

# Gaussian random field-based log odds occupancy mapping

Hongjun Li\*, Miguel Barão\*,<sup>†</sup> and Luís Rato\*

\*Department of Informatics, University of Evora, Evora, Portugal

<sup>†</sup>Control of Dynamical Systems Group, INESC-ID, Lisboa, Portugal

li.hongjun@foxmail.com, mjsb@uevora.pt, lmr@uevora.pt

**Abstract**—This paper focuses on mapping problem with known robot pose in static environments and proposes a Gaussian random field-based log odds occupancy mapping (GRF-LOOM). In this method, occupancy probability is regarded as an unknown parameter and the dependence between parameters are considered. Given measurements and the dependence, the parameters of not only observed space but also unobserved space can be predicted. The occupancy probabilities in log odds form are regarded as a GRF. This mapping task can be solved by the well-known prediction equation in Gaussian processes, which involves an inverse problem. Instead of the prediction equation, a new recursive algorithm is also proposed to avoid the inverse problem. Finally, the proposed method is evaluated in simulations.

**Index Terms**—Binary Bayes filter, Gaussian random field, Log odds occupancy mapping.

## I. INTRODUCTION

When robots explore unknown environments, mapping is the fundamental problem. Without an accurate map, they cannot do further work, such as navigation. Occupancy grid maps [1] have been widely applied to represent environments by many researchers. In occupancy grid maps, environments are divided into many grid cells and it is convenient to do path planning. Normally, the grid cells are assumed to be independent of each other, which leads to inconsistent mapping.

Gaussian random fields have been applied to consider the dependence between grid cells. Gaussian process occupancy map (GPOM) [2] is a continuous occupancy representation of the environment, which overcomes some of the limitations with occupancy grids. It considers the occupancy mapping as a binary classification problem and can predict the classification of unobserved space based on observed space. With increasing the number of training data, Gaussian processes will take more time to deal with the problem. For large-scale environments, training data is divided into small subsets and a mixture of Gaussian processes is presented in [3]. Similarly, local Gaussian processes are applied to the subsets and overlapping clusters is proposed to ensure continuity [4]. Reference [5] proposes a recursive method to update occupancy maps and surface meshes using Gaussian processes and Bayesian Committee Machines. A multi-support kernel, which enables

traditional covariance functions to accept two-dimensional regions, is introduced to reduce the size of covariance matrices and accelerate Gaussian process inference and learning [6]. Reference [7] proposes a nested Bayesian committee machine to online learning 3D occupancy maps using Gaussian processes.

In GPOMs, observations have two possible values: occupied and free. These values are in discrete space. In this paper, the values are extended into continuous space based on log odds form [8]. By log odds form, probability with range  $(0, 1)$  is transformed into  $(-\infty, +\infty)$ . In this paper, a GRF-LOOM is proposed taking the advantages of the log odds form. In Section II, occupancy grid mapping, binary Bayes filter, and log odds form are introduced. The new method is proposed in Section III. The map is regarded as a GRF where the random variables are occupancy probabilities in log odds form. The training data for the GRF is obtained from the result of binary Bayes filter. A GRF model is built based on Bayes rule. Based on Sherman-Morrison equation, a novel algorithm is also proposed to solve the mapping problem, which can avoid the inverse problem. Simulations are done in IV.

## II. BACKGROUND

In occupancy grid mapping, maps are divided into many grid cells as Figure 1. Each grid cell has two possible states: free and occupied. The binary occupancy value of each grid cell specifies whether or not a location is occupied with an object. The darkness of each grid cell corresponds to the likelihood of occupancy. If one grid cell is free surely, the darkness is 0. If it is occupied surely, the darkness is 1.

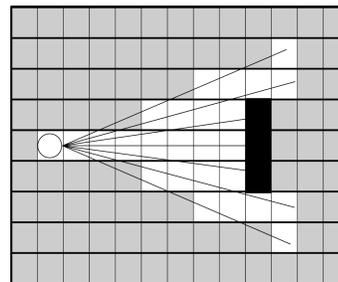


Fig. 1: A grid map

In classical occupancy grid mapping, the states between different grid cells are assumed to be independent of each

This work was supported by EACEA under the Erasmus Mundus Action 2, Strand 1 project LEADER - Links in Europe and Asia for engineering, eEducation, Enterprise and Research exchanges.

other, which is call state dependence in this paper. Every grid is regarded as a binary random variable. The probability that  $m_i$  is occupied is denoted by  $p(m_i)$  and its opposite is denoted by  $p(\bar{m}_i)$ . The odds of the occupancy state is defined as

$$\frac{p(m_i)}{p(\bar{m}_i)}, \quad (1)$$

where  $p(\bar{m}_i) > 0$ . The log odds form [8] is defined as

$$\log \frac{p(m_i)}{p(\bar{m}_i)}. \quad (2)$$

The range of the log odds form is  $(-\infty, +\infty)$ .

Assume measurements in different time  $t$  are represented by  $z^{1:t} = \{z^1, z^1, \dots, z^t\}$ . Based on Bayes rule, the posterior occupancy probability  $p(m_i | z^{1:t})$  of each grid cell is formulated recursively as

$$p(m_i | z^{1:t}) = \frac{p(z^t | m_i)p(m_i | z^{1:t-1})}{p(z^t | z^{1:t-1})}, \quad (3)$$

where  $p(m_i | z^{1:t})$  is a normalizer and formulated as

$$p(z^t | z^{1:t-1}) = p(z^t | m_i)p(m_i | z^{1:t-1}) + p(z^t | \bar{m}_i)p(\bar{m}_i | z^{1:t-1}). \quad (4)$$

$p(z^t | m_i)$  and  $p(z^t | \bar{m}_i)$  are the probability of measurement  $z^t$  conditional on one grid cell  $m_i$ . By analogy with (3), the probability of free state is formulated as

$$p(\bar{m}_i | z^{1:t}) = \frac{p(z^t | \bar{m}_i)p(\bar{m}_i | z^{1:t-1})}{p(z^t | z^{1:t-1})}. \quad (5)$$

Dividing (3) by (5), the odds of  $p(m_i | z^{1:t})$  is formulated as

$$\frac{p(m_i | z^{1:t})}{p(\bar{m}_i | z^{1:t})} = \frac{p(z^t | m_i)}{p(z^t | \bar{m}_i)} \frac{p(m_i | z^{1:t-1})}{p(\bar{m}_i | z^{1:t-1})}. \quad (6)$$

In the same manner, we can obtain the odds forms of  $p(m_i | z^{1:t-1}), \dots, p(m_i | z^1)$ . Combining these odds forms, (6) can be rewritten as

$$\frac{p(m_i | z^{1:t})}{p(\bar{m}_i | z^{1:t})} = \frac{p(z^t | m_i)}{p(z^t | \bar{m}_i)} \dots \frac{p(z^1 | m_i)}{p(z^1 | \bar{m}_i)} \frac{p(m_i)}{p(\bar{m}_i)}, \quad (7)$$

where  $p(m_i)$  and  $p(\bar{m}_i)$  are the initial beliefs of occupied state and free state. In log odds form, it becomes

$$\begin{aligned} \log \frac{p(m_i | z^{1:t})}{p(\bar{m}_i | z^{1:t})} &= \log \frac{p(z^t | m_i)}{p(z^t | \bar{m}_i)} + \dots + \\ &\log \frac{p(z^1 | m_i)}{p(z^1 | \bar{m}_i)} + \log \frac{p(m_i)}{p(\bar{m}_i)}. \end{aligned} \quad (8)$$

In log odds form, the binary Bayes filter is additive.

### III. PROPOSED METHOD

Now the occupancy probability is regarded as an unknown parameter. In order to distinguish from  $p(m_i)$ , the 'true' occupancy probability of the central point of grid cell  $m_i$  is denoted by  $p'(m_i)$ . Its log odds occupancy  $l_i$  is defined as

$$l_i = \frac{p'(m_i)}{1 - p'(m_i)}, \quad (9)$$

which is regarded as one random variable in a GRF. A GRF describes the distribution of random variables in continuous space. Given some training data, a GRF can be used to predict any point in this field.

The noisy training data is  $O_i = \log \frac{p(m_i | z^{1:t})}{p(\bar{m}_i | z^{1:t})}$ , which is the log odds form of a posterior distribution conditional on several observations. It is not easy to decide if one point is observed several times. Occupancy grid mapping is applied to obtain the training data. Every observed grid cell is represented by its central point.

Occupancy grid mapping is efficient computationally with the state independence between  $m_i$ . In the GRF, the random variables  $l_i$  are not independent. However,  $l_i$  is a transformation of the occupancy probability of  $m_i$ , which is a parameter of  $m_i$ . In other words,  $l_i$  is not a function of  $m_i$ . As a result, the state independence and dependence between  $l_i$  can hold at the same time.

As a GRF with noisy observation, the predictive equation with a inverse matrix is given in [9]. When there are more training points, the inverse matrix should be computed again. Computing inverse matrices is a problem to robots with less powerful micro-controllers. In this paper, a new algorithm is described as following.

#### A. GRF model

Assume training points and unknown points of interesting are represented by a vector  $l = [l_1, \dots, l_i, \dots, l_n]^T$ , which is called a log odds occupancy field (LOOF) in this paper. The unknown points need not to be chosen with regular distances. As a GRF, the prior distribution can be formulated as

$$p(l) = \frac{1}{\sqrt{2\pi}^n |K|} \exp\left(-\frac{(l - \mu)^T K^{-1} (l - \mu)}{2}\right), \quad (10)$$

where  $\mu = [\mu_1, \dots, \mu_i, \dots, \mu_n]^T$  is the mean vector and  $K$  is the covariance matrix which is nonsingular. The covariance function are specified by square exponential function

$$\mathcal{C}(c, c') = \exp\left(-\frac{|c - c'|^2}{2\ell}\right), \quad (11)$$

where  $c$  and  $c'$  are the corresponding coordinates of two random variables and  $\ell$  is the length scale.

For one point, its noisy observation is formulated as

$$O_i = l_i + \varepsilon, \quad (12)$$

where  $\varepsilon \sim \mathcal{N}(0, \sigma_i^2)$  is Gaussian distributed with zero mean and variance  $\sigma_i^2$ . If one point is unknown, its observation is

$$O_i = \mu_i + \varepsilon, \quad (13)$$

and  $\sigma_i^2 \rightarrow \infty$ . Assume the coordinate set of training points is denoted by  $\mathcal{I}$ , the likelihood is formulated as

$$p(O|l) = \frac{1}{\prod_{i \in \mathcal{I}} \sqrt{2\pi\sigma_i^2}} \exp\left(-\sum_{i \in \mathcal{I}} (l_i - O_i)^2 / 2\sigma_i^2\right). \quad (14)$$

Assume all the observations is denoted by  $O = [O_1, \dots, O_i, \dots, O_n]^T$ . The likelihood can also be rewritten as

$$p(O | l) = \frac{1}{\prod_{i \in \mathcal{I}} \sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(l - O)^T \Lambda' (l - O)}{2}\right), \quad (15)$$

where  $\Lambda'$  is a diagonal matrix. For the unknown points, they are not considered in the likelihood and the corresponding elements in  $\Lambda'$  are 0.

Based on Bayes rule, the posterior distribution is formulated as

$$P(l | O) = \eta \exp(-E(l)), \quad (16)$$

where  $\eta$  is a constant and

$$E(l) = (l - \mu)^T K^{-1} (l - \mu) / 2 + (l - O)^T \Lambda' (l - O) / 2. \quad (17)$$

### B. Prediction

Maximizing the posterior distribution  $p(l | O)$  or equivalently minimizing  $E(l)$ , the best prediction will be obtained. The derivative of  $E(l)$  with respect to  $l$  is formulated as

$$\frac{d}{dl} E(l) = K^{-1} (l - \mu) + \Lambda' (l - O) \quad (18)$$

$$= K^{-1} (l - \mu) + \Lambda' (l - \mu) - \Lambda' (O - \mu). \quad (19)$$

Let the derivatives  $\frac{d}{dl} E(l)$  be zero, a linear equation set is obtained and formulated as

$$(K^{-1} + \Lambda') (l - \mu) = \Lambda' (O - \mu), \quad (20)$$

where  $K^{-1} + \Lambda'$  is nonsingular. The mapping problem can be solved as

$$\begin{aligned} l &= \mu + (K^{-1} + \Lambda')^{-1} \Lambda' (O - \mu) \\ &= \mu + \mathcal{D} \Lambda' (O - \mu), \end{aligned} \quad (21)$$

where  $\mathcal{D} = (K^{-1} + \Lambda')^{-1}$ .

At the beginning, there is no observation,  $\Lambda'$  is a zero matrix and  $\mathcal{D} = K$ . If one training point is obtained and the corresponding variance is  $\sigma_j^2$ , based on Sherman-Morrison equation, the inverse matrix can be formulated as

$$\mathcal{D} = (K^{-1} + ab^T)^{-1} = K - \frac{Kab^TK}{1 + b^TKa}, \quad (22)$$

where  $a = [\dots, 0, 1, 0, \dots]^T$  and  $b = [\dots, 0, 1/\sigma_j^2, 0, \dots]^T$ . If another training point is obtained and its variance is  $\sigma_*^2$ , we have

$$\mathcal{D} = (K^{-1} + ab^T + a_*b_*^T)^{-1}, \quad (23)$$

where  $a_* = [\dots, 0, 1, 0, \dots]^T$  and  $b_* = [\dots, 0, 1/\sigma_*^2, 0, \dots]^T$ . Based on Sherman-Morrison equation,  $\mathcal{D}_2$  can be solved from (22) without inverse problem. If there are more observations, it can be done in the same way. Now we focus on one row of  $\mathcal{D}$  in (22). Every row is denoted as  $\mathcal{D}_i$  and the element in  $i$  row and  $j$  column is denoted by  $\mathcal{D}_{ij}$ . Every row can be updated by

$$\mathcal{D}_i = K_i - \frac{K_{ij}}{\sigma_j^2 + K_{jj}} K_j. \quad (24)$$

$K$  is the covariance function and it is a nonnegative matrix. When the point with index  $i$  is a little far from the point with index  $j$ , the corresponding covariance  $K_{ij}$  will be close to 0. There is no need to update the corresponding row  $\mathcal{D}_i$ . At the beginning, we only need to keep the elements which are more than a small positive number in  $K$ . This also means there is no need to consider the whole map when only a part of the map should be predicted. When predicting one point, we only need to consider the training points in a small area around it. The chosen area depends on the covariance function. When a new training point is obtained, only the estimation of the points in a small area need to be updated.

## IV. SIMULATION

The simulated map is shown as Figure 2(a). It simulates a corridor with an open door. At a position, there are five measurement directions:  $0, \pm\pi/2$  and  $\pm\pi/4$ . They are relative to the robot direction. The maximum range is 9 grid cells. The trajectory is shown as Figure 2(b). The robot runs from ① to ②.

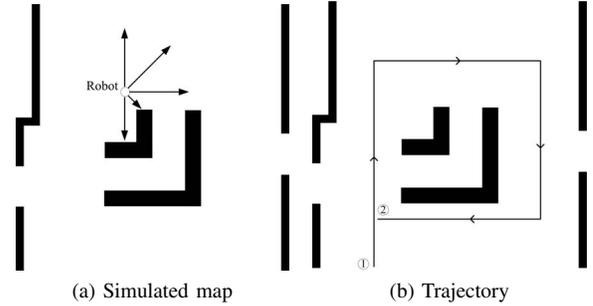


Fig. 2: The simulated map and the trajectory

Following along a beam in the measurement direction, the grid cells are free with  $\log \frac{p(z^1|m_i)}{p(z^1|\bar{m}_i)} = -7$  at least until the measured distance. At the end of one beam, the cell is occupied and  $\log \frac{p(z^1|m_i)}{p(z^1|\bar{m}_i)} = 9$ .  $\log \frac{p(m_i)}{p(\bar{m}_i)}$  is set to 0. The log odds occupancy observations are shown as Figure 3(a) and the corresponding occupancy probabilities are shown as Figure 3(b). Because of the noise of the sensor, the measured distance may be different from the true distance and it is noisy.

Now the central points of observed grid cells are the training data. In this simulation, all the variances  $\sigma_i^2$  are the same as  $\sigma^2$ . Even if one training point is observed many times, the corresponding variance does not change.

First the effect of the length scale  $\ell$  is tested and  $\sigma^2$  is set to 0.1. When  $\ell$  is chosen as 0.5, 1, 1.5, and 2, the prediction results are shown in Figure 4. When the  $\ell$  is smaller, one point is correlated with the points in a smaller area. Meanwhile, its observation is more believable and the prediction is more noise.

Finally, the effect of the variance  $\sigma^2$  is tested.  $\ell$  is fixed to 2. When  $\sigma^2$  is chosen as 0.03 and 0.3, the prediction results are shown in Figure 5. When  $\sigma^2$  is small, it also means the

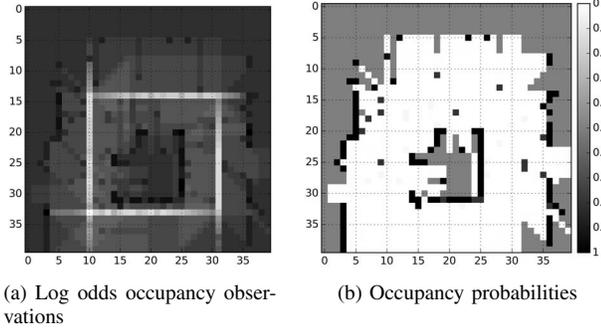


Fig. 3: Log odds occupancy observation and occupancy probabilities

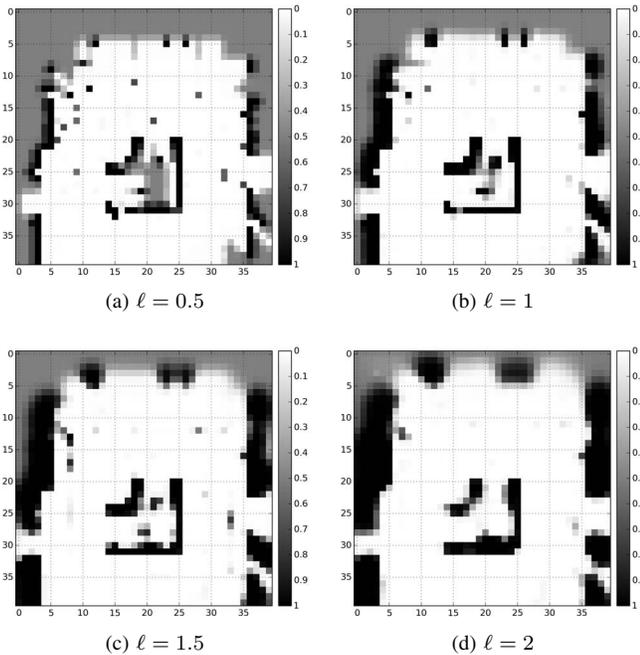


Fig. 4: Prediction results with different  $\ell$

observations are more believable. When becomes bigger, the results are smoother.

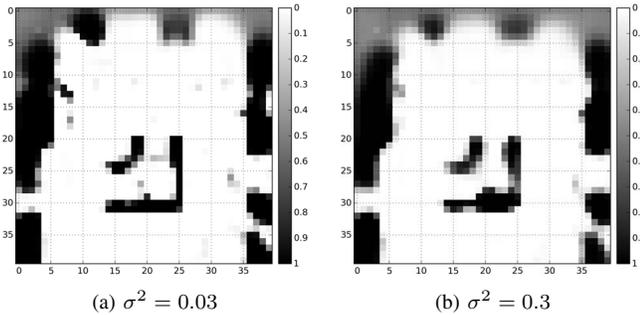


Fig. 5: Prediction results with different  $\sigma^2$

## V. CONCLUSIONS

In this paper, we propose a GRF-based LOOM for static environments. Instead of binary values, continuous values are used to describe the map. The set of occupancy probabilities in log odds form is regarded as a GRF. The posterior occupancy probability in log odds form is the training data for the GRF. Instead of the well-known prediction equation, we propose a novel recursive method, which avoids the inverse problem in the GRFs, to predict the occupancy probabilities of observed space and unobserved space. This recursive method, which makes it possible to implement the GRFs on the robot with less powerful micro-controller, can also be used in other GP or GRF applications. Occupancy grid map is only necessary for training data. After obtaining training data, the GRF-based LOOM can predict any point in continuous space. In the future, we will develop new method to obtain the training data in continue space.

## REFERENCES

- [1] H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 116–121, IEEE, 1985.
- [2] S. T. OCallaghan and F. T. Ramos, "Gaussian process occupancy maps," *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 42–62, 2012.
- [3] S. Kim and J. Kim, "Building occupancy maps with a mixture of gaussian processes," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4756–4761, IEEE, 2012.
- [4] S. Kim and J. Kim, "Continuous occupancy maps using overlapping local gaussian processes," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4709–4714, IEEE, 2013.
- [5] S. Kim, J. Kim, *et al.*, "Recursive bayesian updates for occupancy mapping and surface reconstruction," in *Proceedings of the Australasian Conference on Robotics and Automation*, 2014.
- [6] C. E. Vido and F. Ramos, "From grids to continuous occupancy maps through area kernels," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1043–1048, IEEE, 2016.
- [7] J. Wang and B. Englot, "Fast, accurate gaussian process occupancy maps via test-data octrees and nested bayesian fusion," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1003–1010, IEEE, 2016.
- [8] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.
- [9] C. Rasmussen and C. Williams, "Gaussian processes for machine learning," *Gaussian Processes for Machine Learning*, 2006.