The Navigation of Autonomous Vehicles in Uncertain Dynamic Environments: A Case Study[1]

Myungsoo Jun   Atif I. Chaudhry   Raffaello D'Andrea[2]

Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853-7501, USA

Key Words: Autonomous vehicle, probability map building, uncertain dynamic environments

Abstract

This paper proposes a method for building with multiple vehicles a probability map of uncertain dynamic environments. It is assumed that each vehicle has a limited sensor range and therefore lacks global information. The vehicles share their measurement information to build a probability map. The probability map is updated using sensor information and a priori statistics of the dynamic environment.

1 Introduction

Autonomous vehicles are used to perform hazardous missions such as operations in nuclear power plants, the exploration of Mars, and surveillance of enemy forces. The main difficulty in such uses of autonomous vehicles is that most environments requiring them have uncertain geography, unknown obstacles, unexpected pop-up enemies, etc. This uncertainty in information leads naturally to the consideration of probabilistic models.

How to build a probability map of an environment has been actively studied in the robotics community. The most widespread method for building a probability map of an uncertain environment is grid-based occupancy maps (See [3], [7] and [10]). Occupancy values for each grid cell are determined based on sensor readings and by applying the conditional probability of occupancy using Bayes' rule. These values are determined by the sensor characteristics, the location of the sensors and the measurement methods. In the unmanned aerial vehicle (UAV) community, Hespanha *et al.* [5] constructed a probabilistic map of radar sites by using Bayes' rule and a likelihood function which was defined by considering radar range and other radar characteristics. However, most map building methods consider static environments and little research has been done on map building of uncertain dynamic environments.

Thrun [9] adopted a simple exponential decay factor which puts more weight on recent sensor readings in order to cope with dynamic environments. As the author acknowledges, this approach does not fully model dynamic environments but just adapts to changes. Cox *et al.* [1] used a Bayesian multiple hypothesis framework to build and maintain a world model of a dynamic environment. Their approach, however, focused on modeling dynamic environments by simple geometric primitives and extracting features from multiple hypothesis and sensor readings.

In order to estimate the state of a dynamic system, a system model that describes state transition with time and measurement model that relates the state of the system with noisy measurement are required. A linear or nonlinear state-space model and the Bayesian approach are usually used for a system model and measurement model, respectively. When a linear or nonlinear state-space model of the obstacle is available, a Kalman-Bucy filter or extended Kalman filter can estimate the location of obstacle. However, it is difficult to obtain such a deterministic state-space model under uncertain environment with limited information. Therefore, a probabilistic model of the obstacle should be used for a system model to describe evolution of the state with time, and it can be obtained from *a priori* statistics of obstacle's movement. This paper presents a dynamic map building method incorporating sensor reading and *a priori* statistics of the environment, which estimates location of obstacles. The method assumes that a vehicle can recognize obstacles as static or dynamic from sensor readings. If it detects a moving obstacle, the vehicle estimates a probability distribution of the obstacle's location in the future based on a priori statistics.

The paper is organized as follows: Section 2 explains the notation used throughout the paper. Section 3 provides the main result. It gives an algorithm to build a probability map in uncertain dynamic environments by using sensor readings and a priori statistics of the

---

[2]Corresponding author; email rd28@cornell.edu;

environment. Section 4 describes the simulation platform which includes statistics of obstacle movements, calculation of necessary probability density functions, etc. Simulation results are provided in Section 5. Discussion, future problems and concluding remarks are found in Section 6.

2 Definitions and Notation

Given that there are l moving obstacles. We assume that the number of total obstacles is known beforehand and that obstacles are indistinguishable from each other. Let $\mathcal{R}$ be the region of consideration, $\mathcal{O} = \{1, 2, \cdots, l\}$ the index set of obstacles, and $\mathcal{D} = \{1, 2, \cdots, k\}$, where $k \leq l$, the index set of detected obstacles at current time. Let $\mathcal{V}$ be the region of view region. Note that $\mathcal{D}$ and $\mathcal{V}$ are time-varying. We denote sampling time by Δt, the current time by $T\Delta t$, measured data at k-th sampling time by m_k, and sequence of measured data up to time $T\Delta t$ by $m^{(T)}$, viz., $m^{(T)} = \{m_1, m_2, \cdots, m_T\}$. Measured data m_i is composed of the position and heading angle of the vehicle at time $T\Delta t$, and the two-dimensional Cartesian position and heading angle for each obstacle in view of the vehicle.

3 Probability Map Building

The probability map indicates the probability distribution of obstacle's location. This section describes a method to build a probability map. Suppose that the vehicle receives measurement data at every sampling time, Δt, and has limited measurement range. It is assumed that there are no static obstacles in the environment, as is common for unmanned aerial vehicles. The vehicle updates its probability map of the environment at each measurement time based on what information it has. This information includes sensor readings and a priori statistics of the dynamic environment.

Due to the limited sensor range, the vehicle may not detect all obstacles. Whenever the vehicle senses an obstacle, uncertainty on the location of the obstacle greatly decreases. The uncertainty is not zero due to sensor noise. The vehicle modifies the probability map based on this location information. If the vehicle no longer detects an obstacle, uncertainty of the obstacle's location, which was sensed at a previous time, increases since obstacles are moving. In such a case, the vehicle should estimate the obstacle's location based on a priori statistics on obstacle movement and previous information on the location of the obstacle from sensor readings. It should calculate the probability that obstacles will move to a certain location at the next time step given the current probability map and a pri-

ori statistics. Therefore, the probability map building procedure consists of two steps. The first is measurement update based on sensor readings and a Bayesian measurement model, and the second is time propagation by using a probabilistic system model of the obstacle obtained from a priori statistics. This is similar to measurement update and time update in Kalman-Bucy filters. However, Kalman-Bucy filters use more specific system, which is hard to obtain with uncertain information on the system.

3.1 Measurement Update
The vehicle estimates the posterior distribution of the position (x_T, y_T) and direction of obstacles θ_T at current time T given the data available up to the current time T. Let $\mathbf{x}_T = (x_T, y_T, \theta_T)$. The posterior distribution of the position and direction of obstacles at time $T\Delta t$ given sensor data $m_1, m_2, \cdots, m_T$ can be expressed by

$$f_i(\mathbf{x}_T \mid m_1, m_2, \cdots, m_T) = f_i(\mathbf{x}_T \mid m^{(T)}).$$

Bayes' rule enables us to estimate obstacle position and direction recursively. In order to apply Bayes' rule, we have to assume conditional independence between $f_i(m_k|\mathbf{x})$ and $f_i(m_{k'}|\mathbf{x})$ when $k \neq k'$. This assumption is commonly made in approaches to occupancy grid map building. Note that this assumption is not valid in localization for mobile robots when other moving objects are in the environment [4], [9].

The recursive form of the probability density function for the location of obstacles given measured data can be obtained as follows:

$$\begin{aligned}
&f_i(\mathbf{x}_T \mid m^{(T)}) \\
&= f_i(\mathbf{x}_T \mid m_1, m_2, \cdots, m_T) \\
&= \frac{f_i(m_T \mid m_1, \cdots, m_{T-1}, \mathbf{x}_T) f_i(\mathbf{x}_T \mid m_1, \cdots, m_{T-1})}{f_i(m_T \mid m_1, \cdots, m_{T-1})} \\
&= \frac{f_i(m_T \mid \mathbf{x}_T) f_i(\mathbf{x}_T \mid m_1, \cdots, m_{T-1})}{f_i(m_T \mid m_1, \cdots, m_{T-1})} \\
&= \frac{f_i(m_T \mid \mathbf{x}_T) f_i(\mathbf{x}_T \mid m^{(T-1)})}{f_i(m_T \mid m^{(T-1)})}.
\end{aligned} \tag{6}$$

The term in the denominator of Eq. (6) can be expressed as

$$\begin{aligned}
&f_i(m_T \mid m^{(T-1)}) \\
&= \iiint f_i(m_T \mid \zeta, \xi, \varphi) f_i(\zeta, \xi, \varphi \mid m^{(T-1)}) \, d\zeta \, d\xi \, d\varphi
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
&f_i(\mathbf{x}_T \mid m^{(T)}) \\
&= \frac{f_i(m_T \mid \mathbf{x}_T) f_i(\mathbf{x}_T \mid m^{(T-1)})}{\iiint f_i(m_T \mid \zeta, \xi, \varphi) f_i(\zeta, \xi, \varphi \mid m^{(T-1)}) \, d\zeta \, d\xi \, d\varphi}.
\end{aligned} \tag{7}$$

1. Set $T = 0$ and initialize $DIST_i(x, y, \theta, T)$, $i \in \mathcal{O}$;

2. (Measurement Update) After getting measurement m_T,

$$\alpha_i \longleftarrow f_i(m_T \mid x_T,\, y_T,\, \theta_T) \cdot DIST_i(x, y, \theta, T) \tag{1}$$

$$\beta_i \longleftarrow \iiint f_i(m_T \mid \zeta,\, \xi,\, \varphi) DIST_i(\zeta,\, \xi,\, \varphi, T)\, d\zeta\, d\xi\, d\varphi \tag{2}$$

$$DIST_i(x, y, \theta, T) \longleftarrow \alpha_i / \beta_i \tag{3}$$

for $\forall\, i \in \mathcal{O}$;

3. (Time Propagation)

$$DIST_i(x, y, \theta, T+1) \longleftarrow \iiint f_i(x, y, \theta \mid \zeta, \xi, \varphi, m^{(T)}) DIST_i(\zeta, \xi, \varphi, T)\, d\zeta\, d\xi\, d\varphi \tag{4}$$

for $\forall\, i \in \mathcal{O}$;

4. (Probability Map Building)

$$MAP(x, y, T+1) \longleftarrow 1 - \prod_{i=1}^{l} \left(1 - \int_0^{2\pi} DIST_i(x, y, \theta, T+1)\, d\theta \right); \tag{5}$$

5. Increase T by one and repeat the step 2, 3 and 4;

Algorithm 1: Algorithm for Probability Map Building

Here, $f_i(m_T \mid \mathbf{x}_T)$ is the probability density function that the vehicle will have measurement m_T when an obstacle is at position (x_T, y_T) and in direction θ_T at time T. This is a sensor characteristic and is a function of the location and heading angle of the vehicle since m_T contains the data on the vehicle.

3.2 Time Propagation

Due to limited view range, the vehicle has to estimate the location of obstacles after it loses sight of them. If the vehicle has no information or statistics on obstacle behavior it cannot estimate possible obstacle locations. Therefore, the more information on obstacle behavior is known, the more precise estimation of the future obstacle location is possible. From the total probability theorem, the prior distribution of the position and direction can be given by

$$f_i(\mathbf{x}_{T+1} \mid m^{(T)}) = \iiint \Big(f_i(\mathbf{x}_{T+1} \mid \zeta, \xi, \varphi, m^{(T)})$$
$$\cdot\, f_i(\zeta, \xi, \varphi \mid m^{(T)}) \Big)\, d\zeta\, d\xi\, d\varphi.$$

Here, $f_i(\zeta, \xi, \varphi \mid m^{(T)})$ is the posterior probability density function after measurement at time T in Eq. (7). The term $f_i(\mathbf{x}_{T+1} \mid \zeta, \xi, \varphi, m^{(T)})$ is related to state transition with time and is the probability that the obstacle will be at the position (x_{T+1}, y_{T+1}) and in direction θ_{T+1} at time $(T+1)$ when the vehicle has measurement m_T and the obstacle is at the position (ζ, ξ) and in direction φ at time T. This can be calculated if in-

formation is on the velocity of obstacles, statistics on changes of direction, and so on.

3.3 Probability Map

By combining the measurement update and time propagation procedures described above, the probability map building algorithm in uncertain dynamic environments can be summarized in Algorithm 1. It can be easily noted from the algorithm that the probability map can be constructed recursively.

3.4 Cooperative Map Building

For cooperative map building, we assume that all vehicles share their measurement data with one another, and measurement sampling time is synchronized. Thus, at each sampling time, each vehicle acquires measured data from its own sensors and from all other vehicles, and then performs the measurement update and time propagation procedure. This results in one global probability map.

Sending vehicles only to unexplored areas to minimize exploration time is not effective as the explored area changes in dynamic environments. All areas need to be continually re-explored to update the probability map. For this reason, exploration time is not an appropriate measure of efficiency in dynamic environments. Therefore, mission goals have priority over global map building which should be incorporated into path planning, as in reference [6].

4 Simulation Platform

4.1 Description of Simulation Platform

The game named "RoboFlag" will be used as a simulation platform for validating the algorithm [2]. The game has two teams, Red and Blue. The Red team's objective is to infiltrate Blue's territory, grab the Blue flag and bring it back to the Red Home Zone, and visa versa. See Figure 1. During the game, the following objects will be on the playing field: 8 Red robots, 8 Blue robots, 8 scoring balls, and 8 obstacles. Each robot has its own local information such as its position, orientation, translational velocity and angular velocity. It also has information on the position and orientation of obstacles within its conic sensor range with radius R_s and angle Ω_r. The origin is located at the center of the region and orientation is measured counterclockwise from the horizontal axis with range between 0 and 2π.

In order to apply the algorithm described in the previous section, two probability density functions, $f_i(m_T|\mathbf{x}_T)$ and $f_i(\mathbf{x}_{T+1}|\zeta,\xi,\varphi,m^{(T)})$, are needed. The former means the probability density function implying that the vehicle will have measurement m_T when an obstacle is at the position (x_T, y_T) and in direction θ_T at time T. This is a characteristic of the sensors; thus it can be obtained. The latter implies the probability density function that the obstacle will be at the position (x_{T+1}, y_{T+1}) and in direction θ_{T+1} at time $T+1$ when the vehicle has measurement m_T and the obstacle is at position (ζ, ξ) and in direction φ at time T. This can be obtained if there is information on the velocity of obstacles, statistics of angle changes, and so on.

The dynamics of the obstacles are as follows: the initial positions of l moving obstacles are uniformly distributed on the rectangular region $\mathcal{R}$ with (field width)$\times$(field length). A new random location for each obstacle is determined at time t_{switch}, where t_{switch} is a Poisson process with parameter $\lambda\Delta t$. Each obstacle moves toward its new target positions with velocity v_m at time t_{switch}. The target destination of each obstacle has uniform distribution as well. Therefore, the probability that the obstacle will head to the direction of ψ when switching occurs and it is at the position (v, w) is proportional to the segment length from the point (v, w) to the boundary with angle ψ.

4.2 Calculation of Probabilities

Let $\mathbf{s}$ be the random variable that has value 1 when switching occurs in the time interval $(T\Delta t, (T+1)\Delta t)$ and 0 when switching does not occur. Let us define d_x, d_y and R by

$$d_x(\psi, v) \triangleq \left| x_m - v \cdot \text{sgn}(\cos \psi) \right|,$$

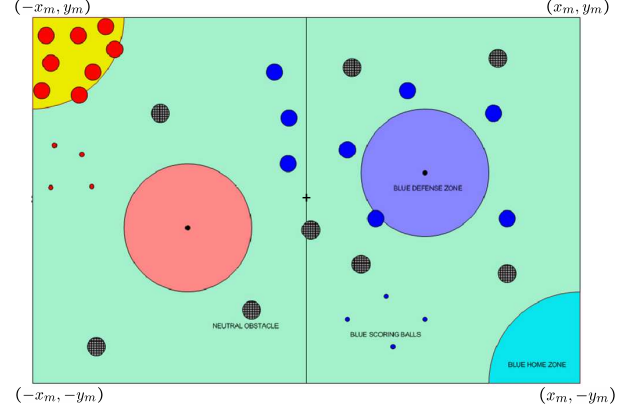$$d_y(\psi, w) \triangleq \left| y_m - w \cdot \text{sgn}(\sin \psi) \right|,$$

Figure 1: Simulation Platform

and

$$R(\psi, v, w) \triangleq \begin{cases} d_x(\psi, v)/|\cos \psi|, & \text{if } d_x|\tan \psi| \leq d_y \\ d_y(\psi, w)/|\sin \psi|, & \text{otherwise} \end{cases},$$

respectively. Then, the probability density function

$$f(\psi \mid v, w, \mathbf{s} = 1, t) = \frac{R(\psi, v, w)}{\int_0^{2\pi} R(\theta, v, w)\, d\theta}, \quad 0 \leq \psi < 2\pi \tag{8}$$

gives the probability that the obstacle will head in the direction of ψ when switching occurs at time $t \in (T\Delta t, (T+1)\Delta t)$ and it is at the position (v, w) at time t.

Now let us calculate the probability density function $f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, m^{(T)})$. It can be said that

$$f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, m^{(T)})$$
$$= f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 1, m^{(T)}) P(\mathbf{s} = 1 \mid \zeta, \xi, \varphi, m^{(T)})$$
$$+ f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 0, m^{(T)}) P(\mathbf{s} = 0 \mid \zeta, \xi, \varphi, m^{(T)})$$

and this can be approximated by

$$f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, m^{(T)})$$
$$\approx f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 1, m^{(T)}) \lambda \Delta t$$
$$+ f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 0, m^{(T)})(1 - \lambda \Delta t) \tag{9}$$

since we assume that t_{switch} is a Possion process and the probability that switching occurs more than twice in the interval with length Δt is very small if Δt is small.

As the obstacle will keep its direction unless switching occurs, it can be said that

$$f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 0, m^{(T)}) = \delta(x - x_0)\delta(y - y_0) \tag{10}$$

where $x_0 = \zeta + (v_m \Delta t)\cos \varphi$ and $y_0 = \xi + (v_m \Delta t)\sin \varphi$. If switching occurs at $t \in (T\Delta t, (T+1)\Delta t)$, it can be

said that

$$f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 1, m^{(T)})$$

$$= \int_{T\Delta t}^{(T+1)\Delta t} \Big(f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 1, m^{(T)}, t)$$

$$\cdot f(t \mid \zeta, \xi, \varphi, \mathbf{s} = 1, m^{(T)}) \Big) \, dt. \quad (11)$$

Since we assume that switching time is a Poisson process with parameter $\lambda \Delta t$ and that $T\Delta t$ is a fixed point, we have (see [8])

$$f(t \mid \zeta, \xi, \varphi, \mathbf{s} = 1, m^{(T)}) = \frac{(\lambda \Delta t) e^{-\lambda \Delta t(t - T\Delta t)}}{1 - e^{-\lambda(\Delta t)^2}}. \quad (12)$$

From Eq. (10) and Eq. (8), we obtain

$$f_i(\mathbf{x}_T \mid \zeta, \xi, \varphi, \mathbf{s} = 1, m^{(T)}, t)$$

$$= \frac{R(\theta, v, w)}{\int_0^{2\pi} R(\psi, v, w) \, d\psi} \delta(x - x_0) \delta(y - y_0) \quad (13)$$

where

$$v = \zeta + v_m(t - T\Delta t) \cos \varphi,$$
$$w = \xi + v_m(t - T\Delta t) \sin \varphi,$$
$$x_0 = v + v_m((T+1)\Delta t - t) \cos \theta,$$
$$y_0 = w + v_m((T+1)\Delta t - t) \sin \theta.$$

5 Simulation

In simulation, the following values were used: $\Delta t = 0.5$, $\lambda = 1.6$, field length $= 60$, field width $= 40$, and $v_m = 5$. The region $\mathcal{R}$ is divided into 1×1 square cells. The vehicle begins with the initial probability map with all cells having values $1/2400$ at time 0. Figure 2 (a) shows the probability map when the vehicle detects an obstacle within its sensor range. As soon as it detects the obstacle, the vehicle turns its heading angle to avoid the obstacle and loses it from sensor range. After that, the vehicle estimates the possible location of the obstacle and updates the probability map. Figure 2 (b)-(d) show the probability maps after time propagation of 0.5, 1 and 1.5 seconds, respectively. All three plots show the directional preference of the obstacle as can be expected from statistics of obstacle movement. Simulations were performed with MatLab. Computation time for the algorithm depends on the number of cells with non-zero probabilities since time propagation procedure calculates probabilities from all cells with non-zero values.

6 Conclusion

A method to build a probability map in uncertain dynamic environments was proposed. The proposed algorithm constructs a probability map by incorporating
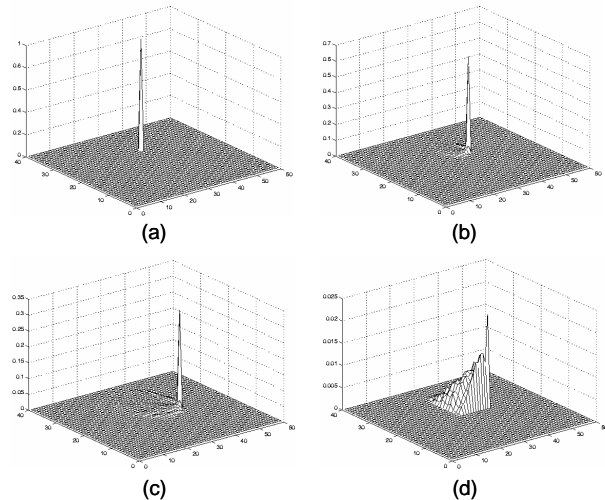

Figure 2: Probability map (a) when the vehicle detects an obstacle. (Measurement update), and (b) when the vehicle loses tracking of the obstacle after 0.5 second time propagation, (c) 1 second time propagation and (d) 1.5 second time propagation.

measurement data from sensor readings and a priori statistics of the environment. This algorithm is recursive; thus it can be implemented easily. How to calculate necessary probability density functions from the statistics of a dynamic environment was also explained.

References

[1] I. J. Cox and J. J. Leonard. Modeling a dynamic environment using Bayesian multiple hypothesis approach. *Artificial Intelligence*, 66:311–344, 1994.

[2] R. D'Andrea. RoboFlag: Rules and regulation. `http://www.mae.cornell.edu/Raff/Research/Autonomous/RoboFlag/RFrules.pdf`, 2002.

[3] A. Elfes. Sonar-based real-world mapping and navigation. *IEEE Trans. on Robotics and Automation*, RA-3(3):249–265, June 1987.

[4] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11:391–427, 1999.

[5] J. P. Hespanha, H. H. Kizilocak, and Y. S. Ateşkan. Probabilistic map building for aircraft-tracking radars. In *Proc. of the American Control Conference*, pages 4381 – 4386, Arlington, VA, 2001.

[6] M. Jun and R. D'Andrea. Path planning for unmanned aerial vehicles in uncertain and adversarial environments. In S. Butenko, R. Murphey, and P. Pardalos, editors, *Cooperative Control: Models, Applications and Algorithms*, chapter 6, pages 95–111. Kluwer, Boston, MA, 2002.

[7] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, pages 61–74, Summer 1988.

[8] A. Popoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 3rd edition, 1991.

[9]   S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99:21–71, February 1998.

[10]   B. Yamauchi and P. Langley. Place recognition in dynamic environments. *J. of Robotic Systems*, 14:107–120, 1997.