

## 3D Environment Exploration with SLAM for Autonomous Mobile Robot Control

ANDRII KUDRIASHOV<sup>1</sup>, TOMASZ BURATOWSKI<sup>2</sup>, JERZY GARUS<sup>3</sup>, MARIUSZ GIERGIEL<sup>4</sup>

<sup>1,2,4</sup>Faculty of Mechanical Engineering and Robotics

AGH University of Science and Technology

30-059 Krakow, Al. Mickiewicza 30

POLAND

<sup>3</sup>Faculty of Mechanical and Electrical Engineering

Polish Naval Academy

81-103 Gdynia, Smidowicza 69,

POLAND

**Abstract:** - In the paper a solution for building of 3D map of unknown terrain for the purposes of control of wheeled autonomous mobile robots operating in an isolated and hard-access area is described. The work environment is represented by a three-dimensional occupancy grid map built with SLAM techniques using LIDAR sensor system. Probabilistic methods such as adaptive Monte Carlo localization and extended Kalman filter are used to concurrently build a map of surroundings and a robot's pose estimation. A robot's displacement and orientation are obtained from odometry and inertial navigation system. All algorithms and sub-systems have been implemented and verified with Robot Operation System with a framework for exploration tasks in multi-level buildings.

**Key-Words:** - mobile robot, SLAM 3D, mapping, exploration, localization

Received: December 2, 2020. Revised: July 13, 2021. Accepted: July 27, 2021. Published: August 11, 2021.

### 1 Introduction

The use of autonomous mobile robots in an unknown environment, where a human being presence is dangerous, has known an increasing interest in recent years. To execute efficiently its missions the robot needs a system to see its surrounding area. Nowadays, the simultaneous localization and mapping (SLAM), one of the most active topics in robotics researches, seems to be the attractive solution to this problem [3, 4, 5, 13, 15, 18, 19, 20]. SLAM is the computational technique which allows the mobile robot to generate a map of the investigated environment using its onboard sensors while, at the same time, this map is concurrently used to find its location within it.

The paper consists of five sections. In Section 2 a brief description of the SLAM problem is presented. A mobile robot's control is written in Section 3. In Section 4 results of numerical study are provided. Conclusions are given in Section 5.

### 2 SLAM principles

The SLAM problem is commonly defined by four vectors: controls  $U$ , observations  $Z$ , map  $m$  and robot's pose  $X$  for a period of time from 0 to  $t$  [8, 12]:

$$\begin{aligned} U_{1:t} &= \{U_1, U_2, U_3 \dots, U_t\} \\ Z_{1:t} &= \{Z_1, Z_2, Z_3 \dots, Z_t\} \\ m & \\ X_{0:t} &= \{X_0, X_1, X_2 \dots, X_t\} \end{aligned} \quad (1)$$

where  $U$ ,  $Z$  are given values,  $m$  and  $X$  calculated or estimated. A pose of the robot during planar motion, according to its kinematics, is defined by the following state vector of positions and orientation  $\mathbf{X}^{(2)} = [x \ y \ \psi]^T$  with respect to 2D coordinate frame.

Probabilistic estimation of the robot's pose and the map for every step is named as probability density function and can be described by the following generic form [12, 17]:

$$p(X_{0:t}, m | Z_{1:t}, U_{1:t}) \quad (2)$$

Currently the most often used SLAM algorithm for robot's pose estimation is adaptive Monte-Carlo localization (AMCL). The AMCL is a particular filter, the variant of Markov localization family. It uses Bayes rule to update beliefs – a probability distribution of robot's localization when it is moving or getting information from sensors [3, 16].

By applying Bayes rule and Markov assumption the robot's pose can be described as recursive posterior estimation and presented in the following form [7]:

$$\begin{aligned} p(X_{0:t}, m | Z_{1:t}, U_{1:t}) &= \\ &= \alpha \cdot p(Z_t | X_{0:t}, m, Z_{1:t-1}, U_{1:t}) \cdot p(X_{0:t}, m | Z_{1:t-1}, U_{1:t}) \end{aligned} \quad (3)$$

where  $\alpha$  is a normalization constant which is used to ensure that  $p(X_t | Z_{1:t}, U_{0:t-1})$  is one over all  $X_t$ .

AMCL is generally carried out in two steps: prediction and correction. Prediction  $p(X_t | U_{t-1}, X_{t-1})$  is given from the robot's previous control  $U_{t-1}$  and  $X_{t-1}$  pose while correction  $p(Z_t | X_t)$  is taken from observation  $Z_t$  in position  $X_t$ .

In three-dimensional space, robot's pose describes a state vector consisting of six coordinates: Cartesian  $x, y, z$  and Euler angles  $\varphi, \vartheta, \psi$ :

$$\mathbf{X}^{(3)} = [x \ y \ z \ \varphi \ \vartheta \ \psi]^T \quad (4)$$

The 3D map applied in this work is represented by occupancy grids, which were introduced as binary cells by Moravec et al [16] and Elfes et al [3]. Each cell is presented by a binary random variable which gives a probability of its occupancy in the following way - 0 for free and 1 for definitely occupied. The cells which have been unobserved yet are fulfilled with some default values (e.g. 0.5). In such a way, the map  $m$  is obtained by a product of each cell occupancy probability:

$$\prod_i^N p(m_{x_i, y_i, z_i}) \quad (5)$$

The 3D mapping Octomap framework developed by Hornung et al [7] is used as the basis for our research. The Octomap, instead of quadtree representation commonly used for 2D occupancy

grids, uses an octree hierarchical data structure, where each volume named node has eight child connections with inner nodes [7]. The specific details of three dimensional mapping techniques with occupancy grids can be found in [5, 8, 12, 17, 21].

For exploration and mapping of multi-level buildings the usage of independent SLAM process for each level is required. For that purpose we applied the AMCL-EKF hybrid pose estimation which had been developed for robust pose estimation in isolated terrain and described in details in [14]. This approach, based on data obtained from LIDAR and inertial navigation unit (IMU) sensing, is used together with odometry technique for local pose estimation by n extended Kalman filter (EKF). The global localization is determined by Rao-Blackwellized 2D SLAM algorithm based on particle filter approach [6] with motion data delivered from EKF estimates. The Rao-Blackwellization approach was introduced by Doucet et al in [2]. Its idea is to factorize the SLAM posterior (2) into  $M$  sampled versions, where each particle is a pose with individual landmarks estimated by low-dimensional EKF:

$$\begin{aligned} p(X_{0:t}, m | Z_{1:t}, U_{1:t}) &= p(X_{0:t}, m_{1:M} | Z_{1:t}, U_{1:t}) = \\ &= (X_{0:t}, m_{1:M} | Z_{1:t}, U_{1:t}) \prod_{i=1}^M p(m_i | Z_{1:t}, U_{1:t}) \end{aligned} \quad (6)$$

The Rao-Blackwellized SLAM can be applied for both grids and landmark based maps, and its application gives much faster solution in comparison to EKF one. Robustness of this approach is shown and confirmed in numerous research works [1, 4, 10, 15].

### 3 Problem formulation

The mobile platform used for implementation and testing is a differential drive robot, the typical one, which has two wheels on common axis. Its orientation is controlled by differences of rotation direction and speed of each wheel (see Figure 1).

The control system is based on kinematics of the robot, where the relation between its linear and angular ( $V_x, V_y, \dot{\theta}$ ) velocities to angular velocity of each wheel ( $\omega_r, \omega_l$ ) is the key. Under assumption that wheel parameters like radius of wheels ( $r = r_r = r_l$ ) are known a kinematic model of differential drive in robot's frame can be described as follows:

$$\begin{bmatrix} V_x \\ V_y \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ 0 & 0 \\ -r/2L & -r/2L \end{bmatrix} \begin{bmatrix} \omega_l \\ \omega_r \end{bmatrix} \quad (7)$$

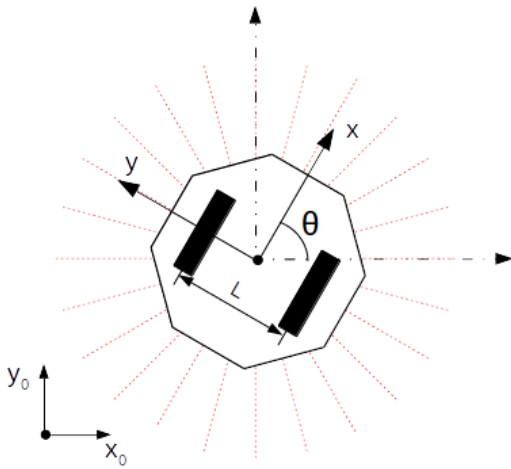


Fig. 1. Two-wheeled mobile robot schema in 2D.

The mobile wheeled robot can operate only in 3 DOF therefore for path planning and state estimation purposes three dimensional state vector can be written as follows:

$$\mathbf{X}^{(3)} = \{\mathbf{X}^{(2)} \quad z \quad \varphi \quad \vartheta\}^T \quad (8)$$

where two-dimensional state vector  $\mathbf{X}^{(2)}$  is used for motion control while the rest of coordinates might be used for robot's tracking in 3D space.

Hence, it is possible to find a hybrid solution suitable to multi-level building exploration. The solution is based on 2D SLAM for precise 2D robot's pose tracking and control as well as on 3D mapping.

For  $\mathbf{X}^{(2)}$  pose estimation purpose the hybrid AMCL-EKF filtering can be applied [14]. Since the mentioned above algorithm besides global  $\mathbf{X}^{(2)}$  also calculates local  $\mathbf{X}^{(3)}$  (see Figure 2), it makes possible to reconstruct global 3D pose and flawlessly use it for further 3D mapping.

#### 4 Implementation and verification

The multilevel 3D inspection system has been implemented and evaluated with ROS software framework and VREP simulator. The wheeled robot and Velodyne LIDAR models which were used in simulation study relate to real equipment that is used in our Lab. The workspace environment scenarios, used in the investigations,

were developed using CAD software and imported to VREP software from STL files (see Fig. 3a and Fig. 3b).

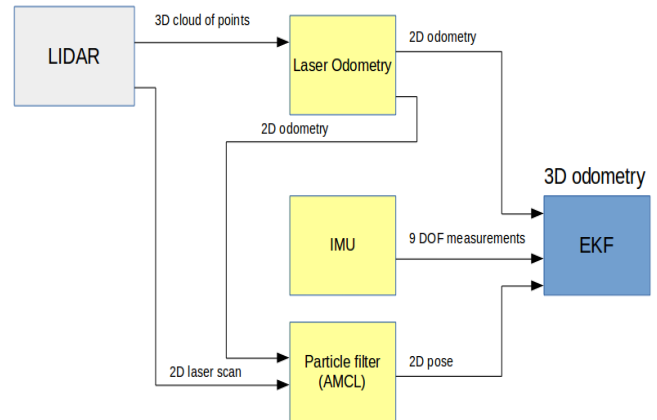
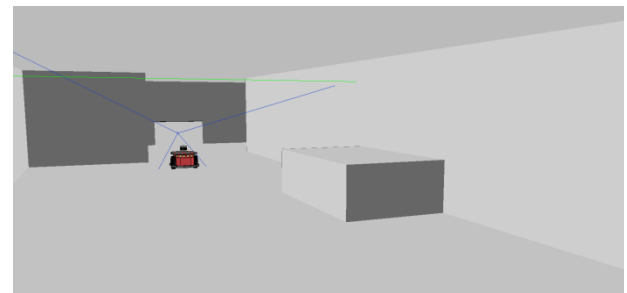
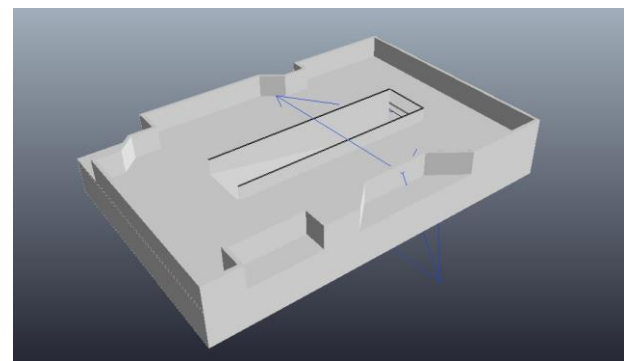


Fig. 2. Local pose estimation: Combined AMCL-EKF techniques for 3 DOF odometry estimations [14].



(a)



(b)

Fig. 3. Multi-level building exploration scene in VREP.

The 3D workspace for the solution regardless of the chosen scene can be also represented as the planar 3D map  $\mathbf{M}_{pl}^{(3)}$  in form of a set of 2D layers, maps of each floor/level of the building connected by the function  $f$  which describes the geometrical relation between neighboring level 2D

maps by the  $l_{z,i}$  (value related to the distance between level bottom surfaces):

$$\mathbf{M}_{pl}^{(3)} = \sum_{i=1}^N f(\mathbf{M}_i^{(2)}, \mathbf{M}_{i-1}^{(2)}, l_{z,i}) \quad (9)$$

Such approach allows to perform 2D SLAM estimation on each level. The main problem is continuous 3D pose tracking and fluent switching between levels. It was solved by applying values of coordinates from (8) to continuous 3D pose tracking under assumption that  $z$  coordinate is related to the  $z$ -axis 'ground-ceiling' distance on the current level and  $\varphi$  angle is used for tracking movement between levels. The  $\mathbf{X}^{(2)}$  is quasi-isolated since on each level  $i$   $\mathbf{X}_{glob,i}^{(2)}$  is connected with others by  $z$  coordinate and is a part of the  $\mathbf{X}_{glob}^{(3)}$  pose creation:

$$\mathbf{X}_{glob,0:t}^{(3)} = \sum_{i=0}^N \left\{ \mathbf{X}_{glob,i}^{(2)} \quad z \quad \varphi \quad \vartheta \right\}_{0:t}^T \quad (10)$$

Since  $\mathbf{X}_{glob}^{(3)}$  and 3D laser data – cloud of points – are known for each while of time the 3D map can be performed. For such conditions the implementation of well-known Octomap algorithm [7] can be applied. However, the calculation of two simultaneous SLAM processes is very costly (time consuming) and the 3D pose is already estimated by the process mentioned before, the only 'mapping' part of 3D SLAM Octomap algorithm is needed. Therefore, the full system can be organized as shown in diagram in Figure 4.

Switching between levels is realized by 'level database' application which stores number of levels information as  $L$  data structure:

$$L = \sum_{i=0}^N l_i \quad (11)$$

where each  $l_i$  structure consists of floor's ground  $l_{gnd}$  and ceiling  $l_{ceil}$  levels, which are given from an octotree map and 2D map of each  $i$  floor  $\mathbf{M}_i^{(2)}$ :

$$l_i = \{l_{gnd} \quad l_{ceil} \quad \mathbf{M}_i^{(2)}\}^T \quad (12)$$

In the initial state  $l_{gnd}$  and  $l_{ceil}$  are calculated and  $z$  variable is tracked to be inside these level borders. When  $z$  starts to change (increase or decrease) along with continuation of the same direction change  $\varphi$  for the system means that the level will be switched, the system saves all data of 2D map for the level. Then, if  $z$  is over the level

limit and  $\varphi$  is close to 'zero' state the level switching is done and the new level initiation procedure is repeated with an additional check under condition that the map of the level already exists in the database. If yes, it is loading immediately and the system works on updating the map instead of building it from a scratch. 3D SLAM with octotree map construction is placed into a separate thread with a conditional independence from 2D mapping. The developed algorithms are shown below as Algorithm 1 and Algorithm 2.

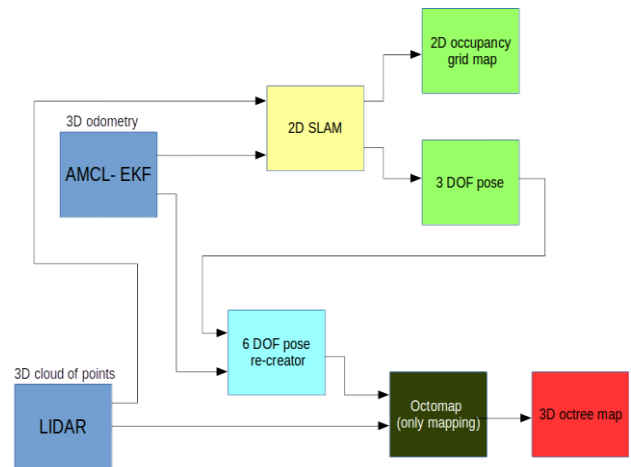


Fig. 4. Developed map building system for a 3D workspace diagram.

Algorithm 1. Multi-level handler. Main loop.

**Require:** 2D SLAM, 3D mapping  
 $i \leftarrow 0$   
 run 2D SLAM handler  
**while** system runs **do**  
    $l_i \leftarrow$  calculate level ( $\mathbf{M}^{(3)}$ )  
   **if**  $l_i \in L$  **then**  
      $\mathbf{M}^{(2)} \leftarrow$  load level ( $l_i$ )  
   **else**  
      $\mathbf{M}^{(2)} \leftarrow 0$   
   **end if**  
    $w \leftarrow$  2D SLAM handler { $w$  - is returned value by 2D SLAM handler associated with level switching}  
   **if**  $w = 1$  or  $w = -1$  **then**  
      $l_i.map \leftarrow \mathbf{M}^{(2)}$   
      $i \leftarrow i + w$   
   **else**  
     **if**  $w = 0$  **then**  
        $\mathbf{M}^{(2)} \leftarrow 0$   
       restart 2D SLAM  
     **end if**  
   **end if**  
**end while**

Algorithm 2. Multi-level handler. 2D SLAM handler.

**Require:** 2D SLAM, multilevel handler main thread  
 $i \leftarrow 0$   
**while** system runs **do**  
  **if**  $z \rightarrow l_{ceil}$  and  $\phi \rightarrow \pi/2$  **then**  
    {all  $z$  and  $\phi$  changes observation must take a while}  
    **repeat**  
       $w \leftarrow 1$   
      **return**  $w$   
    **until**  $z \geq l_{ceil}$  and  $\phi = 0$   
     $w \leftarrow 0$   
    **return**  $w$   
  **else**  
    **if**  $z \rightarrow -\infty$  and  $\phi \rightarrow -\pi/2$  **then**  
      **repeat**  
       $w \leftarrow -1$   
      **return**  $w$   
      **until**  $z \ll l_{gnd}$  and  $\phi = 0$   
       $w \leftarrow 0$   
      **return**  $w$   
    **end if**  
  **end if**  
**end while**

The results of performance check for the proposed solution for the multi-level building scene shown in Figure 3 can be found in Figure 5 where he reconstructed global 3D pose is shown, and Figure 6 where pictured planar 3D map  $M_{pl}^{(3)}$  of the environment.

The 3D map of the whole workspace is built by reduced Octomap algorithm and represented as three-dimensional occupancy grip map and is presented in Figure 7.

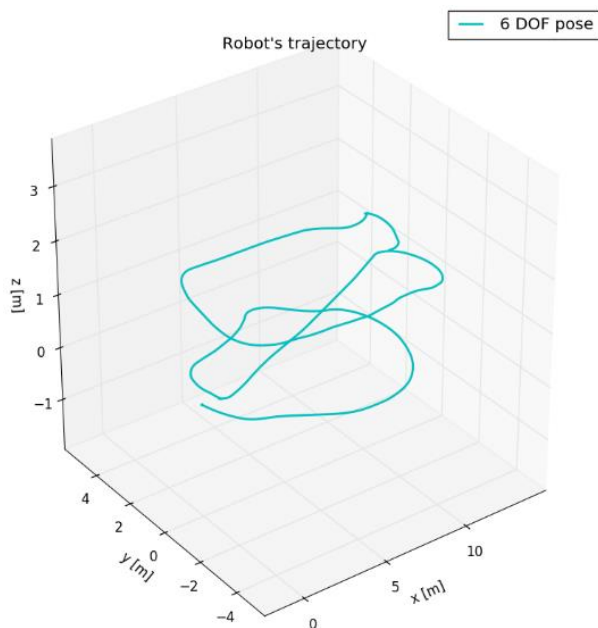


Fig. 5. Global 3D path passed by robot during exploration.

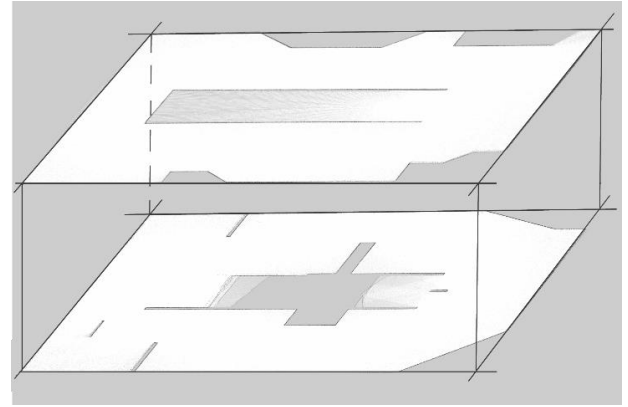


Fig. 6.  $M_{pl}^{(3)}$  map represented by the 2D occupancy grid maps of each level with geometric relations.

Since all maps are represented as a grid, cells might be considered as a graph with fixed topography. Therefore, path planning algorithms like the Dijkstra's algorithm and its extensions can be easily applied. However, the hybrid solution where global path is being calculated by the shortest path finder algorithm like A\* and the local path by a potential field planner is more recommended as it allows to avoid dynamic obstacles during exploration [9, 18, 21].

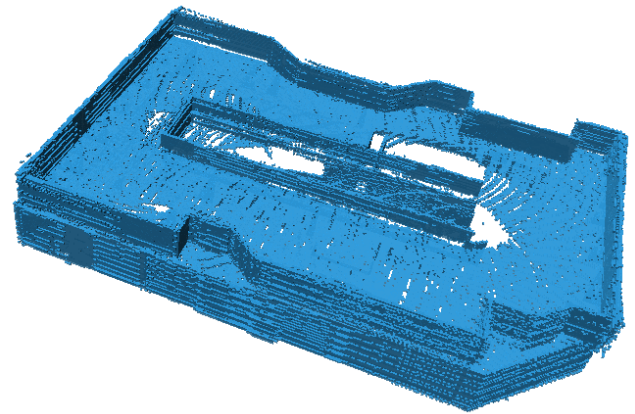


Fig. 7. 3D Octomap built during exploration by the developed multi-level exploration system.

## 5 Conclusions

The system developed and presented in the paper is a working solution of 3D mapping for inspection purposes in the isolated terrain, especially where presence is dangerous and inadvisable for humans.

It can be used for service tasks in public multi-level buildings such as hotels, hospitals and warehouses with traffic requirements between levels and precise pose tracking along with current 3D mapping.

The proposed system is based on laser rangefinder measurements, inertial navigation and two SLAM processes – full 2D SLAM which ensures reliable tracking of pose in isolated terrain and half-way 3D mapping algorithm based on Octomat framework.

The described approach is hierarchical or hybrid, and connects two different world assumptions - static and dynamic – therefore it can be applied to path planning with the best practice usage - shortest path search and reaction motion planning.

The system was implemented and tested with Robot Operation System navigation stack framework. Obtained results confirm correctness of the proposed solution and the possibility of its application in real conditions.

#### References:

- [1] M. Colosi, I. Aloise, T. Guadagnino, D. Schlegel, B. Della Corte, K. O. Arras, G. Grisetti, Plug-and-Play SLAM: A Unified SLAM architecture for modularity and ease of use. In *arXiv preprint*, 2020, arXiv:2003.00754.
- [2] A. Doucet, N. Freitas, K. Murphy, S. Russell, Rao-Blackwellised particle filtering for dynamic bayesian networks. In *Proc. the 16th Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers Inc., 2000, pp. 176–183.
- [3] A. Elfes, Using occupancy grids for mobile robot perception and navigation, *Computer*, Vol. 22, No 6, 1989, pp. 46–57.
- [4] A. Eliazar, R. Parr, DP-SLAM: fast, robust simultaneous localization and mapping without predetermined landmarks. In *Proc. the International Joint Conference on Artificial Intelligence IJCAI'2003*, Vol. 3, 2003, pp. 1135–1142.
- [5] L. Freda, L. M. Gianni, F. Pirri, A. Gaweł, R. Dubé, R. Siegwart, C. Cadena, 3D multi-robot patrolling with a two-level coordination strategy, *Autonomous Robots*, Vol. 43, No 7, 2019, pp. 1747–1779.
- [6] G. Grisetti, C. Stachniss, W. Burgard, Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Transactions on Robotics*, Vol. 23, No 1, 2007, pp. 34–46.
- [7] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, W. Burgard, OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, Vol. 34, No 3, 2013, pp. 189–206.
- [8] A. E. Johnson, S. B. Kang, Registration and integration of textured 3D data, *Image and vision computing*, Vol. 17, No 2, 1999, pp. 135–147.
- [9] S. Kambhampati, L. Davis, Multiresolution path planning for mobile robots. *IEEE Journal on Robotics and Automation*, Vol. 2, No 3, 1986, pp. 135–145.
- [10] C. Kim, H. Kim, W. K. Chung, Exactly Rao-Blackwellized unscented particle filters for slam. In *Proc. IEEE International Conference on Robotics and Automation ICRA'2011*, 2011, pp. 3589–3594.
- [11] R. A. Knepper, S. S. Srinivasa, M. T. Mason, Hierarchical planning architectures for mobile manipulation tasks in indoor environments. In *Proc. of IEEE International Conference on Robotics and Automation ICRA'2010*, 2010, pp. 1985–1990.
- [12] S. Kohlbrecher, J. Meyer, U. Klingauf, O. Stryk, A flexible and scalable SLAM system with full 3D motion estimation. In *Proc. of IEEE International Symposium of the Safety, Security and Rescue Robotics SSR'2011*, 2011, pp. 155–160.
- [13] A. Kudriashov, T. Buratowski, M. Giergiel. Multi-level exploration and 3D mapping with octrees and differential drive robots. *Warsaw University of Technology Press. Electronics*, Vol. 192, 2018, pp. 491–500.
- [14] A. Kudriashov, T. Buratowski, and M. Giergiel, Hybrid AMCL- EKF filtering for SLAM-based pose estimation in rough terrain. In *Proc. the 15th IFToMM World Congress on Advances in Mechanism and Machine Science*, Springer, 2019, pp. 2819–2828.
- [15] M. Montemerlo, S. Thrun, A scalable method for the simultaneous localization and mapping problem in robotics. In *FastSLAM 2.0*, Springer, 2007, pp. 63–90.
- [16] H. P. Moravec, Sensor fusion in certainty grids for mobile robots, *AI Magazine*, Vol. 9, No. 2, 1988, pp. 61–71.
- [17] B. Peasley, S. Birchfield, A. Cunningham, F. Dellaert, Accurate on-line 3D occupancy grids using manhattan world constraints. In *Proc. of IEEE/RSJ International Conference*

*on Intelligent Robots and Systems IROS'2012*, 2012, pp. 5283–5290.

- [18] M. Tarokh, Hybrid intelligent path planning for articulated rovers in rough terrain. *Fuzzy Sets and Systems*, Vol. 159, 2008, pp. 2927-2937
- [19] S. Thrun, W. Burgard, D. Fox, *Probabilistic robotics*. The MIT Press, Cambridge, MA, US, 2005.
- [20] S. Thrun, D. Fox, W. Burgard, F. Dellaert, Robust Monte Carlo localization for mobile robots, *Artificial intelligence*, Vol. 128, No 1-2, 2001, pp. 99-141.
- [21] D. Youakim, P. Cieslak, A. Dornbush, A. Palomer, P. Ridao, M. Likhachev. Multirepresentation, multiheuristic A\* search-based motion planning for a free-floating underwater vehicle-manipulator system in unknown environment. *Journal of Field Robotics*, Vol. 37, No 6, 2020, pp. 925-950.

## **Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)**

This article is published under the terms of the Creative Commons Attribution License 4.0

[https://creativecommons.org/licenses/by/4.0/deed.en\\_US](https://creativecommons.org/licenses/by/4.0/deed.en_US)