Navigational Error Reduction of Underwater Vehicles with Selective Bathymetric SLAM

Roger A. Stuckey *

* *Defence Science and Technology Organisation, Eveleigh, NSW 2015, Australia (e-mail: Roger.Stuckey@dsto.defence.gov.au)*

Abstract: An approach to improve the navigational accuracy of underwater vehicles using patches of bathymetry acquired selectively from the terrain as observations within a Simultaneous Localisation and Mapping (SLAM) framework is presented. Each patch is filtered and stored as a gridded sub-map, or Navigation Cell, in the dynamic system model, along with the vehicle pose. Upon subsequent crossings over the same patch, additional cells are stored, then correlated against the original and the observations updated accordingly. The updated observations are fed into an Extended Kalman Filter (EKF), which updates the vehicle state. The approach has been demonstrated in simulation, using both synthetically generated bathymetry and real terrain data collected from a surface-vessel mounted micro-bathymetric sonar. It was found that the error build-up could be effectively bounded by employing the selective approach, whilst keeping the system state to a manageable size for real-time adoption. The locations of Navigation Cells can be both pre-defined and dynamically located at points where the terrain variability is high. In addition, any Navigation Cell can be chosen by the mission controller to be re-visited for correlation, should the vehicle's estimated state covariance exceed some pre-defined threshold. The paper concludes with some discussion of the advantages and disadvantages of the technique.

Keywords: Autonomous Vehicle, Localisation, Mapping, Terrain, Navigation, Mission Control, Micro-bathymetry.

1. INTRODUCTION

In most AUV systems, underwater navigation is achieved by dead-reckoning of the integrated velocities from the on-board IMU, starting at an initial pose with periodic sojourns to the surface to obtain GPS corrections. However, under certain circumstances, this may be undesirable or impractical, prompting the need for supplementary means of positioning while underwater. Furthermore, established procedures that involve the placement of underwater transponders represent an overhead which may be similarly unfavourable. In recent years, two localisation techniques which utilise other exteroceptive sensor information have become prominent: Terrain-Relative Navigation (TRN) (Nygren, 2005; Meduna et al., 2009) and; Concurrent or Simultaneous Localisation and Mapping (SLAM) (Smith et al., 1997; Williams, 2001). Both have their advantages and disadvantages. For TRN, the main disadvantage is the requirement for a bathymetric map to be known *a-priori* by the navigation system. Numerous variations of the SLAM technique exist, but for many of them, the main disadvantage lies in the necessity to manage a potentially large number of observations and effectively integrate them into the estimated system state (Bosse et al., 2004). More importantly, most featureless SLAM techniques, which use sonar-acquired bathymetry, have been developed with a primary focus toward building self-consistent maps in post-processing (Roman, 2005; Ruiz et al., 2004; Fairfield and Wettergreen, 2008; Barkby et al., 2011), and are not ideally suited for real-time employment on memory-limited hardware. The approach discussed in this paper draws from both techniques. The observations are represented by bathymetric grid-points within sub-maps, or Navigation Cells, at known locations relative to the vehicle and updated using a simple correlation method as with the basic TRN procedure. With the approach described herein, these observations, along with the vehicle pose, are stored directly in the system state for subsequent update by the SLAM algorithm. The Navigation Cells and consequently, the locations for application of the SLAM algorithm, are therefore chosen selectively and may be either pre-defined in the mission plan, or dynamically located according to the variability of the underlying terrain. Unlike the TRN approach, the gridded observations can be updated, along with the vehicle position, on each successive pass. Using this selective approach, the number of observations and hence, the size of the system state, can be kept to a manageable size.

2. THE DYNAMIC SYSTEM MODEL

In order to perform simultaneous localisation and mapping corrections to the system, both the vehicle response and external observations must be modelled. For the vehicle, a simplified coefficient-based dynamical model of the lateral-directional subsystem was employed. The observations are represented by grid-points, which approximate the underlying terrain at each Navigation Cell.

2.1 The Vehicle Model

For many applications of SLAM in the underwater domain, the vehicle is represented by a constant velocity or constant acceleration kinematic model. An initial version of this work was developed using the same representation, but the behaviour of the model was found to lack the short-period response typical of an AUV working at survey speed (Sgarioto, 2008), say 2.5 to 4.5 knots. This work is based on a simplification of the full 6 degree-of-freedom model of the REMUS 100 AUV first published by Prestero (2001) and extended by Sgarioto (2007) incorporating PID controllers for yaw, pitch and thrust. Since we are interested in estimating the vehicle's behaviour in the x–y plane, only the lateral-directional components are modelled. The dynamical subsystem can be expressed in state-space (Healey and Lienard, 1993) as:

$$\dot{\mathbf{x}}_V(t) = \mathbf{f}(\mathbf{x}_V(t), \delta(t)) \tag{1}$$

$$\begin{bmatrix} \dot{x}_V \\ \dot{y}_V \\ \dot{\psi}_V \\ \dot{v}_V \\ \dot{r}_V \end{bmatrix} = \begin{bmatrix} u_{V_0} \cos\psi_V - v_V \sin\psi_V \\ u_{V_0} \sin\psi_V + v_V \cos\psi_V \\ r_V \\ A_{11} v_V + A_{12} r_V \\ A_{21} v_V + A_{22} r_V \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ B_1 \delta_r \\ B_2 \delta_r \end{bmatrix} \tag{2}$$

Here, the subscript V denotes the vehicle component of the system. The vehicle state vector $\mathbf{x}_V(t)$ is composed of the vehicle coordinates x_V and y_V, the azimuth angle ψ_V and the lateral velocity and yaw rate, v_V and r_V, respectively. The control vector $\delta(t)$ is composed of a single lateral control fin displacement term δ_r. The term u_{V_0} represents the trimmed forward velocity and the constant terms A_{ij}, B_i and a, below, are functions of the vehicle hydrodynamic coefficients Y and N, the inertial tensor I and mass m:

$$\mathbf{A} = \frac{1}{a} \begin{bmatrix} I_{zz} - N_{\dot{r}} & Y_{\dot{r}} \\ N_{\dot{v}} & m - Y_{\dot{v}} \end{bmatrix} \begin{bmatrix} Y_v & Y_r - m u_{V_0} \\ N_v & N_r \end{bmatrix} \tag{3}$$

$$\mathbf{B} = \frac{1}{a} \begin{bmatrix} I_{zz} - N_{\dot{r}} & Y_{\dot{r}} \\ N_{\dot{v}} & m - Y_{\dot{v}} \end{bmatrix} \begin{bmatrix} Y_{\delta r} \\ N_{\delta r} \end{bmatrix} \tag{4}$$

$$a = (m - Y_{\dot{v}})(I_{zz} - N_{\dot{r}}) - N_{\dot{v}} Y_{\dot{r}} \tag{5}$$

In discrete form, the predicted vehicle state at time step k can be written in terms of the state at $[k-1]$ and the sample period Δt:

$$\mathbf{x}_V[k] = \mathbf{f}(\mathbf{x}_V[k-1], \delta[k-1]) \tag{6}$$

$$\begin{aligned} x_V[k] &= x_V[k-1] + F_{23}[k-1]\Delta t \\ y_V[k] &= y_V[k-1] - F_{13}[k-1]\Delta t \\ \psi_V[k] &= \psi_V[k-1] + r_V[k-1]\Delta t \\ v_V[k] &= v_V[k-1] + A_{11} v_V[k-1]\Delta t \\ &\quad + A_{12} r_V[k-1]\Delta t + B_1 \delta_r[k-1]\Delta t \\ r_V[k] &= r_V[k-1] + A_{21} v_V[k-1]\Delta t \\ &\quad + A_{22} r_V[k-1]\Delta t + B_2 \delta_r[k-1]\Delta t \end{aligned} \tag{7}$$

Here, the parameters are:

$$\begin{aligned} F_{13}[k] &= -u_{V_0} \sin\psi_V[k] - v_V[k] \cos\psi_V[k] \\ F_{23}[k] &= u_{V_0} \cos\psi_V[k] - v_V[k] \sin\psi_V[k] \end{aligned}$$

In order to incorporate the vehicle model into an Extended Kalman Filter, the vehicle system Jacobians need to be recalculated at each step. For the vehicle state and control components, the Jacobians $\nabla_V \mathbf{f}[k]$ and $\nabla_\delta \mathbf{f}[k]$ of $\mathbf{f}$ evaluated at $[k-1]$ are derived:

$$\nabla_V \mathbf{f}[k] = \mathbf{I}(5) + \mathbf{F}[k-1]\Delta t \tag{8}$$

$$\nabla_\delta \mathbf{f}[k] = [\, 0 \;\; 0 \;\; 0 \;\; \Delta t(B_1) \;\; \Delta t(B_2) \,]^T \tag{9}$$

where $\mathbf{I}(5)$ is the 5×5 identity matrix and:

$$\mathbf{F}[k] = \begin{bmatrix} 0 & 0 & F_{13}[k] & F_{14}[k] & 0 \\ 0 & 0 & F_{23}[k] & F_{24}[k] & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & A_{11} & A_{12} \\ 0 & 0 & 0 & A_{21} & A_{22} \end{bmatrix} \tag{10}$$

The remaining parameters are:

$$\begin{aligned} F_{14}[k] &= -\sin\psi_V[k] \\ F_{24}[k] &= \cos\psi_V[k] \end{aligned}$$

2.2 The Observation Model

Visual SLAM relies on the ability of the image processing component to extract and uniquely identify features within each snapshot of the terrain. Terrain-Relative Navigation obviates this process, although it has its own set of challenges. The main drawback in relying on the 2D bathymetric sonar for navigation is that every unique observation, or sonar return, is made only once as the vehicle passes. As a consequence, the vehicle must pass the same region multiple times in order to make successive updates to the original observations and effectively "close the loop". The other restriction is that patches of terrain are required to be approximated from a sequence of 2D scans in order to conduct a correlation of the terrain in both x and y coordinates (Roman, 2005). This means the observations can only be integrated into the system after the terrain patch is complete and the vehicle has left the Navigation Cell region. At the point when this occurs, the observations have an x and y offset from the vehicle location. In the formulation presented, the observations being referred to are actually the points of a gridded approximation to the terrain patch. This representation was chosen primarily because: (i) It allows for the observations to be directly incorporated into the filter, thereby facilitating correction to the vehicle's map upon each crossing and; (ii) It provides a very simple approximation which is computationally straightforward to interpolate and subsequently correlate. The advantages and disadvantages are further discussed in Section 5. Thus, the observation model $\mathbf{h}$ is posed in terms of the relative position of grid-points, $z_{x,y}$:

$$\mathbf{z}[k] = \mathbf{h}(\mathbf{x}[k]) \tag{11}$$

$$\begin{bmatrix} z_x[k] \\ z_y[k] \end{bmatrix} = \begin{bmatrix} x_V[k] - x_i[k] \\ y_V[k] - y_i[k] \end{bmatrix} \tag{12}$$

Here, x_i, y_i are the grid-point coordinates. Recall that we are only interested in estimating the vehicle's behaviour in the x–y plane, so the vehicle depth is not incorporated. The Jacobians of the vehicle and observation functions $\nabla_V \mathbf{h}[k]$ and $\nabla_i \mathbf{h}[k]$ are:

$$\nabla_V \mathbf{h}[k] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{13}$$

$$\nabla_i \mathbf{h}[k] = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \tag{14}$$

The addition of new observations requires an initialisation function $\mathbf{g}$, which provides an estimate of the absolute grid-point positions, based on the vehicle state and the relative grid-point positions:

$$\mathbf{x}_i[k] = \mathbf{g}(\mathbf{x}_V[k], \mathbf{z}[k]) \tag{15}$$

$$\begin{bmatrix} x_i[k] \\ y_i[k] \end{bmatrix} = \begin{bmatrix} x_V[k] - z_x[k] \\ y_V[k] - z_y[k] \end{bmatrix} \tag{16}$$

The Jacobians of the initialisation functions $\nabla_V \mathbf{g}[k]$ and $\nabla_i \mathbf{g}[k]$ are identical to their observation counterparts.

3. SYSTEM DEFINITION

For integration into the SLAM formulation, the complete system state is composed of the vehicle state (6), plus a set of observation states (11):

$$\mathbf{x}^T[k] = \begin{bmatrix} \mathbf{x}_V{}^T[k] & \mathbf{z}_1{}^T[k] & \mathbf{z}_2{}^T[k] & \cdots & \mathbf{z}_O{}^T[k] \end{bmatrix} \tag{17}$$

For O observation points.

3.1 SLAM Formulation

The model of the vehicle is written in discrete form as:

$$\mathbf{x}_V[k] = \mathbf{f}(\mathbf{x}_V[k-1], \delta[k-1]) + \mathbf{v}_V[k] \tag{18}$$

Where the state and control terms are defined above and the term $\mathbf{v}_V[k]$ represents the system noise and can also accommodate uncertainties in the model itself. The state estimate and control input, are used to generate a prior state estimate. When observations are received by the filter they are fused to produce the posterior estimate. The prior estimate of the vehicle state is expressed as the model evolution of the previous vehicle state estimate and the current control input:

$$\hat{\mathbf{x}}_V^-[k] = \hat{\mathbf{x}}_V[k|k-1] = \mathbf{f}(\hat{\mathbf{x}}_V^+[k-1], \delta[k-1]) \tag{19}$$

The grid-points (landmarks) derived from the terrain are assumed to be static, so the terrain model is simple:

$$\mathbf{x}_i[k] = \mathbf{x}_i[k-1] \tag{20}$$

So is the prior estimate of each grid-point:

$$\hat{\mathbf{x}}_i^-[k] = \hat{\mathbf{x}}_i[k|k-1] = \hat{\mathbf{x}}_i^+[k-1] \tag{21}$$

The sensor (observation) model is expressed as a function of the system state:

$$\mathbf{z}[k] = \mathbf{h}(\mathbf{x}[k]) + \mathbf{w}[k] \tag{22}$$

Here, the observation noise is characterised by the term $\mathbf{w}[k]$. The prior estimate of the observations follows:

$$\hat{\mathbf{z}}^-[k] = \hat{\mathbf{z}}[k|k-1] = \mathbf{h}(\hat{\mathbf{x}}^-[k]) \tag{23}$$

The state and observation noise are uncorrelated with zero mean and have covariances:

$$E[\mathbf{v}[k]\mathbf{v}^T[k]] = \mathbf{Q}[k] \tag{24}$$

$$E[\mathbf{w}[k]\mathbf{w}^T[k]] = \mathbf{R}[k] \tag{25}$$

Similarly, uncertainty in the control inputs can be prescribed:

$$E[\delta[k]\delta^T[k]] = \mathbf{\Delta}[k] \tag{26}$$

The Extended Kalman Filter linearises the propagation of uncertainty about the current state estimate in order to predict the state covariance:

$$\begin{aligned} \mathbf{P}^-[k] = {} & \nabla_\mathbf{x}\mathbf{f}[k]\mathbf{P}^+[k-1]\nabla_\mathbf{x}\mathbf{f}^T[k] \\ & + \nabla_\delta\mathbf{f}[k]\mathbf{\Delta}[k]\nabla_\delta\mathbf{f}^T[k] + \mathbf{Q}[k] \end{aligned} \tag{27}$$

The innovation covariance can then be computed from the state covariance prediction:

$$\mathbf{S}[k] = \nabla_\mathbf{x}\mathbf{h}[k]\mathbf{P}^-[k]\nabla_\mathbf{x}\mathbf{h}^T[k] + \mathbf{R}[k] \tag{28}$$

The innovations measure the difference between actual observations and those predicted:

$$\upsilon[k] = \mathbf{z}[k] - \hat{\mathbf{z}}^-[k] \tag{29}$$

Finally, the posterior estimate of both state and covariance can be determined through an update of the Kalman Filter:

$$\hat{\mathbf{x}}^+[k] = \hat{\mathbf{x}}^-[k] + \mathbf{W}[k]\nu[k] \tag{30}$$

$$\mathbf{P}^+[k] = \mathbf{P}^-[k] - \mathbf{W}[k]\mathbf{S}[k]\mathbf{W}^T[k] \tag{31}$$

Where the Kalman Gain is:

$$\mathbf{W}[k] = \mathbf{P}^-[k]\nabla_\mathbf{x}\mathbf{h}^T[k]\mathbf{S}^{-1}[k] \tag{32}$$

When new features added to the state, its covariance matrix expands:

$$\mathbf{P}^+[k] = \nabla_\mathbf{x}\mathbf{g}[k]\mathbf{P}^*[k]\nabla_\mathbf{x}\mathbf{g}^T[k] \tag{33}$$

Where the prior state covariance matrix $\mathbf{P}^*[k]$ has been augmented with the new observation covariance, $\mathbf{R}[k]$.

3.2 Navigation Cell Correlation

As with visual sensors such as still and video cameras, the micro-bathymetric sonar has the potential to collect a vast amount of data, in the millions. The sonar returns are susceptible to artifacts and shadowing, but they have many advantages over images, including the measurement of actual seafloor depth derived from range and beam angle as well as penetration through the water medium in areas of low visibility. In order to integrate the bathymetric observations into a filter, an approach is necessary to cope with the large amount of data. One such approach, which utilises relative vehicle pose measurements in a delayed state form of the Kalman Filter, has made significant progress in reducing the computational burden (Eustice et al., 2004; Leonard et al., 2002). An alternative approach is to reduce the data into another form, more suitable for inclusion in the system state. For this research, it was decided to employ the latter approach, due to its simplicity, and its ability to update the terrain model upon each successive pass. The specific terrain matching algorithm selected follows the correlation method used by Nygren (2005). In that project, the terrain observations were reduced to a grid and then compared to a prior map of the region, in order to find the vehicle's absolute position on the map. Here we are performing essentially the same correlation, but relative to a grid constructed from previous observations. First, the patch of sonar data - comprised of a sequence of scans, each scan consisting of a set of (range, beam angle) sonar returns - is transformed into Cartesian space and then approximated at points on a uniform grid. This grid must have the same spacing and orientation as the one already stored to facilitate correlation operations between the terrain models as described next. Once the grid has been constructed, it is compared with the previous version using a sum of squared differences in the spatial domain, by incrementing the x and y offset by one grid spacing at a time. For each increment, the correlation function is calculated and when complete, the true offset is determined from the minimum correlation sum, or that which resulted in the best match. The correlation sum must be weighted, since the number of data points for each offset may not be the same and an offset corresponding to a large overlap should be favoured over that with a smaller overlap, but the same unweighted correlation sum. It is also judicious to restrict the range of offsets examined to minimise the possibility of false association with a small number of grid-points. The correlation function utilised has the form:

$$\frac{1}{(\Delta M \times \Delta N)^2} \sum_{m=1}^{\Delta M} \sum_{n=1}^{\Delta N} (d_{m,n}[k_0] - d_{m+\Delta m, n+\Delta n}[k_1])^2 \tag{34}$$

For $1 \le \Delta M \le M$ and $1 \le \Delta N \le N$ overlapping grid-points in x and y axes respectively, (z_x, z_y), where the total number of observations $O = M \times N$. The terms $d_{m,n}[k_0]$ and $d_{m+\Delta m,n+\Delta n}[k_1]$ are the vehicle-relative terrain depths for the grid-points at time-steps k_0 and k_1 respectively. The Navigation Cell offsets for each grid-point are $0 \le \Delta m < M$ and $0 \le \Delta n < N$. Here the weighting is equal to the inverse of the total number of overlapping points squared. Note that this initial formulation will only identify and correct for an offset translation and not an offset rotation; the vehicle's heading remains uncorrected. Once tested in the field, it is planned to further refine the program by employing an Iterative Closest Point (ICP) algorithm instead, which is capable of correlating both the position and heading.

3.3 Mission Control

For many AUV's, such as the REMUS-100, the mission plan is typically defined by a sequence of waypoints for the vehicle to traverse. For navigational correction using bathymetric measurements, there are fundamentally two stages required during a mission: the initial navigation or collection phase in which any number of Navigation Cells are compiled and; the observation or correction phase during which Navigation Cells are revisited and correlated against. It is possible to implement these stages in several ways:

(1) Pre-define the Navigation Cells in the mission plan at points through which the vehicle is programmed to pass through more than once.
(2) Pre-define Navigation Cells in the mission plan and if a navigational measure, the vehicle's positional uncertainty for example, exceeds some critical value, allow the vehicle to return to a cell, make a correction and then resume its mission.
(3) Have the vehicle monitor the terrain variability and upon exceeding some threshold, establish a Navigation Cell. Otherwise, proceed as for option (2).

The basic steps pertaining to the first process are illustrated schematically in Fig. 1.

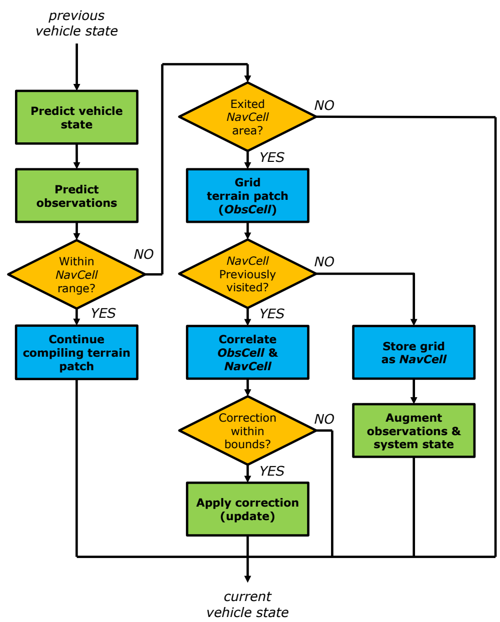

Fig. 1. Mission Control for Pre-defined Navigation Points

The vehicle's estimated proximity to the centre of every Navigation Cell (*NavCell*) is calculated at each step. When the vehicle enters an area surrounding any *NavCell*, relevant action is taken depending on whether it is the first pass or a subsequent one. The area is determined by the lateral range of the sonar at some nominal vehicle altitude. After the first pass, a sequence of sonar scans is compiled and filtered to produce a gridded approximation, the *NavCell*. The grid-points are stored as observations in the system state, along with the defining vehicle pose - that at the vehicle's point of exit from the *NavCell* area. After each successive pass, a similar but temporary grid, the *ObsCell*, is produced and aligned with the *NavCell*, representing observations within the system state. This alignment, determined by the correlation function, constitutes the data-association phase, in which each grid-point observation is either identified with its equivalent in the system state, or flagged as a new observation. Existing observations are then updated via equations (28) to (32) and new ones added with equation (33), effectively increasing the size of the *NavCell*, under the EKF mechanism. In the same step, the estimated vehicle pose is updated through the system state. This updated estimate can then be accessed by the vehicle controller in order for it to make corrections to its track.

4. SIMULATION RESULTS

The SLAM formulation developed in previous sections was applied to the simulation of a bottom-survey mission with sidescan sonar and possibly bathymetric sonar. In Defence, such surveys arise in the course of Mine Countermeasure (MCM) and Rapid Environmental Assessment (REA). In such applications, it is desirable that the vehicle gather data as rapidly and accurately as possible. The sonars perform optimally when the roll angle and yaw rate are close to zero and as such, it is important that the vehicle collects data from the terrain in straight, level motion. Typical mission profiles are therefore comprised of sets of parallel survey lines. Two approaches were demonstrated: Pre-defined Navigation Cells without on-the-fly mission modification by the vehicle and; vehicle-based dynamic establishment of Navigation Cells and subsequent selection of cells for revisiting and correction.

4.1 Pre-defined Navigation Cells

An area of terrain was synthetically generated for the vehicle to traverse and a vehicle mission path constructed in order to demonstrate the procedure. The sonar model characteristics approximated those of an Imagenex Model 837B "Delta T" 1.7MHz multibeam profiling sonar, with 120 beams over 120° at a frame rate of 10fps. Artificial Gaussian sensor noise with a variance of $1.0\times10^{-6}\text{m}^2$ was added to the data scanned from the bathymetry as well as the vehicle's inertial measurements in order to more closely mimic the real system. For the simulations, the vehicle operated at a constant speed of 2.5 knots and a constant depth, with a nominal altitude of 5.0m, which provided a swath width of approximately 17.0m.

The system was augmented with additional observations of heading and yaw rate, which were incorporated into the Kalman Filter to provide more accurate and representative estimate. In a real vehicle, these would typically be provided by the inertial navigation system. The accuracy associated with those measurements was relatively high: 0.01deg and 0.01degs^{-1}, respectively, resulting in a realistic drift of around 0.5% of distance travelled (Roman, 2005).

The vehicle path, pre-defined by waypoints, was constructed from parallel survey lines, each with a single perpendicular cross-line to ensure multiple crossings whilst achieving an acceptable rate of coverage. An example of the terrain, the mission waypoints and the vehicle path is illustrated in Fig. 2. For this mission, the vehicle was first directed to a point at the far end of the survey area and then to follow a lawnmower pattern back to the launch point, periodically crossing over the first transect. Those crossings define locations at which Navigation cells were scheduled to be constructed, the expectation being that *NavCells* are best created early in the mission, when the dead-reckoning error is lowest.

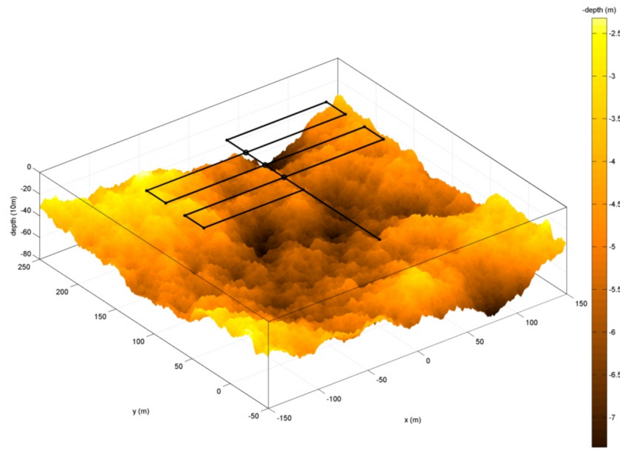

Fig. 2. Vehicle Mission Path and Synthetic Terrain

Another factor in determining the location of the *NavCells* is that it is desirable for them to be positioned along straight transects, so that the vehicle will not be turning when the data for that cell is collected. There are consequently three *NavCells* in this mission. In the correction phase, the *NavCells* can be approached from any angle. However, it is more advantageous to approach along a heading that will maximise the cell overlap - either perpendicular or preferably parallel to the initial track. For the mission presented, the *ObsCell* was created perpendicular to the original *NavCell*, which avoided causing any modification to the mission path. In this simulation, no correction was made on the first *NavCell* crossing; the resolution of the grid effectively limiting the minimum offset able to be applied. On the second crossing, a correction offset was applied, prompting the vehicle controller to regain the correct track. This is shown in Fig. 3, which displays the the actual and estimated vehicle track following correction. The three *NavCells* are depicted as 10×10 grids, incorporating a colormap of the terrain altitude underlying each.

A good indication of the error build-up or uncertainty in the vehicle's position, and one that can be utilised by the vehicle's mission control, is shown in Fig. 4, which graphs the estimated maximum standard deviation of the vehicle's position, $\sigma_{V\max}$, derived from the state covariance

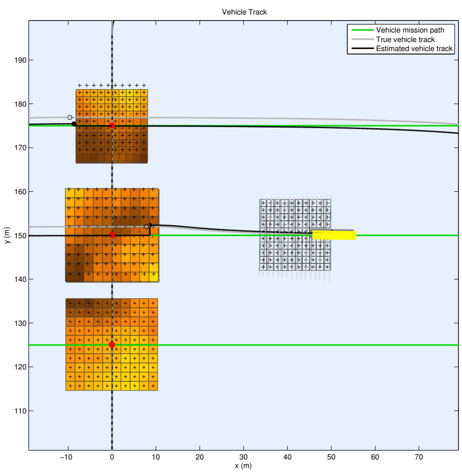

Fig. 3. Estimated and Actual (Corrected) Vehicle Track

matrix, $\mathbf{P}_V^+$, for identical simulations with and without Navigation Cells. In this case, we can see that the presence of *NavCells* has effectively bounded the uncertainty in the vehicle's position.

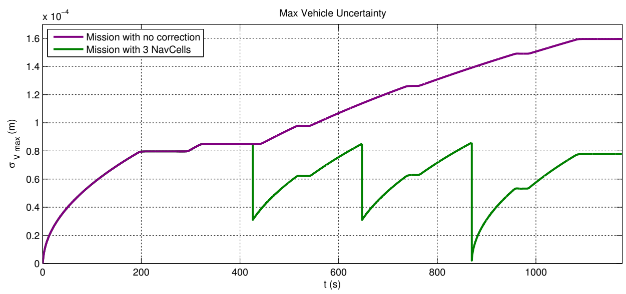

Fig. 4. Estimated Max. Uncertainty in Vehicle Position

4.2 Dynamically Established Navigation Cells

The terrain model employed for this simulation was created by interpolating real sonar data onto a triangular mesh. The data was collected from Iron Cove with the Imagenex multibeam profiling sonar detailed in the previous section. Iron Cove has a relatively benign seafloor, save a trail of sediment mounds along the piles of the overspanning bridge structure. Fig. 5 illustrates the bathymetry collected. The data was post-processed using the MB-System (Caress and Chayes, 1995) software to correct for roll bias and remove erroneous outer-beams. As the data was collected at a relatively shallow depth from a surface-vessel mounted sonar and not corrected for the true sound-speed profile of the water column, each scan has a subtle curve, rendering individual tracks visible in the gridded map. However, these were deemed to have little effect on the behaviour of the navigation algorithm. Likewise, there are a few "holes" in the map where no data was collected, which are simply ignored by the simulation model.

Again, a small amount of artificial sensor noise was added to both the bathymetric and the vehicle's inertial measurements. Two criteria were defined in order for the vehicle to make decisions regarding the creation of *NavCells* and

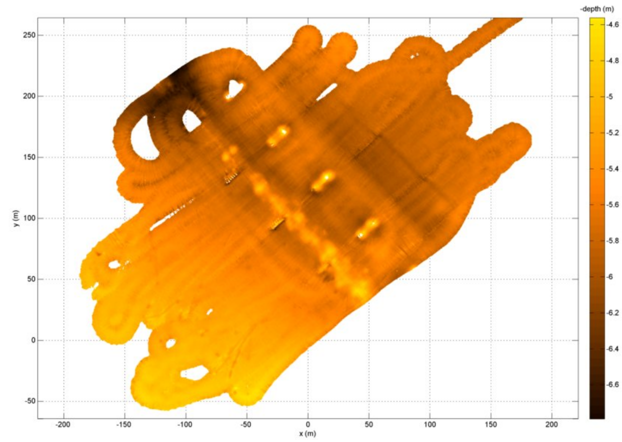

Fig. 5. Iron Cove Bathymetry

subsequently return to them for correction: A threshold of the mean terrain variance, $\bar{\sigma}_T^2$ and; a critical value for the estimated maximum standard deviation, $\sigma_{V\,\mathrm{max}}$. The mean terrain variance was calculated over both x and y directions.

Using a similar methodology for achieving localisation by executing maximally discriminating actions (Fairfield and Wettergreen, 2008), the variability of the terrain is continually assessed from the sonar returns and upon reaching its threshold, a *NavCell* is established. The estimated vehicle state covariance is also monitored continuously. If the latter exceeds a critical value, the vehicle is given an opportunity to return to one of the previously established *NavCells*. The decision whether to return at all and if so, to which *NavCell* could be based on many factors, such as the variability of each *NavCell* and their direct distance from the vehicle. Other factors, such as the orientation of each *NavCell* and the current direction of vehicle travel relative to each could also be considered. If the vehicle does choose to return to one of the *NavCells*, a set of new waypoints is inserted into the mission at the current stage and the vehicle is commanded to update its path, momentarily departing the original mission to attempt a correction. This is accomplished much in the same way that an AUV would normally surface to obtain a GPS correction and then resume its mission. First, a breakpoint is placed at the vehicle's current position in order to establish a new track. Then, two waypoints are inserted on either side of the *NavCell*, forming a path collinear to the original one. As discussed previously, this ensures that the vehicle is aligned with the path in which the *NavCell* was originally surveyed, thereby maximising the probability of a successful correlation. A fourth waypoint, placed at a location along the original mission path, either at or some small distance prior to the first breakpoint, is also necessary to avoid any gaps in the survey data. Consequently, the vehicle departs from the original mission to revisit the selected *NavCell*, perform a correlation and ideally, navigational correction, then return to the original track and resume its mission. Fig. 6 illustrates the augmented vehicle mission for one scenario. Aside from the set of mission waypoints, a scenario comprises various parameters for the vehicle and sensor model, as well as criteria relating to the mission, including the thresholds for terrain variance and vehicle uncertainty. For the same scenario, Fig. 7 shows two graphs of the local terrain variance and vehicle positional uncertainty (estimated maximum standard deviation).

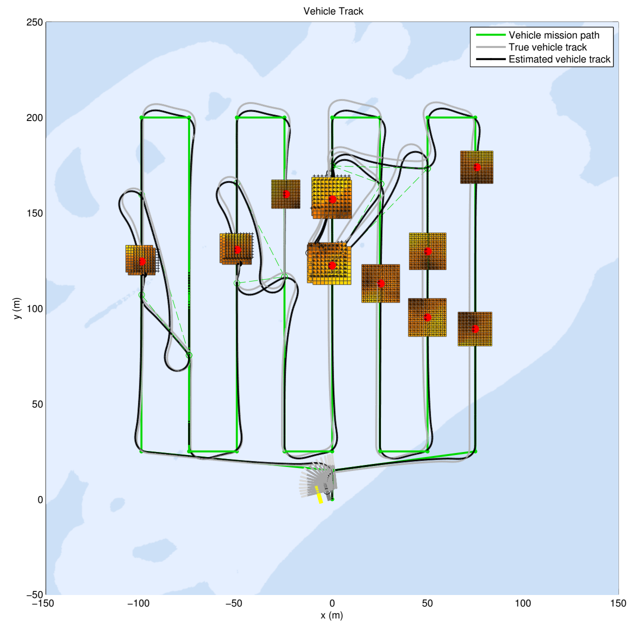

Fig. 6. Vehicle Simulation with Adaptive *NavCell* Creation

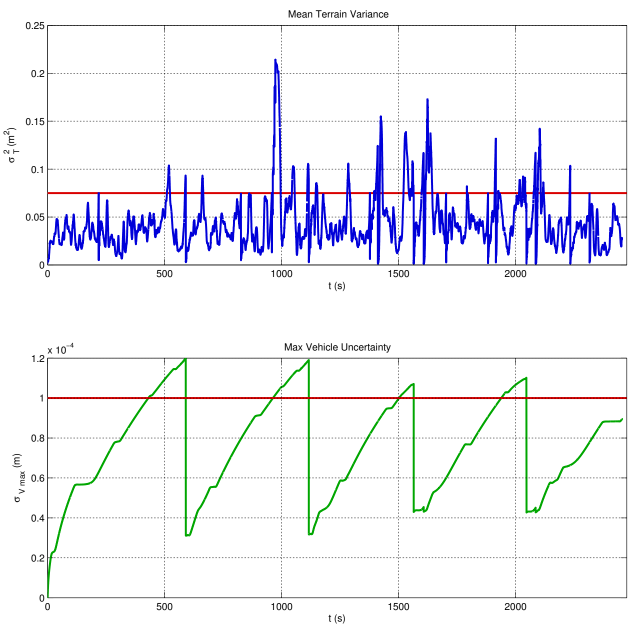

Fig. 7. Terrain Variance & Max. Vehicle Uncertainty for Adaptive *NavCell* Creation

The vehicle mission in Fig. 6 represents a simple lawn-mower pattern consisting of a series of parallel North-South legs progressing in an Easterly direction. Both the terrain variance and vehicle uncertainty are monitored throughout, although no *NavCells* are created and no modifications to the vehicle mission are made in the vicinity of the waypoints during which the vehicle is expected to be turning. As the vehicle traverses along its first parallel N-S leg, it encounters a region of highly varying terrain at around 250s. The local terrain variance exceeds the pre-defined threshold of 0.075m^2 and a *NavCell* is created. On the second parallel N-S leg, at around 420s, the vehicle's estimated maximum standard deviation exceeds the pre-defined threshold of $1.0{\times}10^{-4}$m, triggering a break from the mission to revisit a *NavCell* for positional correction. The vehicle, travelling South, then turns and travels back to the first and only *NavCell*. Upon crossing the *NavCell* at around 590s, a complementary *ObsCell* is created for correlation. The offset correction is applied to the vehicle's estimated position and it then turns once more and proceeds back to the breakpoint to resume its mission. This sequence of events occurs several times during the mission, with a total of 10 *NavCells* being created and 4 correction revisits executed. In this scenario, the mission controller was configured to return the vehicle to the *NavCell* with the largest variance each time. However, it would be prudent to prioritise the *NavCells* according to a weighted sum of the terrain variance and distance from the current location, since the further away each *NavCell*, the more error that would be accumulated in revisiting it. The resulting vehicle uncertainty seen in Fig. 7 follows the familiar sawtooth pattern, which is periodically recovered to a level below the threshold.

Other parameters associated with the scenario include a "cooldown" time, which is reset after each positional correction, in order to prevent the vehicle going into an infinite mission loop. For example, if the error buildup during return from a *NavCell* is sufficient to exceed the threshold of vehicle uncertainty, which would normally trigger another revisit, a cooldown period is necessary to suppress this behaviour. Each time a correction is required, a pre-defined number of upcoming legs in the mission plan are also checked for potential crossings of any existing *NavCells*. There is less justification in creating a revisit detour, if the vehicle is soon due to cross a *NavCell* during the normal progression of its mission.

Understandably, the success of the approach is dependent on the nature of the terrain. Some tuning of the grid size and resolution is required in order to uniquely capture patches of bathymetry for storage as *NavCells*. The grid size is generally determined by the lateral width of the sonar swath at its current depth, less any outer beams deemed unreliable. The grid resolution is more difficult to set, representing a compromise between the minimum offset correction able to be applied and the acceptable size of the system state. However, the main advantage with this approach is the fact that the *NavCells* and therefore, the number of observations, are added to the system parsimoniously - the operator can have a great deal of control over the size of the system state - so it is relatively simple to maximise the effectiveness of the procedure within the bounds of the available computing memory and processing speed.

5. CONCLUSION

A procedure has been developed for the correction of an AUV's estimated position based on the selective establishment of terrain-based Navigation Cells along its mission path. The system equations have been formulated within a SLAM framework to make cumulative corrections to both the vehicle's position and the Navigation Cells compiled. These equations were then implemented in a simulation to demonstrate two possible variants of the approach: One using pre-defined Navigation Cells and; another which allowed the vehicle to dynamically establish Navigation Cells, based on the terrain variability and return to them, contingent on the dead-reckoning error.

The approach showed promise, particularly in areas where the terrain variability was sufficiently high. With a number of pre-defined Navigation Cells, the vehicle's estimated maximum position error was successfully reduced. With dynamically established Navigation Cells, it was possible to limit the positional error.

In summary, the advantages of such an approach in comparison to previous approaches based on Terrain-Relative Navigation and feature-based SLAM are:

- No prior terrain maps are required, although the vehicle's map, once constructed, could be re-used
- More amenable to real-time execution
- Obviates need for feature extraction and subsequent data-association; grid-points are associated through correlation
- Navigation Cells are mutable (in the x–y plane) and updated at each crossing
- Reduced grids are simple to implement and require less storage
- Grid resolution can be readily adjusted to optimise performance & storage

Disadvantages include:

- Local knowledge may be required to devise effective locations for Navigation Cells, if they are to be predefined in the mission plan
- The terrain within Navigation Cells must be sufficiently varying and unique for correlation
- Interpolated grids infer a loss of information from the raw sonar data from which they were constructed
- The vehicle must pass sufficiently close to each Navigation Cell in order to perform correlation

Work is currently underway to implement the algorithm on the secondary controller of a REMUS-100 AUV. When complete, experiments are planned to assess the sensitivity of the mission-critical parameters to the local bathymetry. Future work will also explore different representations of the *NavCells*, including unstructured grids, as well as various algorithms for the correlation phase.

ACKNOWLEDGEMENTS

The author would like to thank the staff at the Center for Autonomous Vehicle Research, Naval Postgraduate School, Monterey, CA, where this project was instantiated. He also acknowledges the work of Suchet Bargoti from the University of Sydney, in implementing an initial draft of the adaptive mission component of the simulation.

REFERENCES

Barkby, S., Williams, S.B., Pizarro, O., and Jakuba, M.V. (2011). A featureless approach to efficient bathymetric slam using distributed particle mapping. *Journal of Field Robotics*, 28(1), 19–39.

Bosse, M., Newman, P., Leonard, J., and Teller, S. (2004). Simultaneous localization and map building in large-scale cyclic environments using the atlas framework. *International Journal of Robotics Research*, 23(12), 1113–39.

Caress, D. and Chayes, D. (1995). New software for processing sidescan data from sidescan-capable multibeam sonars. In *OCEANS '95. MTS/IEEE. Challenges of Our Changing Global Environment. Conference Proceedings.*, volume 2, 997 –1000 vol.2. doi: 10.1109/OCEANS.1995.528558.

Eustice, R., Pizarro, O., and Singh, H. (2004). Visually augmented navigation in an unstructured environment using a delayed state history. In *IEEE International Conference on Robotics and Automation. New Orleans, LA.*

Fairfield, N. and Wettergreen, D. (2008). Active localization on the ocean floor with multibeam sonar. In *OCEANS 2008. Quebec City, QC, Canada. 15-18 Sept. 2008.*

Healey, A.J. and Lienard, D. (1993). Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Journal of Oceanic Engineering*, 18(3), 327–39.

Leonard, J.J., Rikoski, R.J., Newman, P.M., and Bosse, M. (2002). Mapping partially observable features from multiple uncertain vantage points. *International Journal of Robotics Research, vol.21, no.10-11, Oct.-Nov. 2002, 943-75.*, 21, 943–75.

Meduna, D.K., Rock, S.M., and McEwen, R. (2009). Auv terrain relative navigation using coarse maps. In *Unmanned Untethered Submersible Technology Conference.*

Nygren, I. (2005). *Terrain Navigation for Underwater Vehicles.* Ph.D. thesis, Royal Institute of Technology.

Prestero, T. (2001). *Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle.* Ph.D. thesis, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution.

Roman, C. (2005). *Self Consistent Bathymetric Mapping from Robotic Vehicles in the Deep Ocean.* Ph.D. thesis, Massachusetts Institute of Technology.

Ruiz, I.T., de Raucourt, S., Petillot, Y., and Lane, D.M. (2004). Concurrent mapping and localization using sidescan sonar. *IEEE Journal of Oceanic Engineering*, 29(2), 442–56.

Sgarioto, D.E. (2007). Control system design and development for the remus autonomous underwater vehicle. Technical Report 240, Defence Technology Agency.

Sgarioto, D.E. (2008). Steady state trim and open loop stability analysis for the remus autonomous underwater vehicle. Technical Report 254, Defence Technology Agency.

Smith, C.M., Leonard, J.J., Bennett, A.A., and Shaw, C. (1997). Feature-based concurrent mapping and localization for auvs. In *Oceans '97. MTS/IEEE Conference Proceedings. Halifax, NS, Canada. Marine Technol. Soc.. OES. IEEE. MT&T. 6-9 Oct. 1997.*

Williams, S.B. (2001). *Efficient Solutions to Autonomous Mapping and Navigation Problems.* Ph.D. thesis, The University of Sydney.