup 2008: Robot Soccer World Cup XII*, Jun. 2009, pp. 318-330.
- [16] V. Indelman, E. Nelson, N. Michael, F. Dellaert, "Multi-Robot Pose Graph Localization and Data Association from Unknown Initial Relative Poses via Expectation Maximization", in *Proc. ICRA Conf.*, May 2014, pp. 593-600.
- [17] S. Saeedi, L. Paull, M. Trentini, H. Li, "A Neural Network-based Multiple Robot Simultaneous Localization and Mapping", *IEEE Transactions on Neural Networks*, vol. 22, Dec. 2011, pp. 2376-2387.
- [18] P. Koch, S. May, M. Schmidpeter, M. Kuhn, C. Pfitzner, C. Merkl, R. Koch, M. Fees, J. Martin, D. Ammon, A. Nuchter, "Multi-Robot Localization and Mapping based on Signed Distance Functions", in *Proc. ICARSC Conf.*, Apr. 2015, pp. 77-82.
- [19] G. Grisetti, R. Kummerle, C. Stachniss, W. Burgard, "A Tutorial on Graph-Based SLAM", *IEEE Intelligent Transportation Systems Magazine*, vol. 2, 2010, pp. 31-43.