

# Distributed Multi Robot Simultaneous Localization and Mapping with Consensus Particle Filtering

F Hidayat, B R Trilaksono and H Hindersah

Institut Teknologi Bandung Jl. Ganeca No. 10 Bandung

Email: fadhil.2persen@gmail.com, briyanto@lskk.ee.itb.ac.id,  
hilwadi@stei.itb.ac.id

**Abstract.** These paper present distributed computations of joint probabilities SLAM with consensus particle filtering algorithm. We consider groups of robot observe an unknown environment and build the global maps. In this paper, every local map is a global map. Global maps build by using its own information and the information obtained from the other robots. We use particles likelihood as transfer parameters. The information can be transferred to other robot if reach global agreement on particles weight. The global agreement can be obtained by using the Consensus Particle Filtering algorithm, computation is done locally on each robot. After reaching an agreement, then the global map can be built and information of the global map is owned on each robot.

## 1. Introduction

Simultaneous localization and mapping (SLAM) is a method to build a map (mapping) and determine location on the map (localization) simultaneously using single robot or a group of robots. SLAM is part of three main issue in robot navigation that is mapping (mapping), localization (localization), and path planning (path planning) [1]. Mapping is a problem of integrating environmental information obtained from sensor installed on the robot into map representation. The map representation is formed depending on type of sensor used in mapping. On a feature-based SLAM, a map is a collection of landmarks location. Problem of mapping requires the robot localization. Localization is a problem of determining location of the robot on map. Problems of SLAM are a chicken and egg problem. When estimating location the robot requires map information in other words localization is relative to the map. Otherwise map representations can be formed by knowing location of the robot. Problems of localization and mapping is further complicated because there is no initial location information and no landmarks location information.

Researches on SLAM play an important role in the development of robot technology. Solutions for SLAM using single robot has been developed in various forms. SLAM on a single robot can be solved by the probabilistic approach called probabilistic SLAM. The probabilistic SLAM try to find distribution of robots location and map given observation information obtained from sensors, control input, and robots initial starting position. Extended Kalman Filter (EKF) is an algorithm that can be used to find a solution of single-robot probabilistic SLAM [2,3,4,5]. However, solution of EKF SLAM has a problem on the convergence of maps location, computational complexity, and non-linearity [2,6]. This problem is caused by sample-space SLAM is growing with the increasing of environmental observation results. The problem of EKF SLAM can be resolved with a Rao-Blackwellized SLAM (RB SLAM) [2]. Rao-Blackwellized theorem solves the problem by factoring the SLAM problem into



two forms. First, calculate estimated robots location based on observation results, input control and robots initial position. Next is to build maps based on the robots location and maps that have been formed previously. Mapping can be solved analytically while robot localization solved by sampling. This SLAM's factorization solutions resolve the problem that appears on the EKF SLAM by reducing sample-space of SLAM problem.

Algorithms that implement RB SLAM is FastSLAM 1.0 [7] and FastSLAM 2.0 [8]. FastSLAM algorithm applies Particle Filter algorithm to estimate robots location and EKF algorithm to update map using measurement result obtained from sensor. This algorithm represents localization and mapping in a set of weighted particles [9]. Construction of a map based on information represented by the particle with the highest weight.

SLAM now widely implemented in various fields such as robot mapping for indoor or outdoor environment, unmanned aerial vehicle (UAV) mapping and observe certain areas, unmanned underwater vehicle (UUV) to map sea floor, underground robot to map underground, and space robot to map conditions in outer space. However, single robot SLAM has limitation in mapping capabilities only for small environment. The next challenge of SLAM is to map larger environment. The main idea for SLAM with larger environment is to use more than one robot doing SLAM or also called multi-robot SLAM. Challenges arise in multi-robot SLAM are perform coordination between robots, integrate information obtained from different robots into a single consistent map, perform communication among robots [10], and measure relative position between group robots [11].

Research on multi-robot SLAM has been done by Howard (2006) [11] and A. León et.al (2009) [12] in centralized form. Centralized multi robot SLAM requires a robot that acts central processing while non-center robot provide measurement information from its own sensor. Centralized multi robot SLAM has problems in vulnerable to system failure at the robot center and large transfer parameter. This problem can be resolved by applying the distributed multi-robot SLAM. Distributed multi-robot SLAM is a method to distribute computation of SLAM to all robot members without center processing. Research on distributed multi-robot SLAM has been done in [13,14,15] by accumulating local map obtained from each robot member. Transformations of local maps to global map are done by looking at overlap of local maps; this method is implemented in grid-based map. Implementation of distributed multi-robot SLAM to build feature-based SLAM has been done in [16]. This method requires synchronization of the time. Then, Jafri et.al (2011) [17] and Aragues et.al (2011) [18] propose the method that does not need time synchronization.

All of these previous works are focused on merging the map from member of robot groups. In this paper, we propose distributed multi-robot SLAM using global agreement of information that is transferred among the robot member. We apply the Consensus Particle Filter algorithm [19,20] from multi agent target tracking to reach the global agreement of transfer parameter. We use the particle weight representation as transfer parameter and consensus algorithm is computed to reach the global agreement to particles weight. The tutorial of consensus algorithm can be seen at [21].

In the rest of this paper, formulation of the single-robot SLAM and multi robot SLAM is described in section 2. Consensus particle filtering using particle weight parameter is given in section 3. Section 4 provides the simulation results. Finally, conclusions and discussion are summarized in section 5.

## 2. Methodology

Our approaches to distributed multi robot SLAM shown in algorithm 1. This section provides problem formulation of single-robot SLAM and multi-robot SLAM. The SLAM approach is feature-based; therefore the map is represented by a set of 2D landmarks position.

---

### Algorithm 1 Distributed Multi Robot SLAM with Consensus Particle Filtering Algorithm

---

1. **DistributedMultiRobotSLAMAlgorithm**( ):
  2.     **if** robot  $k$  **is connected** with robot  $k' \rightarrow$  **false**
  3.         **SingleRobotSLAM**( )
  4.     **else if** robot  $k$  **is connected** with robot  $k' \rightarrow$  **true**
-

- 
5. *DistributedMultiRobotSLAM()*
  6. **end if**
- 

### 2.1. Single Robot SLAM

Single robot SLAM is implemented when there is no communication or without having transfer data from other robots. Probabilistic SLAM can be denoted as the following equation (1) and (2).

$$p(x_t, m | z_{0:t}, u_{0:t}) \quad (1)$$

$$p(x_{0:t}, m | z_{0:t}, u_{0:t}) = p(m | x_{0:t}, z_{0:t}) p(x_{0:t} | z_{0:t}, u_{0:t}) \quad (2)$$

Where  $x_t$  is vector of robot position at time t,  $m$  is map mad up from location of landmarks,  $z_{0:t}$  is sensor observation, and  $u_{0:t}$  is input control given to the robot. Prediction stage is estimate pose robot based on the previous step, denoted as equation (3).

$$p(x_t, m | z_{0:t}, u_{0:t}, x_0) = \int p(x_t | x_{t-1}, u_t) \cdot p(x_{t-1}, m | z_{0:t-1}, u_{0:t-1}, x_0) dx_{t-1} \quad (3)$$

Update phase make corrections to prediction result using the results of observations, denoted as equation (4).

$$p(x_t, m | z_{0:t}, u_{0:t}, x_0) = \frac{p(z_t | x_t, m) \cdot p(x_t, m | z_{0:t-1}, u_{0:t}, x_0)}{p(z_t | z_{0:t-1}, u_{0:t})} \quad (4)$$

*Bayes*

In FastSLAM 1.0 algorithms [7], robot trajectory represented using weighted sample and map computed analytically. By using samples weight  $w$ , then the joint posterior distribution SLAM represented as equation (5) follows.

$$\{w_t^i, x_{0:t}^i, p(m | x_{0:t}^i, z_{0:t})\}_i^N \quad (5)$$

In equation (5),  $N$  is the number of sample particle, and  $i$  is index of the particle. Prediction phase can be done by spread the sample particle according to proposal distribution (6).

$$x_t^i \sim \pi(x_t | x_{0:t-1}^i, u_t) \quad (6)$$

Map building can be written as equation (7).

$$p(m | x_{0:t}^i, z_{0:t}) = \prod_t^T p(m_k | x_{0:t}^i, z_{0:t}) \quad (7)$$

Particle weight can be computed by using equation (8).

$$w_t^i = w_{t-1}^i p(z_t | x_{0:t}^i, z_{0:t-1}) \left( \frac{p(x_t^i | x_{t-1}^i, u_t)}{\pi(x_t^i | x_{0:t-1}^i, u_t)} \right) \quad (8)$$

Data association problem can be solved by applying equation (9).

$$\hat{n}_t^i = \underset{n_t}{\operatorname{argmax}} w_t^i(n_t) \quad (9)$$

### 2.2. Multi Robot SLAM

Algorithms for distributed multi robot SLAM with consensus particle filtering is shown in algorithm 2.

---

**Algorithm 2** Distributed Multi Robot SLAM with Consensus Particle Filtering Algorithm

---

1. ***DitributedMultiRobotSLAM Algorithm()*** :
  2. *Prediction*  $x_t^i \sim \pi(x_t | x_{0:t-1}^i, z_{0:t}, u_t)$
  3. *Check Connection with Robot k'*
  4. *Measure Relative Position*
  5. *Transfer and get information from robot k'*
  6. *Consensus Particle Filtering on Particles Weight*
  7. *Built global map using approximation of global weight*
- 

To perform multi-robot SLAM, each robot needs to measure relative position of other robot  $k'$  using its own position  $k$ . Method to perform the relative position measurement can be seen at [14]. Assume that there are two robot  $i$  and  $j$  in multi-robot SLAM system communicate each other and

transferring the information  $d_{j,1:t}^{j0}$  and  $d_{i,1:t}^{i0}$  respectively. Superscript  $j0$  and  $i0$  means that the information is form in local frame reference of recipient robot. The objective of measuring relative position of the robot is to find the global frame reference using roto-translation method written as equation (10).

$$X_{j0}^{i0} = X_i^{i0} \oplus X_j^i \ominus X_i^{j0} \quad (10)$$

Each robot compute local particle weight equation (8) locally and transfer the local weight to other robots. Information obtained from robot  $i$  can be written as equation (11) and (12).

$$d_{i,1:t}^{i0} = \langle w_{i,1:t}^{i0}, x_{i,1:t}^{i0}, p(z_t^{i0} | x_{i,0:t}^{i0}, z_{i,0:t-1}^{i0}) \rangle_{i=1}^N \quad (11)$$

$$p(z_{i,t}^{i0} | x_{i,0:t}^{i0}, z_{i,0:t-1}^{i0}) = \int p(z_{i,t}^{i0} | x_{i,0:t}^{i0}, m^{i0}) p(m^{i0} | x_{i,0:t-1}^{i0}, z_{i,0:t-1}^{i0}) dm \quad (12)$$

After receiving the information from other robot, consensus particle filtering is executed to calculate global agreement of particle weight called global weights. When reaching consensus then compute importance weights by using global weight. Hence, equation (8) can be rewritten by equation (13).

$$w_{k,t}^i = w_{k,t-1}^i \prod_{k=1}^K p(z_{k,t}^{i0} | x_{k,0:t}^{(j)i0}, z_{k,0:t-1}^{i0}) \frac{p(x_{k,t}^{(j)} | x_{k,t-1}^{(j)}, u_{k,t})}{\pi(x_{k,t}^{(j)} | x_{k,0:t-1}^{(j)}, u_k)} \quad (13)$$

Data association problem in equation (9) can be rewritten as equation (14).

$$\hat{n}_t^i = \underset{n_t}{\operatorname{argmax}} w_{k,t}^i(n_t) \quad (14)$$

### 3. Consensus Particle Filtering

This paper use consensus particle filtering with consensus particle weight [19] adopted from target tracking problem. Assume there are  $K$  number of agent in agent network and each member of  $K$  having its own scalar value  $s_k$  and its own weight variable  $\omega_{k,k}^{(i)}$ . General consensus algorithm described at algorithm 3.

---

#### Algorithm 3 General Consensus Algorithm

---

1. **Consensus Algorithm**( $s_k, s_{k'}$ ):
  2.  $\xi_k^{(0)} = s_k$
  3. *for*  $i = 1, 2, \dots, \infty$  *then*
  4.  $\xi_k^{(i)} = u(\xi_k^{(i-1)}, \{\xi_{k'}^{(i-1)}\}_{k' \in Nk})$  /\*  $\{\xi_{k'}^{(i-1)}\}_{k' \in Nk} : s_k$  neighboring agents\*/
  5. **broadcast**( $\xi_k^{(i)}$ )
  6. *if* (**consensus**) *then end for*.
- 

We use averaging consensus algorithms for the update function  $u(\cdot)$  written as equation (15).

$$u(\xi_k^{(i)}, \{\xi_{k'}^{(i)}\}_{k' \in Nk}) = \omega_{k,k}^{(i)} \xi_k^{(i)} + \sum_{k' \in Nk} \omega_{k,k'}^{(i)} \xi_{k'}^{(i)} \quad (15)$$

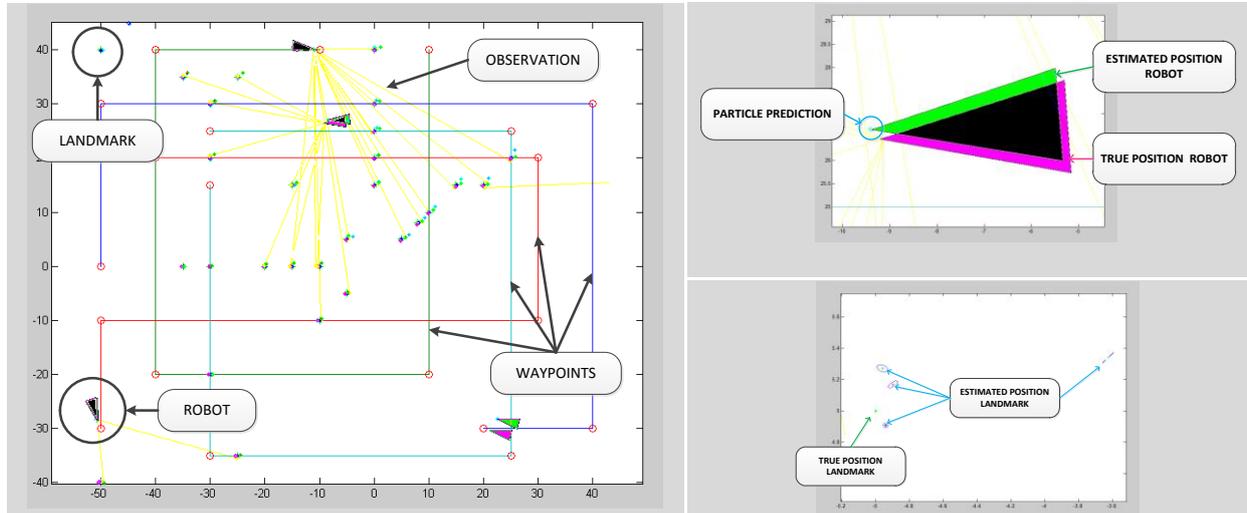
Global weight  $w_t^i$  of particle with index  $i$  at time  $t$  is proportional to product of local likelihood, described in equation (13). The averaging consensus (15) performed for each particle to calculate approximation to average of all particles weight.

$$w_t^{(j)} \triangleq w_{t-1}^{(j)} \frac{1}{K} \sum_{k=1}^K w_{k,t}^{(j)}, \quad j \in \{1, 2, \dots, N\} \quad (16)$$

### 4. Simulation

In order to show the performance of the algorithm, we built simulation of multi-robot SLAM by modified FastSLAM simulation obtained from [22]. The simulation consists of 4 robots exploring 50 x 50 m<sup>2</sup> environment with 29 landmarks. We assume that configuration of each robot are equal. Each robot use 50 particles for prediction phase. The simulation use velocity and steering angle as control variable. Initial noise to velocity is 0.3 m/s and initial noise to steering angle is  $3\pi/180$  radian. Maximum observation range is 30 m and initial noises to observation are 0.1 m for range and  $1\pi/180$

radian for bearing. Time interval between control signals is 0.025 second and time interval between observations is 0.2 second. Maximum distance of communication range between robots is 30 meter. Number of consensus iteration is 5 iterations. The simulation configuration is shown in figure 1.



**Figure 1.** A team consist of 4 robots explore 30 landmarks in 50x50 m<sup>2</sup> environment.

We compare result of the simulation with simulation of single robot SLAM. The comparison result shown in table 1.

**Table 1.** Simulation result.

Description	Loops	Run Time	Landmarks	Map RMSE		
				Location	Position	Orientation
<b>Single Robot</b>	<b>32.412</b>	<b>60.197 s</b>	<b>29 Total</b>			
Robot 1	32.412	± 0,464 s / loop	19 Observed	1,229 m	2,209 m	0,035 π/180 rad
Robot 2	32.412	± 0, 464 s / loop	29 Observed	0,160 m	0,015 m	0,003 π/180 rad
Robot 3	32.412	± 0, 464 s / loop	27 Observed	0,071 m	0,250 m	0,004 π/180 rad
Robot 4	32.412	± 0, 464 s / loop	27 Observed	1,483 m	0,230 m	0,003 π/180 rad
<b>Multi Robot</b>	<b>32.412</b>	<b>72.317 s</b>	<b>29 Total</b>			
Robot 1	32.412	± 0,558 s / loop	29 Observed	0,785 m	1,758 m	0,051 π/180 rad
Robot 2	32.412	± 0,558 s / loop	29 Observed	0,179 m	0,139 m	0,005 π/180 rad
Robot 3	32.412	± 0,558 s / loop	29 Observed	0,139 m	0,336 m	0,006 π/180 rad
Robot 4	32.412	± 0,558 s / loop	29 Observed	0,139 m	1,253 m	0,022 π/180 rad

Number of communication between robots is shown in table 2.

**Table 2.** Number of communication between robots.

Comunication	Robot 1	Robot 2	Robot 3	Robot 4
<b>Robot 1</b>	-	0	88	196
<b>Robot 2</b>	0	-	134	171
<b>Robot 3</b>	88	134	-	256
<b>Robot 4</b>	196	171	256	-

Regarding to the simulation result, multi-robot SLAM with consensus particle filtering improve map estimation. This method also merge the observation between robots.

## 5. Conclusion

We have presented the algorithm for multi-robot SLAM with consensus particle filtering. These methods improve map estimation and merge the observation between robots. However, increasing of computing time cannot be avoided. The increase in computing time depends on the number of iterations used in consensus algorithm.

## Acknowledgements

This work was funded by Ministry of Research, Technology and Higher Education of the Republic of Indonesia. The authors gratefully acknowledge them for providing the funding.

## References

- [1] C. Stachniss, *Robotic Mapping and Exploration*, B. Siciliano, O. Khatib, and F. Groen, Eds. Berlin, Germany: Springer, 2009.
- [2] Hugh Durrant-Whyte and Tim Bailey, "Simultaneous Localisation and Mapping," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99-110, June 2006.
- [3] M.W.M.G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba, "A Solution to the Simultaneous Localization and Map Building (SLAM) Problem," *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*, vol. 17, no. 3, pp. 229 - 241, Agustus 2002.
- [4] Jiang Yan, Liu Guorong, Luo Shenghua, and Zhou Lian, "A Review on Localization and Mapping Algorithm Based on Extended Kalman Filtering," in *International Forum on Information Technology and Application*, vol. 2, Chengdu, 2009, pp. 435-440.
- [5] G. Dissanayake, S. Huang, Z. Wang, and R. Ranasinghe, "A Review of Recent Developments in Simultaneous Localization and Mapping," in *IEEE International Conference on Industrial and Information Systems*, 2011, pp. 477 - 482.
- [6] Tim Bailey and Hugh Durrant-Whyte, "Tutorial Simultaneous Localization and Mapping (SLAM): Part II," *IEEE Robotics & Automation Magazine*, pp. 108-117, September 2006.
- [7] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," in *AAAI National Conference on Artificial Intelligence*, 2002, pp. 593-598.
- [8] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM 2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges," in *IJCAI International Joint Conference on Artificial Intelligence*, 2003, pp. 1151-1156.
- [9] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp, "A Tutorial on Particle Filter for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174-188, February 2002.
- [10] D. Fox et al., "Distributed Multirobot Exploration and Mapping," in *Proceedings of the IEEE*, vol. 94, July 2006, pp. 1325-1339.
- [11] Andrew Howard, "Multi-Robot Simultaneous Localization and Mapping using Particle Filter," *ACM Journal International of Robotics Research*, vol. 25, no. 12, pp. 1243-1256, December 2006.
- [12] A. León, R. Barea, L.M. Bergasa, M. Ocana E. Lopez, and D. Schleicher, "SLAM and Map Merging," *Journal of Physical Agents*, vol. 3, no. 1, pp. 11-23, January 2009.
- [13] S. Carpin, "Fast and Accurate Map Merging for Multi-Robot Systems," *Autonomous Robots*, vol. 25, no. 3, pp. 305-316, 2008.
- [14] L. Carlone, M. K. Ng, J. Du, B. Bona, and M. Indri, "Rao-Blackwellized Particle Filters Multi Robot SLAM with Unknown Initial Correspondences and Limited Communication," in *IEEE International Conference on Robotics and Automation Anchorage Convention District, Anchorage*, 2010, pp. 243 - 249.
- [15] Luca Carlone, Miguel Kaouk Ng, Jingjing Du, Basilio Bona, and Marina Indri, "Simultaneous Localization and Mapping Using Rao-Blackwellized Particle Filters in Multi Robot Systems," *Journal of Intelligent & Robotic Systems*, vol. 63, no. 2, pp. 283 - 307, August 2011.

- [16] Arturo Gil, Oscar Reinoso, Monica Ballesta, and Miguel Julia, "Multi-Robot Visual SLAM Using a Rao-Blackwellized Particle Filter," *Elsevier Robotics and Autonomous System*, vol. 58, pp. 68-80, 2010.
- [17] Syed Riaz un Nabi Jafri, Luca Brayda, and Ryad Chellali, "Distributed Feature Based RBPF Multi Robot SLAM," in *IEEE Conference on Robotics and Biomedics*, Phuket, Thailand, 2011, pp. 7-11.
- [18] R. Aragues, J. Cortes, and C. Sagues, "Distributed Consensus Algorithms for Merging Feature-Based Maps with Limited Communication," *Elsevier Robotics and Autonomous Systems*, vol. 59, pp. 163-180, 2011.
- [19] O. Hlinka, F. Hlawatsch, and P.M. Djuric, "Distributed particle filtering in agent networks: A survey, classification, and comparison," *Signal Processing Magazine*, vol. 30, no. 1, pp. 61 - 81, December 2012.
- [20] O. Hlinka, O. Sluciak, and F. Hlawatsch, "Likelihood Consensus and Its Application to Distributed Particle Filtering," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4334-4349, August 2012.
- [21] R. Olfati-Saber, A. Fax, and R. M. Murray, "Consensus and Cooperation in Networked Multi-Agent System," in *IEEE Proceedings*, 2006.
- [22] Tim Bailey. <https://openslam.org/>. [Online]. <https://openslam.org/>