Adaptive Mobile Robot Navigation and Mapping

Hans Jacob S. Feder
John J. Leonard

Marine Robotics Laboratory
Department of Ocean Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139, USA
{feder,jleonard}@deslab.mit.edu

Christopher M. Smith

Charles Stark Draper Laboratory
Cambridge, Massachusetts 02139, USA
csmith@draper.com

Abstract

The task of building a map of an unknown environment and concurrently using that map to navigate is a central problem in mobile robotics research. This paper addresses the problem of how to perform concurrent mapping and localization (CML) adaptively using sonar. Stochastic mapping is a feature-based approach to CML that generalizes the extended Kalman filter to incorporate vehicle localization and environmental mapping. The authors describe an implementation of stochastic mapping that uses a delayed nearest neighbor data association strategy to initialize new features into the map, match measurements to map features, and delete out-of-date features. The authors introduce a metric for adaptive sensing that is defined in terms of Fisher information and represents the sum of the areas of the error ellipses of the vehicle and feature estimates in the map. Predicted sensor readings and expected dead-reckoning errors are used to estimate the metric for each potential action of the robot, and the action that yields the lowest cost (i.e., the maximum information) is selected. This technique is demonstrated via simulations, in-air sonar experiments, and underwater sonar experiments. Results are shown for (1) adaptive control of motion and (2) adaptive control of motion and scanning. The vehicle tends to explore selectively different objects in the environment. The performance of this adaptive algorithm is shown to be superior to straight-line motion and random motion.

Nomenclature

$\mathcal{F}$	dynamic model
$\mathcal{H}$	observation model
$\mathcal{M}$	transformation relating the Fisher information between time steps recursively
$\mathbf{F}$	Jacobian of the dynamic model
$\mathbf{G}$	process noise scaling matrix (dependent on $\mathbf{u}$)
$\mathbf{H}$	Jacobian of the state-to-observation transformation $\mathbf{h}$
$\mathbf{I}$	Fisher information matrix
$\mathbf{K}$	Kalman filter gain matrix
$\mathbf{L}$	Jacobian of $\mathbf{l}$ with respect to $\mathbf{x}_r$, evaluated at $\mathbf{x}_{r_{k\|k}}$
N	number or element in the state vector; number of features in the map
$\mathbf{P}$	system covariance matrix
$\mathbf{R}$	observation covariance matrix
$\mathbf{T}$	transformation used in the compounding operation
$\mathbf{d_x}$	process noise (assumed to be white and Gaussian)
$\mathbf{d_z}$	observation noise process (assumed to be white and Gaussian)
$\mathbf{h}$	state-to-observation transformation
i	subscript index identifying a feature number
k	time index is shown in subscript
$\mathbf{l}$	transformation used for entering a new element to the state vector
$p(\cdot)$	a probability density function
$\mathbf{u}$	actions, or system control input
U_θ	set of sonar scanning angles
$\mathbf{x}$	actual system state vector
$\hat{\mathbf{x}}$	estimated system state vector
$\hat{\mathbf{x}}_{k\|k}$	estimated state at time step k given all the information up to time step k
$\mathbf{x}_r$	Vehicle state—location (x_r, y_r) and direction ϕ; that is, $\mathbf{x}_r = [x_r \ y_r \ \phi]$
$\mathbf{z}$	measurement vector (range and bearing measurements)
$\hat{}$	signifies an estimated variable

The International Journal of Robotics Research
Vol. 18, No. 7, July 1999, pp. 650-668,

1. Introduction

The goal of concurrent mapping and localization (CML) is to build a map of the environment while simultaneously using that map to enhance navigation performance. CML is critical for many field and service robotics applications. The long-term goal of our research is to develop new methods for CML, enabling autonomous underwater vehicles (AUVs) to navigate in unstructured environments without relying on a priori maps or acoustic beacons. This paper presents a technique for adaptive CML and demonstrates that technique using real sonar data.

Navigation is essential for successful operation of underwater vehicles in a range of scientific, commercial, and military applications (Leonard et al., 1998). Accurate positioning information is vital for the safety of the vehicle and for the utility of the data it collects. The error growth rates of inertial and dead-reckoning systems available for AUVs are usually unacceptable. Underwater vehicles cannot use the global positioning system (GPS) to reset dead-reckoning error growth unless they surface, which is usually undesirable. Positioning systems that use acoustic transponders (Milne, 1983) are often expensive and impractical to deploy and calibrate. Navigation algorithms based on existing maps have been proposed (Tuohy et al., 1996; Lucido et al., 1996), but sufficiently accurate a priori maps are often unobtainable.

Adaptive sensing strategies have the potential to save time and maximize the efficiency of the CML process for an AUV. Energy efficiency is one of the most challenging issues in the design of underwater vehicle systems (Bellingham and Willcox, 1996). Techniques for building a map of sufficient resolution as quickly as possible would be highly beneficial. Survey class AUVs must maintain forward motion for controllability (Bellingham et al., 1994); hence, the ability to adaptively choose a sensing and motion strategy that obtains the most information about the environment is especially important.

Sonar is the principle sensor for AUV navigation. Possible sonar systems include mechanically and electronically scanned sonar, side-scan sonar, and multibeam bathymetric mapping sonar (Urick, 1983). The rate of information obtained from a mechanically scanned sonar is low, making adaptive strategies especially beneficial. Electronically scanned sonar can provide information at very high data rates, but enormous processing loads make real-time implementations difficult. Adaptive techniques can be used to limit sensing to selected regions of interest, dramatically reducing computational requirements.

In this paper, adaptive sensing is formulated as the evaluation of different actions that the robot can take and the selection of the action that maximizes the amount of information acquired. This general problem has been considered in a variety of contexts (Manyika and Durrant-Whyte, 1994; Russell and Norvig, 1995), but not specifically for CML. CML provides an interesting context within which to address adaptive sensing because of the trade-off between dead-reckoning error and sensor data acquisition. The information gained by observing an environmental feature from multiple vantage points must counteract the rise in uncertainty that results from the motion of the vehicle.

Our experiments use two different robot systems. One is an inexpensive wheeled land robot equipped with a single rotating sonar. Observations are made of several cylindrical targets whose location is initially unknown to the robot. Although this is a simplified scenario, these experiments provide a useful illustration of the adaptive CML process and confirm behavior seen in simulation. The second system is a planar robotic positioning system that moves a sonar within a $9 \times 3 \times 1$ m testing tank. The sonar is a mechanically scanned 675-kHz Imagenex model 881 sonar with a 2° beam. These characteristics are similar to alternative models used in the marine industry. Testing tank experimentation provides a bridge between simulation and field AUV systems. Repeatable experiments can be performed under identical conditions; ground truth can be determined to high accuracy.

Section 2 reviews previous research in concurrent mapping and localization and adaptive sensing. Section 3 develops the theory of adaptive stochastic mapping. Sections 4, 5, and 6 describe testing of the method using simulations, air sonar experiments, and underwater sonar experiments, respectively. Finally, Section 7 provides concluding remarks and suggestions for future research.

2. Background

2.1. Stochastic Mapping

Stochastic mapping is a technique for CML that was introduced by Smith, Self, and Cheeseman (1990). In stochastic mapping, a single state vector represents estimates of the locations of the robot and the features in the environment. An associated error covariance matrix represents the uncertainty in these estimates, including the correlations between the vehicle and feature state estimates. As the vehicle moves through its environment taking measurements, the system state vector and covariance matrix are updated using an extended Kalman filter (EKF). As new features are observed, new states are added to the system state vector; the size of the system covariance matrix grows quadratically. Implementing stochastic mapping with real data requires methods for data association and track initiation. Smith, Self, and Cheeseman describe processes for adding new features to the map and enforcing known geometric constraints between different map features but do not perform simulations or experiments using these techniques.

Moutarlier and Chatila (1989) provided the first implementation of this type of algorithm with real data. Their implementation uses data from a scanning laser range finder mounted on a wheeled mobile robot operating indoors. They

implemented a modified updating technique, called relocation fusion, which reduces the effect of vehicle dynamic model uncertainty. More recently, Chatila and colleagues developed improved implementations for outdoor vehicles in natural environments and investigated new topics such as the decoupling of local and global maps (Betgé-Brezetz et al., 1996; Hébert, Betgé-Brezetz, and Chatila, 1996). The number of other researchers who have implemented stochastic mapping with real data has been limited. Using sonar on land robots operating indoors, Rencken (1993) and Chong and Kleeman (1997) implemented variations of stochastic mapping using sonar sensing. Rencken used the standard Polaroid ranging system, whereas Chong and Kleeman used a precision custom sonar array that provided accurate classification of geometric primitives. Current implementations of stochastic mapping are limited to ranges of tens of meters and/or durations of a few hours.

Stochastic mapping assumes a metrical, feature-based environmental representation in which objects can be effectively represented as points in an appropriate parameter space. Other types of representations are possible and have been employed with success. For example, Thrun, Fox, and Burgard (1998) have demonstrated highly successful navigation of indoor mobile robots using a combination of grid-based (Elfes, 1987) and topological (Kuipers and Byun, 1991) modeling. For marine robotic systems, the complexity of representing three-dimensional natural environments with geometric models is formidable. Our hypothesis, however, is that salient features can be found and reliably extracted (Medeiros and Carpenter, 1996), enabling CML. Exhaustively detailed environmental modeling should not be necessary for autonomous navigation.

2.2. Adaptive Sonar Sensing

Adaptive sensing has been a popular research topic in many different areas of robotics. Synonymous terms that have been used for these investigations include active perception (Bajcsy, 1988), active vision (Blake and Yuille, 1992), directed sensing (Leonard and Durrant-Whyte, 1992), active information gathering (Hager, 1990), adaptive sampling (Bellingham and Willcox, 1996), sensor management (Manyika and Durrant-Whyte, 1994), and limited rationality (Russell and Wefald, 1995). A common theme that emerges is that adaptive sensing should be formulated as the process of evaluating different sensing actions that the robot can take and choosing the action that maximizes the information acquired by the robot. The challenge in implementing this concept in practice is to develop a methodology for quantifying the expected information for different sensing actions and evaluate them in a computationally feasible manner given limited a priori information.

Our approach is closest to Manyika and Durrant-Whyte (1994), who formulated a normative approach to multisensor data fusion management based on Bayesian decision theory (Berger, 1985). A utility structure for different sensing actions was defined using entropy (Shannon information) as a metric for maximizing the information in decentralized multisensor data fusion. The method was implemented for model-based localization of a mobile robot operating indoors using multiple-scanning sonar and an a priori map. Feature location uncertainty and the loss of information due to vehicle motion error, which are encountered in CML, were not explicitly addressed.

Examples of the application of adaptive sensing in marine robotics include Singh (1995) and Bellingham and Willcox (1996). Singh formulated an entropic measure of adaptive sensing and implemented it on the Autonomous Benthic Explorer. The implementation was performed using stochastic back-projection, a grid-based modeling technique developed by Stewart (1988) for marine sensor data fusion. Bellingham and Willcox have investigated optimal survey design for AUVs making observations of dynamic oceanographic phenomena (Willcox et al., 1996). Decreased vehicle speeds save power due to the quadratic dependence of drag on velocity, but susceptibility to space-time aliasing is increased.

An interesting motivation for adaptive sonar sensing is provided by the behavior of bats and dolphins performing echolocation. For example, dolphins are observed to move their heads from side to side as they discriminate objects (Au, 1993). Our hypothesis for this behavior is that sonar is more like touch than vision. A useful analogy may be the manner in which a person navigates through an unknown room in the dark. By reaching out for and establishing contact with walls, tables, and chairs, transitions from one object to another can be managed as one moves across the room. Whereas man-made sonar tends to use narrow-band waveforms and narrow beam patterns, bat and dolphin sonar systems use broad-band waveforms and relatively wide beam patterns. A broad beam pattern can be beneficial because it provides a greater range of angles over which a sonar can establish and maintain "contact" by receiving echoes from an environmental feature. The task for CML is to integrate the information obtained from sonar returns obtained from different features as the sensor moves through the environment to estimate both the geometry of the scene and the trajectory of the sensor.

3. Adaptive Delayed Nearest Neighbor Stochastic Mapping

This section reviews the theory of stochastic mapping, derives a metric for performing stochastic mapping adaptively, and describes the delayed nearest neighbor (DNN) method for track initiation and track deletion.

3.1. Stochastic Mapping

Stochastic mapping is simply a special way of organizing the states in an EKF for the purpose of feature relative navigation.

An EKF is a computational efficient estimator for the states of a given nonlinear dynamic system, assuming that the noise processes are well modeled by Gaussian noise and the errors due to linearization of the nonlinear system are small. That is, the EKF for a system provides an estimate of both the state of the system, say $\hat{\mathbf{x}}$, and an estimate of the covariance of the state, say $\mathbf{P}$. The covariance of the state provides an estimate of the confidence in the estimate $\hat{\mathbf{x}}$. A dynamic system is described by a dynamic model, $\mathcal{F}$, which defines the evolution of the system (robot) through time, and an observation model, $\mathcal{H}$, which relates observations (measurements) to the state of the system (robot).

Our implementation of stochastic mapping is based on Smith, Self, and Cheeseman (1990) and is described in more detail in the appendix. We use $\mathbf{x}_k = \hat{\mathbf{x}}_{k|k} + \tilde{\mathbf{x}}_k$ to represent the system state vector $\mathbf{x} = [\mathbf{x}_r^T \ \mathbf{x}_1^T \dots \mathbf{x}_N^T]^T$, where $\mathbf{x}_r$ and $\mathbf{x}_1 \dots \mathbf{x}_N$ are the robot and feature states, respectively, $\hat{\mathbf{x}}$ is the estimated state vector, and $\tilde{\mathbf{x}}$ is the error estimate. Measurements are taken every $t = kT$ seconds, where T is a constant period and k is a discrete time index. The measurements are related to the state by a state-to-observation transform $\mathbf{z}_k = \mathcal{H}(\mathbf{x}_k, \mathbf{d}_\mathbf{z})$, where $\mathbf{d}_\mathbf{z}$ is the measurement noise process. The a posteriori PDF of $\mathbf{x}_k$, given a set of measurements $\mathbf{Z}_k \equiv \{\mathbf{z}_k, \dots, \mathbf{z}_1\}$, can be found from Bayes's rule as

$$p(\mathbf{x}_k|\mathbf{Z}_k) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)\,p(\mathbf{x}_k|\mathbf{Z}_{k-1})}{p(\mathbf{z}_k|\mathbf{Z}_{k-1})}. \tag{1}$$

The distribution $p(\mathbf{z}_k|\mathbf{x}_k)$ is defined as the likelihood function using the likelihood principle (Berger, 1985). By knowing $p(\mathbf{x}_k|\mathbf{Z}_k)$, we can form an estimate $\hat{\mathbf{x}}_k$ of the state.

To perform CML, the state transition (dynamic model) $\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_\mathbf{x})$, in addition to the observation transformation $\mathcal{H}$, must be known. Here, $\mathbf{u}_k$ is the control input (action) at time k and $\mathbf{d}_\mathbf{x}$ is the process noise. If the stochastic processes $\mathbf{d}_\mathbf{z}$ and $\mathbf{d}_\mathbf{x}$ are assumed to be independent, white, and Gaussian, and the state transition $\mathcal{F}$ and observation model $\mathcal{H}$ are both linear, the Bayes least squares (BLS) estimator $\hat{\mathbf{x}}_{k+1|k} = \mathcal{F}(E\{\mathbf{x}_k|\mathbf{Z}_k\}, \mathbf{u})$ will be an efficient estimator of $\mathbf{x}$. However, in the general problem of CML, neither the dynamic model nor the observation model will be linear. Thus, an efficient estimator cannot be obtained. Furthermore, propagating the system's covariance through nonlinear equations does not guarantee that the statistics will be conserved. Thus, to circumvent the problem of transformation of nonlinearities, the nonlinear models of $\mathcal{F}$ and $\mathcal{H}$ are approximated through a Taylor series expansion, keeping only the first two terms. That is,

$$\begin{aligned} \mathcal{F}(\mathbf{x}_k, u_k, \mathbf{d}_\mathbf{x}) &\approx \mathcal{F}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, \mathbf{d}_\mathbf{x}) + \mathbf{F}_x\tilde{\mathbf{x}}_{k|k} \\ \Rightarrow \hat{\mathbf{x}}_{k+1|k} &\approx E\{\mathcal{F}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_\mathbf{x})|\mathbf{Z}_k\}, \end{aligned} \tag{2}$$

where $\mathbf{F}_\mathbf{x} = d\mathcal{F}(\mathbf{x}, \mathbf{u}_k)/d\mathbf{x}|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k}}$ is the Jacobian of dynamic model $\mathcal{F}$ with respect to $\mathbf{x}$, evaluated at $\hat{\mathbf{x}}_{k|k}$. Similarly, the observation model is approximated by

$$\mathcal{H}(\mathbf{x}_k, \mathbf{d}_\mathbf{z}) \approx \mathcal{H}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{d}_\mathbf{z}) + \mathbf{H}_x\tilde{\mathbf{x}}_{k|k-1}, \tag{3}$$

where $\mathbf{H}_\mathbf{x} = d\mathcal{H}(\mathbf{x})/d\mathbf{x}|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1}}$ is the Jacobian of the observation model $\mathcal{H}$ with respect to $\mathbf{x}$ evaluated at $\hat{\mathbf{x}}_{k|k-1}$. These approximations are equivalent to the assumption that the estimated mean at the previous time step, $\hat{\mathbf{x}}_{k|k}$, is approximately equal to the system state $\mathbf{x}_k$ at the previous time step. Once these linearizations have been performed and assuming that the approximation error is small, we can now find the BLS for this linearized system. The result is equivalent to that of the EKF (Gelb, 1973; Bar-Shalom and Li, 1993; Uhlmann, Julier, and Csorba, 1997).

The estimate error covariance $\mathbf{P}_{k|k} = E\{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k^T\}$ of the system takes the form

$$\mathbf{P}_{k|k} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{r1} & \cdots & \mathbf{P}_{rN} \\ \mathbf{P}_{1r} & \mathbf{P}_{11} & \cdots & \mathbf{P}_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nr} & \mathbf{P}_{N1} & \cdots & \mathbf{P}_{NN} \end{bmatrix}_{k|k}. \tag{4}$$

The submatrices $\mathbf{P}_{rr}$, $\mathbf{P}_{ri}$, and $\mathbf{P}_{ii}$ are the vehicle-vehicle, vehicle-feature, and feature-feature covariances, respectively. This form is significant, since it allows us to separate the uncertainty associated with the robot, $\mathbf{P}_{rr}$, as well as the individual features, $\mathbf{P}_{ii}$, and this separation will be used in obtaining a measure of the information in our system.

Thus, the robot and map are represented by a single state vector $\mathbf{x}$ with an associated estimate error covariance $\mathbf{P}$ at each time step. As new features are added, $\mathbf{x}$ and $\mathbf{P}$ increase in size.

In our implementation, we denote the vehicle's state by $\mathbf{x}_r = [x_r \ y_r \ \phi]^T$, and the control input to the vehicle is given by $\mathbf{u} = [u_x \ u_y \ u_\phi]$. For the dynamic model $\mathcal{F}$, we use

$$\begin{aligned} \mathbf{x}_{r_{k+1}} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{G}(\mathbf{u}_k)\mathbf{d}_\mathbf{x} \\ &= \mathbf{x}_{r_k} + \mathbf{T}_{\phi_k}\mathbf{u}_k + \mathbf{G}(\mathbf{u}_k)\mathbf{d}_\mathbf{x}, \end{aligned} \tag{5}$$

where $\mathbf{G}(\mathbf{u}_k)$ scales the noise process $\mathbf{d}_\mathbf{x}$ as a function of the distance traveled; that is,

$$\mathbf{G}(\mathbf{u}_k) = \begin{bmatrix} \gamma_x\sqrt{u_{x_k}^2 + u_{y_k}^2} & 0 & 0 \\ 0 & \gamma_y\sqrt{u_{x_k}^2 + u_{y_k}^2} & 0 \\ 0 & 0 & \gamma_\phi \end{bmatrix} \tag{6}$$

and

$$\mathbf{T}_{\phi_k} = \begin{bmatrix} \cos(\phi_k) & -\sin(\phi_k) & 0 \\ \sin(\phi_k) & \cos(\phi_k) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

The ϕ in this expression comes from the robot position $\mathbf{x}_{r_k}$. The covariance of $\mathbf{d}_\mathbf{x}$ is for convenience set equal to the identity matrix, since any scaling is placed in $\mathbf{G}$. The matrix $\mathbf{G}$ is diagonal by assumption. If the correlations between the

noise processes in the x, y, and ϕ directions are known, it should be included. However, this is a minor point, and this form has been shown to work well for our systems.

Equation (5) does not take into account any of the vehicle's real dynamics. Thus, the model is very general. However, if the vehicle's dynamics were known, they could be used to ensure that the vehicle movements remain realistic. In our experiments and simulation, we will constrain the robot to move only a certain distance each time step, thus making u_x and u_y dependent.

For the observation model, we use

$$\mathbf{z}_k = \mathcal{H}(\mathbf{x}_k, \mathbf{d}_\mathbf{z}) = \mathbf{h}(\mathbf{x}_k) + \mathbf{d}_\mathbf{z}, \tag{8}$$

where $\mathbf{z}_k$ is the observation vector of range and bearing measurements. The observation model $\mathbf{h}$ defines the nonlinear coordinate transformation from state to observation coordinates. The stochastic process $\mathbf{d}_\mathbf{z}$ is assumed to be white, Gaussian, and independent of $\mathbf{x}_0$ and $\mathbf{d}_\mathbf{x}$, and has covariance $\mathbf{R}$.

In our experiments, a sector including an object is scanned over multiple adjacent angles to yield multiple sonar returns that originate from the object. That is, isolated features and smooth surfaces appear as circular arcs, also known as regions of constant depth (Leonard and Durrant-Whyte, 1992), in sonar scans. For the specular returns, an improved estimate of the range and bearing to the object is obtained by grouping sets of adjacent returns with nearly the same range, then taking the mode of this set of angles as the bearing measurement and the range associated with this bearing as the range measurement to the object (Moran, Leonard, and Chryssostomidis, 1997). Rough surfaces (Bozma and Kuc, 1991) yield additional returns at high angles of incidence. These occur frequently in the data from our underwater sonar, but are not processed in the experiments reported in this paper.

In our work, all features are modeled as point features. More complex objects can be described by a finite set of parameters and estimated by using the associated observation model for this parameterization in stochastic mapping. For instance, the modeling of planes, corners, cylinders, and edges is relatively straightforward (Leonard and Durrant-Whyte, 1992).

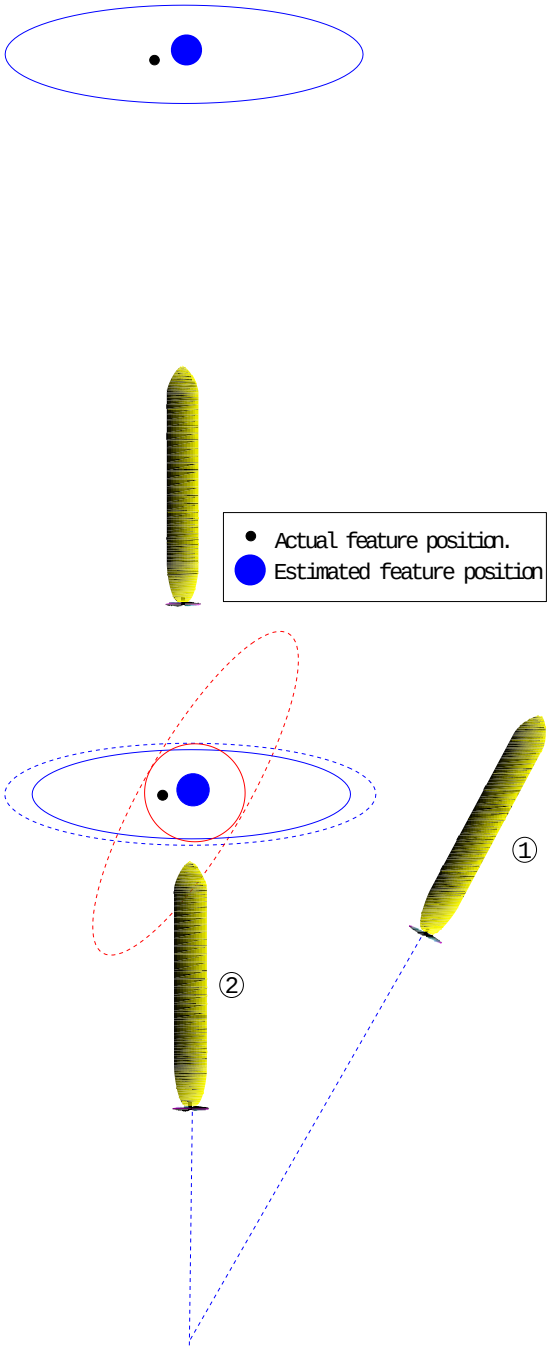


Fig. 1. An autonomous underwater vehicle with no navigational uncertainty estimating the position of an environmental feature. The ellipses denote the certainty (error ellipse) with which the feature position is known. (a) The initial estimate of the feature. (b) Given two possible new allows for a more accurate estimate of the feature's position, as the measurement of the feature is taken from a new angle, resulting in the tilted error ellipse associated with this measurement. Combining this (tilted error ellipse) with the error ellipse from (a), the small dotted circular error ellipse is achieved. Taking the next observation slightly smaller than that of the previous time step in (a), shown by the dotted ellipse.

3.2. Adaptation Step

The goal of adaptive CML is to determine the optimal action given the current knowledge of the environment, sensors, and robot dynamics in a CML framework. To provide an intuitive understanding of this goal, imagine an underwater vehicle with no navigational uncertainty estimating the position of a feature as depicted in Figure 1. As can be seen from this simple example, it is clearly advantageous for the vehicle to take the next measurement from a new direction. By doing so, more information about the feature is extracted, and thus a better estimate can be obtained.

The essence of our model is to determine the action that maximizes the total knowledge (i.e., the information) about the system in the presence of measurement and navigational uncertainty. By *adaptively* choosing actions, we mean that the next action of the robot is chosen so as to maximize the robot's information about its location and all the features' locations (the map).

The "amount" of information contained in eq. (1) can be quantified in various ways. Fisher information is of particular interest and is related to the estimate of the state $\mathbf{x}$ given the observations. The Fisher information for a random parameter is defined (Bar-Shalom and Fortmann, 1988) as the covariance of the gradient of the total log probability; that is,

$$\begin{aligned}\mathbf{I}_{k|k} &\equiv E\{(\nabla_{\mathbf{x}} \ln p(\mathbf{x}, \mathbf{Z}_k))(\nabla_{\mathbf{x}} \ln p(\mathbf{x}, \mathbf{Z}_k))^T\} \\ &= -E\{\nabla_{\mathbf{x}} \nabla_{\mathbf{x}}^T \ln p(\mathbf{x}_k, \mathbf{Z}_k)\},\end{aligned} \tag{9}$$

where $\nabla_{\mathbf{x}} = [\frac{d}{dx_1} \ \cdots \frac{d}{dx_N}]^T$ is the gradient operator with respect to $\mathbf{x} = [x_1 \ \cdots x_N]$; thus, $\nabla_{\mathbf{x}} \nabla_{\mathbf{x}}^T$ is the Hessian matrix.

Strictly speaking, the Fisher information is defined only in the non-Bayesian view of estimation (Bar-Shalom and Fortmann, 1988); that is, in the estimation of a nonrandom parameter. In the Bayesian approach to estimation, the parameter is random with a (prior) probability density function. However, as in the non-Bayesian definition of the Fisher information, the inverse of the Fisher information for random parameter estimation is the Cramér-Rao lower bound for the mean square error.

Applying Bayes's rule to eq. (9)—that is, $p(\mathbf{x}_k, \mathbf{Z}_k) = p(\mathbf{x}_k, \mathbf{z}, \mathbf{Z}_{k-1}) = p(\mathbf{x}_k, \mathbf{Z}_{k-1}) \cdot p(\mathbf{z}|\mathbf{x}_k)$—and noting the linearity of the gradient and expectation operators, we obtain a Fisher information update:

$$\begin{aligned}E\{\nabla_{\mathbf{x}} \nabla_{\mathbf{x}}^T \ln p(\mathbf{x}_k, \mathbf{Z}_k)\} &= E\{\nabla_{\mathbf{x}} \nabla_{\mathbf{x}}^T \ln p(\mathbf{x}_k, \mathbf{Z}_{k-1})\} \\ &\quad + E\left\{\nabla_{\mathbf{x}} \nabla_{\mathbf{x}}^T \ln p(\mathbf{z}_k|\mathbf{x}_k)\right\} \\ -\ \mathbf{I}_{k|k} &= \mathbf{I}_{k|k-1} - E\left\{\nabla_{\mathbf{x}} \nabla_{\mathbf{x}}^T \ln p(\mathbf{z}_k|\mathbf{x}_k)\right\}.\end{aligned} \tag{10}$$

The first term on the right represents the Fisher information, $\mathbf{I}_{k|k-1}$, before the last measurement, whereas the second term corresponds to the additional information gained by measurement $\mathbf{z}_k$.

If we obtain an efficient estimator for $\mathbf{x}$, the Fisher information will simply be given by the inverse of the error covariance of the state. Thus, under the assumption that eqs. (2) and (3) hold, the inverse of error covariance $\mathbf{P}$ will be an estimate for the Fisher information of the system; that is

$$\mathbf{I} \approx \mathbf{P}^{-1}.$$

Under this assumption, and using eq. (5), the transformation $\mathcal{M}$ relating $\mathbf{I}_{k|k}$ to $\mathbf{I}_{k+1|k}$ can be found. By combining this with eq. (10), we have a recursive Fisher information update that depends on the actions $\mathbf{u}$ (inputs). $\mathcal{M}$ will generally depend on the state $\mathbf{x}_{k+1}$ as well, which is not available. By invoking the assumption of eq. (2), $\mathbf{x}_{k+1}$ is replaced by the estimate $\hat{\mathbf{x}}_{k+1|k}$. Thus, $\mathcal{M}$ can be used to give us the optimal action $\mathbf{u}_k$ to take given our model and assumptions.

By optimal we mean that, under the assumption that the EKF is the best estimator for our state, the action $\mathbf{u}$ that maximizes the knowledge (information) of the system given the current knowledge of the system can be determined. This is not necessarily the optimal action for the actual system.

At each time step, the algorithm seeks to determine the transformation $\mathcal{M}$ and, from this, infer the optimal action $\mathbf{u}_k$. Combining the vehicle prediction and EKF update step of stochastic mapping, $\mathcal{M}$ is

$$\mathbf{I}_{k+1|k+1} = (\mathbf{F}_{\mathbf{x}} \mathbf{I}_{k|k}^{-1} \mathbf{F}_{\mathbf{x}}^T + \mathbf{G}(\mathbf{u}_k)\mathbf{G}(\mathbf{u}_k)^T)^{-1} + \mathbf{H}_{\mathbf{x}}^T \mathbf{R}^{-1} \mathbf{H}_{\mathbf{x}}, \tag{11}$$

where $\mathbf{F}_{\mathbf{x}}$ and $\mathbf{H}_{\mathbf{x}}$ are the Jacobian of $\mathbf{f}$ and $\mathbf{h}$ with respect to $\mathbf{x}$ evaluated at $\hat{\mathbf{x}}_{k|k}$ and $\hat{\mathbf{x}}_{k+1|k}$, respectively. The first term on the right of eq. (11) represents the previous information of the system, as well as the loss of information that occurs due to the action $\mathbf{u}_k$. The second term represents the additional information gained by the system due to observations after the action $\mathbf{u}_k$. Since this quantity is a function of $\mathbf{x}_{k+1}$, which is unknown, we approximate $\mathbf{x}_{k+1}$ by $\hat{\mathbf{x}}_{k+1|k}$ by the assumption of eq. (2). The action that maximizes the information can be expressed as

$$\mathbf{u}_k = \max_{\mathbf{u}} \mathbf{I}_{k+1|k+1} = \min_{\mathbf{u}} \mathbf{P}_{k+1|k+1}. \tag{12}$$

The information is a matrix, and we require a metric to quantify the information. Furthermore, it is desired that this metric have a simple physical interpretation.

For CML, it is desirable to use a metric that makes explicit the trade-off between uncertainty in feature locations and uncertainty in the vehicle position estimate. To accomplish this, we define the metric by a cost function $C(\mathbf{P})$, which gives the total area of all the error ellipses (i.e., highest probability density regions) and is thus a measure of our confidence in our map and robot position. That is,

$$\begin{aligned}C(\mathbf{P}) &= \pi \textstyle\prod_j \sqrt{\lambda_j(\mathbf{P}_{rr})} + \pi \sum_{i=1}^N \prod_j \sqrt{\lambda_j(\mathbf{P}_{ii})} \\ &= \pi \sqrt{\det(\mathbf{P}_{rr})} + \pi \textstyle\sum_{i=1}^N \sqrt{\det(\mathbf{P}_{ii})}\ ,\end{aligned} \tag{13}$$

where $\lambda_j(\cdot)$ is the jth eigenvalue of its argument and det is the determinant of its argument. Turning back to Figure 1b, eq. (13) minimizes the error ellipses, thus yielding a lower cost for position 1 than for position 2 as makes intuitive sense.

The action to take is obtained by evaluating eq. (12) over the action space of the robot using the metric in eq. (13). This yields an adaptive stochastic mapping algorithm. This procedure optimizes the information locally at each time step; thus, the adaptation step performs a local optimization. Note

that the action space of the robot is not limited to motion control inputs. Other actions and constraints can readily be included in the control input $\mathbf{u}$, such as the measurements that should be taken by the sonar. For this, we control the set of angles U_θ that are scanned by the sonar.

In principle, eq. (12) can be solved in symbolic form using Lagrange multipliers. However, the symbolic matrix inversion required in eq. (11) is very tedious and results in a very large number of terms. Furthermore, as more features are added, the inversion scales exponentially in the number of calculations. Numerical methods, however, can be used to evaluate eq. (12) quite efficiently. The experiments below do not use a numerical technique such as the simplex method to perform this optimization; however, this could be readily incorporated in the future.

3.3. Data Association, Track Initiation, and Track Deletion

Section 3.1 outlined the stochastic mapping approach to feature-based navigation. To employ stochastic mapping in a real-world scenario, we have to be able to extract features from the environment. Sonar is notorious for exhibiting dropouts, false returns, no returns, and noise. Thus, addressing the problem of data association is critical for the validity of the observation model (eq. (8)) and in employing a stochastic-mapping-based approach to CML. For this purpose, data association, initiation of new feature estimates (i.e., track initiation), and the removal of outdated feature estimates (track deletion) for the mobile robot are described here.

The problem of data association is that of determining the source of a measurement. Obtaining correct data association can pose a significant challenge when making observations of environmental features (Smith, 1998). In our implementation, data association is performed using a gated nearest neighbor technique (Bar-Shalom and Fortmann, 1988). The initiation of new feature tracks is performed using a DNN initiator. The DNN initiator is similar in spirit to the logic-based multiple target track initiator described by Bar-Shalom and Fortmann (1988). One important difference in our method is that the vehicle's position is uncertain, and this uncertainty has to be included when performing gating and finding the nearest neighbor. It is assumed that a sonar return originates from not more than one feature.

To include the vehicle's uncertainty in performing data association using a nearest neighbor gating technique, we need to transform the vehicle's uncertainty into measurement space and add this uncertainty to the measurement uncertainty. We assume that the true measurement of feature i at time k conditioned on $\mathbf{Z}_{k-1}$ is normally distributed in measurement space. Furthermore, it is assumed that the transformation from vehicle space to measurement space retains the Gaussianity of the estimated state. Under these assumptions, one may define the innovation matrix $\mathbf{S}_i$ for feature i as

$$\mathbf{S}_i = \mathbf{H}_{\mathbf{x}_i} \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{ri} \\ \mathbf{P}_{ir} & \mathbf{P}_{ii} \end{bmatrix}_{k|k} \mathbf{H}_{\mathbf{x}_i}^T + \mathbf{R},$$

with

$$\mathbf{H}_{\mathbf{x}_i} = \left. \frac{d\mathbf{h}_i([\mathbf{x}_r\ \mathbf{x}_i])}{d[\mathbf{x}_r\ \mathbf{x}_i]} \right|_{[\mathbf{x}_r\ \mathbf{x}_i]=[\hat{\mathbf{x}}_{r_{k+1|k}}\ \hat{\mathbf{x}}_{i_{k+1|k}}]}, \tag{14}$$

where $\mathbf{h}_i$ is the observation model for feature i. $\mathbf{H}_{\mathbf{x}_i}$ is the linearized transformation from vehicle space to measurement space. The nearest neighbor gating is performed in innovation space. That is, defining the innovation $\boldsymbol{\nu} = \mathbf{z} - \mathbf{z}_i$, the validation region, or gate, is given by

$$\boldsymbol{\nu}^T \mathbf{S}_i^{-1} \boldsymbol{\nu} \leq \gamma. \tag{15}$$

The value of the parameter γ is obtained from the χ^2 distribution. For a system with 2 DoF, a value of $\gamma = 9.0$ yields the region of minimum volume that contains the measurement with a probability of 98.9% (Bar-Shalom and Fortmann, 1988). This validation procedure defines where a measurement is expected to be found. If a measurement is outside this region, it is considered too unlikely to arise from feature i. If several measurements gate with the same feature i, the closest (i.e., most probable) one is chosen.

In performing track initiation, all measurements that have not been matched with any feature over the last N time steps are stored. That is, any measurement that was not matched to a known feature is a potential new feature. At each time step, a search for clusters of more than $M \leq N$ measurements over this set of unmatched measurements is performed. For each of these clusters, a new feature track is initiated. A cluster is defined as at most one measurement at each time step that gates according to eq. (15) with all other measurements in the cluster. For our systems, where the probability of false returns is relatively low and the probability of detection is relatively high, values of $M = 2$ or 3 and $N = M + 1$ are sufficient.

A track deletion capability is also incorporated to provide a limited capability to operate in dynamic environments. When a map feature is predicted to be visible but is not observed for several time steps in a row, it is removed from the map (Leonard, Cox, and Durrant-Whyte, 1992). This is motivated by assuming a probability of detection $P_D < 1$. Thus, if the feature has not been observed over the last r time steps during which an observation was expected of the feature, the probability of the feature being at the expected location is $(1 - P_D)^r$, assuming that the observations are independent. Thus, setting a threshold on r is equivalent to setting a threshold on the probability that the feature still exists at the predicted location.

With the incorporation of these data association methods, the complete adaptive DNN stochastic mapping algorithm is summarized as follows:

1. *State projection*. The system state (vehicle and features) is projected to the next time step using the state transition model $\mathcal{F}$ along with the control input $\mathbf{u}_k$.
2. *Gating*. The feature closest to each new measurement is determined and gated with the feature, and nonmatching measurements are stored.
3. *State update*. Reobserved features update the vehicle and feature tracks using the EKF.
4. *New feature generation*. New features are initialized using the DNN data association strategy.
5. *Old feature removal*. Out-of-date features are deleted.
6. *Adaptation step*. Determine the next action to take, $\mathbf{u}_k$, by optimizing eq. (13).

An outline of the algorithm for performing adaptive augmented stochastic mapping is shown in Figure 2.

4. Simulation Results

The algorithm described above has been extensively tested in simulation. In these simulations, as well as in the experiments to follow, a numerical approximation was performed by evaluating eq. (13) at a fixed number of points in the action space. The robot was constrained to move a distance of 0, 10, or 20 cm at each time step, and could only turn in increments of 22.5°. Furthermore, the vehicle was constrained to get no closer than 40 cm to the features (PVC tubes), since the sonar signal in this range becomes unreliable. In all these simulations, range error was assumed to have a standard deviation of 2 cm, whereas the standard deviation for the bearing was 10°. Furthermore, the sonar could move in increments of 0.9° between each measurement. Thus, a complete scan of 360° consisted of 400 sonar returns. The standard deviation for the vehicle odometry was set to 5% of the distance traveled. The angle uncertainty was set to 1°. These parameters were chosen because they resemble the situation for the air sonar

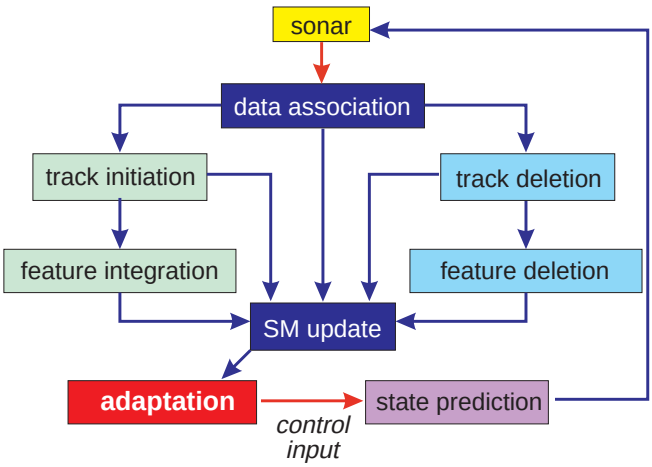


Fig. 2. Structure of the adaptive delayed nearest neighbor stochastic mapping algorithm.

experiments in Section 5. Two different types of simulation are described.

4.1. Adaptive Vehicle Motion

In these simulations, it was assumed that the robot stopped and took a complete 360° scan of the environment before continuing. The algorithm chose where to move adaptively. The algorithm took advantage of the fact that the measurements have different certainties in range and bearing, thus forming an error ellipse. Note, however, that if the observations of a target have nearly equal standard deviations in all directions, the robot will not move, since the loss of information from odometric error is larger than the information gained by taking a measurement from a different location.

Figure 3 shows the three typical paths of the robot in the presence of two features (8.4-cm radius PVC tubes) as a result of adaptation. The robot started at (0,0), and the paths (solid, dashed, and dashed-dotted lines) occurred with approximately equal frequency. The dotted lines around the PVC tubes denote the constraints placed on the robot for how close it could come to the PVC tubes while still obtaining valid sonar returns. (This is a limitation of the standard Polaroid sonar driver circuit.) The resulting 95% confidence ellipses are drawn for the middle path, for the estimated position of the center of the features, and for the robot's position. The robot's true position is indicated by a +, whereas the estimated position is shown by a $\times$. The robot stopped moving after about 15 time steps, since more information would be lost than gained by moving. Also note that the robot moved greater distances in the beginning before taking the next scan than it did at the end of the run; more information was gained by moving and taking a measurement from a different location in the beginning than near the end of a run.

Figure 4 shows the average, over 2000 runs, of the total cost (as defined by eq. (13)) of the system under adaptation as a percentage of the total cost of the system when moving randomly (solid line) or moving along the negative x-axis (dashed line) without adaptation. Moving along the negative x-axis is a worst case scenario; moving in a straight line from the initial position in a direction between the two features would practically be identical to the adaptive run, thus presenting an upper bound for straight-line motion. In all the runs without adaptation, the robot moved a distance of 10 cm at each time step. As can be seen, the adaptation procedure obtains a cost of about 60% of the nonadaptive strategies after about 8 time steps. The lower cost signifies that the adaptive strategy has obtained more information about the environment and, thus, has produced a more accurate estimate of the robot's, as well as the features', position. The random motion slowly started catching up after the 8th time step, and by the 50th time step, it had achieved a cost about 15% higher than that of the adaptive strategy. When moving along the negative x-axis, the cost actually starts increasing after about the 15th time step,

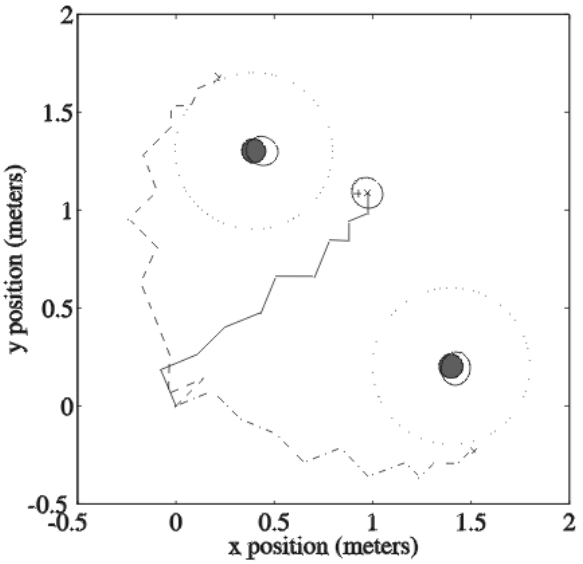


Fig. 3. Three typical paths taken by the robot in simulations of adaptive concurrent mapping and localization in the presence of two PVC tubes (filled circles) of radius 8.4 cm. Of 1000 simulated runs, the robot chose to go to the left, to the right, and through the middle with about equal frequency. The 95% confidence level of the map for the middle path is shown by the ellipses. The dotted circles define how close to the PVC tubes the robot is allowed to be (see text for symbols).

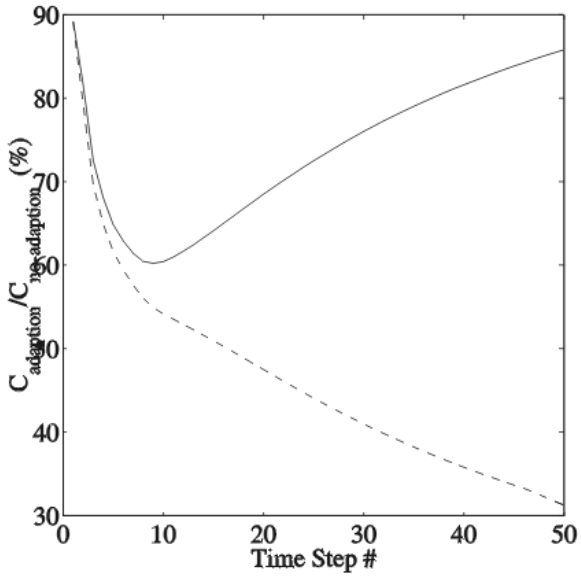


Fig. 4. Simulation result showing the cost function $C(\mathbf{P})$ when performing adaptation divided by the cost function when not performing adaptation. Two cases in which no adaptation was performed are shown: the solid line denotes the case in which the robot moved randomly, whereas the dashed line denotes the case in which the robot moved along the negative x-axis.

since the robot was so far away from the features that it lost more information by moving than it gained by sensing at each time step. This is due to the poor angular resolution of the sonar.

Figure 5 shows the sensor trajectory for a simulation involving eight objects. Figure 6 shows the east and north vehicle location errors and associated error bounds as a function of time during the simulation. After an initial loop around four of the objects is executed, the error bounds converge and the sensor wanders back and forth over a small area.

4.2. Adaptive Scanning and Motion

In using sonar for mapping an environment, one is limited by the relatively slow speed with which measurements can be acquired. Thus, we imposed the additional constraint that the sonar could scan only an angle of 15° (i.e., 50% more than the measurement bearing standard deviation) at each time step. The algorithm was required to decide where to direct the attention of the robot. The algorithm thereby adaptively decided where to look and where to move. This was implemented in the framework outlined above by adding an additional action u_θ to be controlled and solving eq. (13) given the constraints of the scan angle.

To compare the simulation results to the experiments and the previous simulations, the simulations were, as before, conducted over 50 time steps. However, under the adaptive strat-

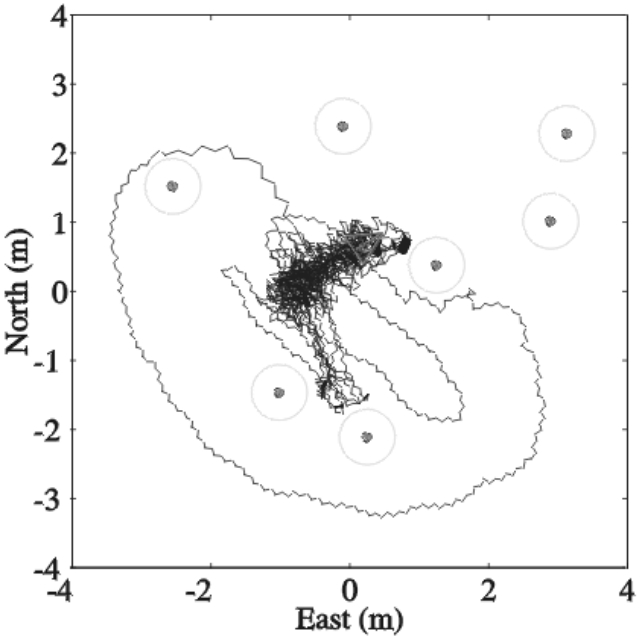


Fig. 5. Path of vehicle performing adaptive motion among multiple objects.

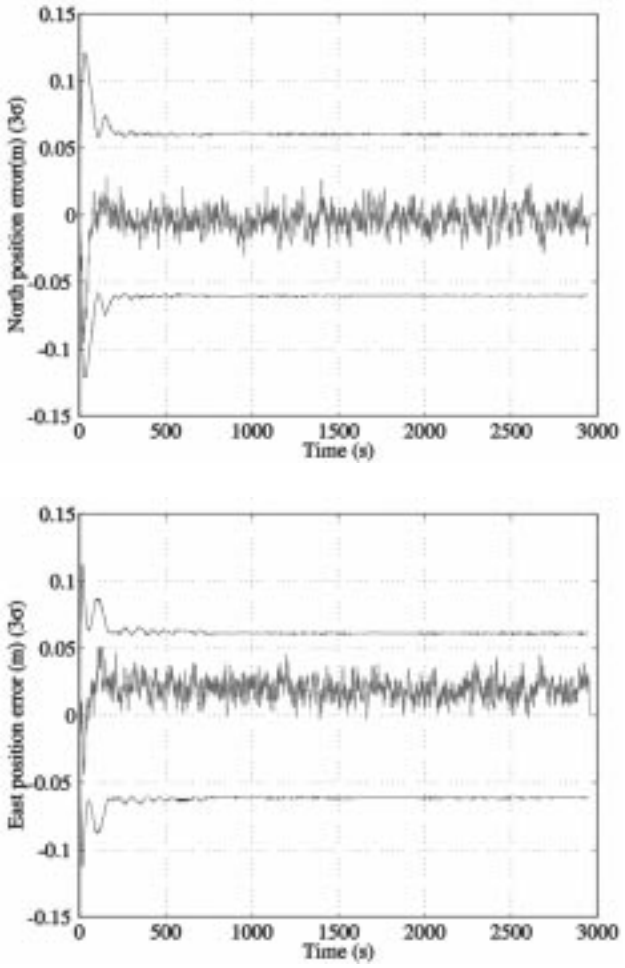


Fig. 6. Position estimate and 3σ bounds for adaptive motion among multiple features. As can be seen, a steady state is reached after about 500 s.

egy, only a 15° scanning angle was chosen at each time step. This is equivalent to obtaining 17 sonar returns. Without adaptation, a complete scan was taken at every time step, generating 400 sonar returns. Figure 7 shows the relative cost of adaptive sensing and motion with and without adaptation. The solid line denotes the case in which the robot moved randomly without adaptation, whereas the dashed line denotes the case in which the robot moved along the negative x-axis without adaptation as a function of sonar returns. The dotted vertical line indicates the point at which the adaptive case was terminated as 50 time steps was reached. As can be seen from Figure 7, the adaptive method obtained a map with high confidence after relatively few sonar returns. After obtaining a total of 20,000 returns, moving randomly and along the negative x-axis achieved only 93% and 33% of the confidence levels, respectively, that the adaptive method obtained with 850 sonar returns. Thus, the adaptive strategy required fewer measurements and achieved a higher confidence level than any of the strategies without adaptation. The actual vehicle's motion was similar to that shown in Figure 3.

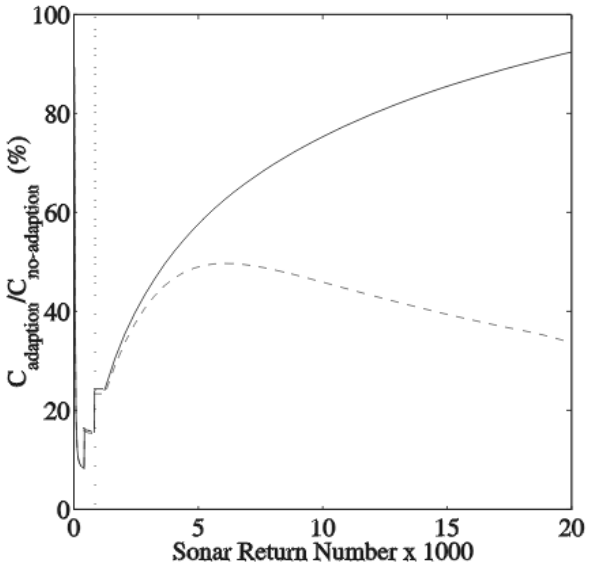


Fig. 7. Advantage of adaptive sensing and motion control. The cost function when performing adaptation divided by the cost function when not performing adaptation is plotted. Two cases in which no adaptation was performed are shown: the solid line denotes the case in which the robot moved randomly, whereas the dashed line denotes the case in which the robot moved along the negative x-axis. The vertical dotted line indicates the point at which the adaptive method had completed its 50 time steps and terminated.

Table 1 compares all the strategies on the basis of how many sonar returns and how many time steps are required before the map reaches a specified confidence level. As expected, the adaptive sensing and motion strategy required the fewest number of sonar returns to reach a given confidence level. However, it used more time steps than the adaptive motion strategy alone, since under the adaptive motion and sensing strategy only one feature was measured at each time step, whereas under the adaptive motion strategy, both features were measured at each time step.

5. Air Sonar Experimental Results

The algorithm was implemented on a Nomad Scout robot equipped with a 50-kHz Polaroid 6500 series ultrasonic sensor mounted on a stepping motor that rotated the sensor in 0.9° increments (see Figure 8). The error in the sensor and the vehicle odometry was assumed to be the same as that used in the simulations. The same constraints were employed as well. The sonar returns were assumed to be mainly specular; therefore, regions of constant depth were extracted from the scans (Leonard and Durrant-Whyte, 1992). In these experiments, rather than using the DNN track initiator, tracks were initiated from the first scan only. Figure 9 shows a rough model of the room and the configuration of the robot and

Table 1. Resources Needed to Achieve a Given Cost C in Simulation

	$C = C_e$		$C = C_r$	
Strategy	Returns	Steps	Returns	Steps
Adaptive sensing and motion	100	6	300	18
Adaptive motion	1600	4	5200	13
No adaptation, random motion	3200	8	20000	50
No adaptation, line motion	3600	9	∞	∞

$C_e = 0.038$ m^2 is the minimum cost achieved when moving in one direction during the experiment. $C_r = 0.0019$ m^2 is the minimum cost achieved in the simulations when moving randomly.

Fig. 8. The Nomad Scout robot with the Polaroid ultrasonic sensor mounted on top.

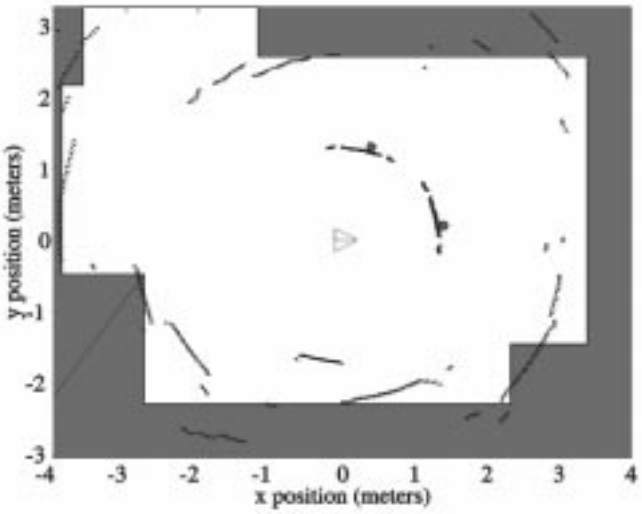


Fig. 9. The returns from the Polaroid sensor (dots) with a rough room model superimposed. The robot is drawn as a triangle, and the PVC tubes are indicated by small filled circles.

features (PVC tubes of known radius) in the experimental setup. The dots indicate individual sonar returns of the Polaroid sensor from a complete scan of the environment. Each complete scan of the environment consisted of 400 sonar returns. In these experiments, 15 motion steps were performed in each run.

Figure 10 shows the advantage of adaptation for a representative Nomad run, which is similar to the simulation of Figure 4. As can be seen from this figure, the advantage of performing adaptation is clear. Furthermore, the experimental result was well within the one-standard-deviation bound of the simulated predicted result over 2000 runs.

Figure 11 shows the advantage of performing adaptive motion and sensing over a straight-line movement along the negative x-axis for the Nomad Scout robot (solid line). The simulated result is shown by a dashed line along with the one-standard-deviation bound for 2000 simulated runs. As can be seen, the experimental cost ratio was within the one standard deviation of the simulated value. The adaptive strategy produced a high confidence map after relatively few measurements, which is consistent with the simulations in Figure 7. The nonadaptive method reached a confidence level of only 50% of the adaptive level, even after more than 20 times as many measurements were taken.

Table 2 shows the number of time steps and sonar returns required under each strategy to reach a map with some maximum specified cost C. As expected, the adaptive sensing and motion strategies required orders of magnitude less sonar returns than any of the other strategies. However, the adaptive motion strategy also used fewer time steps to reach a specified confidence level. When comparing this table of experimental results to Table 1, we see that they are consistent.

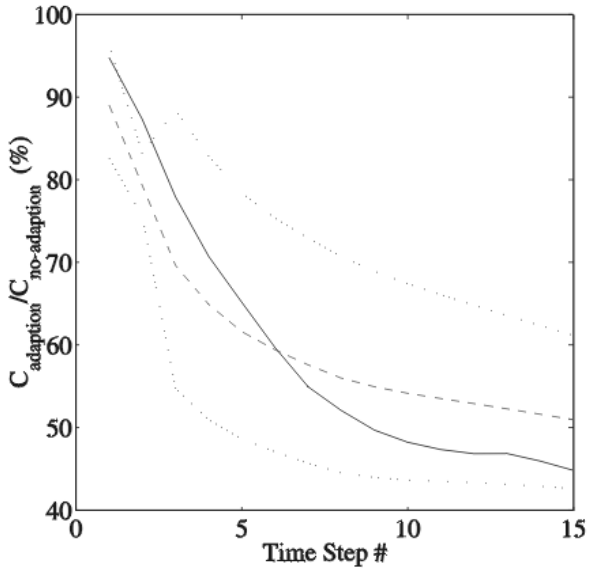


Fig. 10. Comparison of cost function $C(\mathbf{P})$ with and without adaptation for a representative Nomad experiment. The solid line is the ratio of the cost when performing adaptation divided by the cost of moving in a straight line along the negative x-axis. The dashed line denotes the average simulation result over 2000 runs, and the dotted line denotes the one-standard-deviation bound.

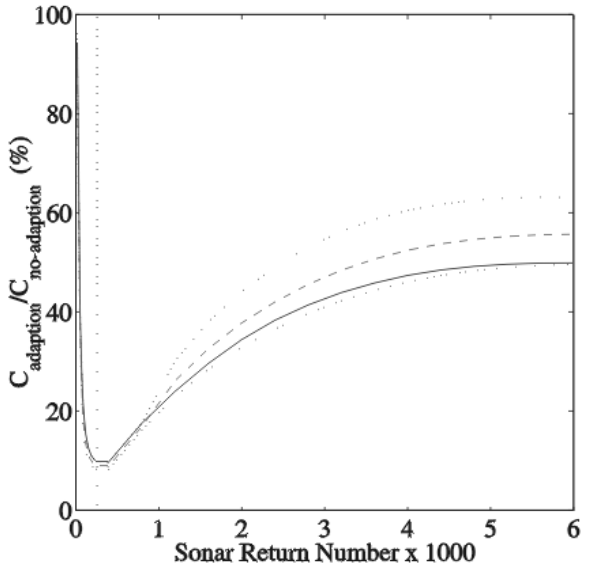


Fig. 11. Advantage of adaptive sensing and motion in a representative Nomad experiment. The solid line is the cost of adaptive sensing and motion divided by the cost of moving in a straight line along the negative x-axis. The dashed line is the average of 2000 simulated runs, and the dotted lines show the one-standard-deviation bound. The dotted vertical line indicates the time at which the adaptive case terminated as 15 time steps were completed.

Table 2. Number of Scans and Time Steps Required to Achieve a Cost of $C_e = 0.038$ m^2 or Less for a Representative Experiment

Strategy, $C = C_e$	Returns	Steps
Adaptive sensing and motion	100	6
Adaptive motion	2000	5
No adaptation, line motion	6000	15

The cost C_e was chosen to be the minimum cost achieved during the no-adaptation strategy.

6. Underwater Sonar Experimental Results

The second type of experiments conducted to test the adaptive stochastic mapping algorithm used a narrow-beam 675-kHz sector scan sonar mounted on a planar robotic positioning system, as shown in Figure 12. The positioning system was controlled by a Compumotor AT6450 controller card. The system was mounted on a $3.0 \times 9.0 \times 1.0$ m testing tank. The system was executed on a PC, and controller software and CML code were integrated to obtain a closed-loop system for performing CML. At each time step, eq. (13) was minimized over the action space of the robot to choose the motion and scanning angles of the sensor.

The experiments were designed to simulate an underwater vehicle equipped with a sonar that can scan at any direction relative to the vehicle at each time step. Conducting complete 360° scans of the environment at every time step is slow with a mechanically scanned sonar and computationally expensive with an electronically scanned sonar. For these experiments, we envisioned a vehicle mounted with two sonar systems, one forward looking for obstacle avoidance and one that can scan at any angle for localization purposes. The forward-looking sonar was assumed to scan an angle of ±30°. The scanning sonar was limited to scan over $U_\theta = [-15°, 15°]$ at each time step. The scanning was performed in intervals of 0.15°. The sonar systems were modeled to have a standard deviation in bearing of 10.0° and 2.0 cm in range. Between each scan by the sonar, the vehicle could move between 15 cm and 30 cm. The lower limit signifies a minimum speed at which the vehicle could move before loosing control, whereas the upper limit signifies the maximum speed of the vehicle. The vehicle was constrained to turn in increments of only 22.5°. Furthermore, we assumed that the vehicle was equipped with a dead-reckoning system with an accuracy of 10% of distance traveled and an accuracy of 1.0° in heading.

Figure 13 shows a typical scan taken by the sonar from the origin. The crosses show individual sonar returns. The circles show the features (PVC tubes). The dotted circles around the features signify the minimum allowable distance between the vehicle and the features. The triangle shows the position of the sensor. Circular arc features were extracted from the sonar scans using the technique described by Leonard and Durrant-Whyte (1992).

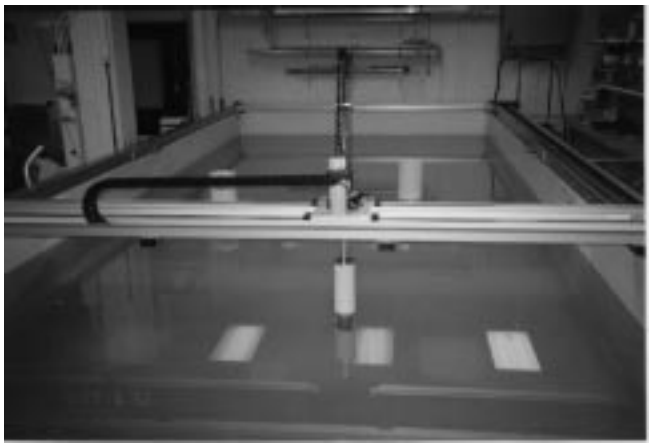


Fig. 12. The planar robotic positioning system and sector scan sonar used in the underwater sonar experiments. The water in the tank is approximately 1 m deep. The transducer was translated and rotated in a horizontal plane midway through the water column.

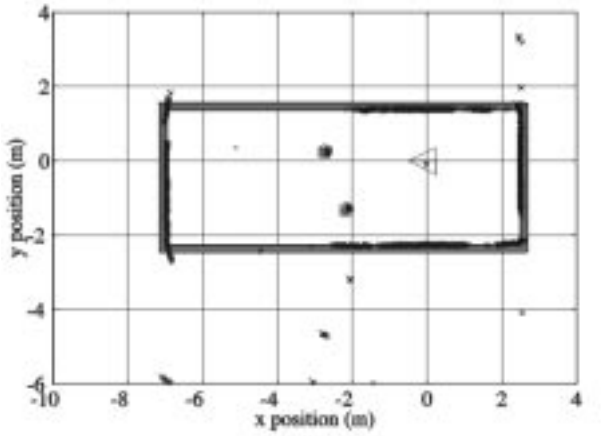


Fig. 13. The returns from the underwater sonar for a 360° scan of the tank from the origin. The crosses show individual returns. The small circles identify the positions of the features (PVC tubes), with a dotted 5-cm outside circle drawn around them to signify the minimum allowable distance between the sonar and the features. The sonar was mounted on the carriage of the positioning system, which served as a simulated autonomous underwater vehicle. The location of the sonar is shown by a triangle. The outline of the tank is shown in gray.

Figure 14 shows the sensor trajectory for a representative underwater sonar experiment with the complete algorithm. The sensor started at the origin and moved around the tank using the adaptive stochastic mapping algorithm to decide where to move and where to scan. Based on minimization of the cost function in eq. (13), the vehicle selected one target to scan to provide localization information. In addition, at each time step, the sonar was also scanned in front of the vehicle for obstacle avoidance. Figure 15 shows the x and y errors for the experiment and associated error bounds. No sonar measurements were obtained from approximately time step 75 to time step 85 due to a communication error between the sonar head and the host PC. Figure 16 shows the cost as a function of time. Solid vertical lines in Figures 15 and 16 indicate the time steps when features of the environment were removed. Figure 17 plots the vehicle position error versus time for the stochastic mapping algorithm in comparison to dead reckoning.

Figures 18 through 20 show the results of a nonadaptive experiment in which the vehicle moved in a straight line in the negative x-direction with the two objects present throughout the experiment. Without adaptive motion, the observability of the features was degraded and the y estimate seemed to diverge. Comparing Figure 19 with Figure 15, we observe that the nonadaptive strategy had a maximum 3σ confidence level of about 0.8 m and 0.2 m in the x and y positions, respectively, whereas the maximum 3σ confidence level under adaptation was about 0.17 m and 0.15 m in the y and x positions, respectively. A further indication of the advantage of the adaptive approach can be seen by comparing Figure 20 with Figure 17.

Interestingly, over a range of different operating conditions, the system selectively explores different objects in the environment. For example, Figure 21 shows the sensor path for two different underwater sonar experiments under similar conditions in which an exploratory behavior can be observed.

7. Conclusion

This paper considered the problem of adaptive sensing for feature-based CML by autonomous underwater vehicles. An adaptive sensing metric was incorporated within a stochastic mapping algorithm and tested via air and underwater experiments. This is the first time, to our knowledge, that a feature-based CML algorithm has been implemented with underwater sonar data.

We introduced a method for performing adaptive CML in a priori unknown environments for any number of features. The adaptive method was based on choosing actions that, given the current knowledge, would maximize the information gained in the next measurement. This approach can easily be implemented as an extra step in a stochastic mapping algorithm for CML. The validity and usefulness of the approach were verified in simulation and in experiments with air and underwater sonar data.

Based on the air sonar experiments, we feel confident that the simulation can accurately predict experimental outcomes. For example, the three typical paths for the robot shown in Figure 3 are representative of both adaptive simulations and real data experiments, and the plots comparing the performance

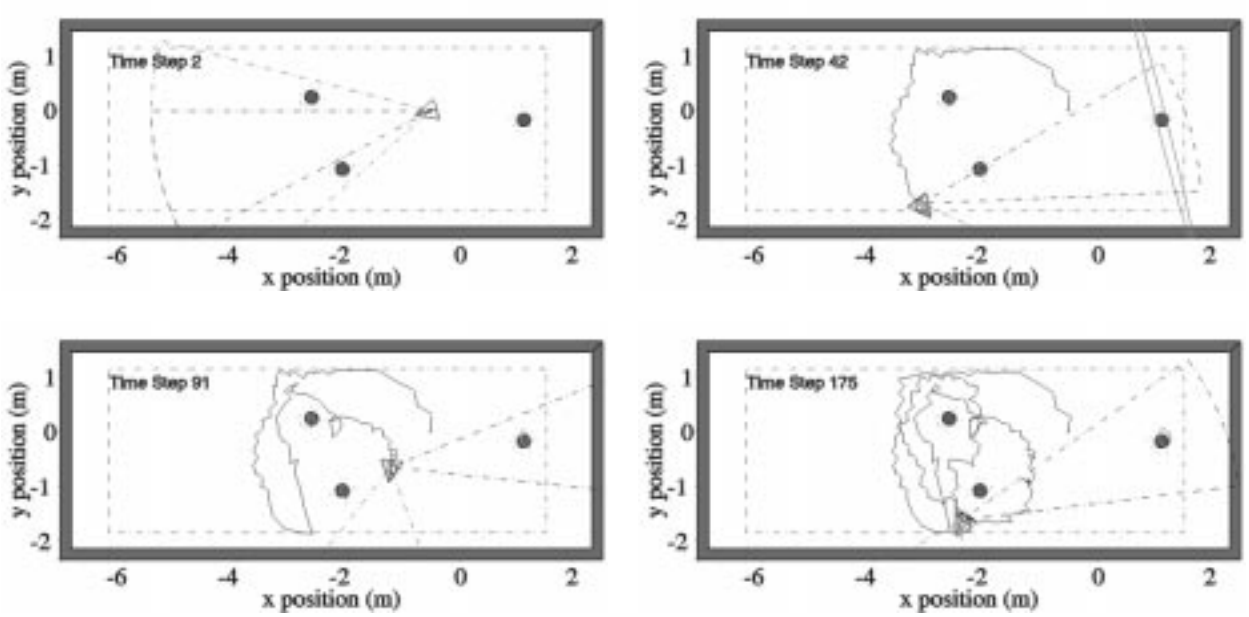


Fig. 14. Time evolution of the sensor trajectory for an adaptive stochastic mapping experiment with the underwater sonar. The feature on the right was added at the 40th time step, and the two features on the left were removed near the end of the experiment. The vehicle started at the origin and moved adaptively through the environment to investigate different features in turn, maximizing eq. (13) at each time step. The filled circles designate the feature locations and are surrounded by dotted circles that designate the standoff distance used by an obstacle avoidance routine. Similarly, the dashed-dotted rectangle designates the standoff distance to the tank walls. The dashed-dotted line represents the scanning region selected by the vehicle at the last time step. The large triangle designates the vehicle's position. The vehicle was constrained from moving outside the dashed-dotted lines to avoid collisions with the walls of the tank. Sonar returns originating from outside the dashed-dotted lines were rejected.

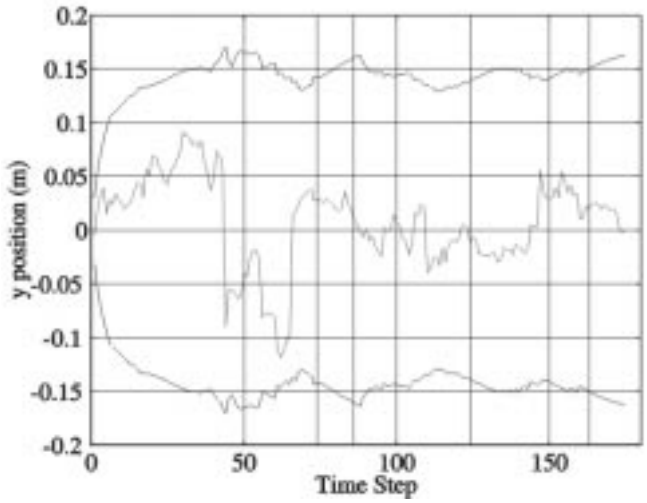
with and without adaptation are similar. The advantage of performing adaptive CML is notable when adapting only the motion of the vehicle (Figs. 4 and 10). However, more substantial gains are obtained when performing adaptive motion *and* sensing. This is apparent from Figures 7 and 11, where the number of sonar returns required to obtain a given confidence level is an order of magnitude fewer than when not performing adaptation.

The adaptive sensing technique employed here is a local method. At each cycle, only the next action of the robot is considered. By predicting over an expanded time horizon, one can formulate global adaptive mapping and navigation. For example, one can consider how to determine the best path between the robot's current position and a desired goal position. In this case, the space of possible actions grows tremendously, and a computationally efficient method for searching the action space will be essential.

In future research, we will integrate adaptive sensing within a hybrid estimation framework for CML in development in our laboratory (Smith and Leonard, 1997; Smith, 1998). In addition, we will perform experiments with more complex objects and develop methods that incorporate additional criteria for adaptation to address effects such as occlusion, rough surface scattering, and multiple reflections.

CML is a compelling topic for investigation because there are so many open issues for future research. Key questions include the following:

- How can representation of complex, natural environments be incorporated within a metrically accurate, feature-based mapping approach?
- How can we reliably extract features from sensor data?
- What are the trade-offs between global versus local, absolute versus relative, and metrical versus topological mapping?
- How can we select which types of environmental features are most useful as navigation landmarks?
- How can data association ambiguity be effectively addressed?
- How can computational complexity be overcome in extending CML to long duration missions over large areas?
- In which situations can long-term bounded position errors be achieved?
- How can CML be extended to dynamic environments in which previously mapped environmental objects can move or disappear and new features can appear in previously mapped areas?
- How can CML be integrated with path planning and obstacle avoidance?



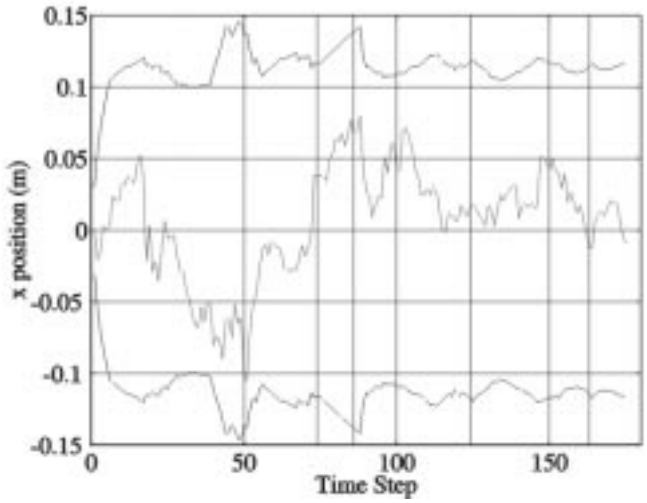


Fig. 15. Position errors in the x and y directions and 3σ confidence bounds for adaptive underwater experiment.

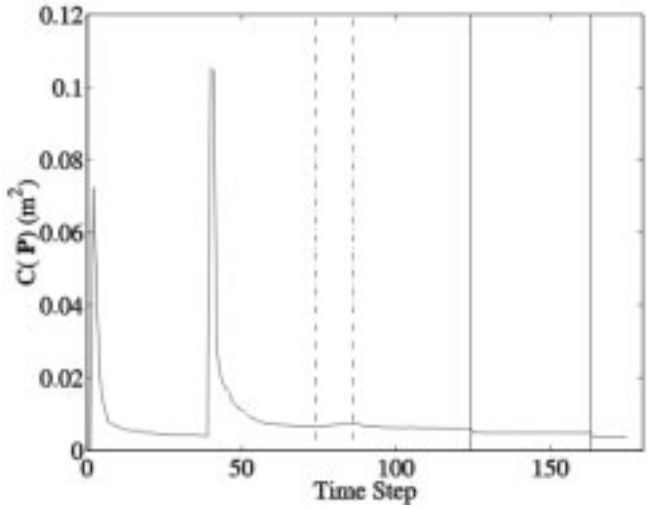


Fig. 16. Cost as a function of time for the adaptive underwater experiment. The cost increased at approximately the 40th time step when the third object was inserted into the tank and was observed for the first time. During the time interval between the two dashed vertical lines, no sonar data were obtained due to a serial communications problem between the PC and the sonar head. The two solid vertical lines designate the time steps during which features were removed from the tank, to simulate a dynamic environment.

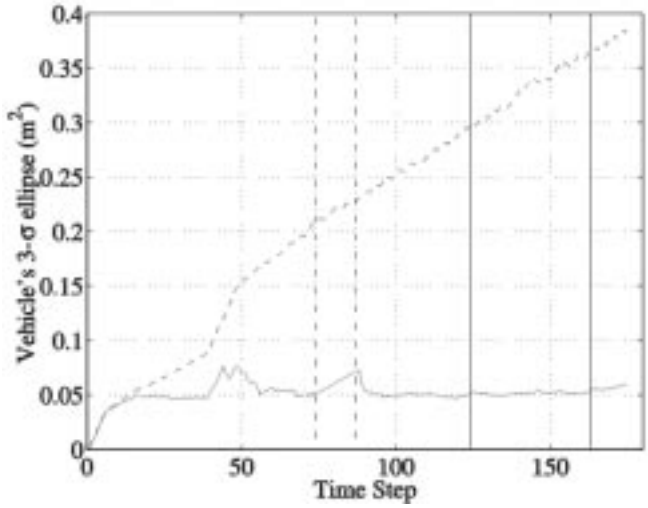


Fig. 17. Vehicle position error versus time for dead reckoning (dashed line) and the adaptive delayed nearest neighbor stochastic mapping algorithm (solid line). During the time interval between the two dashed vertical lines, no sonar data were obtained due to a serial communications problem between the PC and the sonar head. The two solid vertical lines designate the time steps when features were removed from the tank, to simulate a dynamic environment.

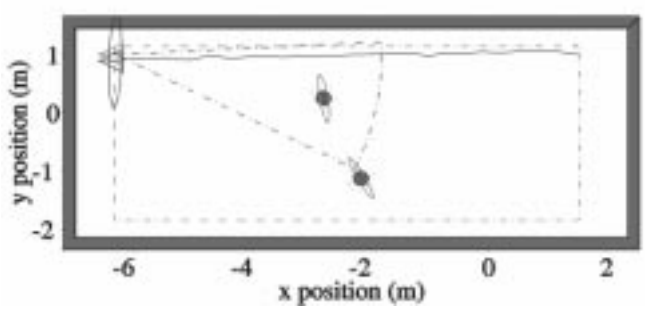


Fig. 18. Sensor trajectory for a nonadaptive experiment in which the vehicle moved in a straight line. While accurate location information was obtained in the x direction, the concurrent mapping and localization process diverged in the y-direction. The dashed-dotted rectangle indicates the standoff distance from the tank walls used by an obstacle avoidance routine.

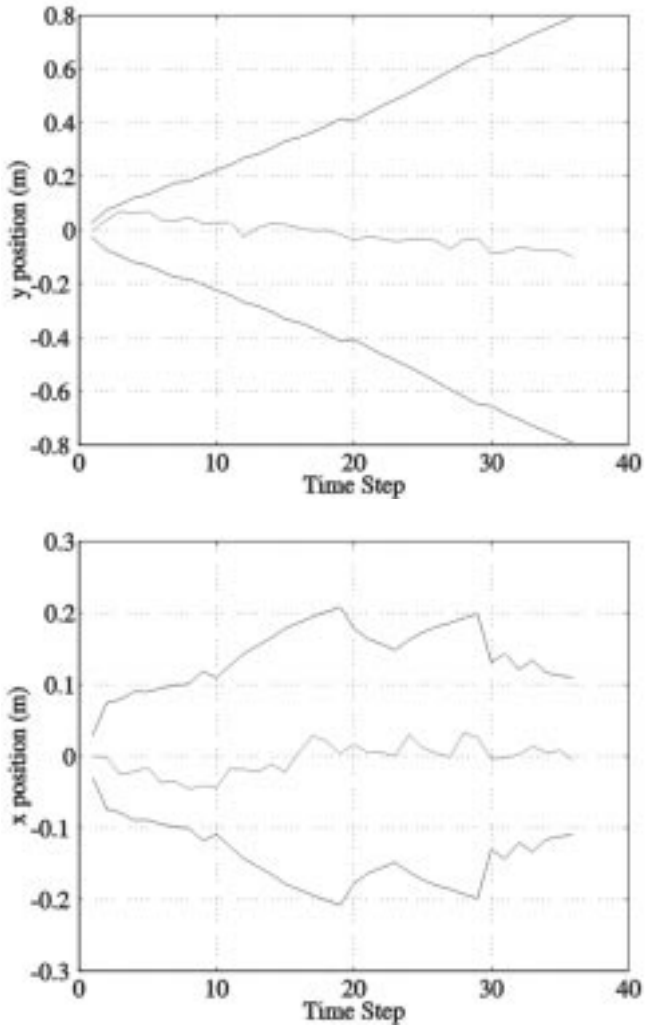


Fig. 19. Position errors in the x and y directions and 3σ confidence bounds for the nonadaptive underwater experiment.

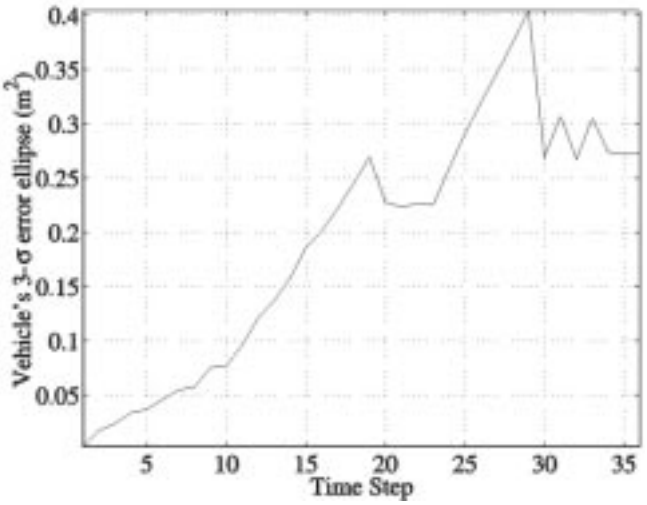


Fig. 20. Vehicle error as a function of time for the nonadaptive underwater experiment.

Although this paper has focused on an adaptive sensing metric for improved state estimation performance in CML, we believe that other types of sensing strategies can help answer many of these questions. We have shown that adaptive sensing can save time and energy and reduce the amount of data that needs to be acquired. However, we believe that adaptive strategies have the potential to improve state estimation robustness, ease data association ambiguity, prevent divergence, and facilitate recovery from errors. Investigation of adaptive strategies for these purposes will be addressed in future research.

Appendix: Stochastic Mapping

This appendix provides more detail on the stochastic mapping method, based on the work of Smith, Self, and Cheeseman (1990). Stochastic mapping is simply a special way of organizing the states in an EKF and providing a consistent way of adding new states to the systems as more features are observed and estimated. In a Kalman filter, through the combination of a dynamic model and an observation model, an estimated state $\mathbf{x}$ and associated estimated covariance $\mathbf{P}$ are produced. In this paper, the dynamic model for the robot was given by

$$\begin{aligned} \mathbf{x}_{r_{k+1|k}} &= \mathbf{f}(\mathbf{x}_{r_k}, \mathbf{u}_k) + \mathbf{G}(\mathbf{u}_k)\mathbf{d}_\mathbf{x} \\ &= \mathbf{x}_{r_{k|k}} + \mathbf{T}_{\phi_k} + \mathbf{G}(\mathbf{u}_k)\mathbf{d}_\mathbf{x}, \end{aligned}$$

where, as in the main text, k defines the time index. Since all features are assumed to be static, there are no dynamics for the features; thus,

$$\mathbf{x}_{i_k} = \hat{\mathbf{x}}_{i_k}.$$

The noise scaling matrix $\mathbf{G}$ is defined in eq. (6), and the transformation matrix $\mathbf{T}$ is defined in eq. (7). As in the main text,

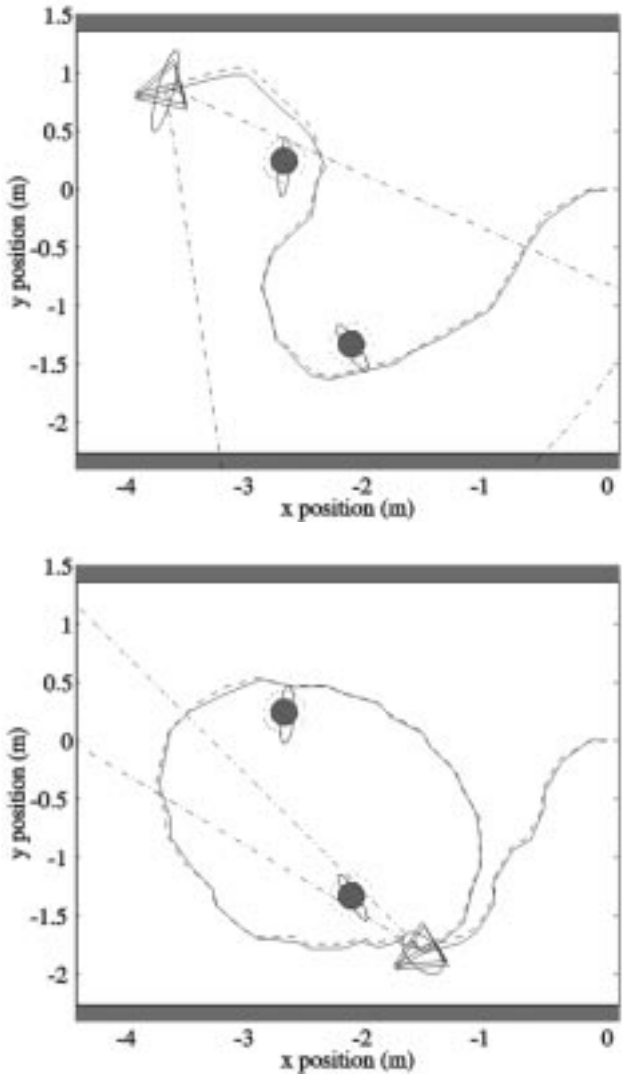


Fig. 21. Two representative stochastic mapping experiments that exhibited adaptive behavior. In each figure, the solid line shows the estimated path of the sensor and the dashed line shows the actual path. The triangle indicates the final position of the sensor. The filled disks indicate the locations of the features (PVC tubes). The ellipses around the features and the sensor are the 3σ contours; that is, the 99% highest confidence regions. The sonar view is indicated by the dashed-dotted line. In the left figure, the sensor had a scanning angle of $[-30°, 30°]$. The sensor started at (0,0) and adaptively determined the path to take and the direction to scan, resulting in "exploratory" behavior. The sensor first moves over to one of the objects, turns, and moves around the second. The experiment illustrated in the right figure was similar, with the exception that the sonar was able to scan an area of only $[-7.5°, 7.5°]$ each time step. Again, we can see that exploratory behavior emerged as the sensor attempted to maximize the information it obtained about the environment. In these experiments, rather than using the delayed nearest neighbor track initiator, tracks were initiated from the first scan only. In addition, the robot was constrained to turn a maximum of 30° at each time step in 15° increments.

the state vector $\mathbf{x} = [\mathbf{x}_r\ \mathbf{x}_1 \ldots \mathbf{x}_N]$ consists of the robot's state $\mathbf{x}_r$ and all the N feature states denoted by $\mathbf{x}_1 \ldots \mathbf{x}_N$. The control input is given by $\mathbf{u}$, and the process noise is given by $\mathbf{d_x}$.

As the robot moves around in its environment, the features are observed through the robot's sensor. The relation between the current state and the observations is given by the observation model

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{d_z}, \tag{16}$$

where $\mathbf{z}$ is range and bearing measurements to the features in the environment and $\mathbf{h}$ is the nonlinear transformation from the Cartesian coordinates of $\mathbf{x}$ and the polar coordinates of $\mathbf{z}$.

As the robot moves around in its environment, the system evolves through (1) robot displacement, (2) new feature integration, and (3) reobservation of features. Each of these steps of stochastic mapping is presented below.

Robot Displacement

When the robot moves a distance given by $\mathbf{u}_k$, the robot state, $\mathbf{x}_r$, estimated at time step $k+1$, is given by taking the expectation of the dynamic model given above:

$$\hat{\mathbf{x}}_{r_{k+1|k}} = \hat{\mathbf{x}}_{r_{k|k}} + \mathbf{T}_{\hat{\phi}_{k|k}}\mathbf{u}_k \ \text{ and } \hat{\mathbf{x}}_{i_{k+1|k}} = \hat{\mathbf{x}}_{i_k}.$$

Under the assumption of eq. (2), that this approximation error is small, the Fisher information is updated through the linearized system by

$$\mathbf{I}_{k+1|k} = (\mathbf{F_x}\mathbf{I}_{k|k}^{-1}\mathbf{F}_\mathbf{x}^T + \mathbf{G}(\mathbf{u}_k)\mathbf{G}(\mathbf{u}_k)^T)^{-1}.$$

Here, $\mathbf{F_x}$, is the Jacobian of $\mathbf{f}$ with respect to $\mathbf{x}$ evaluated at $\hat{\mathbf{x}}_{k|k}$.

New Feature Integration

If the robot observes a new feature $\mathbf{z}_{\text{new}} = [r\ \theta]$ with respect to the its reference frame, a new feature state $\hat{\mathbf{x}}_{N+1}$ is estimated and incorporated by

$$\hat{\mathbf{x}}_{N+1} = \mathbf{l}(\hat{\mathbf{x}}_{k|k}, \mathbf{z}_{\text{new}}) = \begin{bmatrix} \hat{x}_r + r\cos(\phi + \hat{\theta}) \\ \hat{y}_r + r\sin(\phi + \hat{\theta}) \end{bmatrix}.$$

The new feature is integrated into the map by adding this new state to $\mathbf{x}$ and $\mathbf{I}$; that is,

$$\hat{\mathbf{x}}_{k+1|k} = \begin{bmatrix} \hat{\mathbf{x}}_{k|k} \\ \hat{\mathbf{x}}_{N+1} \end{bmatrix}$$

$$\mathbf{P}_{N+1N+1} = \mathbf{L}_{\mathbf{x}_r}\mathbf{P}_{rr}\mathbf{L}_{\mathbf{x}_r}^T + \mathbf{L_z}\mathbf{R}\mathbf{L}_\mathbf{z}^T$$

$$\mathbf{P}_{N+1\,i} = \mathbf{P}_{i\,N+1}^T = \mathbf{L}_{\mathbf{x}_r}\mathbf{P}_{ri},$$

where $\mathbf{L}_{\mathbf{x}_r}$ and $\mathbf{L}_{\mathbf{z}}$ constitute the Jacobian of $\mathbf{l}$ with respect to the robot state $\mathbf{x}_r$ evaluated at $\hat{\mathbf{x}}_{r_{k|k}}$ and $\mathbf{z}_{\text{new}}$ evaluated at $\mathbf{z}_{\text{new}}$. The covariance of the state of the new features is given a priori by $\mathbf{R}$. The structure of the covariance for the system, $\mathbf{P}$, is given in eq. (4), and its inverse is a good estimate for the Fisher information of the system; that is, $\mathbf{P} = \mathbf{I}^{-1}$.

Reobservation of Features

When a feature i is reobserved, we use the prediction step of the EKF to update the vehicle's state and the map. By introducing the following definitions,

$$\begin{aligned} \tilde{x}_i &= x_{i_{k+1|k}} - x_{r_{k+1|k}} \\ \tilde{y}_i &= y_{i_{k+1|k}} - y_{r_{k+1|k}}, \end{aligned}$$

the observation model $\mathbf{h}$ (eq. (16)) for feature i takes the form

$$\begin{aligned} \mathbf{z}_{i_k} &= \begin{bmatrix} r_{i_k} \\ \theta_{i_k} \end{bmatrix} = \begin{bmatrix} \sqrt{\tilde{x}_i^2 + \tilde{y}_i^2} \\ \tan^{-1}(\tilde{y}_i/\tilde{x}_i) - \phi_k \end{bmatrix} + \mathbf{d}_{z_i} \\ &= \mathbf{h}_i(\mathbf{x}_k) + \mathbf{d}_{\mathbf{z}i}. \end{aligned}$$

The noise process $\mathbf{d}_{z_i}$ is assumed to be white and Gaussian with covariance $\mathbf{R}_i$. If the n features $i_1 \ldots i_n$ are reobserved, the observation model becomes

$$\mathbf{z}_k = \begin{bmatrix} \mathbf{z}_{i_{1k}} \\ \vdots \\ \mathbf{z}_{i_{nk}} \end{bmatrix}, \ \mathbf{h} = \begin{bmatrix} \mathbf{h}_{i_1} \\ \vdots \\ \mathbf{h}_{i_n} \end{bmatrix}, \ \mathbf{R} = \begin{bmatrix} \mathbf{R}_{i_n} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{R}_{i_n} \end{bmatrix},$$

with the Jacobian of $\mathbf{h}$ given by $\mathbf{H}_{\mathbf{x}} = \mathbf{H}_{\mathbf{x}}(\hat{\mathbf{x}}_{k+1|k}) = d\mathbf{h}(\mathbf{x})/d\mathbf{x}|_{\mathbf{x}=\hat{\mathbf{x}}_{k+1|k}}$. These matrices are used in the update step of the EKF as follows:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1}(\mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k})) \\ \mathbf{I}_{k+1|k+1} &= \mathbf{I}_{k+1|k} + \mathbf{H}_{\mathbf{x}}^T \mathbf{R}^{-1} \mathbf{H}_{\mathbf{x}}, \end{aligned}$$

where $\mathbf{K}_{k+1}$ is the EKF gain given by

$$\mathbf{K}_{k+1} = \mathbf{I}_{k+1|k}^{-1} \mathbf{H}_{\mathbf{x}}^T (\mathbf{H}_{\mathbf{x}} \mathbf{I}_{k+1|k}^{-1} \mathbf{H}_{\mathbf{x}}^T + \mathbf{R})^{-1}.$$

Computational Issues

To prevent instability in the coding of the EKF, it is advised that the square root filter or the Josephson form covariance update (Bar-Shalom and Li, 1993) be employed. In these implementations, the Josephson form was used, which ensures the positive definiteness of the EKF.

To prevent the EKF from becoming overconfident, extra process noise was added to the feature position. That is, at each time step, each feature was assumed to have a process noise including the same order of magnitude as $\mathbf{d}_{\mathbf{x}}$. This was placed in the $\mathbf{G}$ matrix.

Acknowledgments

This research was funded in part by the Naval Undersea Warfare Center, Newport, Rhode Island. The first author acknowledges the support of NFR (Norwegian Research Council) through grant 109338/410. The second author acknowledges the support of the Henry L. and Grace Doherty Assistant Professorship in Ocean Utilization and NSF Career Award BES-9733040. Furthermore, the authors would like to thank Seth Lloyd for fruitful discussions and comments. The authors would also like to thank Jan Meyer, Jong H. Lim, Marlene Cohen, and Tom Consi for their help with the Nomad hardware integration, Professor J. Milgram for the use of the testing tank, Paul Dambra for the installation of the positioning system, and Imagenex Technology Corporation and Paul Newman for their help with the sonar integration.

References

Au, W. 1993. *The Sonar of Dolphins*. New York: Springer-Verlag.

Bajcsy, R. 1988. Active perception. *Proc. IEEE* 76:996–1005.

Bar-Shalom, Y., and Fortmann, T. E. 1988. *Tracking and Data Association*. New York: Academic Press.

Bar-Shalom, Y., and Li, X.-R. 1993. *Estimation and Tracking: Principles, Techniques, and Software*. Artech House.

Bellingham, J. G., Goudey, C. A., Consi, T. R., Bales, J. W., Atwood, D. K., Leonard, J. J., and Chryssostomidis, C. 1994. A second generation survey AUV. *IEEE Conference on Autonomous Underwater Vehicles*, Cambridge, MA.

Bellingham, J. G., and Willcox, J. S. 1996. Optimizing AUV oceanographic surveys. *IEEE Conference on Autonomous Underwater Vehicles*, Monterey, CA.

Berger, J. O. 1985. *Statistical Decision Theory and Bayesian Analysis/*, 2nd ed. Springer-Verlag.

Betgé-Brezetz, S., Hébert, P., Chatila, R., and Devy, M. 1996. Uncertain map making in natural environments. *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 1048–1053.

Blake, A., and Yuille, A., eds. 1992. *Active Vision*. Cambridge, MA: MIT Press.

Bozma, O., and Kuc, R. 1991. Characterizing pulses reflected from rough surfaces using ultrasound. *J. Acoustical Soc. Am.* 89(6):2519–2531.

Chong, K. S., and Kleeman, L. 1997. Sonar feature map building for a mobile robot. *Proc. IEEE Int. Conf. Robotics and Automation*.

Elfes, A. 1987. Sonar-based real-world mapping and navigation. *IEEE J. Robotics Automation* RA-3:249–265.

Gelb, A. C. 1973. *Applied Optimal Estimation*. Cambridge, MA: MIT Press.

Hager, G. 1990. *Task-Directed Sensor Fusion and Planning: A Computational Approach*. Boston: Kluwer Academic.

Hébert, P., Betgé-Brezetz, S., and Chatila, R. 1996. Decoupling odometry and exteroceptive perception in building a global world map of a mobile robot: The use of local maps. *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 757–764.

Kuipers, B. J., and Byun, Y. 1991. A robot exploration and mapping strategy based on a semantic heirarchy of spatial representations. *Robotics Autonomous Sys.*, pp. 47–63.

Leonard, J. J., Bennett, A. A., Smith, C. M., and Feder, H.J.S. 1998. Autonomous underwater vehicle navigation. *IEEE ICRA Workshop on Navigation of Outdoor Autonomous Vehicles*, Leuven, Belguim.

Leonard, J. J., Cox, I. J., and Durrant-Whyte, H. F. 1992. Dynamic map building for an autonomous mobile robot. *Int. J. Robotics Research* 11(4):286–298.

Leonard, J. J., and Durrant-Whyte, H. F. 1992. *Directed Sonar Sensing for Mobile Robot Navigation*. Boston: Kluwer Academic.

Lucido, L., Popescu, B., Opderbecke, J., and Rigaud, V. 1996. Segmentation of bathymetric profiles and terrain matching for underwater vehicle navigation. *Proc. of Second Annual World Automation Conf.*, Montpellier, France.

Manyika, J. S., and Durrant-Whyte, H. F. 1994. *Data Fusion and Sensor Management: A Decentralized Information-Theoretic Approach*. New York: Ellis Horwoord.

Medeiros, M., and Carpenter, R. 1996. High resolution array signal processing for AUVs. *AUV 96*, pp. 10–15.

Milne, P. H. 1983. *Underwater Acoustic Positioning Systems*. London: E.F.N. Spon.

Moran, B. A., Leonard, J. J., and Chryssostomidis, C. 1997. Curved shape reconstruction using multiple hypothesis tracking. *IEEE J. Ocean Eng.* 22(4):625–638.

Moutarlier, P., and Chatila, R. 1989. Stochastic multisensory data fusion for mobile robot location and environment modeling. *5th Int. Symp. on Robotics Research*, Tokyo.

Rencken, W. D. 1993. Concurrent localisation and map building for mobile robots using ultrasonic sensors. *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, Yokohama, Japan, pp. 2192–2197.

Russell, S., and Norvig, P. 1995. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ: Prentice Hall.

Russell, S., and Wefald, E. 1995. *Do the Right Thing: Studies in Limited Rationality*. Cambridge, MA: MIT Press.

Singh, H. 1995. An entropic framework for AUV sensor modelling. Ph.D. thesis, Massachusetts Institute of Technology.

Smith, C. M. 1998. Integrating mapping and navigation. Ph.D. thesis, Massachusetts Institute of Technology.

Smith, C. M., and Leonard, J. J. 1997. A multiple hypothesis approach to concurrent mapping and localization for autonomous underwater vehicles. *Int. Conf. on Field and Service Robotics*, Sydney.

Smith, R., Self, M., and Cheeseman, P. 1990. Estimating uncertain spatial relationships in robotics. In Cox, I., and Wilfong, G. (eds.) *Autonomous Robot Vehicles*. Springer-Verlag.

Stewart, W. K. 1988. Multisensor modeling underwater with uncertain information. Ph.D. thesis, Massachusetts Institute of Technology.

Thrun, S., Fox, D., and Burgard, W. 1998. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning* 31:29–53.

Tuohy, S. T., Leonard, J. J., Bellingham, J. G., Patrikalakis, N. M., and Chryssostomidis, C. 1996. Map based navigation for autonomous underwater vehicles. *Int. J. Offshore Polar Eng.* 6(1):9–18.

Uhlmann, S. J., Julier, M., and Csorba, J. K. 1997. Nondivergent simultaneous map building and localisation using covariance intersection. Navigation and control technologies for unmanned systems II.

Urick, R. 1983. *Principles of Underwater Sound*. New York: McGraw-Hill.

Willcox, J. S., Zhang, Y., Bellingham, J. G., and Marshall, J. 1996. AUV survey design applied to oceanic deep convection. *IEEE Oceans*, pp. 949–954.