Feature-Based Concurrent Mapping and Localization for AUVs

Christopher M. Smith[1] *John J. Leonard*[1] *Andrew A. Bennett*[1] *Christopher Shaw*[2]

[1] Department of Ocean Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139, U.S.A.

[2] Naval Undersea Warfare Center
Newport, Rhode Island 02841, U.S.A.

Abstract - One of the primary problems in marine robot navigation is the growth of uncertainty. Sensory measurements of the environment provide an enticing source of information about vehicle location. Various current approaches to AUV sensor data fusion fall short of incorporating environmental measurements in navigation estimation to improve navigation performance in unmapped environments. We present a unified approach to using environmental measurements to map an unknown environment and localize the vehicle within that map. First, we discuss the importance of our feature-based approach to concurrent mapping and localization (CM&L). Innovative aspects of this algorithm, including feature modeling and decision dependencies, are highlighted. We then present our feature-based CM&L algorithm. Finally, we draw conclusions about the challenges in implementing this algorithm and the performance gains expected for AUV navigation.

Keywords:
underwater robotics; data fusion; navigation; feature-based modeling.

I. NAVIGATION INCORPORATING ENVIRONMENTAL INFORMATION

Autonomous underwater vehicle (AUV) navigation is plagued by the insufficiency of proprioceptive information to bound error growth. The kinds of information available to the vehicle from itself, such as dead reckoning and inertial navigation systems (INS), provide information about the derivatives of vehicle position, and so are subject to uncorrectable drift. To navigate accurately for missions of any substantial length (in duration, path length, or number of maneuvers), some form of ground-fixed relative positioning information must be obtained. In land robotics, this requirement can often be met by the global positioning system (GPS). AUVs can surface to reset their navigation systems with GPS, but navigational error will grow whenever they are away from the surface. Often, acoustic beacons are deployed in a known arrangement to allow ground-fixed position measurements (e.g. LBL or USBL navigation). While these solutions work in some scenarios, there are missions where predeployment of an acoustic array or surface breaching to acquire GPS fixes are undesirable or impossible.

There is, however, another source of considerable information regarding ground-fixed relative position available during most missions: the environment through which the vehicle is moving. Most AUVs are equipped with some form of organic sonar (e.g. for measuring altitude, obstacle avoidance, or mapping the environment). If, in such a case, an accurate map of the bathymetry through which the vehicle traveled were available, bathymetric data could provide the vehicle with ground-fixed relative position information [8]. The question arises, though, what constitutes an accurate map and how can errors be dealt with. Even given these answers, there remain missions in areas where no map of sufficient accuracy or detail is available.

The project of concurrent mapping and localization (CM&L) is to extend this idea to unknown environments. In such a case, no prior map is available, but distinctive features can be identified and mapped as they are encountered. Concurrent to mapping these environmental features, subsequent measurements of these features can provide ground-fixed relative position information to improve vehicle navigation. Concurrent mapping and localization is, therefore, a unified approach to incorporating environmental information into vehicle navigation.

There are several current approaches which attempt to incorporate environmental information in order either to map the environment or to aid in navigation. Leonard *et al.* [1, 3, 2] have developed multiple hypothesis tracking for mapping unknown environments. This approach starts with a model-based decomposition of the environment and stresses using the physics of the sensing process to analyze and fuse sensor data in a theoretically consistent manner. Multiple competing hypotheses are compared within a Bayesian framework to provide increased explanatory power when distinguishing between possible alternatives. However, vehicle navigational uncertainty is not accounted for and can quickly lead to estimate divergence.

The stochastic map has been proposed by Smith *et al.* [6] and partially implemented by Moutarlier and Chatila [4] using a land robot. This augments vehicle state to track features in a Kalman filter. Model-level interactions, such as feature track initiation and measurement explanation, are treated in a somewhat *ad hoc* manner. Also, there is no attempt to resolve ambiguities in the source of measurements (the data association problem).

Stewart [7] investigated stochastic backprojection as a method for probabilistically mapping where sensor data is coming from. The measurements are probabilistically smeared to account for various forms of uncertainty (measurement noise, process noise, and vehicle navigation uncertainty). A cell-based representation of the environment is used; there are no discrete features postulated.

These approaches to incorporating environmental information into navigation fail to adequately address operation in unknown environments. While Uhlmann [9] has considered some of the implementation issues for CM&L (in particular, nonlinear estimation and uncertain correlations), he stops short of addressing the problem in a more general way.

There are three key elements needed for a viable approach to concurrent mapping and localization: explicit handling of vehicle uncertainty, delayed decisions based on multiple, competing hypotheses, and a feature-based representation of the environment. The first of these needs has been demonstrated by Moran [3]. The second holds in particular if a feature-based representation is used. In this case, delayed decisions are essential in providing proper explanatory power for decision-making.

The need for a feature-based representation is not obvious, and will be addressed in the next section. Then we will consider two innovative aspects of our CM&L algorithm: feature modeling and decision dependency. This will lead to a discussion of the algorithm itself. We will conclude with an outline of the challenges posed by the implementation of this algorithm and the expected navigational benefits of CM&L.

II. WHY HAVE FEATURES?

We have proposed a feature-based approach to incorporating environmental measurements into navigation, but it is *prima facie* unobvious why the environment should be represented in discrete features. There are many reasons for choosing such a representation scheme, and we explore a number of the most compelling here.

As stated above, cell-based representations seek to determine *where* particular measurements come from. Feature-based approaches additionally try to estimate *what* measurements come from, or, what processes give rise to particular measurements. Part of the motivation here is based on how the environment is modeled. In many cases, the physical processes underlying sensor measurement can be broken into discrete mathematical models. For example, sonar returns are characterized as specular or diffuse. Feature-based representation of the environment allows the use of multiple models to describe the measurement processes for different parts of the environment.

One reason for avoiding a cell-based approach is the effect of data smearing. Cell-based approaches often smear measurements onto a region of the map to account for measurement and navigational uncertainty. However, these two types of uncertainty are fundamentally different. Navigational uncertainty is an *a posteriori* amalgam of measurement and process noises. Measurement noise is stipulated *a priori*. By combining these uncertainties for data smearing, information is lost. If a feature-based approach is taken, a distinction is made between modeling features themselves and mapping the features globally. Characterization of a feature and relative positioning with nearby features can be obtained with low uncertainty (near the level of measurement noise) even when the vehicle navigational uncertainty is high. Thus, the vehicle can acquire information about a feature while yielding little information about the global map. This advantage is particularly useful when the vehicle can later back out map information (e.g. by relocating using previously discovered features). In a cell-based approach, the local information would be lost by reducing the information content to global levels of certainty.

Two final reasons for a feature-based approach arise from its inherent discretization of the environment. First, features are localized in extent, and feature size can (and probably will) be matched to sensor footprint. While field-oriented processing of data yields positional information (the depth is seven meters and the seven meter contour runs here), localized interpretations often go further in localizing the vehicle (the vehicle is on the east side of this hill). Second, features can be grouped together to form features, i.e. a network of features can itself be a feature. In the parlance of object-oriented analysis, features form a homomorphic hierarchy. This is important in providing feature distinguishability. If dots are features, they all look pretty much alike. But if we can form constellations of dots as features, recognizing particular features (constellations, and, therefore, individual dots) becomes much easier.

All of these reasons have led us to the conclusion that a feature-based representation is appropriate. This will especially be true in unknown environments, where existing maps do not tempt us into using a cell-based approach. Even when prior maps are available, however, we can use feature models to extract a feature-based representation from what is, essentially, a field- or cell-based representation.

A feature-based representation does, of course, require some knowledge in order to model the features. Although one can envision developing an adaptive conception of features, for now, this requirement means some *a priori* knowledge about the *kind* of environment to be encountered, even if specific details (i.e. a map) are unnecessary. In a way, our feature-based representation is a way to encode our *a priori* knowledge about the environment. In the next section, we will develop a generalized method for dealing with this knowledge.

III. FEATURE MODELING

In order to have a feature-based environmental representation, we need an understanding of what a feature is. Questions about what features are, how they can be detected, measured, and tracked, and what parts of the environment they represent fall into the category of feature ontology. Feature ontology is the sum of our *a* *priori* knowledge about the environmental context of the vehicle. We have developed a generalized framework for dealing with features; feature ontology is encapsulated in a Markov network of state observers. Essentially, an observer is a plant model for the vehicle or a feature, allowing a current estimate to be projected into a prediction of the plant (vehicle or feature track) at the next time cycle. Since what we are doing is predicting the future, and we are already committed to multiple, competing hypotheses in other portions of the algorithm, there is no reason not to hedge our bets by making more than one prediction. There are a number of reasons why this might be appealing. First, there may be more than one model for the feature in question, and we are unsure about which model to use for this prediction step. This might occur, for example, when we have separate dynamic models for steady motion and maneuvering, and are not sure whether the feature (or target) will begin a maneuver in this time step. Second, there may be some question about what feature model best represents the phenomenology so far received from the feature. For example, in Moran [3], plane and curved features are indistinguishable given a single measurement. The implementation is a small network of ontic possibilities. When a measurement is received that might be a plane or a curve, it is considered a plane. Subsequent predictions are split, allowing projection as if the feature were a curve as well as if it were a plane. Since there is this firm distinction between the ways in which we model the environment and the environment itself, the observers present a noumenal framework, our *a priori* understanding of the environmental context of the vehicle.

An observer network is simply that, a set of observers which are connected together by our understanding of the ontic possibilities and their consequences. Every state estimate is formed based on a particular observer.

$$\phi_i \mapsto x_k^l; \tag{1}$$

the observer ϕ_i gives rise to the state l at time k. Initial track states, or root states, are formed by an observer from a given measurement. All other states are transformations of existing state estimates using a particular plant model, the observer function ϕ. Each observer has one or more consequent observers:

$$\phi_i^+ = \{\phi_j\}_j \,. \tag{2}$$

This set of consequent observers operates on the state to produce a set of projected states at the next time cycle.

$$\left\{{}_l x_{k+1}^m\right\}_m = \left\{\phi_j\left(x_k^l\right)\right\}_j, \tag{3}$$

or,

$$\Phi\left(x_k^l\right) = \left\{{}_l x_{k+1}^m\right\}_m = {}_l X_{k+1}, \tag{4}$$

where ${}_l X_{k+1}$ is the set of states at time $k+1$ derived from state l, x_k^l, and m is an index over these states.

Given a state estimate from a particular observer, the next projection will be based on the consequent observers of that state's observer. An observer can have itself as a consequent. As a particular, and simple, example, consider a feature model for a point object. The state estimate of the object is simply the location (x, y) of the point. The observer does not alter the estimate (although it might increase the covariance matrix). A model for stationary points might consist of a single observer with itself as its only consequent.

IV. DECISION DEPENDENCY

Feature track trees are essentially a what-if game. Each state is our (in some sense) best estimate given the measurements, our *a priori* knowledge, and a series of decisions about what has happened. The need for such decisions arises due, primarily, to two factors. First, the stochastic nature of the problem is not limited to state estimation; we are also estimating what features are present. The set of features which describe the vehicle's environmental context is itself a random set. Because of this, discrete decisions about what features are present must be made in addition to the continuous decisions (or estimates) regarding the states of these proposed features. Second, because we are taking a multiple hypothesis approach to data association, we purposely allow decisions to be resolved in multiple ways until further evidence can confirm one hypothesized outcome. Because states are conditioned on these decisions, it only makes sense to compare states which have compatible dependencies on this set of decisions. In our what-if game, each decision we make to arrive at a state estimate splits the world into a number of possibilities. If we are to compare two states, there must not be any conflicts in the set of conditionals which their decision dependencies represent.

The typical hypothesis is an association of the vehicle, a feature track, and a measurement. Hypotheses and their formation is discussed in detail below. The important issue in considering decision dependencies is that a decision is made any time multiple competing hypotheses are available to describe the source of a measurement or the disposition of a feature. Once a single hypothesis remains, the decision is confirmed, and all state estimates depend on the same outcome, the confirmed hypothesis for that decision. If more than one hypothesis remains, the decision is tentative. Since all states depend on confirmed decisions in the same way, they may be ignored when determining whether two states have compatible decision dependencies. So, the set of tentative decisions Δ is the set of possible decisions δ upon which a state may depend.

Each state x has a set of decisions upon which it is dependent,

$$\Delta(x) = \{\delta_i\}\,. \tag{5}$$

In order for the state to obtain, each of these decisions δ_i must be resolved to the necessary hypothesis,

$$\delta_i(x) = \theta_j. \tag{6}$$

For two states to have compatible decision dependencies, there can be no decision where the states differ in their assumed outcome. Otherwise, the two states are compatible. We can define a function $d(x, \xi)$ returning the number of incompatible decisions between two states, x and ξ. This is, in some sense, the ontological distance between the states. State compatibility is then affirmed if this distance is zero:

$$d(x, \xi) = 0 \quad \text{iff} \quad \forall\, \delta_i \in \Delta(x), \Delta(\xi), \\ \delta_i(x) = \delta_i(\xi). \tag{7}$$

Note that there can be decisions that a state is not dependent on. The estimate represented by that state is then independent of the outcome of that particular decision. The state is orthogonal to the decision. By maintaining decision dependencies in this way (by allowing states to be orthogonal to certain decisions), we maintain as much separability as is possible for the states. For example, we above used the metaphor that having two competing hypotheses for a decision outcome was like splitting the world into two cases. However, we do not need to split our entire model into two new models, because much of our estimate structure can be orthogonal to the decision, and thus valid whatever the outcome of that particular decision. Thus, decision dependencies result in reducing the order of the problem by maximizing separability of states and decisions.

V. THE CONCURRENT MAPPING AND LOCALIZATION ALGORITHM

As shown in Figure 1, three processes, measurement, navigation, and feature mapping, are combined to improve estimates and to provide global explanations for what has happened. In the stochastic map of Smith *et al.* [6], vehicle and feature track models are combined in a single (large) state vector. For CM&L, vehicle and feature tracks are separated to enable a multiple hypothesis approach. In each time cycle, predicted vehicle states $X_{k|k-1}$ and feature states $\Xi_{k|k-1}$ are projected from the previous cycle. These predicted states are compared with sensor measurements Z using a gating function , to check for possible matches. Hypotheses Θ are formed to cover the following possibilities: a measurement has come from a feature, a measurement is spurious, a feature is not detected, or a measurement is the result of an unknown feature. Feature track states are updated $\Xi_{k|k}$ based on these hypotheses Θ, the predicted vehicle states $X_{k|k-1}$, and the measurements Z. The hypotheses Θ are combined to form global assignments Ω. Each assignment is a set of compatible hypotheses which explains the source of each measurement (a particular known feature, a new feature, or no feature) and the disposition of each feature (a particular measurement or a miss). Vehicle track states are updated $X_{k|k}$ based upon the possible global assignments Ω, the predicted feature track states $\Xi_{k|k-1}$, and the measurements Z. A Bayesian framework is used to evaluate the likelihood of each of these states, hypotheses, and assignments [5]. Determination of what has, in fact, happened, is made at the global level by rejecting unlikely assignments. The result of this determination is a pruning of assignments. This produces, teleologically, a removal of unsupportable hypotheses and states.

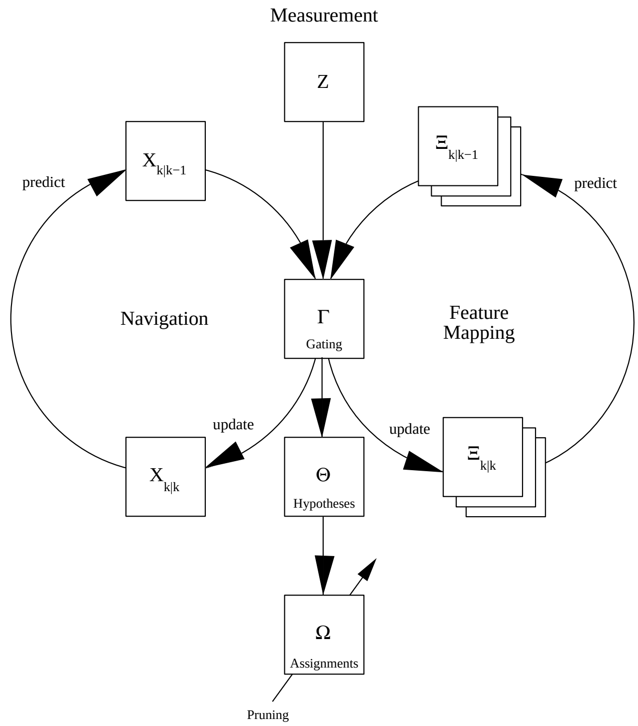

Figure 1. Process flow for feature-based CM&L. Information from relative measurements is used to update vehicle navigation and feature models simultaneously. Hypotheses and assignments are formed to allow *a posteriori* situational assessment at a global level.

A. *State Projection*

As discussed above, states can be projected to the next time cycle using the observer networks. Each vehicle or feature track state produces a set of at least one projected state. When more than one projection is made, the likelihood of each projected state is modified by the *a priori* probabilities assigned to their generative observers. Thus, the vehicle produces a set of predicted states from its set of updated states for the previous time cycle,

$$X_{k+1|k} = \Phi\left(X_{k|k}\right). \tag{8}$$

Likewise, each feature track i produces a similar set based on its updated states,

$${}^{i}\Xi_{k+1|k} = \Phi\left({}^{i}\Xi_{k|k}\right). \tag{9}$$

B. *Feature Track Updating*

Feature track updating is based on the hypotheses which are available to explain the disposition of each feature. Either a feature has given rise to one of the measurements (a match hypothesis), or it was not detected (a miss hypothesis). To reduce the number of unlikely matches, a gating function , is used. The gating function combines a predicted vehicle state, a predicted feature state, and a measurement to produce a statistical "distance" and determines whether it falls within the valid gate, that is, does not exceed the gating threshold γ:

$$, \ (x, \xi, z) < \gamma. \tag{10}$$

A common approach is to normalize the Kalman innovation to produce the Mahalanobis distance, which has a chi-square distribution. For a match hypothesis to be formed, there must be at least one combination of predicted vehicle state and predicted feature state such that the two states are compatible,

$$d\left(x_{k|k-1}, \xi_{k|k-1}\right) = 0, \tag{11}$$

and they gate with the measurement,

$$, \ \left(x_{k|k-1}, \xi_{k|k-1}, z_k\right) < \gamma. \tag{12}$$

If a match hypothesis is formed, all compatible vehicle and feature predicted track state pairs produce an updated feature track state using the measurement,

$$\xi_{k|k} = K\left(\xi_{k|k-1}, x_{k|k-1}, z_k\right), \tag{13}$$

where K represents a traditional Kalman update using the feature and vehicle models to form an augmented state vector. The predicted feature state is also carried through to account for the miss hypothesis. Updates to other feature estimates due to correlation may then be added to recover the traditional Kalman estimate.

Hypotheses are also formed to explain measurement sources. A measurement source may turn out to be a known feature (a match hypothesis as described above), no feature (a spurious measurement hypothesis), or a previously unmodeled feature (a new feature hypothesis). In this last case, a new feature track is initiated using the observer network for the feature being proposed. If there is more than one kind of feature, there can be more than one new feature hypothesis for a measurement.

C. *Vehicle Updating*

Since the vehicle takes part in all measurements, vehicle track states are updated on the basis of global assignments rather than individual hypotheses. Assignments are consistent, exhaustively explanatory sets of hypotheses. An assignment is a global explanation for what happened during a time cycle to produce the data received. The disposition of each feature and the source of each measurement must therefore be explained by hypotheses. Consistency ensures that there is one and only one explanation proffered by the assignment. For example, one could not say that measurement z comes from feature η and also hold that feature η was not detected. Cox and Leonard [1] discuss an efficient method for forming all possible assignments given a set of hypotheses in detail.

Once assignments are formed, possible consistent combinations of vehicle track states, feature track states, and measurements are used to perform Kalman individual updates:

$$x_{k|k} = K\left(x_{k|k-1}, \xi_{k|k-1}, z_k\right). \tag{14}$$

Again, additive correction terms can be used to reconstruct the optimal Kalman estimate, in which measurements are traditionally processed in a batch and correlated states are concurrently updated.

D. *Pruning*

Pruning is an essential step for any multiple hypothesis estimation scheme; otherwise, the number of prospective states increases exponentially [1]. Pruning is based on assignment likelihood. There are three essential methods for rejecting unlikely explanations. First, a likelihood threshold may be used, so that explanations which are very unlikely are rejected quickly. Second, the number of assignments allowed for each cycle may be limited to the k best. This prevents excessive horizontal spread of the track trees, so that the most likely possibilities can be concentrated on more fully. Finally, a choice can be forced after n timesteps. Since the primary goal of multiple hypothesis tracking is to delay decisions until enough corroborative evidence is produced to make an unambiguous (or at least less ambiguous) choice among alternative explanations, it makes sense to institute a deadline for gathering such corroborative evidence.

All of these methods rank assignments based on their probabilistic likelihood [5]. The likelihood of an assignment is simply the sum of the likelihoods of its hypotheses. Hypothesis likelihood is calculated from the likelihoods of the possible combinations of states which support the hypothesis. State likelihood is passed causally during estimation, and is altered whenever multiple outcomes are possible.

VI. IMPLEMENTATION CHALLENGES AND BENEFITS

We have posed the question of how to incorporate environmental measurements into vehicle navigation. A consideration of previous efforts has identified three ingredients for success: explicit modeling of vehicle navigational uncertainty, consideration of multiple competing explanatory hypotheses, and a feature-based representation of the environment. We discussed several reasons for taking a feature-based approach, including the inability of cell-based methods to retain local information and the enhanced positional distinguishability provided by feature hierarchies. Two innovative aspects of the concurrent mapping and localization algorithm, a generalized treatment of feature modeling and state compatibility tracking using decision dependencies, were presented. Finally, an outline of the CM&L algorithm was presented.

There are several challenges remaining in the implementation of this algorithm. We are currently assessing CM&L performance in simulation. Software implementation poses, first of all, the challenge of capturing the considerable (yet, as we have seen, necessary) intrinsic complexity of the algorithm, while stripping away any extrinsic complexities. Proper tracking of the causal dependencies among states, hypotheses, and assignments is essential. Efficient pruning strategies are needed to

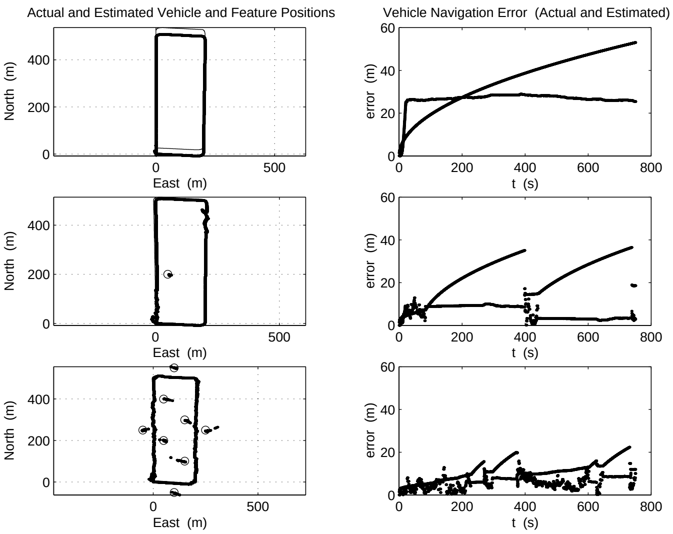

Figure 2. Simulation sample run showing the benefit of using environmental cues to reduce navigational error. In each of the three runs, the vehicle travels in a rectangle; the number of observable features changes in each run. The left graphs show the estimated (thick) and actual (thin) vehicle tracks and the estimated (points) and actual (circles) feature positions. The right graphs show estimated (upper) and actual vehicle navigation error. In the first run, only dead-reckoning is available for navigation information. In the second run, one feature is present. In the third run, eight features are present. The presence of features not only reduces the navigation error, but also improves the uncertainty estimate.

enable real-time performance capability [1].

Initial experimentation has been carried out with a partial implementation of CM&L. In this implementation, only stationary point features are considered, and data association ambiguity is removed. Figure 2 shows some interesting results from this simulation. The same commanded vehicle track is used for three runs during which a forward-looking sonar encounters, respectively, zero, one, and eight features. The right-hand figures display the estimated and actual vehicle navigation error (vehicle position covariance). There are two things to notice as the number of features is increased. First, navigation error is reduced. The dramatic effect of reacquiring a known feature is visible in the one-feature case. Second, the estimate of the navigation error is able to track the actual error more closely. This is due to the fundamentally inaccurate method of accounting for unknown dynamic and environmental forces using increased process noise. By incorporating information from a more accurate sensing model, errors and deficiencies in the vehicle dynamic model are overcome. These initial results provide tantalizing evidence that feature-based concurrent mapping and localization will provide an enabling capability for AUVs navigating in unknown environments.

ACKNOWLEDGEMENT

This research has been funded by the Office of Naval Research.

REFERENCES

[1] I. J. Cox and J. J. Leonard. Modeling a dynamic environment using a Bayesian multiple hypothesis approach. *Artificial Intelligence*, 66(2):311–344, April 1994.

[2] J. J. Leonard, B. A. Moran, I. J. Cox, and M. L. Miller. Underwater sonar data fusion using an efficient multiple hypothesis algorithm. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 2995–3002, May 1995.

[3] B. A. Moran. *Underwater Shape Reconstruction in Two Dimensions*. PhD thesis, Massachusetts Institute of Technology, 1994.

[4] P. Moutarlier and R. Chatila. An experimental system for incremental environment modeling by an autonomous mobile robot. In *1st International Symposium on Experimental Robotics*, Montreal, June 1989.

[5] D. B. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, AC-24(6), December 1979.

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

[7] W. K. Stewart. *Multisensor Modeling Underwater with Uncertain Information*. PhD thesis, Massachusetts Institute of Technology, 1988.

[8] S. T. Tuohy. *Geophysical Map Representation, Abstraction, and Interrogation for Underwater Vehicle Navigation*. PhD thesis, MIT, 1993.

[9] J. Uhlmann. *Dynamic Map Building and Localization: New Theoretical Foundations*. PhD thesis, University of Oxford, 1995.