Proceedings Paper DOI: 10.58190/icontas.2023.59

PROCEEDINGS OF

INTERNATIONAL CONFERENCE ON NEW TRENDS IN APPLIED SCIENCES

https://proceedings.icontas.org/

International Conference on New Trends in Applied Sciences (ICONTAS'23), Konya, December 1-3, 2023.

VERIFICATION OF GRID BASED FASTSLAM WITH MULTIPLE CANDIDATES OF PARTICLES

Reo YASUDA
Tokyo City University, 0009-0005-2527-1725
g2281459@tcu.ac.jp

Hidetoshi OYA
Tokyo City University, 0000-0003-1046-5832
hide@tcu.ac.jp

Yoshikatsu HOSHI
Tokyo City University, 0009-0000-6216-7170
hoshi@tcu.ac.jp

ABSTRACT: In this paper, we propose a strategy for improving the accuracy of own positions in FastSLAM (Fast-Simultaneous Localization And Mapping). One can see that estimation of self-position and mapping are important for an autonomous mobile robot. As one approach to do this, GPS (Global Positioning System) based self-positioning methods have been proposed. However, GPS-based method has disadvantage that positioning is difficult in the environment where radio waves cannot be received. On the other hand, it is well known that SLAM can perform self-positioning by sensing the external environment based on LiDAR sensors, cameras and other devices. Since SLAM does not require online communication, estimate of self-position and mapping can be performed without being affected by the driving environment. FastSLAM is one of the SLAM algorithms, and samples the path using a particle filter. Each particle represents one possible motion path of robots in the particle filter, and the observed information is used to calculate the weight of each particle and evaluate each path. In FastSLAM, when self-location is determined, a particle is selected by comparing its likelihood as evaluated by the likelihood function. Since this depends on observed information, the reliability of the estimate depends on the accuracy of the sensor. Thus, there is possibility that there are more useful particles among those not selected. In this paper, we propose a new FastSLAM algorithm that selects multiple candidates of particles in FastSLAM algorithm. In the proposed approach, multiple candidates are stored and estimates of self-position is determined by using multiple candidates in the next step. By maintaining multiple particles, more robust self-position estimation and mapping can be performed comparing with the conventional FastSLAM. In this paper, we show the proposed algorithm and numerical simulations are shown to evaluate the performance of the proposed algorithm.

Key words: SLAM, FastSLAM, Particle filter, LiDAR, Multiple candidates of particles

INTRODUCTION

When an autonomous mobile robot is assigned the task of moving to a given destination, it is crucial to perform accurate self-localization and map creation. The use of GPS sensors is well known method for self-localization (Kumar, S., & Moore, K. B., 2002). However, in environments where radio signals cannot be received, such as tunnels, indoor settings and so on, traditional GPS-based positioning becomes challenging. For this problem, Simultaneous Localization and Mapping (SLAM) has been proposed (Fujimoto, M. et al., 2019; Thrun, S. et al., 2005). SLAM does not rely on GPS positioning and can achieve both localization and mapping unaffected by the characteristics of the operating environment (Luo, J. et al., 2018). Among the various techniques for SLAM, the FastSLAM algorithm which incorporates particle filtering has gained attention, because it can address two primary problems: online SLAM problem and full SLAM one (Thrun, S. et al., 2005). However, it is well known that many implementations of FastSLAM require prior knowledge of features (Saitoh, T., & Kuroda, Y., 2009), such as obstacles observed by sensors. Therefore, there is a need for a method that can perform self-localization and environmental map creation without relying on prior knowledge of features. One such approach is "grid based FastSLAM," which integrates the concept of an occupancy grid map, representing the probability of the presence of obstacles, with FastSLAM (Ogawa, T. et al., 2021; Yasuda, R. et al., 2023).

In this paper, we propose a method to improve the accuracy of self-localization and map creation in grid based FastSLAM. In our approach multiple candidates for particles during the self-localization process are considered and these are used to decide the final estimate in the subsequent step. In FastSLAM, the maximum likelihood method is commonly used to determine estimates of self-localization. However, In the present case, we consider the uncertainty in the posterior distribution of the particles when the estimate of self-localization is determined. Then, this information is preserved for the calculations in the next step, so that we can achieve better estimation giving consideration to observation errors. In this study, we vary the number of candidates of values retained and conduct numerical simulations using two different environmental maps to examine how the increase in candidates of estimates affects the estimation accuracy. Estimation accuracy will be evaluated using the Root Mean Squared Error (RMSE).

FastSLAM ALGORITHM

FastSLAM with Single Particle

We introduce the FastSLAM algorithm similar to the one described in reference (Ogawa, T. et al., 2021). The computational process of FastSLAM involves creating an initial set of $\boldsymbol{M}$ candidates of self-localization, denoted as $\boldsymbol{X}^{[\boldsymbol{k}]}$ for $\boldsymbol{k} = \boldsymbol{1}, \dots, \boldsymbol{M}$. Additionally, Corresponding candidates of environmental maps denoted as $\boldsymbol{m}^{[\boldsymbol{k}]}$ for these candidates of self-localizations are prepared. We refer to the combination of these candidates of self-localizations and candidates of environmental maps, denoted as $\left[\boldsymbol{X}^{[\boldsymbol{k}]}, \boldsymbol{m}^{[\boldsymbol{k}]}\right]$, as particles. The candidates of environmental maps $\boldsymbol{m}^{[\boldsymbol{k}]}$ for each particle are structured as occupancy grid maps similar to the environmental map represented as two-dimensional grids. The occupancy grid maps consist of a grid-like structure and use from "0" to"1" to indicate the absence or presence of obstacles, respectively. Grid cells in unexplored areas are assigned a value of 0.5. Therefore, the initial values for all grid cells in both the environmental maps $\boldsymbol{m}$ and $\boldsymbol{m}^{[\boldsymbol{k}]}$, are set to 0.5. It is worth noting that, in FastSLAM, the estimation of self-localization is obtained by updating particles in accordance with the following steps:

STEP 1 (Motion)

Calculate the estimated self-localization $\boldsymbol{X}_{\boldsymbol{t}}^{[\boldsymbol{k}]}$ at time step $\boldsymbol{t}$ based on the self-localization of particles $\boldsymbol{X}_{\boldsymbol{t-1}}^{[\boldsymbol{k}]}$ at time step $\boldsymbol{t-1}$ and the input $\boldsymbol{u}_{\boldsymbol{t}}$ at time step $\boldsymbol{t}$.

STEP 2 (Update)

Update the environmental map $\boldsymbol{m}^{[\boldsymbol{k}]}$ with the positions where obstacles are detected at time step $\boldsymbol{t}$ using the observations $\boldsymbol{z}_{\boldsymbol{t}}$ obtained from sensors and the self-localization estimate $\boldsymbol{X}^{[\boldsymbol{k}]}$ calculated in *STEP 1*.

STEP 3 (Likelihood)

Evaluate the similarity between the environmental maps $\boldsymbol{m}^{[\boldsymbol{k}]}$ for each obtained particle and the environmental map $\boldsymbol{m}$ from the previous time step. This evaluation is used to calculate the likelihood $\boldsymbol{\omega}^{[\mathbf{k}]} = \boldsymbol{p}\left(\boldsymbol{z}_{\boldsymbol{t}} | \boldsymbol{X}_{\boldsymbol{t}}^{[\boldsymbol{k}]}, \boldsymbol{m}^{[\boldsymbol{k}]}\right)$. The likelihood $\boldsymbol{\omega}^{[\boldsymbol{k}]}$ can be calculated as follows:

$$\bar{m} = \frac{1}{2N} \sum_{i=1}^{N} \left(m_i + m_i^{[k]}\right) \tag{1}$$

$$\omega^{[k]} = \frac{\sum_{i=1}^{N} (m_i - \bar{m}) \cdot \left(m_i^{[k]} - \bar{m}\right)}{\sqrt{\sum_{i=1}^{N} (m_i - \bar{m})^2 \cdot \left(m_i^{[k]} - \bar{m}\right)^2}} \tag{2}$$

Here, $\boldsymbol{N}$ represents the total number of grid cells in the occupancy grid map, and $\boldsymbol{m}_{\boldsymbol{i}}$ denotes the value of the grid cell in the environmental map.

STEP 4 (Selection)

Select the particle with the maximum likelihood as the self-localization estimate for time step t. The self-localization estimates X_t at time step t is based on the self-localization $X^{[k]}$ associated with the particle that has the highest likelihood. Furthermore, by using on the obtained self-localization estimate X_t and the sensor observations z_t, the positions of surrounding obstacles are calculated and update the environmental map m is updated.

STEP 5 (Resampling)

Resample particles by selecting them with a probability proportional to the likelihood $\omega^{[k]}$ calculated in *STEP 3*. This resampling process helps update and maintain a set of particles that better represent the current state of the system. In the conventional method (Ogawa, T. et al., 2021), each particle's estimated position is evaluated using past estimations and the likelihood function (*STEP 3*). Furthermore, in *STEP 4*, the single particle with the highest likelihood is selected, and the position information and environmental map associated with that particle are used as the estimation for that time step.

FastSLAM with Multiple Particles

In the conventional FastSLAM (Ogawa, T. et al., 2021), the metric used to obtain the estimate in *STEP 4* is based solely on observation likelihood. As a result, if high-precision sensors are used, the estimation reliability is higher. However, typical sensors include observation noise, which leads to error accumulation as the estimation process progresses, resulting in larger estimation errors. In this study, we address this issue by introducing a new approach. We retain a set of $\boldsymbol{P}$ particles ($\boldsymbol{p = 1, 2, \ldots, P}$) as candidates of estimates, which are determined on the basis of the likelihood function. These candidates of estimates are represented as $\boldsymbol{\widehat{X}_t^{[p]}}$ for self-localization and $\boldsymbol{\widehat{m}_t^{[p]}}$ for the environmental map. The likelihood calculation uses the candidates of the environmental map $\boldsymbol{\widehat{m}_{t-1}^{[p]}}$ to perform the calculation and determine the estimate of self-localization at time $\boldsymbol{t-1}$ along with the environmental map $\boldsymbol{m}$. This approach allows us to perform calculations that consider the uncertainty of the posterior distribution by determining self-localization in this manner. The overall algorithm is same to the updating methods described from *STEP 1* to *STEP 5*, but *STEP 3* and *STEP 4* are modified as follows:

STEP 3 (Likelihood Calculation and Determination)

We evaluate the similarity by calculating the relationship between the obtained environmental map $\boldsymbol{m^{[k]}}$ for each particle and the candidates of environmental map $\boldsymbol{\widehat{m}_{t-1}^{[p]}}$. This evaluation is used to calculate the likelihood $\boldsymbol{\omega^{[k,p]} = p\left(z_t | X_t^{[k]}, \widehat{m}_t^{[k,p]}\right)}$. The likelihood $\boldsymbol{\omega^{[k,p]}}$ can be calculated as:

$$\overline{m}^{[k,p]} = \frac{1}{2N}\sum_{i=1}^{N}\left(\widehat{m}_i^{[p]} + m_i^{[k]}\right) \tag{3}$$

$$\omega^{[k,p]} = \frac{\sum_{i=1}^{N}\left(\widehat{m}_i^{[p]} - \overline{m}^{[p]}\right)\cdot\left(m_i^{[k]} - \overline{m}^{[p]}\right)}{\sqrt{\sum_{i=1}^{N}\left(\widehat{m}_i^{[p]} - \overline{m}^{[p]}\right)^2\cdot\left(m_i^{[k]} - \overline{m}^{[p]}\right)^2}} \tag{4}$$

Additionally, using the candidates of number p associated with the highest likelihood from time step t, we determine the candidates of estimate $\widehat{X}_{t-1}^{[p]}$ for the self-localization at time step $t-1$, for the environmental map m.

STEP 4 (Retention of Candidates of Estimates)

Based on the likelihood obtained in *STEP 3*, retain P candidates of estimates $\widehat{X}_t^{[p]}$ from the self-localization $X_t^{[k]}$ associated with each particle. In addition, calculate the positions of surrounding obstacles from the obtained candidates of estimates and sensor observations z_t, update the candidates of environmental map $\widehat{m}_t^{[p]}$, and retain it.

SIMULATION

Simulation Environment

In this paper, we consider two simulation environments, and these are shown in Figures 1 and 2. Each of the environmental maps represents a 5 [m] by 5 [m] area. In these figures blue point is the starting position and we set waypoints to navigate while avoiding obstacles. During the passage of waypoints in sequence, self-localization and mapping based on FastSLAM are performed. In this simulation is considered, we have provided the initial settings in Table 1. The resolution of the map indicates the side length of each grid cell in the occupancy grid map. Consequently, the number of grid cells, $\boldsymbol{N}$, is calculated as $\boldsymbol{N = 2,500}$, which is obtained by squaring 50 (resulting from dividing one side of the 5 [m] area by a resolution of 0.1). Furthermore, we use 30 or 50 particles

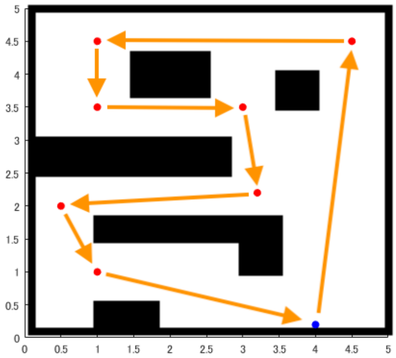

Figure 1. A Environmental map 1 and paths of movement.

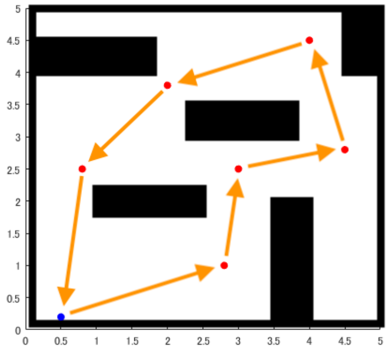

Figure 2. A Environmental map 2 and paths of movement.

Table1. Initial values set(Robot).

Initial angle ($\boldsymbol{\theta}$)	0 [rad]
Velocity ($\boldsymbol{v}$)	0.2 [m/s]
Steering limit ($\boldsymbol{\phi}$)	0.4 [rad]
Wheelbase ($\boldsymbol{l}$)	0.08 [m]

Table 2. Initial values set(Sensor・Map).

Sampling time	0.1 [s]
Sensing distance	1 [m]
Sensing angle	Front 180 [°]
Map resolution	0.01 [$\mathbf{m^2}$]

and their initial values from a normal distribution with a mean of 0 and a standard deviation of 0.1 based on the actual self-localization of the robots.

Robot Model

For robot model, a four-wheel drive model (Taruumi, Y. et al., 2013) described by:

$$[\dot{x} \quad \dot{y} \quad \dot{\theta}]^{\top} = \left[\boldsymbol{v}\ \mathbf{cos}\,\boldsymbol{\theta} \quad \boldsymbol{v}\ \mathbf{sin}\,\boldsymbol{\theta} \quad \frac{\boldsymbol{v}}{\boldsymbol{l}}\mathbf{tan}\,\boldsymbol{\phi}\right] \tag{5}$$

Here, $(\dot{x}, \dot{y})$ represents coordinates of the robot's position, and the other variables are physical quantities shown in Table 2. In the case of a real-world robot, the robot and its sensors are subject to disturbances. Therefore, in this study, the errors are introduced in the form of distance errors proportional to the generated ratio following a normal distribution with a mean of 0 and a standard deviation of 0.1. These errors affect observation distances for the sensors, and observation angles.

Evaluation Method

The evaluation is based on the root mean square error (RMSE) of the actual robot's position $\boldsymbol{X_t}$ and the estimated position $\widetilde{\boldsymbol{X}}_t$ at each step in FastSLAM. RMSE for the total of $\boldsymbol{S}$ steps during FastSLAM execution are defined as:

$$\mathbf{RMSE} = \sqrt{\frac{1}{S}\sum_{t=1}^{S}\left\|X_t - \widetilde{X}_t\right\|^2} \tag{6}$$

Each verification is performed 100 times, and the variability in the results is evaluated using box-and-whisker plots.

Simulation Results

These simulations are conducted with $\mathbf{30}$ and $\mathbf{50}$ particles for two different environmental maps. We set $\boldsymbol{P}$ for each number of particles as follows:

(i) The total number of particles is $\mathbf{30}$ particles, the number of candidates of particles ($\boldsymbol{P}$) is $\mathbf{1, 2, 3, 4, 5, 10, 15, 20}$ **and** $\mathbf{25}$.

(ii) The total number of particles is $\mathbf{50}$ particles, the number of candidates of particles ($\boldsymbol{P}$) is $\mathbf{1, 2, 3, 4, 5, 10, 15, 20, 25}$ **and** $\mathbf{30}$.

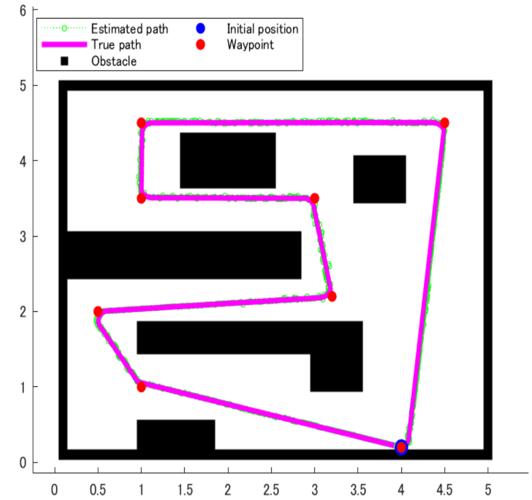

Figure 3. Robot movement trajectory(map 1).

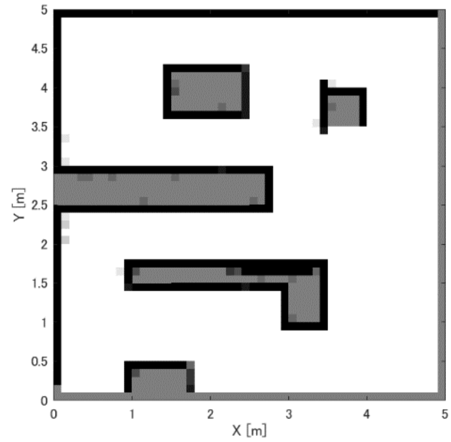

Figure 4. Output of mapping(map 1).

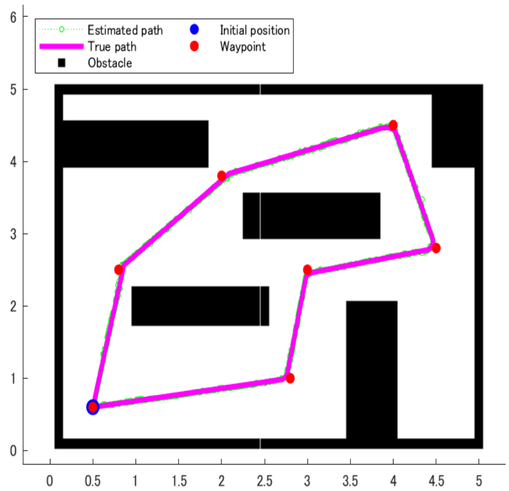

Figure 5. Robot movement trajectory(map 2).

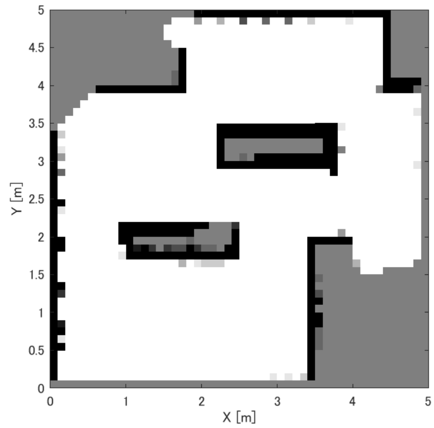

Figure 6. Output of mapping(map 2).

In Figure 3, we show the result for the first environmental map with $\mathbf{50}$ particles and $\mathbf{10}$ candidates of particles, and red line is the actual robot trajectory and green line is the estimated trajectory. Figure 4 is the corresponding map created by the robot. Similarly, in Figure 5, we used the second environment map with $\mathbf{50}$ particles and $\mathbf{10}$ candidates of particles. Figure 6 shows the map created by the robot for this environment. Figures 3 and 5 demonstrate that self-localization performs well, estimating the robot's position accurately from the beginning. Figures 4 and 6 illustrate that the created maps also accurately capture obstacles within the sensor range and are in good agreement with the actual environment maps. In order to analyze simulation results with $\mathbf{50}$ particles, Simulation one creates out $\mathbf{100}$ times and created box-and-whisker plots so as to show the variability in the results. Please note that the results for simulations with $\mathbf{30}$ particles are omitted due to spaces. Figure 7 shows that the RMSE and one can so that RMSE decrease monotonically as the number of candidates of particles increases. The variance of RMSE also decreases as the number of candidates of particles increases. Figure 8 presents similar trends, we find that RMSE decreases with increasing candidates of particles, and the variability also decreases. Figure 9 represents the average of RMSE for different numbers of candidates of particles based on $\mathbf{100}$ simulation runs. This indicates that increasing the number of candidates of particles results in lower average of RMSE, and once a certain threshold of candidates of particles is reached. Namely further increase has leads undesirable for improvement. These results collectively demonstrate that increasing the number of candidates of particles can reduce variability and improve the accuracy of self-localization. However, beyond a certain point, the effect of increasing the number of candidates of particles becomes less pronounced.

CONCLUSION

In this study, we have shown verification result for a grid based FastSLAM for self-localization where multiple candidates are maintained, and the determination of estimates is performed in the subsequent steps. The verification has been performed using two different environmental maps. The results of each verification using

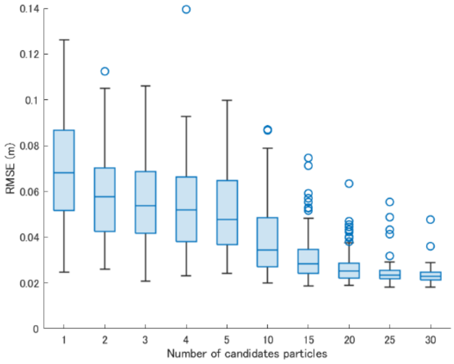

Figure 7. Dispersion of RMSE in 50 particles(map 1).

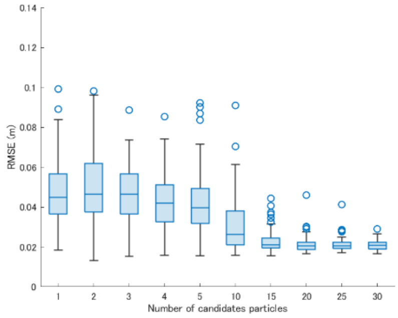

Figure 8. Dispersion of RMSE in 50 particles(map 2).

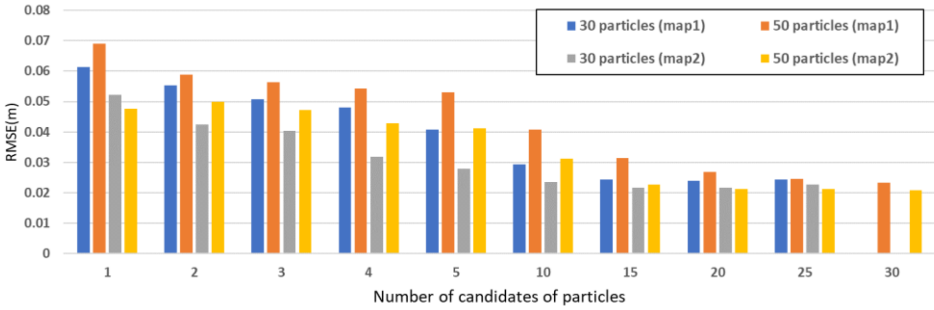

Figure 9. Average RMSE for each validation.

box-and-whisker plots to illustrate the variability has been shown and increasing the number of candidates of particles leads to a reduction in the variance of the RMSE and improvement the self-localization accuracy. From the above, the estimation error can be reduced by calculating multiple candidates of values. Moreover, the effectiveness of increasing the number of candidates of particles has been diminished beyond a certain threshold has been presented.

For future works, there is room to consider a method of candidate of choice, such as a weighted average of particles, and an approach to determine self-localization estimates in cases where multiple candidates are involved.

REFERENCES

Fujimoto, M., Nawa, S., Kawaguchi, N., Araki, N., Sato, T., & Konishi, Y. (2019). Performance Improve Method for Backward Path-Tracking Control of an Articulated Vehicle and its Implementation. *IEEJ Transactions on Electronics Information and Systems*, 139(4), 305-312. DOI: 10.1541/ieejeiss.139.305

Luo, J., Sun, L., & Jia, Y. (2018). A new FastSLAM algorithm based on the unscented particle filter. In *Proceedings of the 2018 Chinese Control And Decision Conference*, 1259–1263. DOI:10.1109/CCDC.2018.8407322

Kumar, S., & Moore, K. B. (2002). The Evolution of Global Positioning System (GPS) Technology. *Journal of Science Education and Technology*, 11(1), March 2002, 59-80.

Ogawa, T., Hoshi, Y., & Oya, H. (2021). Improved self-position estimation accuracy of grid based FastSLAM. *Proceedings of the Institute of Electronics, Information and Communication Engineers general conference*, 2021, 8.

Saitoh, T., & Kuroda, Y. (2009). FastSLAM based Global Path Planning Strategy in Unknown Environment. *Research reports, School of Science and Technology, Meiji University*, 40(96), 1-8.

Taruumi, Y., Hanajima, N., Dai, J., & Takashima, A. (2013). Driving Control of Car-Like Robots by Path-Generating Regulator. *Transactions of the JSME (in Japanese)*, 79(801), 1693-1708. DOI: 10.1299/kikaic.79.1693

Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic Robotics.* MIT Press.

Yasuda, R., Hoshi, Y., & Oya, H. (2023). An Improvement Strategy for Self-Position Estimation Accuracy Based for Grid-Based FastSLAM. *Proceedings of the 67th Annual Conference of the Institute of Systems, Control and Information Engineers*. Kyoto, Japan, 1161-1163.