Robotica (2014) volume 32, pp. 757–782. © Cambridge University Press 2013
doi:10.1017/S0263574713001033

Active exploration using a scheme for autonomous allocation of landmarks†

Jing Yuan*, Yalou Huang, Fengchi Sun and Tong Tao

Department of Automation, Nankai University, Tianjin 300071, P. R. China

(Accepted October 28, 2013. First published online: December 3, 2013)

SUMMARY

In this paper, we focus on the unknown environments without artificial landmarks and features, such as disaster situations and polar regions. An approach to active exploration based on an on-line scheme for autonomous allocation of landmarks is proposed. Specifically, the robot carries along with itself some landmarks which are to be allocated during the exploration according to some heuristic rules. The utility of landmark allocation is analyzed and calculated. Then the active exploration is converted into a problem of multi-objective optimization. The objective function includes three weighted terms: the accuracy of localization and mapping, the coverage rate of the unknown environment and the utility of the allocated landmarks. By solving this optimization problem, control inputs of the robot are computed to guarantee that accurate localization, high-quality mapping and complete exploration can be achieved simultaneously. Moreover, supplementation and redundancy elimination of the allocated landmarks are executed to make a complete and non-redundant coverage for the environment. Finally, some landmarks, together with a device for allocating these landmarks, are developed. Both experiment and simulation results are presented to demonstrate the effectiveness of the proposed approach.

KEYWORDS: Active exploration; Landmark allocation; EKF; Multi-objective optimization.

1. Introduction

The capability of localization and mapping is a prerequisite for successful navigation and exploration of a mobile robot in unknown environments, for which the map information is not known *a priori*. Simultaneous Localization and Mapping (SLAM) provides an effective solution to the above issue. Since Smith *et al.*[1] first proposed a stochastic mapping approach using an extended Kalman filter (EKF), the research of the SLAM has attracted considerable attention in the community of robotics for almost 20 years. The most popular solution to the SLAM is the EKF,[2–4] which has been used successfully in many robotic applications. In the Kalman filter family, Unscented Kalman filter (UKF),[5] postponement,[6] Compressed EKF filter (CEKF),[7] SEIF filter,[8] Relative Map Filter (RMF)[9] and Divide and Conquer SLAM (D&C SLAM)[10] are the alternatives to achieve effective solutions. UKF can lead to more accurate results than EKF because it approximates the posterior mean and covariance accurately to the second-order term of the Taylor series for any nonlinearity. Postponement works by updating a constant-sized data set based on the current measurements, which can be used to effect updates on all unobserved parts of the map at a later stage. As a result, the full map update can be postponed. CEKF algorithm stores and maintains all the information gathered in a local area. The information can then be transferred to the rest of the global map. SEIF approximates the information matrix by a sparse form that allows $O(1)$ update. RMF recursively estimates relative features which are invariant to robot position, thereby removing the correlations between the robot and the features, as well as feature–feature correlations. D&C SLAM works for local maps and carries out map joining

* Corresponding author. Email: nkyuanjing@gmail.com

† This paper was partially presented at the 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, May 12–17.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

in a hierarchical fashion. It can compute both state and its covariance, with no approximations, and with the advantage of providing a more precise and consistent estimate.

Another solution is the Bayesian estimate approach that does not assume Gaussian probability distributions. FastSLAM[11–14] is an efficient approach to SLAM based on particle filtering. Some other efficient algorithms have also been proposed, such as Rao–Blackwellized particle filters (RBPF),[15,16] Graph-SLAM,[17,18] Topological SLAM,[19–21] EM-based Incremental SLAM[22] and Prediction-based SLAM (P-SLAM).[23]

However, most of the existing approaches assume that pre-placed artificial landmarks or sufficient natural features are available in the environment. Unfortunately, in some extreme environments, such as disaster situations and polar regions, there exists no pre-placed landmark, while feature extraction is also extremely difficult. For example, the vision sensors cannot gather distinct features in polar regions because environmental features are too rare to be extracted from snow and ice backgrounds. In view of this problem, Kleiner *et al.*[24] proposed a Radio Frequency Identification (RFID)-based exploration approach, wherein the robot allocated RFID tags onto the ground as landmarks. However, the robot placed the RFID tags in a relatively arbitrary way. The utility of the allocated tags and the relationship among the positions of these tags were not considered.

In addition, motion planning is an important problem in exploration because the accuracy and efficiency of the SLAM can be largely improved if a robot moves along an appropriate motion trajectory. However, most existing SLAM algorithms do not take the motion planning into account. That is, the motion trajectory of the robot is predetermined rather than gradually created through an optimization process. To address this problem, Feder *et al.*[25] proposed an adaptive SLAM approach, wherein the information theory was used to decide the motion of the robot in order to acquire more useful information for localization and mapping. Based on Feder *et al.*'s[25] work, several methods of active/adaptive SLAM have been proposed.[26–28] However, most of these assumed that the robot had unlimited sensing ability.[27,28] Furthermore, only the accuracy of the SLAM was considered in the cost function of the optimization problem. Fairfield and Wettergreen[29] proposed an active strategy for SegSLAM using a submap-based approach. The distribution over robot poses and submaps was estimated by a particle filter. Then active actions of the robot were computed to reduce the uncertainty quantified as information entropy. Leung *et al.*[30] considered the trajectory planning problem for the indoor environment SLAM, wherein a motion trajectory, generating minimum uncertainty and maximum coverage, was calculated by using model predictive control algorithms.

Our method has been originally presented in Yuan *et al*.[31] It integrated landmark allocation, motion planning and SLAM into a problem of active exploration in an environment without landmarks and features. The problem was addressed by solving a multi-objective optimization.

The exploration approach proposed in this paper is the outgrowth of the method given in Yuan *et al*.[31] In contrast to our previous work, i.e. Yuan *et al*.[31], we increase the following three aspects. First, a complete proof for validating the utility measure of landmark allocation is presented in this paper (see the proof of Theorem 1) to build a theoretical foundation for the proposed method. Second, design and development of both landmarks and a device to allocate them are given in detail to support the proposed theoretical method. Third, we give additional simulation results and present experimental results to show the effectiveness and the practical applicability of the proposed method.

In this paper we take into account an unknown environment without landmarks and features. Exploration in such an environment is somewhat tough. For example, when a person enters a completely unfamiliar environment without conspicuous features, he/she would make some marks as the points of reference to avoid missing him/her way. When he/she revisits a place which has been visited before, he/she can use the marks to localize himself/herself. In the same way, the robot can also mark the unknown environment with some landmarks to help itself for localization. In our scheme, the robot decides when and where a new landmark should be allocated, according to some elegantly designed rules. In contrast to the method in given Kleiner *et al.*[24], the robot can allocate landmarks in an active and intelligent way, rather than in an optional way. The utility of landmark allocation is analyzed and proved completely. By converting the problem of active exploration into a multi-objective optimization, a trade-off between the coverage rate of the unknown space, the accuracy of the SLAM and the utility of the landmark allocation is achieved. Moreover, a complete and non-redundant coverage with the allocated landmarks is considered to provide a complete solution. Using this scheme, the robot can allocate the landmarks and explore the environment elegantly. At last, we develop a new type of landmarks consisting of ultrasonic and RFID units, as well as a

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

mechanical device for allocating the landmarks. The remainder of this paper is organized as follows: The EKF-based SLAM is reviewed in Section 2, and the utility of landmark allocation is analyzed and proved in Section 3. Section 4 describes the proposed exploration strategy. Landmarks, together with a device for allocating them, are designed and developed in Section 5. Both simulation and experimental results are provided in Section 6. The paper is concluded in Section 7.

2. EKF-Based SLAM

When a mobile robot moves through an unknown environment, the relative pose between the robot and the surrounding landmarks can be directly observed. These resultant observations are then used to simultaneously estimate the robot pose $X_R(k)$ and the corresponding positions of the landmarks $X_i(k)$ $(i = 1, 2 \cdots N)$. The whole process is called the SLAM. For the SLAM, the state vector to be estimated at time k is assumed to contain both robot pose and positions of landmarks, namely $X(k) = [X_R^T(k), X_M^T(k)]^T$, where $X_R(k) = [x_R(k), y_R(k), \theta_R(k)]^T$ denotes the robot pose, and $X_M(k) = [X_1^T(k), \cdots X_N^T(k)]^T$ is referred to as the position vector of the landmarks, with $X_i(k) = [x_i(k), y_i(k)]^T$ $(i = 1, 2 \cdots N)$ representing the position of the ith landmark. The covariance matrix associated with the state estimate is denoted by $P(k)$. The motion model and the perception model are given by

$$X(k+1) = f(X(k), u(k)) + v(k), \tag{1}$$

$$Z(k+1) = h(X(k+1)) + w(k), \tag{2}$$

where $f(\cdot)$ and $h(\cdot)$ are respectively the nonlinear process model and the observation model; $u(k)$ and $Z(k+1)$ denote the control input and the perception data and $v(k)$ and $w(k)$ represent motion noise and measurement noise with covariance $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ respectively.

For the range and bearing sensors, the perception model is given by

$$Z(k) = \begin{bmatrix} r_i(k) \\ \phi_i(k) \end{bmatrix} = \begin{bmatrix} \sqrt{[x_R(k) - x_i(k)]^2 + [y_R(k) - y_i(k)]^2} \\ \arctan \dfrac{y_R(k) - y_i(k)}{x_R(k) - x_i(k)} - \theta_R(k) \end{bmatrix} + \begin{bmatrix} w_r(k) \\ w_\phi(k) \end{bmatrix}, \tag{3}$$

where $Z(k)$ provides a measurement of range $r_i(k)$ and bearing $\phi_i(k)$ to an observed landmark $l_i(x_i(k), y_i(k))$ relative to the robot.

In the following statements, we use $\hat{X}(p|q)$ to denote the state estimate at time p using all the observations collected up to time q, where $q \le p$. We give the EKF-based SLAM process as follows. The reader is referred to Thrun *et al.*[32] and Choset *et al.*[33] for a complete description.

(1) *Prediction*

$$\hat{X}(k+1|k) = f(\hat{X}(k|k), u(k)), \tag{4}$$

$$P(k+1|k) = \nabla_X f \cdot P(k|k) \cdot \nabla_X^T f + Q(k), \tag{5}$$

$$\hat{Z}(k+1|k) = h(\hat{X}(k+1|k)), \tag{6}$$

where $\nabla_X f = \partial f / \partial X \big|_{X = \hat{X}(k|k)}$ expresses the Jacobian matrix of $f(\cdot)$ with respect to $\mathbf{X}(k)$.

(2) *Update*

$$\hat{X}(k+1|k+1) = \hat{X}(k+1|k) + W(k+1) \cdot \gamma(k+1), \tag{7}$$

$$P(k+1|k+1) = P(k+1|k) - W(k+1) \cdot S_{\gamma\gamma}(k+1|k) \cdot W^T(k+1), \tag{8}$$

where $\nabla_X h = \partial h / \partial X|_{X=\hat{X}(k|k)}$ represents the Jacobian matrix of $h(\cdot)$ with respect to $\mathbf{X}(k)$, $\gamma(k+1)$ and $S_{\gamma\gamma}(k+1|k)$ denote the innovation and its covariance matrix respectively, and $W(k+1)$ is the Kalman gain,

$$\gamma(k+1) = Z(k+1) - \hat{Z}(k+1|k), \tag{9}$$

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

$$S_{\gamma\gamma}(k+1|k) = \nabla_X h \cdot P(k+1|k) \cdot \nabla_X^T h + R(k+1), \tag{10}$$

$$W(k+1) = P(k+1|k) \cdot \nabla_X^T h \cdot S_{\gamma\gamma}^{-1}(k+1|k). \tag{11}$$

When the robot observes a new landmark l_j at time k, the state of this landmark is initialized by

$$\begin{bmatrix} \hat{x}_j(k \mid k) \\ \hat{y}_j(k \mid k) \end{bmatrix} = g(k) = \begin{bmatrix} \hat{x}_R(k \mid k) + d(k) \cdot \cos(\phi(k) + \hat{\theta}_R(k \mid k)) \\ \hat{y}_R(k \mid k) + d(k) \cdot \sin(\phi(k) + \hat{\theta}_R(k \mid k)) \end{bmatrix}, \tag{12}$$

where $d(k)$ and $\phi(k)$ denote the measurements of the range and the bearing from l_j respectively.

3. Landmark Allocation

The main contribution of this paper is presented in this section, i.e. how to allocate landmarks in the environment. We only take into account the environments without any artificial landmarks and natural features. Note that the proposed scheme can also apply to the environments with insufficient artificial landmarks or natural features which fail to provide indispensable observations for the robot in most cases.

3.1. Rules of landmark allocation

We design four heuristic rules for landmark allocation, which are as follows:

Rule 1: The robot should allocate a new landmark immediately at the current position (chosen as the origin of the robot coordinate system in this paper) once it can observe no landmark.

Rule 2: If it is predicted that the accuracy of the SLAM could be improved enough after a new landmark is allocated at the current position, the robot should really allocate this landmark.

Rule 3: All the allocated landmarks should cover the environment entirely, that is, the robot can observe landmarks at any position.

Rule 4: There should be as few redundant landmarks as possible left in the environment after the landmark allocation has been finished. According to Rule 3, the robot could allocate sufficient landmarks to fully cover the environment with them. It has a possibility that too many redundant landmarks are allocated into the environment. As a result, considerable effort has to be spent in maintaining the redundant landmarks.

3.2. Utility of landmark allocation

During the exploration, if the robot cannot observe any landmarks for a period of time, the uncertainty of localization and mapping will increase rapidly due to the accumulation of odometry errors over time. To show this fact, Fig. 1 gives a simulation result of the EKF-based SLAM. At the initial stage, the robot could observe many landmarks and the uncertainty of the SLAM kept at a low level. Nevertheless, at the later stage, the robot could observe no landmarks. As a result, the error and covariance rapidly accumulated to large levels, even though the robot observed some new landmarks at the last stage. Figure 2 gives a simulation result to show the utility of landmark allocation, in which the robot started from $X_r(0) = [0, 0, 0]^T$ and tracked a predetermined straight-line path. Two cases are respectively displayed in Figs. 2(a) and (b). In the first case, the robot allocated only one landmark l_1 at the origin. In the second one, the robot at first allocated a landmark l_1' at the origin, and then it allocated another landmark l_2' at a new position, with the result that the uncertainty of landmark l_1' became much smaller than that of landmark l_1. From this simulation study, it can be seen that allocation of an extra landmark had a substantial impact on the accuracy of the resulting map.

For the EKF-based SLAM, we give an evaluation for the utility of landmark allocation as follows.

THEOREM 1. *Assume that the robot has allocated a set of landmarks, denoted by* $L(k-1) = \{l_i,\ i = 1, 2 \cdots N_{k-1}\}$, *up until time* $k-1$. *We denote the system state vector at time* k *by* $X(k) = [X_R^T(k), X_1^T(k), \cdots X_{N_{k-1}}^T(k)]^T$, *which contains the robot pose and the positions of the landmarks in* $L(k-1)$. *If the robot allocates a new landmark* $l_{N_{k-1}+1}$ *at time* k, *the state vector containing the robot pose and the positions of the landmarks in the set* $L(k) = L(k-1) \cup \{l_{N_{k-1}+1}\}$ *is denoted by* $\bar{X}(k) = [X^T(k), X_{N_{k-1}+1}^T(k)]^T$. *Let* $\hat{X}(k+1 \mid k+1)$ *denote the estimate of the system state* X *at time* $k+1$, *without allocating the new landmark*

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

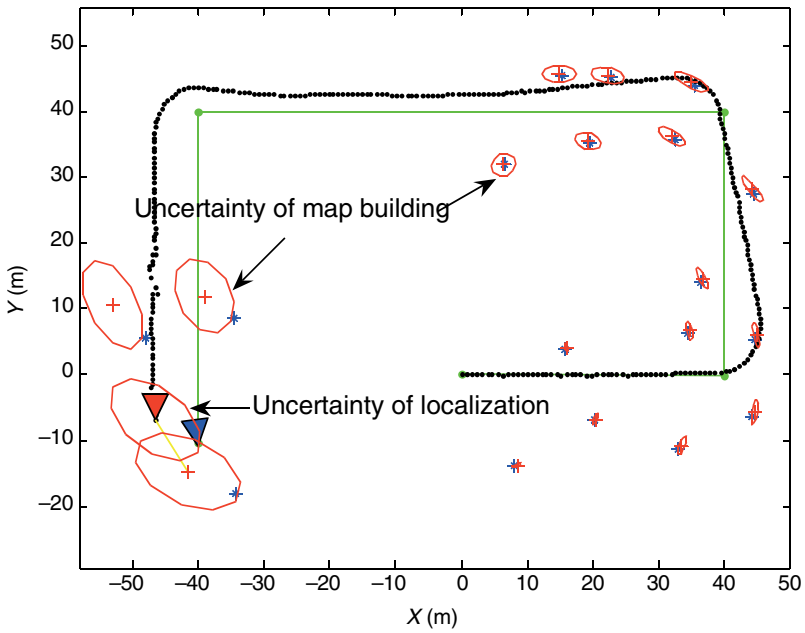


Fig. 1. (Colour online) Accumulation of error and uncertainty.

$l_{N_{k-1}+1}$ *at time k. The covariance matrix associated with* $\hat{X}(k+1\,|\,k+1)$ *is* $P(k+1\,|\,k+1) = \begin{bmatrix} P_{RR}(k+1\,|\,k+1) & P_{RM}(k+1\,|\,k+1) \\ P_{MR}(k+1\,|\,k+1) & P_{MM}(k+1\,|\,k+1) \end{bmatrix}_{(2N_{k-1}+3)\cdot(2N_{k-1}+3)}$. *Let* $\hat{X}_{\text{new}}(k+1|k+1)$ *denote the sub-vector of the estimate* $\hat{\bar{X}}(k+1|k+1) = [\hat{X}^T_{\text{new}}(k+1|k+1), \hat{X}^T_{N_{k-1}+1}(k+1|k+1)]^T$ *for the state* $\bar{X}(k+1)$ *with allocating the new landmark* $l_{N_{k-1}+1}$. $\hat{X}_{\text{new}}(k+1|k+1)$ *consists of the estimates of the robot pose and the positions of the first* N_{k-1} *landmarks. The covariance matrix* $P_{\text{new}}(k+1\,|\,k+1)$ *associated with* $\hat{X}_{\text{new}}(k+1|k+1)$ *is a sub-matrix of* $\bar{P}(k+1\,|\,k+1)$, *where* $\bar{P}(k+1\,|\,k+1) = \begin{bmatrix} P_{\text{new}}(k+1\,|\,k+1)_{(2N_{k-1}+3)\cdot(2N_{k-1}+3)} & B(k+1\,|\,k+1) \\ B^T(k+1|k+1) & A(k+1\,|\,k+1) \end{bmatrix}_{(2N_{k-1}+5)\cdot(2N_{k-1}+5)}$ *is the covariance of* $\hat{\bar{X}}(k+1|k+1)$, $A(k|k) = \nabla_R g(k) \cdot P^T_{RR}(k|k)\ \nabla_R g^T(k)$, $B(k|k) = \begin{bmatrix} P^T_{RR}(k|k)\cdot\nabla_R g^T(k) \\ P^T_{RM}(k|k)\cdot\nabla_R g^T(k) \end{bmatrix}$, $g(k)$ *is given in Eq. (12). When the robot observes the landmark* $l_{N_{k-1}+1}$ *at time* $k+1$, *we can conclude that* $P_{\text{new}}(k+1\,|\,k+1) < P(k+1|k+1)$.

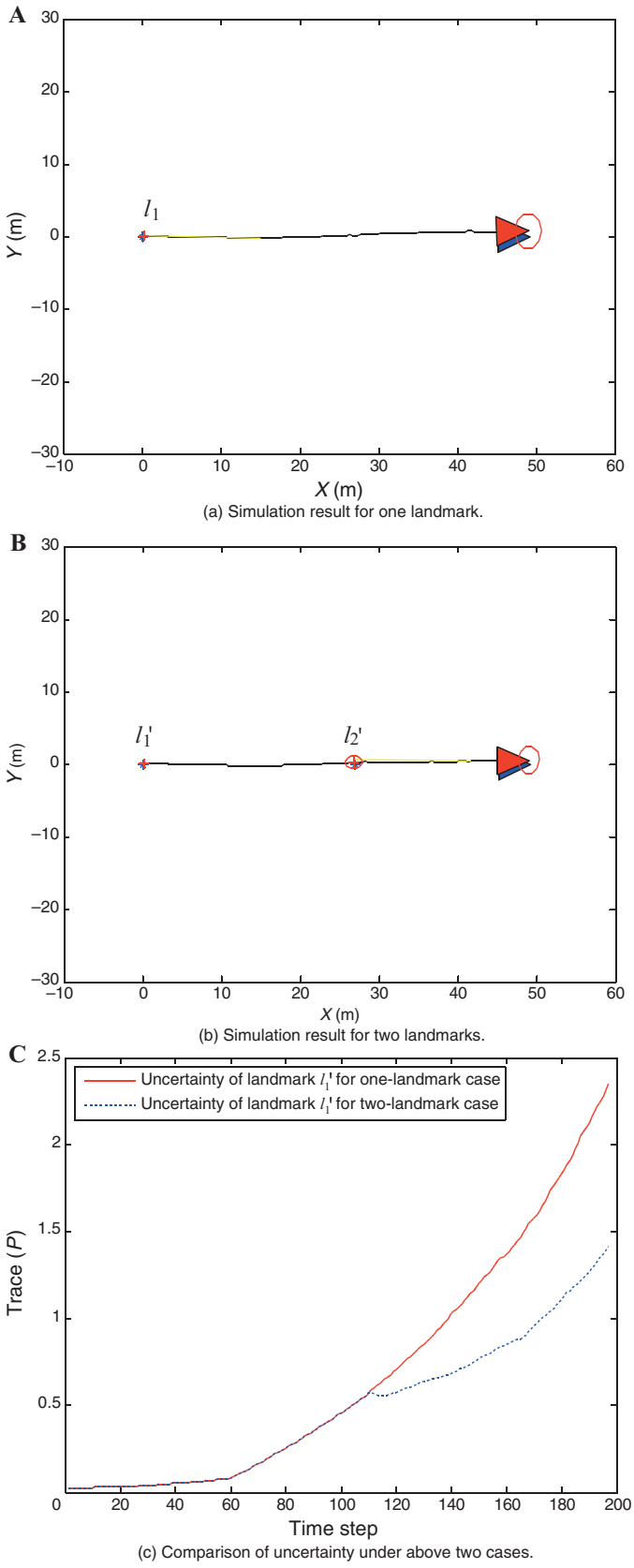
Proof. If the robot does not allocate the new landmark $l_{N_{k-1}+1}$ at time k, the covariance matrix at time $k+1$ is □

$$\begin{aligned} P(k+1|k+1) &= P(k+1|k) - W(k+1)\cdot S_{\gamma\gamma}(k+1|k)\cdot W^T(k+1) \\ &= P(k+1\,|\,k) - P(k+1\,|\,k)\cdot\nabla_X h^T \cdot S_{\gamma\gamma}(k+1\,|\,k)^{-1}\cdot\nabla_X h\cdot P(k+1\,|\,k). \end{aligned}$$

If the robot allocates the new landmark $l_{N_{k-1}+1}$ at time k, then the covariance matrix of the new state vector is denoted by

$$\begin{aligned} \bar{P}(k|k) &= \begin{bmatrix} P_{R,R}(k|k) & P_{R,M}(k|k) & P_{R,l_{N_{k-1}+1}}(k|k) \\ P_{M,R}(k|k) & P_{M,M}(k|k) & P_{M,l_{N_{k-1}+1}}(k|k) \\ P_{l_{N_{k-1}+1},R}(k|k) & P_{l_{N_{k-1}+1},M}(k|k) & P_{l_{N_{k-1}+1},l_{N_{k-1}+1}}(k|k) \end{bmatrix} \\ &= \begin{bmatrix} P_{\text{new}}(k|k) & B(k|k) \\ B^T(k|k) & A(k|k) \end{bmatrix}. \end{aligned}$$

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Simulation result for one landmark.

(b) Simulation result for two landmarks.

(c) Comparison of uncertainty under above two cases.

Fig. 2. (Colour online) Utility of landmark allocation. The robot allocated one landmark l_1 at the origin in (a), while allocated two landmarks l_1' and l_2' in (b). Comparison of uncertainty between l_1 and l_1' is given in (c).

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

After the new landmark $l_{N_{k-1}+1}$ is allocated, we set $\nabla_X \bar{f} = \begin{bmatrix} \nabla_X f & 0 \\ 0 & I_{2\times 2} \end{bmatrix}$, $\nabla_X \bar{h} = \begin{bmatrix} \nabla_X h & 0 \\ C & D \end{bmatrix}$, where $C = [\nabla_{X_R} h, 0, \cdots 0]$, $D = \nabla_{l_{N_{k-1}+1}} h$, and $\bar{Q} = \begin{bmatrix} Q & 0 \\ 0 & \tilde{Q} \end{bmatrix}$ is the uncertainty arising from the motion noise of the robot, as defined in Section 2.

Then we calculate $\bar{P}(k+1|k+1)$ by

$$\bar{P}(k+1|k+1) = \bar{P}(k+1|k) - \bar{P}(k+1|k) \cdot \nabla_X \bar{h}^T \cdot \bar{S}_{\gamma\gamma}^{-1}(k+1\,|\,k) \cdot \nabla_X \bar{h} \cdot \bar{P}(k+1|k), \quad (13)$$

where $\bar{S}_{\gamma\gamma}(k+1\,|\,k)$ is the covariance matrix of the innovation. We calculate the two terms on the right-hand side of Eq. (13) as follows:

$$\begin{aligned} \bar{P}(k+1|k) &= \nabla_X \bar{f} \cdot \bar{P}(k|k) \cdot \nabla_X \bar{f}^T + \bar{Q}(k) \\ &= \begin{bmatrix} \nabla_X f \cdot P(k|k) \cdot \nabla_X f^T & \nabla_X f \cdot B \\ B^T \cdot \nabla_X f^T & A \end{bmatrix} + \begin{bmatrix} Q & 0 \\ 0 & \tilde{Q} \end{bmatrix} \\ &= \begin{bmatrix} P(k+1|k) & \nabla_X f \cdot B \\ B^T \cdot \nabla_X f^T & A + \tilde{Q} \end{bmatrix}. \end{aligned} \quad (14)$$

Substituting Eq. (14) into the second term on the right-hand side of Eq. (13), we get

$$\begin{aligned} &\bar{P}(k+1|k) \cdot \nabla_X \bar{h}^T \cdot \bar{S}_{\gamma\gamma}^{-1}(k+1\,|\,k) \cdot \nabla_X \bar{h} \cdot \bar{P}(k+1|k) \\ &= \begin{bmatrix} \bar{M}_1 \cdot \bar{S}_{\gamma\gamma}^{-1} \cdot \bar{M}_1^T & \bar{M}_1 \cdot \bar{S}_{\gamma\gamma}^{-1} \cdot \bar{M}_2^T \\ \bar{M}_2 \cdot \bar{S}_{\gamma\gamma}^{-1} \cdot \bar{M}_1^T & \bar{M}_2 \cdot \bar{S}_{\gamma\gamma}^{-1} \cdot \bar{M}_2^T \end{bmatrix}. \end{aligned} \quad (15)$$

In Eq. (15), $\bar{M}_1$, $\bar{M}_2$ and $\bar{S}_{\gamma\gamma}^{-1}$ are given by

$$\begin{bmatrix} \bar{M}_1 \\ \bar{M}_2 \end{bmatrix} = \begin{bmatrix} P(k+1|k) \cdot \nabla_X h^T & P(k+1|k) \cdot C^T + \nabla_X f \cdot B \cdot D^T \\ B^T \cdot \nabla_X f^T \cdot \nabla_X h^T & B^T \cdot \nabla_X f^T \cdot C^T + (A + \tilde{Q}) \cdot D^T \end{bmatrix},$$

$$\bar{S}_{\gamma\gamma}^{-1} = (\nabla_X \bar{h} \cdot \bar{P}(k+1|k) \cdot \nabla_X \bar{h}^T + \bar{R})^{-1} = \begin{bmatrix} S_{\gamma\gamma} & E \\ E^T & F \end{bmatrix}^{-1} = \begin{bmatrix} G & H \\ H^T & J \end{bmatrix},$$

where

$$\begin{aligned} S_{\gamma\gamma} &= \nabla_X h \cdot P(k+1|k) \cdot \nabla_X^T h + R(k+1), \\ E &= \nabla_X h^T \cdot P(k+1|k) \cdot C^T + \nabla_X h^T \cdot \nabla_X f \cdot B \cdot D^T, \\ F &= C \cdot P(k+1|k) \cdot C^T + D \cdot B^T \cdot \nabla_X f^T \cdot C^T + C \cdot \nabla_X f \cdot B, \\ &\quad \cdot D^T + D \cdot (A + \tilde{Q}) \cdot D^T, \\ G &= S_{\gamma\gamma}^{-1} + S_{\gamma\gamma}^{-1} \cdot E \cdot (F - E^T \cdot S_{\gamma\gamma}^{-1} \cdot E)^{-1} \cdot E^T \cdot S_{\gamma\gamma}^{-1}, \\ H &= -S_{\gamma\gamma}^{-1} \cdot E \cdot (F - E^T \cdot S_{\gamma\gamma}^{-1} \cdot E)^{-1}, \\ J &= (F - E^T \cdot S_{\gamma\gamma}^{-1} \cdot E)^{-1}. \end{aligned}$$

We rewrite $\bar{P}(k+1|k+1)$ in the form of a block matrix as

$$\bar{P}(k+1|k+1) = \begin{bmatrix} P_{\text{new}}(k+1|k+1) & B(k+1|k+1) \\ B^T(k+1|k+1) & A(k+1|k+1) \end{bmatrix}.$$

Substituting Eqs. (14) and (15) into Eq. (13), we obtain

$$P_{\text{new}}(k+1|k+1) = P(k+1|k) - \bar{M}_1 \cdot \bar{S}_{\gamma\gamma}^{-1} \cdot \bar{M}_1^T. \quad (16)$$

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

We set $\bar{M}_{11} = P(k+1\,|\,k)\cdot\nabla_X h^T$, $\bar{M}_{12} = P(k+1\,|\,k)\cdot C^T + \nabla_X f\cdot B\cdot D^T$, then

$$\begin{aligned}\bar{M}_1\cdot\bar{S}_{\gamma\gamma}^{-1}\cdot\bar{M}_1^T &= \bar{M}_{11}\cdot G\cdot\bar{M}_{11}^T + \bar{M}_{12}\cdot H^T\cdot\bar{M}_{11}^T + \bar{M}_{11}\cdot H\cdot\bar{M}_{12}^T + \bar{M}_{12}\cdot J\cdot\bar{M}_{12}^T\\ &= P(k+1|k)\cdot\nabla_X h^T\cdot S_{\gamma\gamma}^{-1}\cdot\nabla_X h\cdot P(k+1|k) + [I - P(k+1|k)\\ &\quad\cdot\nabla_X h^T\cdot S_{\gamma\gamma}^{-1}\cdot\nabla_X h]\cdot[P(k+1|k)\cdot C^T + \nabla_X f\cdot B\cdot D^T]\cdot(F - E^T\\ &\quad\cdot S_{\gamma\gamma}^{-1}\cdot E)^{-1}\cdot[P(k+1|k)\cdot C^T + \nabla_X f\cdot B\cdot D^T]^T\cdot[I - P(k+1|k)\\ &\quad\cdot\nabla_X h^T\cdot S_{\gamma\gamma}^{-1}\cdot\nabla_X h]^T. \end{aligned}\tag{17}$$

In Eq. (17), we set

$$\begin{aligned}\Delta P &= [I - P(k+1|k)\cdot\nabla_X h^T\cdot S_{\gamma\gamma}^{-1}\cdot\nabla_X h]\cdot[P(k+1|k)\cdot C^T\\ &\quad + \nabla_X f\cdot B\cdot D^T]\cdot(F - E^T\cdot S_{\gamma\gamma}^{-1}\cdot E)^{-1}\cdot[P(k+1|k)\cdot C^T\\ &\quad + \nabla_X f\cdot B\cdot D^T]^T\cdot[I - P(k+1|k)\cdot\nabla_X h^T\cdot S_{\gamma\gamma}^{-1}\cdot\nabla_X h]^T. \end{aligned}\tag{18}$$

By substituting Eq. (18) into Eq. (17), we get $\bar{M}_1\cdot\bar{S}_{\gamma\gamma}^{-1}\cdot\bar{M}_1^T = P(k+1\,|\,k)\cdot\nabla_X h^T\cdot S_{\gamma\gamma}^{-1}\cdot\nabla_X h\cdot P(k+1\,|\,k) + \Delta P$, where both $\bar{S}_{\gamma\gamma}$ and $\bar{S}_{\gamma\gamma}^{-1}$ are symmetric and positive definite matrices, thus $J > 0$ and $\Delta P > 0$, then we rewrite Eq. (16) as

$$\begin{aligned}&P_{\text{new}}(k+1|k+1)\\ &= P(k+1|k) - [P(k+1|k)\cdot\nabla_X h^T\cdot S_{\gamma\gamma}^{-1}\cdot\nabla_X h\cdot P(k+1|k) + \Delta P] < P(k+1|k+1)\\ &= P(k+1|k+1) - \Delta P. \end{aligned}\tag{19}$$

From Eq. (19), the covariance of the system state decreases after a new landmark l_{N_k+1} is allocated. □

4. Active Exploration Based on Landmark Allocation

The task of exploration generally has several objectives, including the map coverage rate, the map accuracy and sometimes the exploration efficiency. This implies that control inputs of the mobile robot should be elegantly chosen such that the robot can detect maximal unknown space and minimize the uncertainty of the system state under estimation. The exploration efficiency can be evaluated by the traveling distance. But for the sake of clarity, we only take into account the map coverage rate and the accuracy of SLAM. Note that the traveling distance can also be easily integrated into the cost function.

The level of uncertainty plays a key role in localization and map building. Low uncertainty of the robot pose and landmark positions can help the robot to achieve correct data association easily. In addition, when a new landmark is observed, the current robot pose estimate is added to the relative observation between the robot and the landmark in order to form an absolute position estimate of the landmark. If the estimate of the robot pose has some uncertainty before the landmark estimate is initialized, the landmark estimate will inherit the robot position uncertainty. Similarly, any landmarks that are initialized must have at least the initial robot position uncertainty as part of their overall position uncertainty. Thus, if the robot pose estimate has high uncertainty, it would initialize and maintain the map at a high-level uncertainty, especially when loop closure cannot be detected.

On the other hand, further observations of landmarks can generally increase the accuracy of the SLAM. Localization of inaccurate landmarks is generally more effective if the robot travels between inaccurate landmarks and well-known landmarks because the correlation between the good and poor landmarks increases. However, if the robot revisits known landmarks too often, the coverage time would then increase.

The active exploration proposed in this paper refers to a tight coupling between the tasks of landmark allocation, SLAM and motion planning. Our approach to active exploration calls for

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

a balanced evaluation of alternative motion actions from the viewpoint of the uncertainty of the localization and mapping, the coverage rate of the unknown space and the utility of the allocated landmarks.

During the exploration, the robot continuously plans its motion and determines where a landmark should be allocated. Simultaneously, the robot estimates the positions of the allocated landmarks and localizes itself based on the observations of the landmarks.

4.1. Uncertainty of localization and mapping

Uncertainty of localization and mapping is calculated by

$$\begin{aligned} C(P(k+1|k+1)) &= \pi \prod_{j=1}^{3} \sqrt{\lambda_j(P_{RR}(k+1|k+1))} + \pi \sum_{i=1}^{N_k} \prod_{j=1}^{2} \sqrt{\lambda_j(P_{ii}(k+1|k+1))} \\ &= \pi \sqrt{\det(P_{RR}(k+1|k+1))} + \pi \sum_{i=1}^{N_k} \sqrt{\det(P_{ii}(k+1|k+1))}, \end{aligned} \tag{20}$$

where $P_{ii}(k+1|k+1)$ denotes the covariance of the ith landmark, it is a diagonal sub-matrix of $P_{MM}(k+1\,|\,k+1)$, and $\det(\cdot)$ and $\lambda_j(\cdot)$ respectively denote the determinant and the eigenvalues of the related matrix.

4.2. Coverage rate

We use an occupancy grid map to represent the environment. In a grid map, each grid cell is classified as follow:

Known: the cell has been detected by the robot's sensor;

Unknown: the cell has not been detected.

Assume that the onboard sensor of the robot can detect a circular region with a radius of $d_{\max}$. Frontiers are defined as the regions on the boundary between the known and the unknown space.[34] A frontier cell is a known, i.e. already explored cell, which is an immediate neighbor of an unknown, i.e. unexplored cell. When the robot moves to a frontier cell, it can see into unexplored space and add the new information to its map. As a result, the map built by the robot expands, pushing forward the boundary between the known and the unknown space. By moving to the frontiers successively, the robot can continuously increase its knowledge of the world. For simplicity, the robot only uses its current pose $\hat{X}_R(k|k)$ estimated by the EKF to calculate the frontiers. If a cell is within the sensing range of the robot, it will be labeled as "known," otherwise "unknown." When all the cells of the environment with a given boundary are labeled as "known," the robot can know that the environment is fully explored.

Frontier cells between explored and unexplored environments are calculated using the method proposed in Yamauchi.[34] Based on the frontiers, the coverage rate of the environment is evaluated by calculating the predictive new area, $Area_{\text{new}}(k+1|k)$, of the unknown regions that will be explored by the robot in the next step,

$$Area_{\text{new}}(k+1|k) = Area_{\text{cell}} \cdot N_{\text{sensor_new}}(k+1|k), \tag{21}$$

where $Area_{\text{cell}} = a^2$ denotes the area (or the size) of a cell, and a is the grid resolution. $N_{\text{sensor_new}}$ represents the predictive number of unexplored cells that will fall into the sensing region around the predictive robot pose, $\hat{X}_R(k+1\,|\,k)$.

4.3. Utility of the allocated landmark

We define the trace of the matrix ΔP, denoted by $\text{tr}(\Delta P)$ as an evaluation of the utility provided by the newly allocated landmark. The control inputs of the robot are to be chosen to maximize $\text{tr}(\Delta P)$ such that the accuracy of localization and mapping can be improved as quickly as possible.

4.4. Objective function

For simplicity, we assume that the landmark is placed at the origin of the coordinate frame attached to the robot. The objective function is obtained by combining the accuracy of the SLAM, the coverage

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

rate and the utility of the allocated landmark.

$$J(u(k), u(k+1), \ldots u(k+H-1)) = \sum_{i=1}^{H} \left[\omega_1 \cdot \frac{C(P_{\min})}{C(P(k+i \mid k+i))} + \omega_2 \cdot \frac{Area_{\text{new}}(k+i|k+i-1)}{Area_{\max}} + \omega_3 \cdot \frac{\text{tr}(\Delta P(k+i|k+i))}{\text{tr}(P(k+i|k+i))} \right], \tag{22}$$

where H is the predictive horizon. A small horizon may be greedy whereas a large horizon may result in expensive computation consumption. Through a number of simulations, we set $H = 3$ to achieve satisfactory performance. $P_{\min}$ represents the lower bound[2] of the covariance matrix associated with the state estimate. $Area_{\max} = \pi d_{\max}^2$ is the area of the sensing disc of the robot sensor, and ω_i $(i = 1,\ 2,\ 3)$ is the importance weight. Thus, the active control input sequence can be calculated by

$$[u^*(k), u^*(k+1), \cdots u^*(k+H-1)] = \arg\max_{u \in U} J(u(k), u(k+1), \cdots u(k+H-1)), \tag{23}$$

where $U = \{u = [u_1, u_2]^T | u_{1\min} \le u_1 \le u_{1\max}, u_{2\min} \le u_2 \le u_{2\max}\}$ is the set of feasible control inputs. In our approach, the feasible control inputs are generated by discretizing the velocity space into a small set of fixed discrete linear and angular velocities in order to simplify the implementation (see Section 6.1). In contrast to our choice, the feasible control inputs can also be designed as straight ahead, left and right by one grid cell distance, which depend on the size of the cell. The advantage of our choice is that the control inputs used in our approach can be determined only by the motion ability of the robot, rather than by the size of the cell.

Then we use $\{u^*(k), u^*(k+1), \cdots u^*(k+H-1)\}$ to calculate the total utility of the allocated landmark by

$$\text{Tr} = \sum_{i=1}^{H} \frac{tr(\Delta P(k+i \mid k+i))}{tr(P(k+i|k+i))}. \tag{24}$$

If $Tr \ge \Delta P_t$, the robot will allocate a new landmark at the current position, where ΔP_t is the threshold determined by the experiments. And then the robot uses only $u^*(k)$ as the control input at time k. At time $k+1$, the optimal control input sequences are calculated by Eq. (23) again and only $u^*(k+1)$ is carried out. If $Tr < \Delta P_t$, no landmark is required to be allocated. In this case, we set ω_3 as zero in Eq. (22) and optimize only the first two terms of the objective function. After computing the optimal control input sequence $[u^{*'}(k), u^{*'}(k+1), \cdots u^{*'}(k+H-1)]$, the robot executes $u^{*'}(k)$ as the current control input. During the previous process, if the robot observes no landmark, it will allocate a new landmark right away according to Rule 1. The above process is repeated till the robot sensor covers the whole environment.

In order to avoid the local minimum, we give a simple but effective strategy as follows. If $C(P_{MM}(k+1|k+1))/N_k \le C_t$ and $Area_{\text{new}}(k+1|k) = 0$, which means the uncertainty of the map is small enough and the robot does not acquire new information from the unknown space, then the robot will move toward its nearest frontier cell until the condition $C(P_{MM}(k+1 \mid k+1))/N_k > C_t$ or $Area_{\text{new}}(k+1|k) > 0$ is satisfied, where C_t is a given threshold value.

4.5. Supplementation and redundancy elimination for landmarks

To distinguish the concepts of "complete exploration" and "complete coverage by landmarks," we give the following definitions.

Definition 1 (*Complete exploration*): If the whole environment has already been covered by the onboard sensor of the robot, we call the exploration as complete.

Definition 2 (*Complete coverage by landmarks*): At any positions, if the robot can always observe landmarks, we call that the landmark allocation is complete, or the environment is completely covered by the landmarks.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

An essential purpose of landmark allocation is to cover the environment completely with sufficient landmarks. However, after the environment has been completely explored using the motion-planning algorithm proposed in Section 4.4, some local regions may not yet be covered by landmarks. When the robot re-enters these regions, it cannot observe any landmarks. As a result, it fails to localize itself. In this case, the robot should supplement some landmarks in these regions of interest to achieve a resulting complete coverage of landmarks for the environment.

On the other hand, more than one allocated landmarks may be observed at some positions, i.e. each of these positions is covered by multiple landmarks. Note that by using the proposed scheme, all the landmarks are estimated very accurately when the environment is completely explored. Thus, only one landmark is useful and the others are redundant. The robot should delete these redundant landmarks to eliminate redundancy.

Definition 3 (*Non-redundancy*): For a given local region, if the robot can observe *at most one* landmark at any position in this region, then we will call the landmark allocation in this region as non-redundant.

The robot carries out the tasks of landmark supplementation and redundancy elimination in order to generate complete and non-redundant coverage with the landmarks. To address this problem, we adopt the algorithm proposed in Jiang *et al.*[35] This algorithm was developed to solve the minimal cover set problem in the field of wireless sensor networks. In the tasks of supplementation and redundancy elimination, the landmarks and the robot are respectively viewed as equivalents of the sensor nodes and the target to be detected in a wireless sensor network. Because the problem of computing the minimal cover set is NP-hard, the algorithm in Jiang *et al.*[35] provided a sub-optimal solution. Here we refer to this algorithm and acquire a sub-optimal solution to supplementation and redundancy elimination accordingly. As a result, the robot can observe as few landmarks as possible at any position in the environment.

Assume that the robot allocates a set of landmarks $L_{\text{active}} = \{l_1, l_2 \cdots l_n\}$ after it completely explores the environment. The robot can observe the landmark l_j if it enters the disc around l_j with a radius of $d_{\max}$. For clarity, this region is denoted by S_j. We define $L_{\text{add_temp}}$ as a temporary landmark set that contains candidate landmarks to be supplemented. The initial state of $L_{\text{add_temp}}$ is empty. The algorithm of supplementation and redundancy elimination is presented as follows:

1. The robot calculates Voronoi partition of the environment for the landmark set $L_{\text{complete}} = L_{\text{active}} \cup L_{\text{add_temp}}$. We denote the Voronoi polygon associated with l_j as $VP_j(L_{\text{complete}})$. $VV_j(L_{\text{complete}})$ is referred to the set consisting of all the vertexes of $VP_j(L_{\text{complete}})$, and $VN_j(L_{\text{complete}})$ denotes the set of Voronoi neighbors of the landmark l_j. Figure 3 shows the result of the Voronoi partition, in which l_i is one of the Voronoi neighbors of l_j.
2. If one of the vertexes of the $VP_j(L_{\text{complete}})$ associated with $l_j \in L_{\text{complete}}$ cannot be covered by S_j, the robot will add a imaginary landmark $l_{j,\text{add}}$ at this vertex and add it into $L_{\text{add_temp}}$.
3. Repeat steps (1) and (2) until the whole environment is covered by L_{complete}.
4. Execute the algorithm for calculating redundant nodes in Jiang *et al.*[35] to identify the set of redundant landmarks denoted as $L_{\text{redundant}}$. The details are given as follows.
   If $VV_j(L_{\text{complete}}) \subset S_j$ $(\forall l_j \in L_{\text{complete}})$, the environment will be covered by L_{complete}. When a landmark l_i is deleted from L_{complete}, it can only affect $VP_j(L_{\text{complete}} \backslash l_i)$ $(l_j \in VN_i(L_{\text{complete}})$, $L_{\text{complete}} \backslash l_i = L_{\text{complete}} - \{l_i\})$, while the Voronoi polygons of the other landmarks do not change. Thus, l_r can be regarded as a redundant landmark if the condition of $VV_j(L_{\text{complete}} \backslash l_r) \subset S_j$ holds for any $l_j \in VN_r(L_{\text{complete}})$. For a redundant landmark l_r, if its Voronoi neighbor set contains at least one redundant landmark, we call that l_r is dependently redundant, otherwise independently redundant. Independently redundant landmarks can be deleted directly. In what follows, we consider the dependently redundant landmarks. We first construct an undirected graph $RG(V_r, E_r)$ for all the dependently redundant landmarks. The nodes V_r of $RG(V_r, E_r)$ are the dependently redundant landmarks. The edges of $RG(V_r, E_r)$ are denoted by $E_r = \{(v_{ri}, v_{rj}) \mid v_{ri}, v_{rj} \in V_r, v_{rj} \in$. That is, if v_{rj} is a Voronoi neighbor of v_{ri}, we will connect v_{rj} with v_{ri} and add (v_{ri}, v_{rj}) into E_r. Then we use the greedy algorithm presented in Hochbaum[36] to determine the maximal independent set (MIS) of $RG(V_r, E_r)$. The landmarks in the MIS can be deleted and the others in $RG(V_r, E_r)$ are useful for complete coverage. Finally, $L_{\text{redundant}}$ is composed of independently redundant landmarks and the MIS of $RG(V_r, E_r)$.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

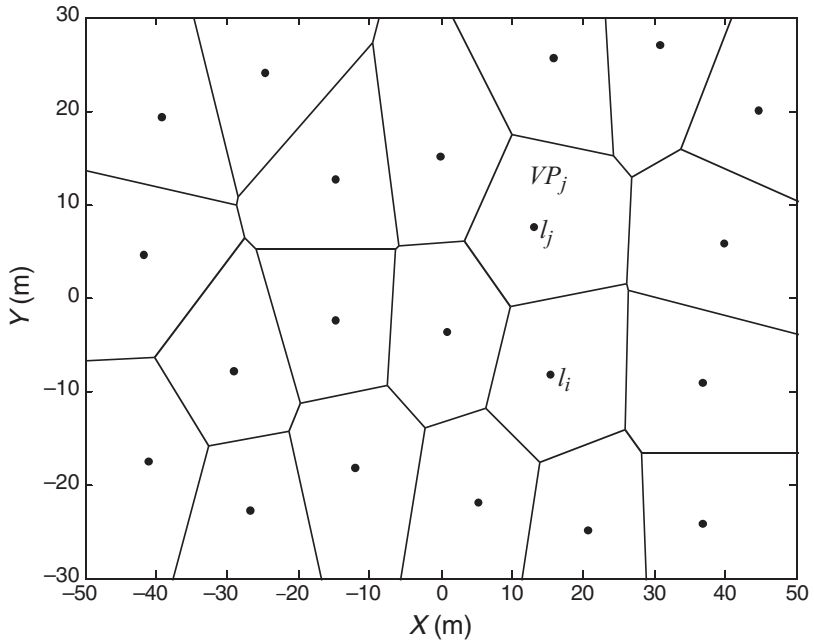


Fig. 3. Voronoi partition of the environment. VP_j is the Voronoi polygon associated with the landmark l_j. l_i is one of the Voronoi neighbors of l_j.

5. The set $L_{\text{add_temp}}$ is composed of imaginary candidate landmarks to be supplemented. Thus, the set of landmarks to be really supplemented is $L_{\text{add_allocation}} = L_{\text{add_temp}} - (L_{\text{add_temp}} \cap L_{\text{redundant}})$. The robot successively moves to each one of the positions in $L_{\text{add_allocation}}$, in ascending order of the distance from the robot position to the landmark position, and then places a landmark there.

When the algorithm terminates, the set of landmarks left in the environment is $L_{\text{final}} = L_{\text{complete}} - L_{\text{redundant}}$. The set $L_{\text{delete}} = L_{\text{active}} \cap L_{\text{redundant}}$ contains the landmarks required to be really removed. The robot only needs to delete them from its memory.

5. Development of landmarks and the device for allocating landmarks

5.1. Landmark design

A two-dimensional (2D) laser rangefinder can only scan objects beyond a certain volume and height, such that miniature landmarks on the ground cannot be detected. Therefore, we develop a kind of sensor device consisting of landmarks and the associated detector, instead of using laser rangefinder. This device can provide range and bearing measures between the robot and the detected landmarks.

We design the landmark as an embedded system, as shown in Fig. 4. We adopt the modularized design scheme, in which the architecture and the circuit of the detector and those of each landmark are exactly the same, i.e. the detector and the landmark can replace each other. The landmark/detector is composed of a microprocessor, an RFID transceiver, an ultrasonic transceiver and related interface circuits. The RFID module and the ultrasonic module are integrated in each landmark to provide the landmark-identifier (ID) information and range measurement simultaneously. The Time Difference of Arrival (TDOA) between RF and ultrasonic signals is computed to estimate the distance between the landmark and the detector.

We use a combination of RF and ultrasound signals to enable the detector to determine the distance to the landmarks. We achieve this by taking advantage of the fact that the speed of sound in air (about 344 m/s at room temperature) is much smaller than the speed of light (RF) in air. The landmark periodically transmits an RF message containing a unique ID. At the start of the RF message, the landmark concurrently transmits a narrow ultrasonic pulse. When the detector hears the RF signal, it turns on its ultrasonic receiver. It then listens for the ultrasonic pulse, which will usually arrive a

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

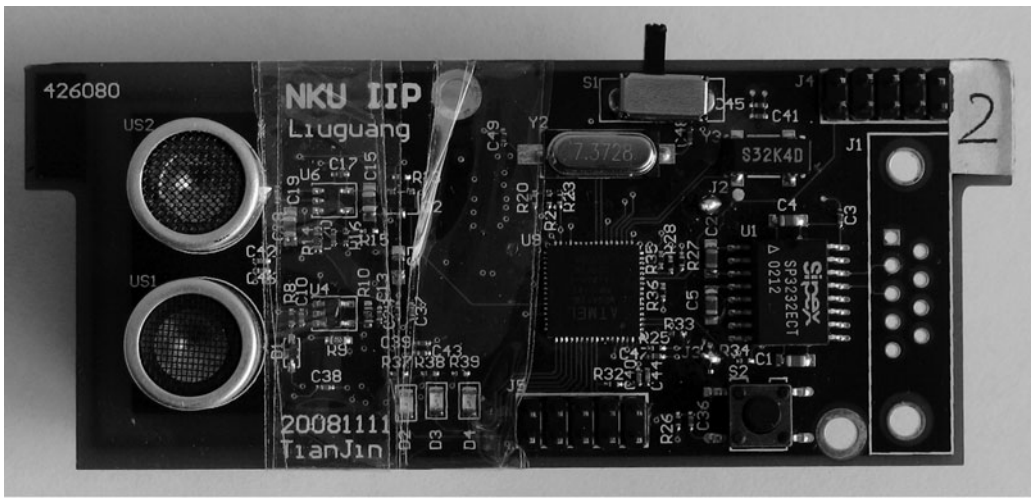


Fig. 4. Hardware circuit of the landmark/detector.

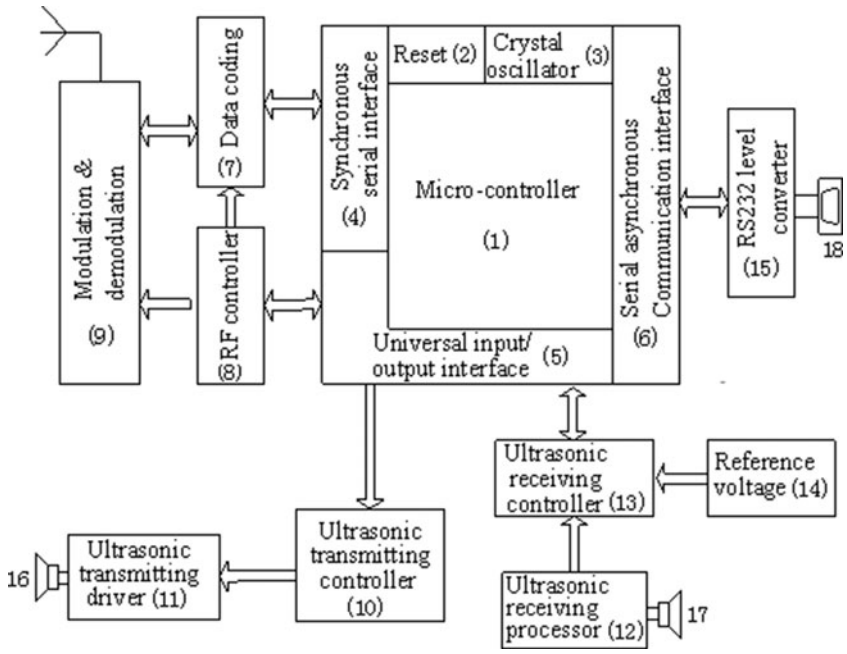
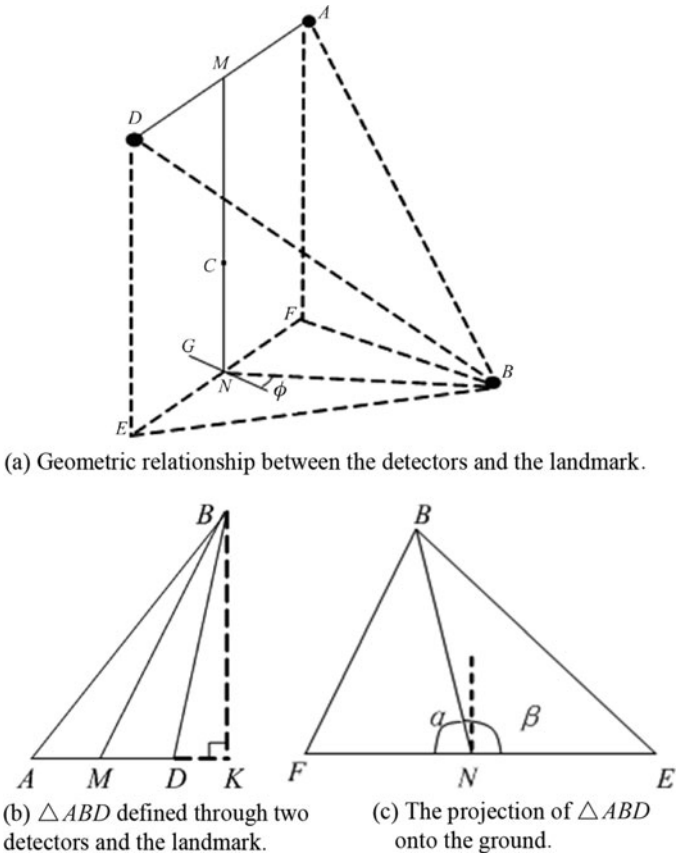


Fig. 5. Structure diagram of the landmark/detector.

short time later. The detector uses the time difference between the receipt of the RF information and the ultrasonic signal to determine the distance to the landmark. To reduce power consumption of the landmark, the ultrasonic pulse does not carry any data. The detector passively listens to landmark transmissions and computes the distance to the landmark. Figure 5 shows the structure diagram of the landmark/detector. Five modules are embedded into each landmark, including a microprocessor and related interface circuits, an RS232 serial communication unit, an RF transceiver, an ultrasonic transmitter and an ultrasonic receiver.

The microprocessor and its interface circuits are composed of a micro-controller (U1), a reset circuit (U2) and a crystal oscillator circuit (U3). The RS232 serial communication unit comprises a serial asynchronous communication interface (U6) and an RS232-level converter (U15). The RF transceiver contains a synchronous serial interface (U4), a universal input/output interface (U5), a

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Geometric relationship between the detectors and the landmark.

(b) $\triangle ABD$ defined through two detectors and the landmark.

(c) The projection of $\triangle ABD$ onto the ground.

Fig. 6. Schematic drawing for the localization principle. C denotes the center of the robot. A and D denote respectively the two detectors, and B denotes the allocated landmark. E and F are respectively the projections of A and D onto the ground. $|BN|$ and ϕ are the range and the bearing of the landmark B.

data coding unit (U7), an RF controller (U8) and a modulation and demodulation units (U9), wherein U7, U8 and U9 are realized with the chip CC1000. The ultrasonic transmitter is composed of a transmitting controller unit (U10) and a transmitting driver unit (U11). The ultrasonic receiver is made up of a receiving processor unit (U12), a receiving controller unit (U13) and a reference voltage unit (U14).

5.2. Localization principle

As shown in Fig. 6(a), C denotes the center of the robot platform, N is the projection of C onto the groundplane and B denotes the observed landmark. If two detectors are installed at positions A and D with the same given height, and satisfy$|AM| = |MD|$, the range $|BN|$ and the bearing ϕ of the observed landmark can be calculated by using the law of cosines. Since both lines AF and DE are orthogonal to the ground, from Fig. 6(b) we get,

$$|AB|^2 = (|AD| + |DK|)^2 + |BK|^2, \quad |BD|^2 = |DK|^2 + |BK|^2. \tag{25}$$

$$\begin{aligned} |BM|^2 &= (|MD| + |DK|)^2 + |BK|^2 \\ &= (|AD|/2 + |DK|)^2 + |BK|^2 \\ &= |DK|^2 + |BK|^2 + |AD|^2/4 + |AD| \cdot |DK| \\ &= \frac{|BD|^2 + |AB|^2}{2} - \frac{|AD|^2}{4}. \end{aligned} \tag{26}$$

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

Besides, $|AD|$ and $|MN|$ are known, and $|AB|$, $|BD|$ can be measured, thus $|BN|$ can be determined by

$$|BN| = \sqrt{|BM|^2 - |MN|^2} = \sqrt{\frac{|BD|^2 + |AB|^2}{2} - \left(|MN|^2 + \frac{|AD|^2}{4}\right)}. \tag{27}$$

The derivation of ϕ is given as follows. According to the law of cosines, we get Eqs. (28) and (29) from Fig. 6(c),

$$\cos\alpha = \frac{|BN|^2 + |FN|^2 - |BF|^2}{2 \cdot |BN| \cdot |FN|}, \tag{28}$$

$$\cos\beta = \frac{|BN|^2 + |EN|^2 - |BE|^2}{2 \cdot |BN| \cdot |EN|}. \tag{29}$$

Substituting Eqs. (28) and (29) into the following equation,

$$\begin{aligned} \cos\alpha - \cos\beta &= 2\cos\alpha \\ &= \frac{|BE|^2 - |BF|^2}{2 \cdot |BN| \cdot |FN|} \\ &= \frac{(|BD|^2 - |DE|^2) - (|AB|^2 - |AF|^2)}{2 \cdot |BN| \cdot |FN|} \\ &= \frac{|BD|^2 - |AB|^2}{2 \cdot |BN| \cdot |FN|}. \end{aligned} \tag{30}$$

Then, we get,

$$\alpha = \arccos\left(\frac{|BD|^2 - |AB|^2}{4 \cdot |BN| \cdot |FN|}\right). \tag{31}$$

Thus,

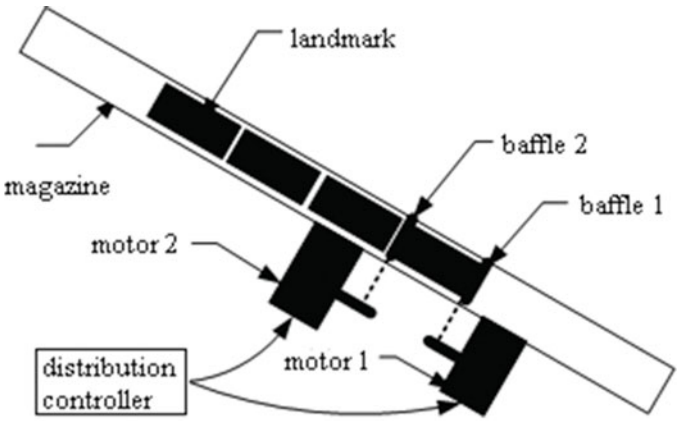
$$\phi = \frac{\pi}{2} - \arccos\left(\frac{|BD|^2 - |AB|^2}{4 \cdot |BN| \cdot |FN|}\right). \tag{32}$$

Using the range $|BN|$ and the bearing ϕ, the robot can estimate the landmark position with the EKF described in Section 2.

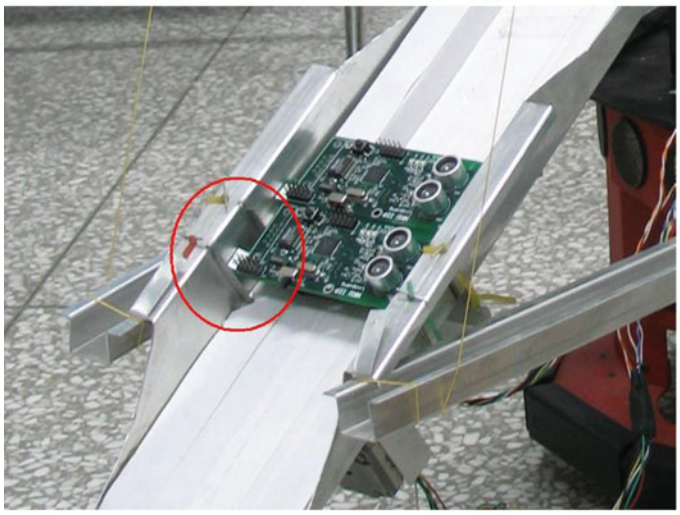
5.3. Design of the device for allocating landmarks

The device for allocating landmarks is made up of a magazine, two conventional motors, two metal baffle plates and a controller, as illustrated in Fig. 7. Multiple landmarks to be allocated in turn

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Design principle.



(b) Top-view photo.

Fig. 7. (Colour online) Device for landmark allocation.

are arranged in the magazine. In our experimental system, the controller for allocating landmarks is realized by 8051 MCU. Figure 7(a) depicts the working principle of the designed device. Each time the mechanism is triggered, the controller drives motor 2 to drop down baffle 2. After a landmark slides down between two baffles, baffle 2 will come back to the initial position. Then baffle 1 is dropped down by motor 1 such that the landmark falls onto the ground. Figure 7(b) shows the top-view photo of the allocating device.

6. Experimental Results

6.1. Exploration with a real robot

The experimental platform is a Pioneer 3-DX mobile robot equipped with some landmarks and the mechanical device for allocating landmarks, as depicted in Fig. 8. We fixed two landmark detectors onto a plastic shelf. Since the ultrasonic sensor transmits cone-shaped signals, we let the central axis of the ultrasonic signals be perpendicular to the ground. As a result, the sensing region of the detector formed a disc on the ground. The height of the detectors above the ground was 2.16 m, thus the detecting radius of the sensing disc was about 1.82 m.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

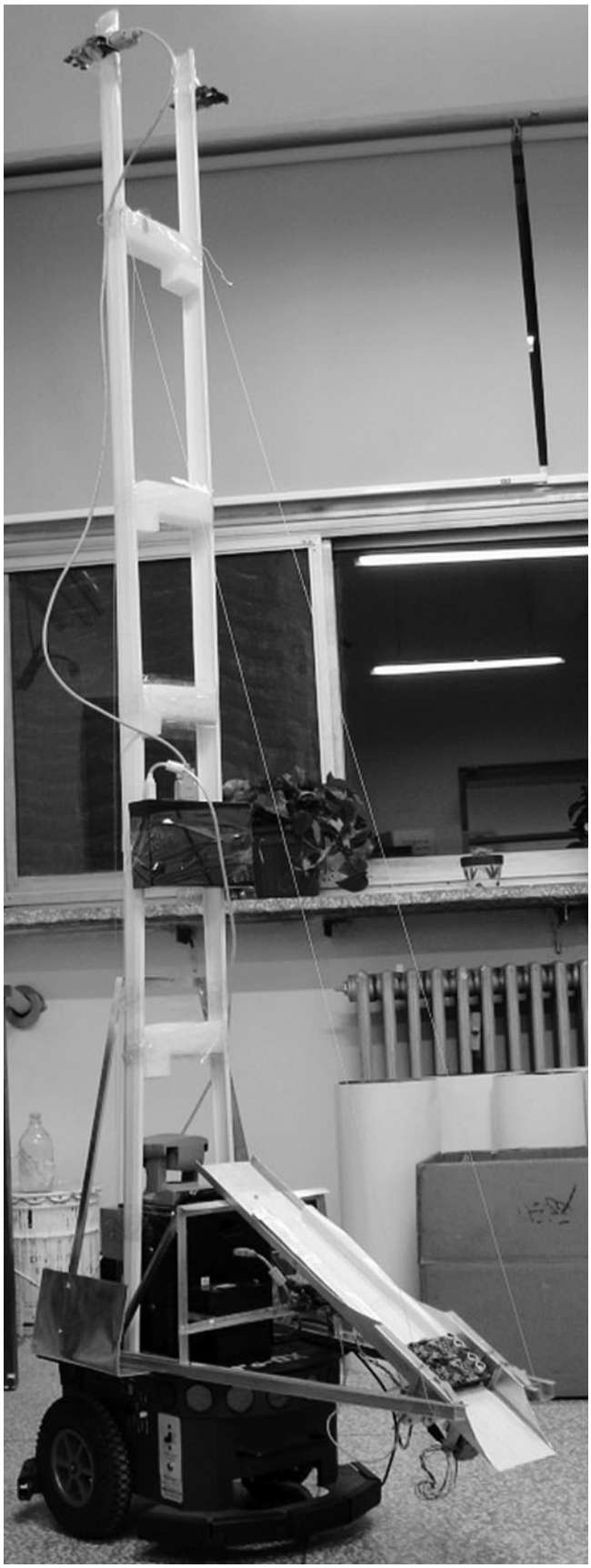
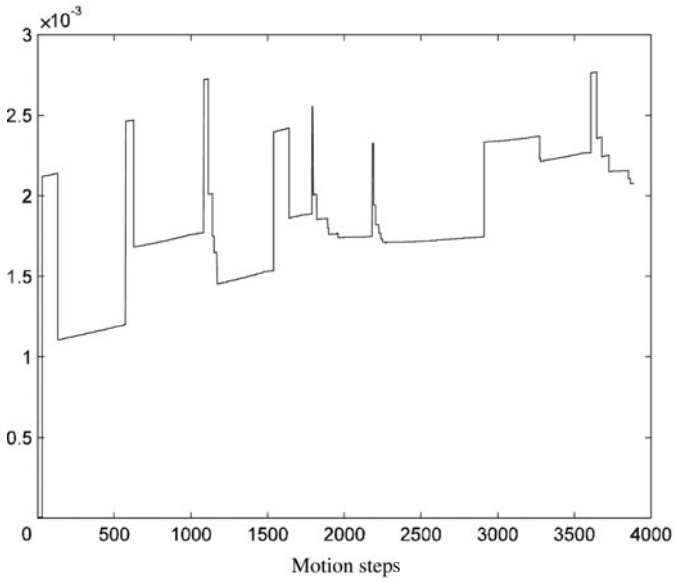


Fig. 8. Experimental platform.

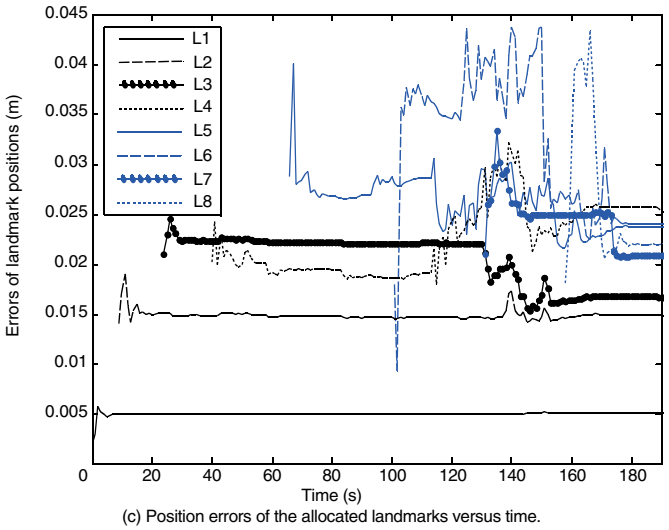
https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Experiment result of landmark allocation.



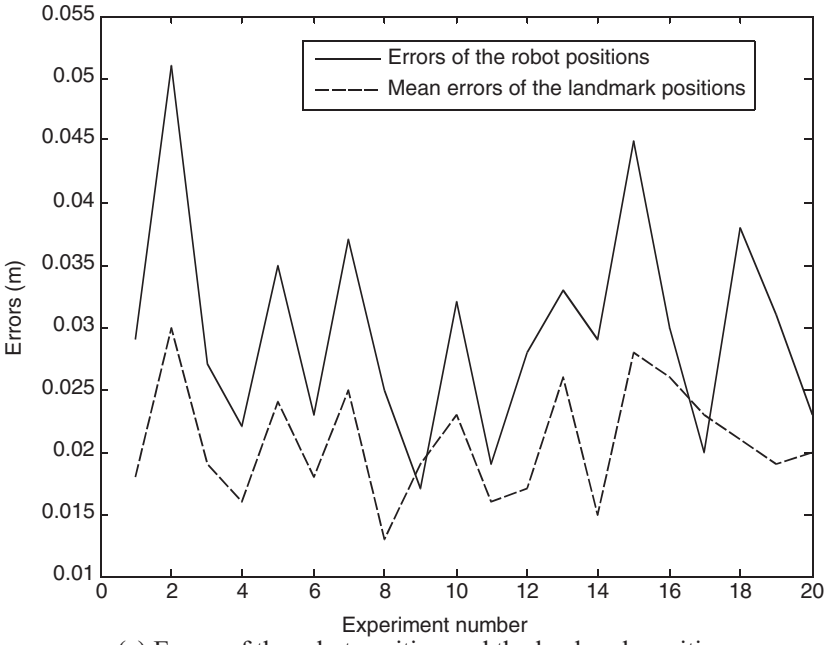
(b) Uncertainty of robot pose.



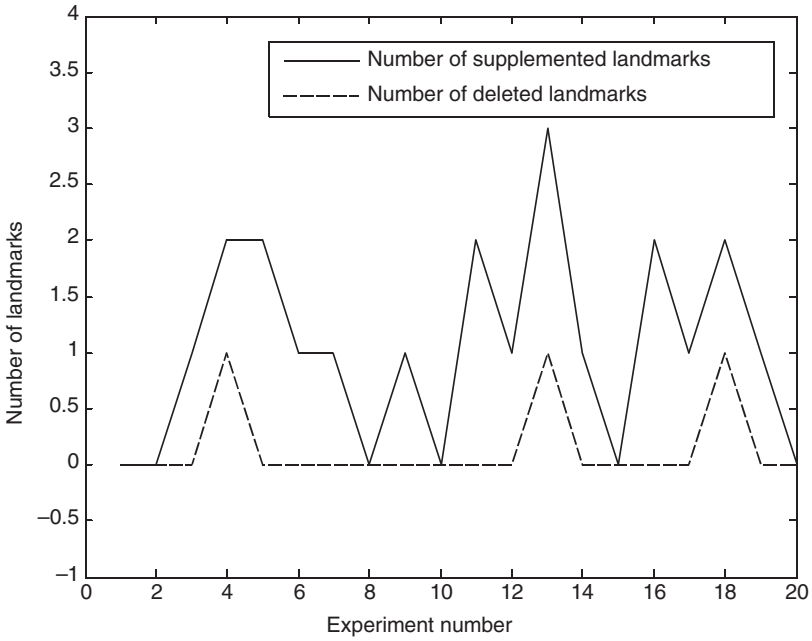
(c) Position errors of the allocated landmarks versus time.

Fig. 9. (Colour online) One of the experimental results. The robot allocated eight landmarks, which could cover the environment. The uncertainty of the robot pose and the errors of the landmarks are respectively shown in (b) and (c).

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Errors of the robot position and the landmark positions.



(b) The number of supplemented and deleted landmarks.

Fig. 10. Statistical result of 20 experiments.

Experiments were conducted in our lab with a size of 10 m × 4 m. The feasible control inputs were $u_1 = \{0.1, 0.2, 0.5\}$ m/s, $u_2 = \{-\pi/2, -3\pi/8, -\pi/4, -\pi/8, 0, \pi/8, \pi/4, 3\pi/8, \pi/2\}$ rad/s. The weights in Eq. (22) were $\omega_1 = \omega_2 = \omega_3 = 1/3$. Figure 9 provides the experimental result. The robot allocated eight landmarks when the environment was completely explored, as shown in Fig. 9(a). These landmarks could cover the whole environment and no landmark was redundant. Figure 9(b) shows the uncertainty of the robot pose. It can be seen that the uncertainty of the robot pose remained on a reasonably low level during the whole exploration process. When the experiment terminated, the ground truth of the robot position and the allocated landmark positions were measured. The error of the robot position was 0.031 m. Figure 9(c) shows the position errors of eight landmarks versus time.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

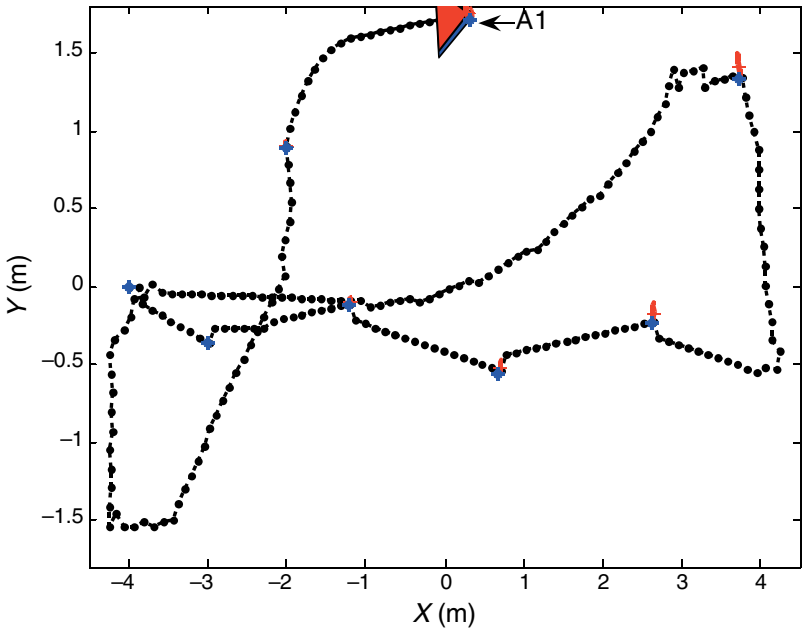


Fig. 11. (Colour online) One of the simulation results of landmark allocation. The robot allocated seven landmarks and supplemented one landmark.

We performed the experiments in the same environment for 20 times. The robot position errors and the mean absolute errors of all the allocated landmarks in 20 experiments are shown in Fig. 10(a). These errors were measured when the robot stopped exploring the environment. Figure 10(b) shows the number of supplemented and deleted landmarks.

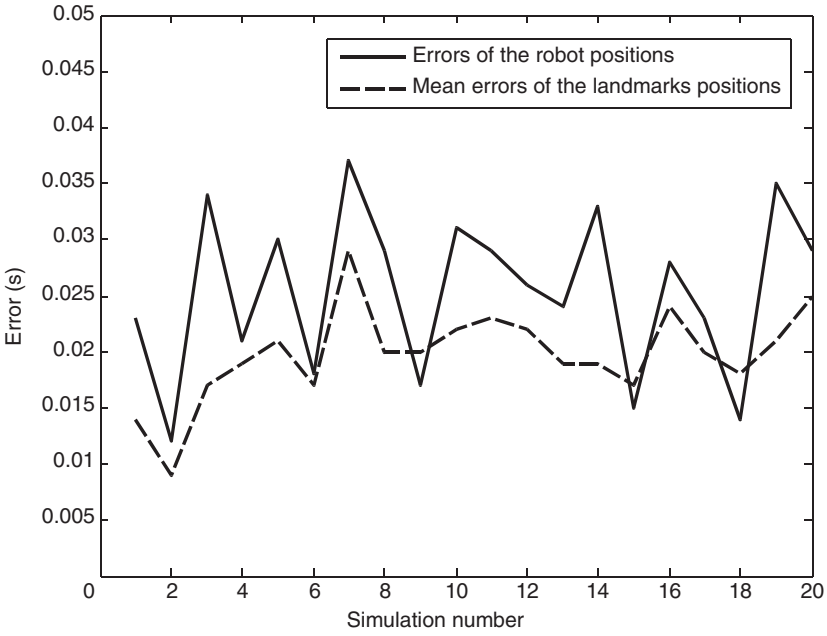
We also conducted the simulations under the same conditions for 20 times. Figure 11 gives one of the simulation results, in which the robot supplemented one landmark A1, and no landmark was redundant. The robot position errors and the mean absolute errors of all the allocated landmarks in 20 simulations are shown in Fig. 12(a). These errors were also computed when the robot stopped exploring the environment. Figure 12(b) shows the number of supplemented and deleted landmarks.

It is shown from the simulation and experiment results that landmark supplementation was implemented in most cases, but redundant landmarks appeared in only three experiments and three simulations. The reason is that in such a small-scale environment, the robot only needed to travel short distance to cover the environment with the on-board sensor. As a result, some regions, especially the regions adjacent to the environment boundary, could not be covered by the allocated landmarks, although they were already covered by the robot sensor. When this happened, the robot had to supplement some new landmarks into these regions to achieve a complete coverage for the environment with the landmarks.

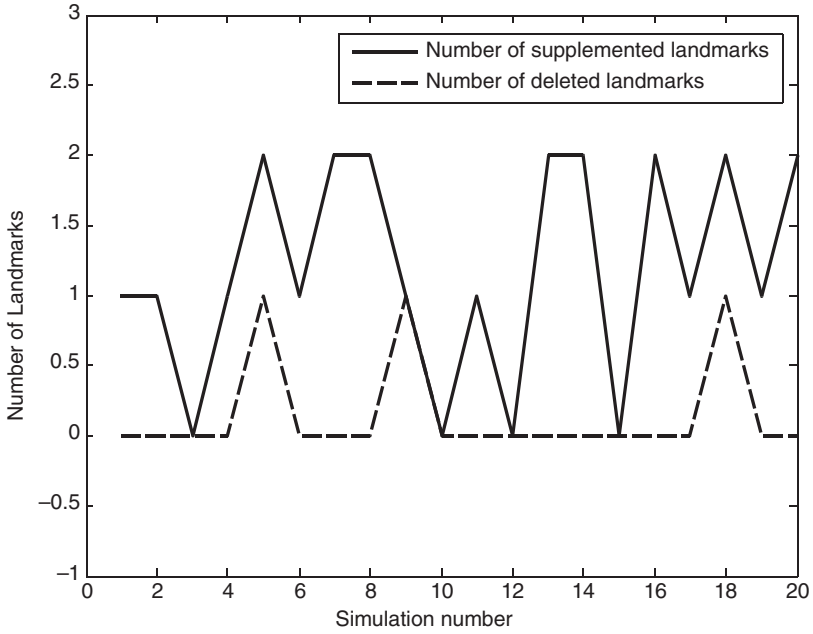
6.2. Simulation results in a large-scale environment

To get a more quantitative assessment, we performed several simulation experiments in an environment with a size of 100 m × 60 m. The initial pose of the robot was $X_R(0) = [0, 0, 0]^T$, the control period was $T = 0.25$ s, $\omega_1 = \omega_2 = \omega_3 = 1/3$, $d_{\max} = 15$ m, $Q = \text{diag}\{0.3^2, 0.3^2, 0.05^2, 0.3^2, \cdots, 0.3^2\}$, $R = \text{diag}\{0.2^2, 0.04^2\}$, $C_t = 0.1$, $H = 3\,T$ and $\Delta P_t = 0.25$. For simplicity, the control inputs were discretized as $u_1 = \{0.5, 1.0, 1.5, 2.0\}$ m/s, $u_2 = \{-\pi/2, -3\pi/8, -\pi/4, -\pi/8, 0, \pi/8, \pi/4, 3\pi/8, \pi/2\}$ rad/s. Figure 13 shows the simulation result in which "—" represents the estimated motion trajectory of the robot, "*" represents the exact positions of the landmarks and "+" denotes the estimated results. If the robot found that a collision with the environment border could happen, it would turn back with 180° to avoid collisions. By the time the robot completed active exploration, 26 landmarks had been allocated. Then one landmark

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Errors of robot position and landmark positions.



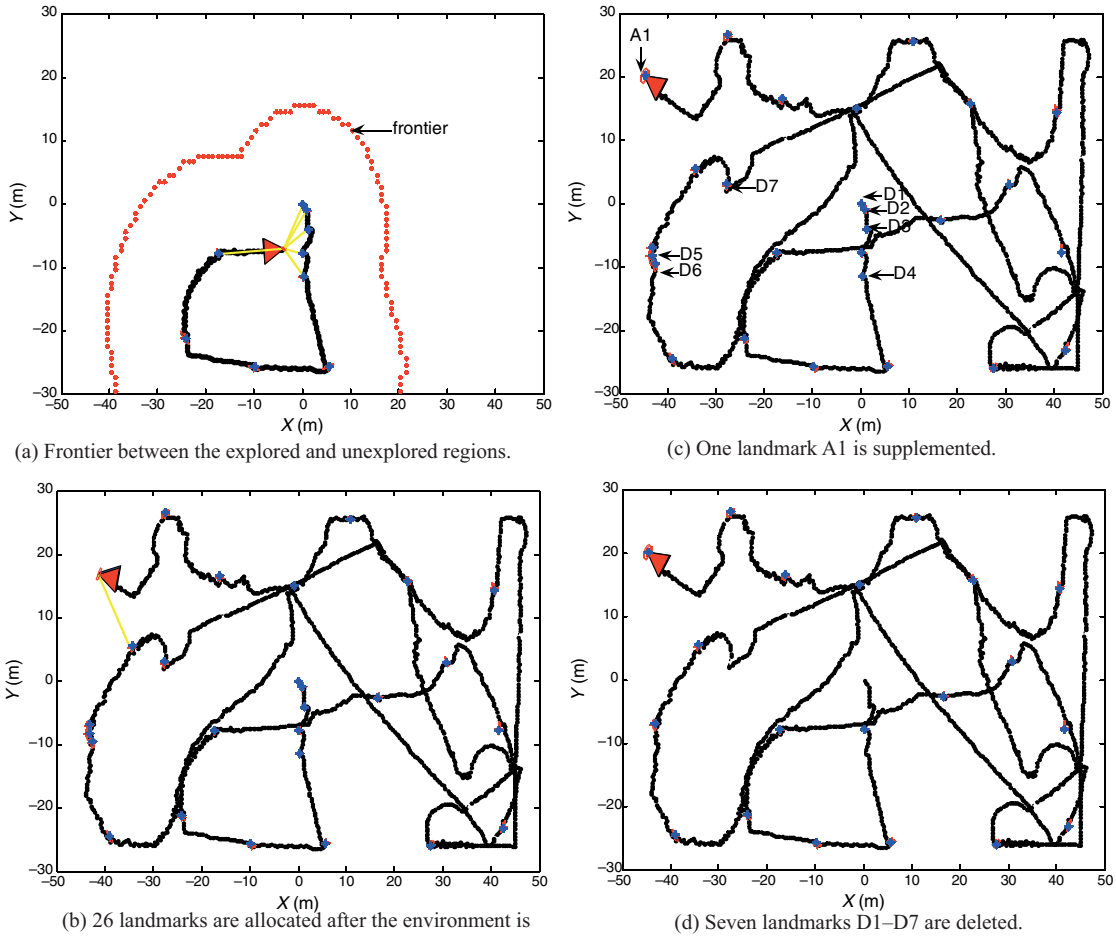
(b) The number of supplemented and deleted landmarks.

Fig. 12. Statistical result of 20 simulations.

A1 was supplemented and seven landmarks D1–D7 were removed. Finally, the whole environment was covered by 20 landmarks. Figure 14 gives another simulation result, in which the robot allocated 25 landmarks and no landmark was required to be supplemented, since the whole environment had been covered by these landmarks, and then six landmarks D1–D6 were deleted. Finally, the whole environment was covered by 19 landmarks. The above simulation results demonstrate that our approach to the active exploration is effective and stable.

During the simulations we found that the parameter ΔP_t had some impact on experimental results. Thus, we also implemented a set of simulations from $\Delta P_t = 0.1$ to 0.60 with an interval of 0.05. For each given ΔP_t, 20 trials were done and the mean error and uncertainty were calculated. The results are shown in Fig. 15. Furthermore, the average number of supplemented landmarks and their standard

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Frontier between the explored and unexplored regions.

(c) One landmark A1 is supplemented.

(b) 26 landmarks are allocated after the environment is covered by robot sensor.

(d) Seven landmarks D1–D7 are deleted.

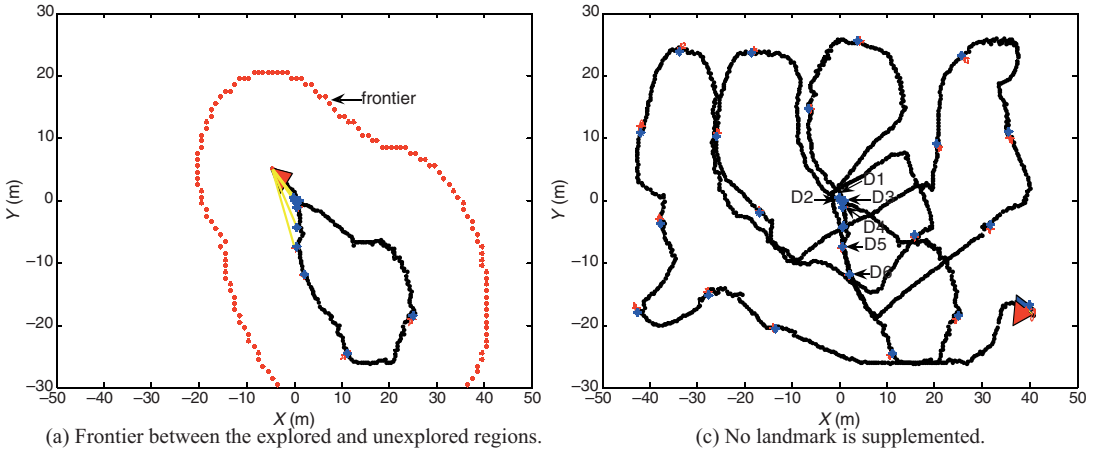
Fig. 13. (Colour online) One simulation result of active exploration. The robot allocated 26 landmarks, supplemented one landmark and deleted seven landmarks.

deviations, together with the average number of deleted landmarks and their standard deviations are shown in Fig. 16 respectively.

It can be seen from Figs. 15 and 16 that the number of landmarks allocated during the active exploration process reduced as ΔP_t increased, so did the accuracy of localization and mapping, and the robot needed to supplement more landmarks. If we chose ΔP_t between 0.20 and 0.25, the robot allocated an appropriate number of landmarks, and a sufficiently accurate SLAM could also be achieved. In some environments, deletion of redundant landmarks is inaccessible since the extra processing time is required for this operation. In such cases, we can increase ΔP_t accordingly to tighten the condition for landmark allocation. As a result, the robot allocates few redundant landmarks. Besides, deletion of a small quantity of redundant landmarks is not very necessary, because they can also help the robot to improve the accuracy of localization and mapping.

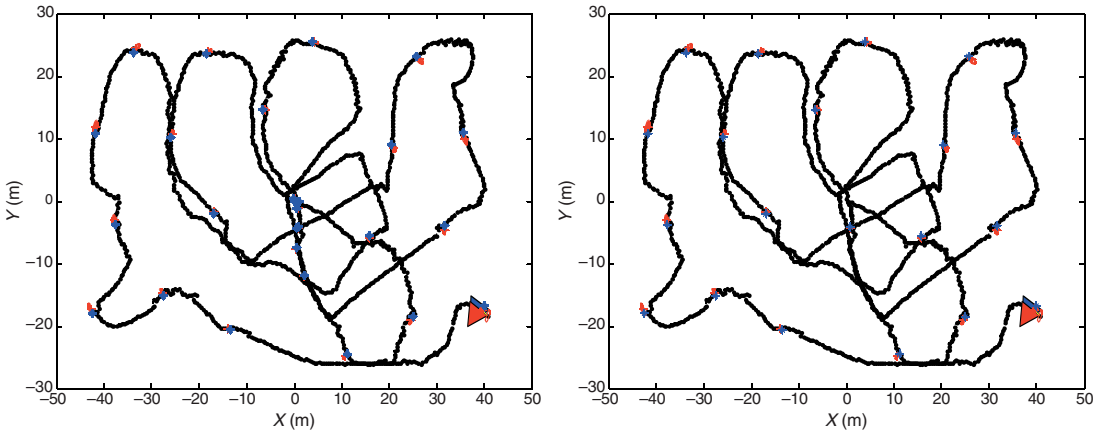
Compared with the experiment and simulation results in Section 6.1, it is shown that the robot deleted more and supplemented fewer landmarks in a large-scale environment, since in such an environment, the robot had to travel a much longer distance to cover the environment with its sensor. Especially, the robot might travel long distance from one boundary region of the environment to the opposite one in order to detect all the boundary regions. As a result, lot of local regions could be traversed for multiple times and multiple landmarks could be allocated at each of these regions. On the contrary, the robot could cover a small-scale environment with fewer scans and shorter distance. Thus, most local regions could only be traversed once and fewer landmarks were allocated such that somewhat more landmarks had to be added.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press



(a) Frontier between the explored and unexplored regions.

(c) No landmark is supplemented.



(b) 25 landmarks are allocated after the environment is covered by robot sensor.

(d) Six landmarks D1–D6 are deleted.

Fig. 14. (Colour online) Another simulation result of active exploration. The robot allocated 25 and deleted six landmarks.

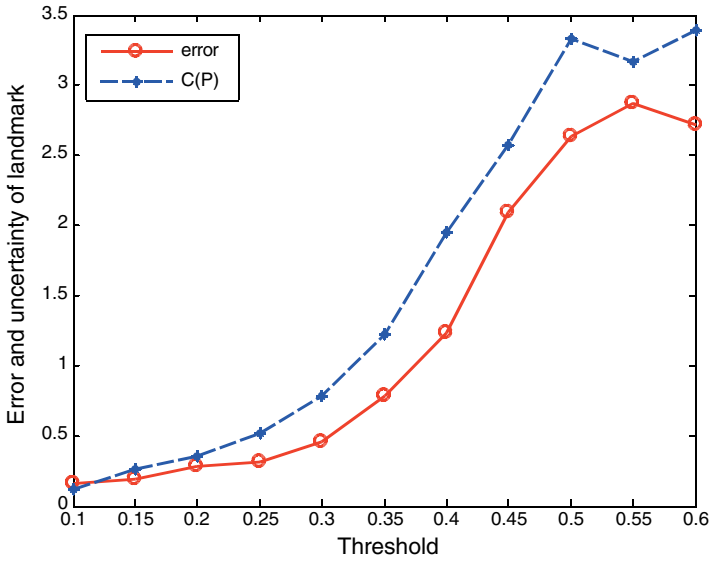


Fig. 15. (Colour online) Error and uncertainty affected by the threshold ΔP_t.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

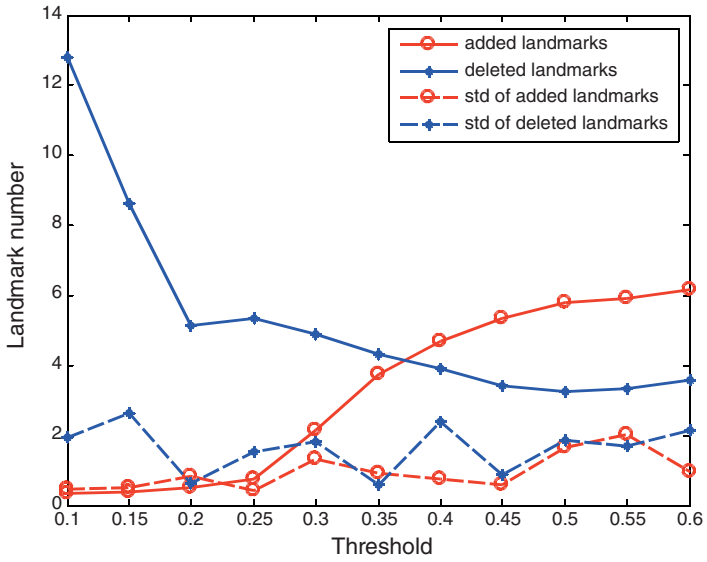


Fig. 16. (Colour online) Number of landmarks affected by the threshold ΔP_t.

7. Conclusions

In this paper, the active exploration and SLAM have been solved based on a novel approach to autonomous allocation of landmarks in unknown environments without sufficient features and artificial landmarks. Multiple objectives, including the accuracy of SLAM, the coverage rate and the predictive utility of the allocated landmarks, were integrated to guide the robot to allocate landmarks at appropriate positions. Simultaneously, the robot estimated the positions of the allocated landmarks and used them for self-localization. Then the completeness and non-redundancy of the landmark allocation were achieved by adding and removing some landmarks at right positions.

Using the proposed scheme, the robot can accomplish landmark allocation, localization and mapping simultaneously such that it can explore the environments in the absence of sufficient landmarks/features. Thus, our scheme can promote the classical SLAM to more general exploration for unknown environments. Furthermore, the proposed approach can also be extended to deal with the SLAM using other sensors, such as the vision sensors and the RFID.

In the future research we will investigate the possibility of extending this method to explore unknown environment with obstacles. In such a case, the robot will allocate landmarks and localize itself based on the landmark positions, simultaneously building the map for the obstacles.

Acknowledgements

This paper was sponsored in part by Natural Science Foundation of China (grant nos.: 60805031, 61175085, 61175083), Natural Science Foundation of Tianjin (grant no.: 10JCYBJC07600).

References

1. R. Smith, M. Self and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotics," **In**: *Autonomous Robot Vehicle* (I. J. Cox and G. T. Wilfon, eds.), (Springer-Verleg, New York, NY, 1990) pp. 167–193.
2. G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. Robot. Autom.* **17**(3), 229–241 (2001).
3. A. Chatterjee, "Differential evolution tuned fuzzy supervisor adapted extended Kalman filtering for SLAM problems in mobile robots," *Robotica* **27**(3), 411–423 (2009).
4. F. A. Cheein, F. di Sciascio, G. Scaglia and R. Carelli, "Towards features updating selection based on the covariance matrix of the SLAM system state," *Robotica* **29**(2), 271–282 (2011).
5. J. Andrade-Cetto, T. Vidal-Calleja and A. Sanfeliu, "Unscented Transformation of Vehicle States in SLAM," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain (2005) pp. 324–329.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

6. J. Knight, A. Davison and I. Reid, "Toward Constant Time SLAM Using Postponement," **In**: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, HI, USA (2001) pp. 406–412.
7. J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE Trans. Robot. Autom.* **17**(3), 242–257 (2001).
8. S. Thrun, Y. Liu, D. Koller, A. Y. Ng, Z. Ghahramani and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *Int. J. Robot. Res.* **23**(7), 693–716 (2004).
9. A. Martinelli, N. Tomatics and R. Siegwart, "Open Challenges in SLAM: An Optimal Solution Based on Shift and Rotation Invariants," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, New Orleans, LA, USA (2004) pp. 1327–1332.
10. L. M. Paz, J. D. Tardos and J. Neira, "Divide and conquer: EKF SLAM in O(n)," *IEEE Trans. Robot.* **24**(5), 1107–1120 (2008).
11. M. Montemerlo, S. Thrun and D. Koller, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," *Proceedings of the AAAI National Conference on Artificial Intelligence*, Edmonton, Canada (2002).
12. 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," *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, Acapulco, Mexico (2003).
13. M. Montemerlo and S. Thrun, "Simultaneous Localization and Mapping with Unknown Data Association Using FastSLAM," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, Taipei, Taiwan (2003) pp. 1985–1991.
14. C. Kim, R. Sakthivel and W. K. Chung, "Unscented FastSLAM: a robust and efficient solution to the SLAM problem," *IEEE Trans. Robot.* **24**(4), 808–820 (2008).
15. C. Kim, H. Kim and W. K. Chung, "Exactly Rao–Blackwellized Unscented Particle Filters for SLAM," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China (2010) pp. 3589–3594.
16. H. C. Lee, S. H. Lee, M. H. Choi and B.H. Lee, "Probabilistic map merging for multi-robot RBPF-SLAM with unknown initial poses," *Robotica* **30**(2), 205–220 (2012).
17. S. Thrun and M. Montemerlo, "The GraphSLAM algorithm with applications to large-scale mapping of urban structures," *Int. J. Robot. Res.* **25**(5/6), 403–430 (2005).
18. J. Levinson and S. Thrun, "Robust Vehicle Localization in Urban Environments Using Probabilistic Maps," **In**: *Proceedings of IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA (2010) pp. 4372–4378.
19. H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization," *IEEE Trans. Robot. Autom.* **17**(2), 125–137 (2001).
20. S. Tully, G. Kantor, H. Choset and F. Werner, "A Multi-Hypothesis Topological SLAM Approach for Loop Closing on Edge-Ordered Graphs," **In**: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, USA (2009) pp. 4943–4948.
21. T. Tao, S. Tully, G. Kantor and H. Choset, "Incremental Construction of the Saturated-GVG for Multi-Hypothesis Topological SLAM," **In**: *Proceedings of IEEE International Conference on Robotics and Automation*, Shanghai, China (2011) pp. 3072–3077.
22. S. Thrun, W. Burgard and D. Fox, "A Real-Time Algorithm for Mobile Robot Mapping with Applications to Multi-Robot and 3D Mapping," **In**: *Proceedings of IEEE International Conference on Robotics and Automation*, San Francisco, CA, USA (2000) pp. 321–328.
23. H. J. Chang, C. S. George Lee, Y. H. Lu and Y. Charlie Hu, "P-SLAM: Simultaneous localization and mapping with environmental-structure prediction," *IEEE Trans. Robot.* **23**(2), 281–293 (2007).
24. A. Kleiner, J. Prediger and B. Nebel, "RFID Technology-Based Exploration and SLAM for Search and Rescue," **In**: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China (2006) pp. 4054–4059.
25. H. Feder, J. Leonard and C. Smith, "Adaptive mobile robot navigation and mapping," *Int. J. Robot. Res.* **18**(7), 650–668 (1999).
26. F. Bourgault, A. Makarenko and S. Williams, "Information-Based Adaptive Robotic Exploration," **In**: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland (2002) pp. 540–545.
27. S. Huang, Z. Wang and G. Dissanayake, "Time Optimal Robot Motion Control in Simultaneous Localization and Map Building (SLAM) Problem," **In**: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan (2004) pp. 3110–3115.
28. S. Huang, N. M. Kwok, G. Dissanayake, Q. P. Ha and G. Fang, "Multi-Step Look-Ahead Trajectory Planning in SLAM: Possibility and Necessity," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain (2005) pp. 1103–1108.
29. N. Fairfield and D. Wettergreen, "Active SLAM and Loop Prediction with the Segmented Map Using Simplified Models," **In**: *Proceedings of the 7th International Conference on Field and Service Robotics*, Cambridge, MA, USA (2009) pp. 173–182.
30. C. Leung, S. Huang and G. Dissanayake, "Active SLAM in Structured Environments," *Proceedings of the IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA (2008) pp. 1898–1903.

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press

31. J. Yuan, Y. Huang, F. Sun and T. Tao, "Active Exploration Using Scheme of Autonomous Distribution for Landmarks," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, Kobe, Japan (2009) pp. 4169–4174.
32. S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics* (MIT Press, Cambridge, MA, 2005).
33. H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki and S. Thrun, *Principles of Robotic Motion: Theory, Algorithms and Implementation* (MIT Press, Cambridge, MA, 2004).
34. B. Yamauchi, "A Frontier-Based Approach for Autonomous Exploration," **In**: *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Monterey, CA, USA (1997) pp. 146–151.
35. J. Jiang, L. Fang, H. Zhang and W. Dou, "An algorithm for minimal connected cover set problem in wireless sensor networks," *J. Softw.* 17(2), 175–184 (2006).
36. D. Hochbaum, *Approximation Algorithms for NP-Hard Problem* (PWS, Cambridge, MA, 1995).

https://doi.org/10.1017/S0263574713001033 Published online by Cambridge University Press