23

6-DoF Navigation Systems for Autonomous Underwater Vehicles

Andrew Lammas, Karl Sammut and Fangpo He
Flinders University
Australia

1. Introduction

Navigation is one of the most critical factors in determining the operational suitability of any unmanned vehicle for its designated environment. In fully autonomous vehicles, due to the lack of a human operator to perform the navigation task, there is a fundamental requirement to incorporate estimation techniques that can provide the desired information necessary for navigation. Such information includes position, attitude, and velocity of the vehicle. In most vehicles, estimating this information is a relatively straight forward process due to the information available from the measurement sensors, particularly position information from Global Positioning System (GPS) sensors.

This chapter will focus on the more difficult problem of navigation, particularly positioning, where GPS reception is limited or non-existent. One example where GPS is only partially available is the case of an Autonomous Underwater Vehicle (AUV) where the vehicle is forced to use dead-reckoning in between GPS sightings in order to navigate accurately. Though an underwater environment is used within the context of this example, the techniques examined can be equally applied anywhere where GPS reception is compromised by other electromagnetically opaque mediums, satellite availability or even deliberate denial.

When using dead reckoning, due to the integrative nature of the position estimate, the error of the position estimate will grow unbounded. The growth in this error is caused by errors in the velocity and attitude estimates which are in turn affected by the accuracy of the navigational filter and the accuracies of the measurements observing these states. The accuracy of the measurement sensors is typically related to cost, thus some of the effects can be negated through more precise (expensive) equipment. For a given cost the single biggest factor affecting the accuracy of the navigational estimate is the navigation filter itself.

There have been numerous papers on the accuracy of various algorithms employed in the generic problem of state estimation (Arulampalam et al., 2001). Research pertaining to navigation algorithms for underwater vehicles have almost exclusively concentrated on ways to improve positioning accuracy below the surface through the use of ranging information other than GPS (Caiti et al., 2004). The methodology most prevalent in the literature is Simultaneous Localization And Mapping (SLAM) (Choset & Nagatani, 2001; Dissanayake et al., 2001)where information used for mapping, such as sonar 'pings', is fed

back into localization estimate as the map is being built. This process correlates the position of the vehicle with the map, thereby improving both positioning accuracy and map information.

When an AUV is in an environment that is devoid of any distinctive seabed features, SLAM may perform poor, as it is the motion of the AUV relative to these features that provides most of the information in the localization aspect of SLAM. In this context the navigation algorithm has to estimate the position of the vehicle solely using dead reckoning and thus the filter becomes the most significant factor in determining the accuracy of the position estimate. This brings out the critical question addressed in this chapter, which filters are best at minimizing the growth error in the position estimate using dead reckoning and which are best at reacquiring an accurate position estimate when GPS or other positioning information become available again after a period of dead reckoning.

The Bayesian filter algorithm, specifically the recursive Bayesian estimation, is chosen as the class of filters to be examined in this chapter for the following reasons. Bayesian filtering in general is advantageous in that the filter represents an estimate of a state's probability distribution function (pdf) rather than a particular estimate of a state (Ristic et al., 2004), and as such can inherently accommodate uncertain models and incorporate uncertainties of noisy measurements thus making the filter more robust. This robustness comes at the cost of increased computational demand due to the necessity of estimating the whole error distribution rather than a single solution. The recursive Bayesian filter algorithm is in comparison to its batch method equivalents more memory and computationally efficient since, due to its recursive nature, only information pertaining to the current estimate needs to be stored and processed. Recursive methods also have the capability of achieving higher accuracy since the estimate at a particular time is based on all the previous measurements as opposed to a fixed window of measurement, as used in batch methods. The computational efficiency, robustness to noise, and improved accuracy of the recursive Bayesian filter make it the ideal filter architecture for real-time state estimation for a navigation system.

The recursive Bayesian estimation algorithm in itself is, however, intractable due to the infinite number of solutions in a continuous solution space. In this chapter the filters that will be considered are all implementations that are based on the recursive Bayesian filter architecture, yet the algorithms are made tractable by applying simplifying assumptions or approximations to the pdf being estimated. These implementations consist of three main classes: Kalman Filters (KF), Particle Filters (PF), and Grid-based Filters (GF). The filters that are assessed in this chapter belong to the KF and PF classes since the GF class is more applicable for discrete or bounded solution spaces (Doucet et al., 2001).

This chapter aims to present a comprehensive analysis of a variety of KF and PF implementations by implementing these filters to estimate the navigational state of an AUV in a simulated environment. In the test-scenario, shown in Fig. 1, the simulated vehicle first manoeuvres on the surface before diving down to 50 meters to perform a raster scan covering an area of 500 m x 500 m and finally returning to the surface at the point where it had originally submerged. GPS position measurements are only available on the surface. This scenario has been chosen to compare the two most critical aspects of any navigational estimator, namely the error growth and integration of new observations after a period of absence. In the case of AUV navigation this increased estimation error can cause some filters to perform poorly or not at all when presented with new position information from the GPS that is inconsistent with their own estimate.

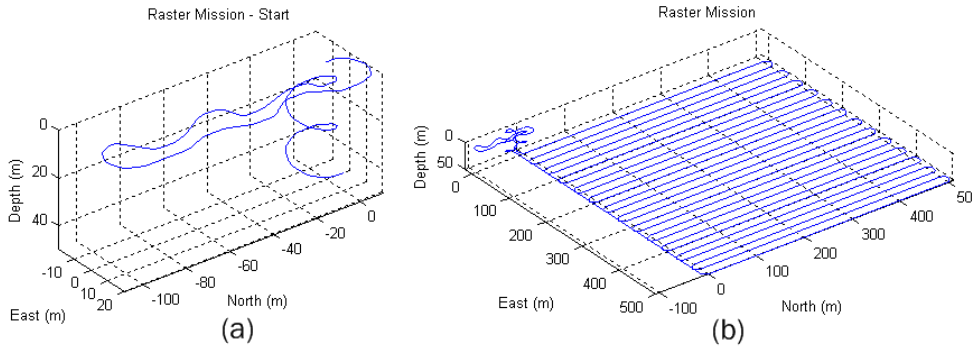


Fig. 1. Simulated raster-scan mission trajectory

This chapter will first cover the requirements of and limitations placed on a navigation system, and then review the concepts and algorithms used by the Bayesian filters in the particular context of the navigation estimation case. This will include:

- Requirements – what information is required of the navigation system by other systems (e.g., motion control systems) on the vehicle, and what limitations are placed on the system due to measurement and computational constraints.
- Dynamics – the kinematic and kinetic models used in defining the motion of a free moving rigid body vehicle, i.e., a vehicle with six degrees of freedom (6-DoF).
- Sensors – the instruments used to acquire the set of observation measurements from which the vehicle's state vector can be estimated, as well as the mathematical models that relate the sensor measurements to the states being estimated.
- Estimators – a representative set of KF and PF variants that have been used for vehicular navigation.

The chapter will then conclude with a comparison of the selected filters based on the aforementioned test scenario illustrated in Fig. 1, using the vehicle dynamics and sensor models provided.

2. Requirements

A navigation system is required to maintain a navigational solution at all times during a mission. A navigation solution, x, in (3) is an estimate of the pose (state) of the vehicle in (1). This estimate consists of translational, p^n, and rotational, Θ, position (i.e., position and attitude) coordinates η, in (2), and translational, v_o^b, and rotational, ω_{nb}^b, velocity coordinates, υ, in (2).

$$v_o^b = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \quad \omega_{nb}^b = \begin{bmatrix} p \\ q \\ r \end{bmatrix}, \quad p^n = \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix}, \quad \Theta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{1}$$

$$\upsilon = \begin{bmatrix} v_o^b \\ \omega_{nb}^b \end{bmatrix}, \qquad \eta = \begin{bmatrix} p^n \\ \Theta \end{bmatrix} \tag{2}$$

$$x = \begin{bmatrix} \upsilon \\ \eta \end{bmatrix} \tag{3}$$

This estimate is built on the stream of measurements being acquired by sensors that can be related to the states of interest. These measurements may be consistent or inconsistent over the duration of a mission. If the measurements are inconsistent then the navigation system maintains estimates of the states using knowledge of the dynamics of the system and the remaining measurements. This most commonly occurs while estimating position when GPS is unavailable, but can also affect velocity estimates when velocity measurements are unavailable due to the limited range of the acoustic based velocity sensors.
On an AUV there are a variety of power and physical size constraints on the vehicle due to mission requirements, and these constraints present themselves as computational restrictions on any embedded algorithms being used onboard, including the navigation system. Most of the vehicle power budget is highly prioritised for propulsion and control actuation, making the budget for computation limited and tightly constrained. This situation has been mitigated somewhat since the introduction of Li-ion battery technology in that the amount of power available on a vehicle has now increased significantly. There have also been significant improvements in the performance and efficiency of the computing platforms used, through advancements in computer architectures and silicon technology. Over the past few years several alternatives that could be used as suitable computational platforms for AUVs have been developed. These platforms include:
CPU – Central Processing Unit – the ubiquitous x86 based compact PCI 104 architecture which is currently the most commonly used architecture for mobile robotics.
DSP – Digital Signal Processors – microprocessor optimised for digital signal processing which constitutes the majority of calculations in a navigation system.
FPGA – Field Programmable Gate Arrays – reconfigurable integrated circuits enabling parallel architectures and optimal custom-built hardware implementation.
GP-GPU – General Purpose - Graphic Processor Units – essentially highly parallel data processing units which are suitable for digital signal processing and control applications.

3. Dynamics

To estimate the dynamic state of an AUV, first the equations that describe its motion must be defined and as such the dynamic navigation equations of a 6-DoF vehicle are presented. These equations are used to define the vehicle's kinematic and kinetic behaviour. They cover the co-ordinate frames used to represent the state information and the transformations that convert information represented in one frame to another frame. Rigid body dynamics and equations that model the hydrodynamic forces and moments acting on the vehicle are also included.

3.1 Reference Frames

Before reviewing the principles of navigation in underwater environments, certain reference frames need to be defined. A reference frame is a co-ordinate frame in which information can be represented. The frames that are used in underwater navigation are:

- ECEF – The Earth Centered Earth Fixed reference frame is a three dimensional Euclidean space with an x_e, y_e, z_e cartesian representation whose origin is at the centre of the earth

and is stationary in reference to the earth's surface. As illustrated in Fig. 2a, the x_e axis of this reference frame projects out of the earth's surface at 0^0N 0^0E, the y_e axis out at 0^0N 90^0E, and the z_e axis out at 90^0N the geographic north pole. This frame is used in long distance navigation over significant portions of the earth surface, as any point in the vicinity of the earth can be defined in this frame.

- ECI – The Earth Centered Inertial reference frame represents the true inertial frame in the confines of the earth. It is much like ECEF but does not rotate with the earth, rather the x_e and y_e axes point to specific points in the celestial sphere. Although the ECI frame is used for inertial measurements, in the case of slow moving vehicles and for short missions ECEF can be considered as inertial.
- NED – The North East Down reference frame is defined as a 'flat earth' frame in that it makes use of the assumption that for relatively small distances the surface of the earth can be considered to be flat. The origin of the NED is placed at some arbitrary point on the earth reference ellipsoid. The axes of this frame, as the name suggests, points north, east and down for axes x_n, y_n, and z_n, respectively, as shown in Fig. 2a. This frame is used when the flat earth assumption holds as it simplifies the equations in comparison to ECEF.
- BODY – The body-fixed frame is a reference frame which is fixed to the vehicle. The origin of the frame is located at a convenient position in relation to the vehicle usually the centre of gravity (or buoyancy in the case of underwater vehicles). The axes of this frame project out of the front (bow), right (starboard) and bottom (keel) of the vehicle for the x_b, y_b, and z_b axes, respectively, as presented in Fig. 2b. The body-fixed frame is used to define the dynamics of the vehicle as well as the measurements as many of the measurement sensors are attached to the vehicle.

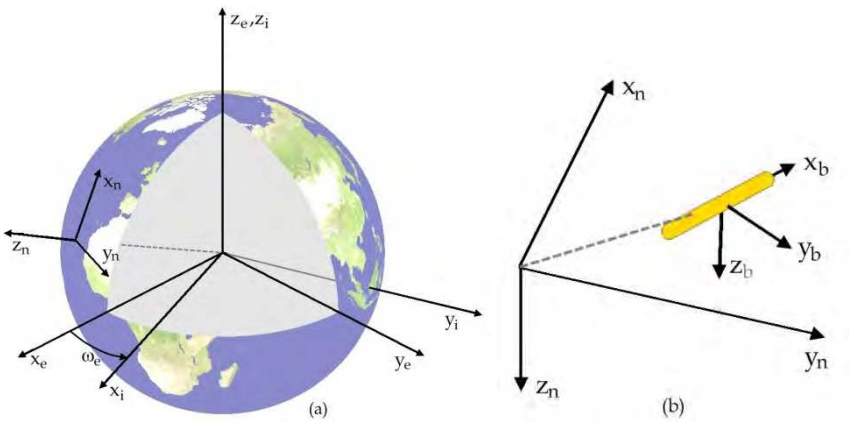


Fig. 2. Reference frames ECI, ECEF and NED (a), NED and BODY frame (b)

3.2 NED to Body-Fixed Frame Conversion

While it is convenient to represent the vehicle's velocities in the body-fixed frame, it is more convenient to represent the vehicle's position in the NED frame. The consequence of using two frames is the necessity of transforming information between the frames. In the case of 6 DoF dynamics of a free moving body, the velocity information represented in the body-fixed

frame, υ, must be rotated such that the same velocity information is represented in the NED frame, $\dot{\eta}$. To rotate the velocity information, each of the vectors, v_o^b and ω_{nb}^b, are applied to two rotational matrices, $R_b^n(\Theta)$ and $T_\Theta(\Theta)$, respectively.

3.2.1 Linear Velocity Transformation

The linear velocity transformation matrix $R_b^n(\Theta) \in \mathbb{R}^{3\times3}$ transforms the translational velocities defined in the body-fixed frame, v_o^b, into the velocities in the NED frame, $\dot{p}^n$, based on the rotational differences between the two frames represented using Euler angle notation Θ (4). $R_b^n(\Theta)$ consists of three rotations, one for each Euler angle (5), and is defined in (6) using the common convention zyx (Titterton & Weston, 2004), where v_o^b is rotated by ϕ, then by θ, and then by ψ. The order of rotations as shown in (6) is not arbitrary, due to the compounding effect of the rotational order.

$$\dot{p}^n = R_b^n(\Theta)v_o^b \tag{4}$$

$$R_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}, R_{y,\theta} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, R_{z,\psi} = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

$$R_b^n(\Theta) := R_{z,\psi}R_{y,\theta}R_{x,\phi} \tag{6}$$

$$R_b^n(\Theta) = \begin{bmatrix} \cos\psi\cos\theta & -\sin\psi\cos\phi + \cos\psi\sin\theta\sin\phi & \sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi \\ \sin\psi\cos\theta & \cos\psi\cos\theta + \sin\phi\sin\theta\sin\psi & -\cos\psi\sin\phi + \sin\theta\sin\psi\cos\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{7}$$

3.2.2 Angular Velocity Transformation

The angular velocity transformation matrix $T_\Theta(\Theta) \in \mathbb{R}^{3\times3}$ transforms the body-fixed rotational velocity information, ω_{nb}^b, into the NED frame, $\dot{\Theta}$, based on the rotational difference between the frames (8). Similar to (4), the transformation conforms to the zyx rotational ordering as shown in (9) for the same reasons given for the linear velocity transformation. For clarity, consider $T_\Theta^{-1}(\Theta)$, given in (10), in which $\dot{\psi}$ is rotated by $R_{y,\theta}$, added to $\dot{\theta}$, then the sum rotated by $R_{x,\phi}$ and added to $\dot{\phi}$, as described in (9). This effect is in contrast to (4) where the whole vector is rotated.

$$\dot{\Theta} = T_\Theta(\Theta)\omega_{nb}^b \tag{8}$$

$$\omega_{nb}^b = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_{x,\phi}^T \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_{x,\phi}^T R_{y,\theta}^T \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} := T_\Theta^{-1}(\Theta)\dot{\Theta} \tag{9}$$

$$T_\Theta^{-1}(\Theta) = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \cos\theta\sin\phi \\ 0 & -\sin\phi & \cos\theta\cos\phi \end{bmatrix} \Rightarrow T_\Theta(\Theta) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \tag{10}$$

3.2.4 Unit Quaternion

Unit quaternion is a four element replacement of the three element Euler angle representation. The main motivation for quaternions is to avoid singularities, commonly called 'gimbal lock', that can occur in the three element representation when θ=±90°. When the pitch attains these values, the rotational matrix $T_\Theta(\Theta)$ become undefined due to the trigonometric functions used in the rotation. A quaternion is a four element array that is described as a four dimensional vector in a complex space (Fossen, 2002), or alternatively a complex number (11) with one real and three complex components. Replacing the Euler infomation in (3) with (11) yields an alternative state vector, x (12). Using a complex space representation is advantageous due to the implied orthogonality of a complex number.

$$q = \eta_0 + \varepsilon_1 i + \varepsilon_2 j + \varepsilon_3 k = [\eta_0 \quad \varepsilon_1 \quad \varepsilon_2 \quad \varepsilon_3]^T \tag{11}$$

$$x = [v \quad p^n \quad \mathrm{q}]^T \tag{12}$$

A unit quaternion is essential for representing frame rotations as any non-unit magnitudes will scale the transformations rather than a pure rotational transformation. A unit quaternion must satisfy (13) and as such a normalization would consist of (14).

$$q^T q = 1 \tag{13}$$

$$\tilde{q} = \frac{q}{q^T q} \tag{14}$$

When applying the same principle to unit quaternions as to Euler angles in defining (4), (15) results. As evident in (15), an advantage of the unit quaternion representation is the absence of trigonometric relationships in the rotation matrix $R_b^n(\mathrm{q})$ thus simplifying the computation. When applying unit quaternion definition to (8), (16) is produced. Equation (16), similar to (15), produces a result that contains no trigonometeric functions and likewise simplifies the computation.

$$R_b^n(\mathrm{q}) = \begin{bmatrix} 1-2(\varepsilon_2^2+\varepsilon_3^2) & 2(\varepsilon_1\varepsilon_2-\varepsilon_3\eta_0) & 2(\varepsilon_1\varepsilon_3+\varepsilon_2\eta_0) \\ 2(\varepsilon_1\varepsilon_2+\varepsilon_3\eta_0) & 1-2(\varepsilon_1^2+\varepsilon_3^2) & 2(\varepsilon_2\varepsilon_3-\varepsilon_1\eta_0) \\ 2(\varepsilon_1\varepsilon_3-\varepsilon_2\eta_0) & 2(\varepsilon_2\varepsilon_3+\varepsilon_1\eta_0) & 1-2(\varepsilon_1^2+\varepsilon_2^2) \end{bmatrix} \tag{15}$$

$$T_{\mathrm{q}}(\mathrm{q}) = \begin{bmatrix} -\varepsilon_1 & -\varepsilon_2 & -\varepsilon_3 \\ \eta_0 & -\varepsilon_3 & \varepsilon_2 \\ \varepsilon_3 & \eta_0 & -\varepsilon_1 \\ -\varepsilon_2 & \varepsilon_1 & \eta_0 \end{bmatrix} \tag{16}$$

3.3 Kinematics

Kinematics are the equations of motion that relate the state of any body at one point in time to another point in time due the geometric nature of space and the motion of the object in that space. For a free body in three dimensional space, the 6-DoF equations of motion comprising the three translational and three rotational degrees of freedom (Titterton & Weston, 2004), are defined as (17).

$$\dot{\eta} = J(\eta)\upsilon \quad \Rightarrow \quad \begin{bmatrix} \dot{p}^n \\ \dot{\Theta} \end{bmatrix} = \begin{bmatrix} R_b^n(\Theta) & 0_{3\times3} \\ 0_{3\times3} & T_\Theta(\Theta) \end{bmatrix} \begin{bmatrix} v_o^b \\ \omega_{nb}^b \end{bmatrix} \tag{17}$$

Alternatively, applying (17) to a quaternion form (11) yields (18).

$$\dot{\eta} = J(\eta)\upsilon \quad \Rightarrow \quad \begin{bmatrix} \dot{p}^n \\ \dot{q} \end{bmatrix} = \begin{bmatrix} R_b^n(q) & 0_{3\times3} \\ 0_{3\times3} & T_\Theta(q) \end{bmatrix} \begin{bmatrix} v_o^b \\ \omega_{nb}^b \end{bmatrix} \tag{18}$$

Equation (17) or (18) can be used to define the motion of any freely moving object in three dimensional space as it does not consider how the motion is created. This results in (18) being valid for any mobile platform regardless of context.

3.4 Kinetics

Kinetics, now commonly known as analytical dynamics or just dynamics, can be described as a branch of mathematics that deals with the motion of bodies and the influences that create the motion, namely forces and torques. Kinetics, in contrast to kinematics, incorporates the causes of the motion as well as the motion itself.

The basic premise of dynamics is expressed in Newton's Second Law $F = ma$. Expanding Newton's Second Law to three dimensional space, as found in a freely moving body with mass M and force F_i exerted upon it, and using the principle of superposition of forces result in the generalized kinetic equation (19).

$$M\dot{\upsilon} = \sum_{i=1}^{n} F_i \tag{19}$$

The components of (19), M and F_i, are defined by two distinct effects, rigid body dynamics and hydrodynamics.

3.4.1 Rigid Body Dynamics

Rigid body dynamics apply Newtonian mechanics using only forces involved with the 'rigid' body of the vehicle and do not include forces produced by motion of the vehicle through the medium. In (20), the effect from the intrisic interia of the vehicle, M_{RB}, and the forces apparent due to the motion of the reference frame, namely the 'coriolis force', $C_{RB}(\upsilon)\upsilon$, and the control forces, τ_{RB}, are modelled.

$$M_{RB}\dot{\upsilon} + C_{RB}(\upsilon)\upsilon = \tau_{RB} \tag{20}$$

3.4.2 Vehicle Hydrodynamics

When describing the motion of a vehicle through a fluid, in this case an AUV through water, there is a generalized equation given in (21) which expands from (20) and parameterizes this motion (Fossen, 2002).

$$M\dot{\upsilon} + C(\upsilon)\upsilon + D(\upsilon)\upsilon + g(\eta) = \tau + w \tag{21}$$

where M and $C(\upsilon)\upsilon$ add the effect of 'added mass' (Fossen, 2002) to (20), i.e. $M = M_{RB} + M_A$ and $C(\upsilon) = C_{RB}(\upsilon) + C_A(\upsilon)$, while $D(\upsilon)\upsilon$ characterises hydrodynamic drag, $g(\eta)$ does the same for buoyancy and gravitational effects, and w models external disturbances.

3.4.3 Navigation Equations

Collating (21) and (17) or (18), a concise representation of the 6-DoF dynamics of an underwater vehicle, accurate to the second order, can be represented in (22).

$$\begin{aligned} \dot{\upsilon} &= M^{-1}\big(\tau - C(\upsilon)\upsilon - D(\upsilon)\upsilon - g(\eta)\big), \\ \dot{\eta} &= J(\eta)\upsilon \end{aligned} \tag{22}$$

4. Sensors

An AUV employs a variety of exteroreceptive and proprioceptive sensors to obtain information about the surrounding environment and the vehicle itself. This section will review some of the most common sensors that can provide information related to (3). The review will cover the information available from these sensors, limitations of these sensors and sources of noise. AUVs are particularly susceptible to limitations in sensors since the ranges and capabilities of the sensors are limited to what can be carried by the vehicle based on power, size, and cost constraints.

4.1 IMU

An Inertial Measurement Unit (IMU) is an orthogonal triad of accelerometers and gyroscopes that can detect the three translational accelerations and rotational velocities experienced by any free moving body in three dimensional space. Integrating this information yields position and attitude in a process called dead reckoning. These sensors are defined as inertial sensors as they use the inertia of an internal reference mass or structure to generate the measurements and, as a consequence, they are completely self-contained.

There are two main configurations, namely 'inertial-frame' IMUs and 'strapdown' IMUs. An inertial-frame IMU mechanically maintains its orientation using the gyroscopes, while the accelerometers stay in the same frame effectively mechanically integrating the rotational rates. Being mechanical, these systems are large and power hungry and thus unsuitable for small AUVs. In strapdown IMUs, all the sensors are attached to the vehicle and the information is computationally integrated. The advent of the microprocessor makes this integration a relatively trivial process.

4.1.1 Accelerometers

An accelerometer measures translational acceleration in the inertial frame of the sensor (vehicle). In principle, accelerometers measure acceleration by measuring the deflection of a mass attached to a spring. The most suitable technology for use in an IMU is MEMS accelerometers. MEMS accelerometers use either a pendulous mass or a resonant beam structure. In the pendulous mass architecture, as illustrated in Fig. 3a, a capacitance is generated that is related to the displacement of the proof mass. In the resonant beam

structure, variations in the resonant frequency of the beam due to acceleration induced loading are measured. These values are converted to voltages and subsequently converted to digital representations.

4.1.2 Gyroscopes

Gyroscopes are devices that can measure rotational velocity in the inertial frame. For use in small underwater vehicles, gyroscopes can be classified into two classes, optical based and MEMS technology based. Optical based technologies, such as ring-laser gyroscopes and fibre-optic gyros, all use the same principal. A structured light beam is first split and then sent in opposing directions around a circular path, resulting in a phase difference between the two beams whenever a rotational change occurs – this is known as 'the Sagnac effect'. The intensity of the recombined beams is related to the rotational rate. MEMS gyroscopes use a resonant beam or ring structure. A resonant structure will tend to vibrate in the same direction and any vibration not parallel or orthogonal to this axis is related to rotational motion – this is known as 'the coriolis effect' and is shown in Fig. 3b. This motion can be detected as a capacitance changes.

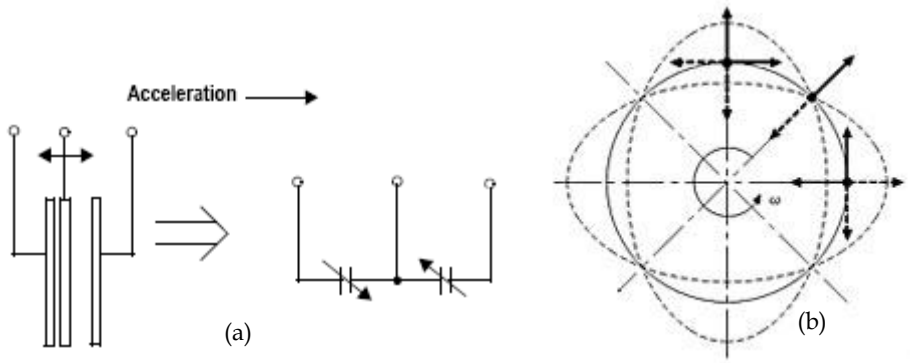


Fig. 3. MEMS Accelerometer (a), and Gyroscope (b) Structures

4.1.3 MEMS Sensors

While MEMS sensors are smaller, cheaper, and use much less power than their counterparts, they have one downside, i.e., reduced accuracy. The errors generated from the non-idealities in MEMS sensors include:

- Fixed-Bias (b_i) - a bias in the measurement due to offset in the zero of the sensor and/or A/D biases (m/s^2).
- Scale Factor (a_i) - a scaling constant which defines the relationship between the measured voltage or binary value and the perceived acceleration (m/s^2/V or m/s^2/LSB). This can include second or higher order terms.
- Cross Coupling (ϵ_i) - is a matrix that represents the sensitivity of a sensor to motion or fields orthogonal to its desired axis of sensitivity.
- Temperature based (c) - some of the errors mentioned previously, specifically bias and scale errors, can also be affected by the ambient temperature.
- Quiescent Noise (w) - is the noise that occurs when the instrument is sensing a zero state. This value is used to define the random error, *noise*, in the output signal.

Combining these effects to define a relationship between (3) and the measurement produces (23).

$$\beta_{INS} = \begin{bmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & a_3 \end{bmatrix} \begin{bmatrix} 1 & -\epsilon_\psi & \epsilon_\theta \\ \epsilon_\psi & 1 & -\epsilon_\phi \\ -\epsilon_\theta & \epsilon_\phi & 1 \end{bmatrix} \beta + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} T + w \tag{23}$$

where

$$\beta_{INS} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \text{ and } \beta = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \text{ for gyroscopes, and}$$

$$\beta_{INS} = \begin{bmatrix} \alpha_x \\ \alpha_y \\ \alpha_z \end{bmatrix} \text{ and } \beta = \begin{bmatrix} I_{3\times3} & \begin{matrix} 0 & -x_a^b & y_a^b \\ x_a^b & 0 & -z_a^b \\ -y_a^b & z_a^b & 0 \end{matrix} \end{bmatrix} \dot{v} \text{ for accelerometers}$$

Most modern MEMS sensors have factory calibration for many of these errors, however, some of the errors are not static but change very slowly over the lifetime of the device in a process known as 'ageing'. These residual errors can be accounted for using various offline or online parameter estimation techniques. Assuming the sensors are calibrated, (23) simplifies to (24).

$$\beta_{INS} = \beta + w \tag{24}$$

4.2 Compass

A compass is a sensor that measures the earth's magnetosphere. The core of most modern IC based magnetometers uses the 'Hall Effect' principle. The Hall Effect dictates that in presence of a magnetic field, the electron flow in a conductor will be distorted orthogonally to the current flow, and the magnetic field producing a voltage difference across the conductor that is proportional to the magnetic field.

MEMS based magnetometers have many of the same errors as MEMS IMU components such as scale, C_{sf}, and misalignment, C_m, but have two extra sources of error due to the interaction of the sensed magnetic field with nearby magnetic materials:

- Hard Iron Errors (δB^b) - The largest error induced in the magnetic field tends to be unwanted magnetic fields produced by surrounding electrical equipment or magnetic abnormalities in the environment. These hard iron errors produce a bias in the resulting measurement due to the summation of the desired and interfering magnetic fields.
- Soft Iron Errors (C_{si}) - When non-magnetized magnetic materials are exposed to an external magnetic field, they will distort the field in vicinity of the material.

Combining these errors to define a relationship between the measurement and (3) produces (25).

$$B^b = C_m C_{sf} C_{si} \left(R_b^{n^T}(\Theta) B^n + \delta B^b \right) + w \tag{25}$$

Most of the errors that occur in MEMS based magnetometers can be removed through calibration. The only effect that cannot be calibrated for are hard iron errors due to local magnetic abnormities which must be removed using online estimation (Gebre-Egziabher et

al., 2001) if the error is significant enough to be of concern. Once calibration or estimation has removed the above errors (25) simplifies to (26).

$$B^b = R_b^{nT}(\Theta)B^n + w \tag{26}$$

4.3 GPS

The Global Positioning System (GPS) is a global navigation satellite system (GNSS) which consists of an array of satellites that is used by a receiver to triangulate the receiver's position anywhere in the world depending on satellite availability (Tsui, 2005).

GPS operates on the principle that, given a common time reference, a range to a satellite can be calculated by the time difference between transmission from the satellite and reception by the receiver of a radio signal multiplied the speed of light. This measurement is called a pseudo-range (PR), as the measurement is a range calculated from a time difference rather than a direct range measurement. Using three satellite range measurements and positions, $p_{s_i}^e$, the three values that constitute a position in 3D space, p^e, can be solved using simultaneous equations (27).

A common time reference does not exist, however, as the receiver only has a quartz crystal oscillator whereas each of the satellites have atomic clocks that are accurate to a few nanoseconds being periodically corrected by the GPS control segment. This problem can be solved by including a receiver clock offset, δt_r, into the triangulation problem (27). Since four unknowns now exist, a minimum number of four satellites are required for a position fix (Tsui, 2005).

$$\begin{aligned} PR_1 &= \delta t_r + \left\| p^e - p_{s_1}^e \right\|, \\ PR_2 &= \delta t_r + \left\| p^e - p_{s_2}^e \right\|, \\ &\vdots \\ PR_n &= \delta t_r + \left\| p^e - p_{s_n}^e \right\|. \end{aligned} \tag{27}$$

The main sources of error in this system are:

- Clock Inaccuracies (δt_r) - as mentioned previously there exists a clock bias due to clock drift that is compensated for in the positioning algorithm. Remaining inaccuracies in the clock can create ranging errors of up to two meters.
- Satellite Geometry – the apparent angle between satellites as perceived by the receiver can have a significant effect on the triangulation problem. If the angles are small, any error in the range measurement from one of the satellites can significantly affect the resolved position with errors as large as 100 - 150 meters.
- Satellite Orbits – Even though GPS satellites are very accurately placed in orbit, errors in this orbit can still occur. 'Ephemeris data' transmitted within the GPS signal contains correction information, limiting the effect of orbit errors to less than two meters.
- Multipath Effect –The line of sight 'desired' signal between satellite and receiver can be corrupted by delayed multipath reflections of itself. Typically the error for multipath is no more than a few meters.
- Atmospheric Effects – The speed of the GPS signals are affected by their passage through the atmosphere, particularly the ionosphere which slows down electromagnetic waves proportional to $1/f^2$ (Tsui, 2005). This has motivated the use of dual frequency receivers

which receive a GPS signal on two frequencies. The amount that each signal is slowed by can be calculated using the time difference between the reception of the GPS signals on the two frequencies.
- Relativistic Effects – Due to relativistic effects, the atomic clock onboard the satellite runs slower than on earth and so is tuned before launch to account for these effects.
- Selective Availability (SA) – This refers to the inclusion of pseudo noise by the US government to restrict access to high quality position information. SA was removed in May 2000.

Although the addition of a second frequency can be used to compensate for ionspheric errors, it does nothing to compensate for some of the other effects or for ionospheric errors in single frequency receivers. In such cases, the use of systems, like WAAS or EGNOS (Tsui, 2005), can be used to provide coarse corrections for atmospheric, clock, and orbit related errors.

4.4 SONAR

SOund Navigation And Ranging (SONAR) is an acoustic method for determining the range and bearing of a target or of the sonar device itself, if multiple targets are at known locations.

There are two types of sonar: active and passive. In this chapter, the discussion will be restricted to active sonar systems as they are the most pertinent in relation to navigation.

Active sonar assumes that sound has a finite speed, c, in any medium, thus a range to a feature, p_t^n, can be determined if the time difference between transmission and reception, δt, can be measured. By using a transceiver to transmit a focused sound pulse (ping) and receive the return signal (echo), target information can be extracted from the amplitude, round trip time, and transceiver azimuth (28) of the echo.

$$\frac{1}{2}c\delta t = \|p^n - p_t^n\| \tag{28}$$

There are several varieties of active sonar systems, the most common are scanning sonar, side-scan sonar, and multibeam (bathymetric) sonar systems.

The errors in any sonar system are highly nonlinear, the most significant being:
- Refraction – the path of the sonar ping can become curved, changing the ensonified area, due to the non uniformity of the temperature and density of the water over the trajectory of the ping.
- Scattering – when the angle of incidence of the ping to the target is not ideal, the reflection is diffuse rather than specular, resulting in a poor return with the possibility of multipath reflections producing errant echoes.
- Speed of sound – any error in the value of the sound speed due to temperature, salinity, or density fluctuations, will produce a proportional error in the range calculation.

Many of the above errors cannot be calibrated for and must be accounted for using robust signal processing and estimation algorithms within the sonar.

4.5 DVL

A Doppler Velocity Log (DVL) is a form of downward looking sonar that rather than simply measuring the round-trip time of the sonar ping, it can also measure the Doppler shift, $f_0 - f_r$, in the return signal which is related to the speed of the vehicle relative to the seabed, $v_{s,r}$, and the speed of the wave in the medium, c, via (29).

$$v_{s,r} = \frac{(f_0 - f_r)c}{f_0} \tag{29}$$

DVLs typically use four transponders that all approximately point down but are each offset in a different direction by a given angle, usually 30°, as illustrated in Fig. 4. These four measurements provide enough information to determine the velocity of the transponder in three dimensions relative to some other reference (30). If the seafloor is in range of the DVL's beams, the velocity relative to the ground can be determined. If not, the DVL can determine its velocity to a given body of water below the vehicle.

$$v_{s,r} = \begin{bmatrix} \cos\psi_{t1}\sin\theta_{t1} & \sin\psi_{t1}\sin\theta_{t1} & \cos\theta_{t1} \\ \cos\psi_{t2}\sin\theta_{t2} & \sin\psi_{t2}\sin\theta_{t2} & \cos\theta_{t2} \\ \cos\psi_{t3}\sin\theta_{t3} & \sin\psi_{t3}\sin\theta_{t3} & \cos\theta_{t3} \\ \cos\psi_{t4}\sin\theta_{t4} & \sin\psi_{t4}\sin\theta_{t4} & \cos\theta_{t4} \end{bmatrix} v_n^b \tag{30}$$

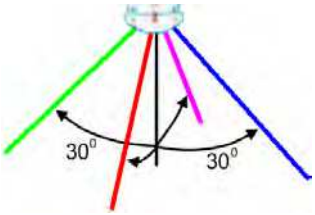


Fig. 4. DVL Transponders

DVLs suffer from the same error sources as sonar. As a consequence, many of these errors cannot be calibrated for and must be accounted for using robust signal processing and estimation algorithms.

4.6 CTD

A CTD is a sensor triad comprising a conductivity, a temperature, and a pressure sensor. The CTD can thus determine the salinity and density, as well as the temperature of the surrounding water. The first two parameters are crucial for correcting the sonar range information. AUV depth, h, below the surface of the water can be determined using (31) which is dependent on the pressure above atmospheric pressure, $P - P_0$, and density of the water, ρ. Depending on required accuracy and operating environment, the density of water can be assumed to be constant making (31) purely proportional to pressure. For more accurate results, the surrounding water density and salinity can be determined from CTD measurements using high-order polynomials fitted to experimental data as described in 'The International Equation of State of Seawater 1980' (IESS 1980).

$$h = \frac{P - P_0}{\rho g} \tag{31}$$

4.7 Acoustic Positioning

There have been several attempts to augment the onboard navigation sensors by the use of acoustic beacons. Typical schemes include long baseline (LBL), short baseline (SBL), and ultra-short baseline (USBL) positioning systems. In LBL systems, four GPS positioned beacons are located at the corners of an AUV's operating site. The AUV determines its position by transmitting a single pulse and then working its distance from each beacon from the time taken for the signal to be retransmitted by the beacons and received by the AUV. SBL systems use three or more separate beacons that are centrally connected to a surface vessel to provide synchronised timing. An initiator signal is sent by one of the beacons and a reply, generated by the AUV, is received by all beacons. The position of the AUV is then calculated and sent back to the AUV. USBL systems, conversely, use only a single beacon which is rigidly mounted on a precisely located object, and a receiver array mounted on the AUV. Distance from the beacon is determined from the signal return time, and direction by measuring the phase shift of the reply signal on the receive array.

Acoustic positioning systems have their drawbacks, the most significant of which is that such systems restrict the autonomous nature of the vehicle since the vehicle has to remain in an area that is covered by the beacons system. Other problems with acoustic positioning occur due to multipath reflections, occlusion and interference from other transponders in the vicinity.

5. Estimators

Navigation estimation systems are filtering mechanisms that process the available measurements to obtain an estimate of the required information with minimal noise corrupting the solution. This section will review the use of filters that employ a recursive Bayesian technique. This set of filters encompasses the Kalman filter family which includes the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF), as well as the particle filter family which includes the Sequential Importance Re-sampling (SIR) filter, the Regularized Particle filter, the Monte Carlo Localizer (MCL), and the Measurement-Assisted Partial Re-sampling (MAPR) Particle filter.

5.1 Bayesian Filtering

The general Bayesian filter approach defines how the pdf (probability distribution function) of a given state's estimate evolves over time based on the state dynamics (i.e., 'state model'), observations of the state (measurements), and the uncertainties of each 'observation model'(Ristic et al., 2004). The principle of Bayesian filtering states that:

If state and observation models are known as (32) and (33), respectively, then the state estimate, $p(x|Z)$, can be propagated from time k-1 to k via (34) and (35).

$$x_k = f_{k-1}(x_{k-1}, v_{k-1}) \tag{32}$$

$$z_k = h_k(x_k, w_k) \tag{33}$$

$$p(x_k|Z_{k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|Z_{k-1})dx_{k-1} \quad (34)$$

$$p(x_k|Z_k) = \frac{p(z_k|x_k)p(x_k|Z_{k-1})}{p(z_k|Z_{k-1})} \quad (35)$$

where

$$p(z_k|Z_{k-1}) = \int p(z_k|x_k)p(x_k|Z_{1:k-1})\, dx_k$$

These equations can conveniently be defined as a two-phase system in which a prediction of the new state value is made based on the previous value and the state dynamics (34) and the prediction is then corrected by any available observations (35).

5.2 Kalman Filter

The Kalman filter is an implementation of the general recursive Bayesian filtering approach. It is a recursive filter for linear systems and is generally considered to be a Linear Quadratic Estimator (LQE). The Kalman filter is a computable approximation of the general equation (32)-(35) and achieves this by placing linear and Gaussian assumptions on (32)-(33), thus resulting in (36)-(37). These assumptions ensure that for all times, (34)-(35) will be Gaussian and thus can be represented using the mean and covariance (Ristic et al., 2004).

$$x_k = F_{k-1}x_{k-1} + v_{k-1} \quad (36)$$

$$z_k = H_k x_k + w_k \quad (37)$$

The Kalman filter applies Bayesian filtering to (36)-(37), and defines (38)-(43).

$$\hat{x}_{k|k-1} = F_{k-1}\hat{x}_{k-1|k-1} \quad (38)$$

$$P_{k|k-1} = Q_{k-1} + F_{k-1}P_{k-1|k-1}F_{k-1}^T \quad (39)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\left(z_k - H_k\hat{x}_{k|k-1}\right) \quad (40)$$

$$P_{k|k} = P_{k|k-1} - K_k S_k K_k^T \quad (41)$$

where

$$S_k = R_k + H_k P_{k|k-1} H_k^T \quad (42)$$

$$K_k = P_{k|k-1}H_k^T S_k^{-1} \quad (43)$$

5.2.1 EKF

The Extended Kalman Filter (EKF) is a modification of the standard Kalman Filter to accommodate the use of nonlinear equations in defining (32)-(33). This is accommodated by the use of the Jacobian operator to provide a linear approximation, (52)-(53), of (44)-(45) at x. An important note in relation to accuracy is that, the Jacobians of (52)-(53) are accurate to a first-order Taylor series expansion of (44)-(45). Replacing (44)-(45) with (52)-(53), where required, produces (46)-(53).

$$x_k = f_{k-1}(x_{k-1}) \quad (44)$$

$$z_k = h_k(x_k) \quad (45)$$

$$\hat{x}_{k|k-1} = f_{k-1}\left(\hat{x}_{k-1|k-1}\right) \tag{46}$$

$$P_{k|k-1} = Q_{k-1} + \hat{F}_{k-1} P_{k-1|k-1} \hat{F}_{k-1}^T \tag{47}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\left(z_k - h_k\left(\hat{x}_{k|k-1}\right)\right) \tag{48}$$

$$P_{k|k} = P_{k|k-1} - K_k S_k K_k^T \tag{49}$$

where

$$S_k = R_k + \hat{H}_k P_{k|k-1} \hat{H}_k^T \tag{50}$$

$$K_k = P_{k|k-1} \hat{H}_k^T S_k^{-1} \tag{51}$$

$$\hat{F}_{k-1} = \left[\nabla_{x_{k-1}} f_{k-1}^T(x_{k-1})\right]^T \Big|_{x_{k-1}=\hat{x}_{k-1|k-1}} \tag{52}$$

$$\hat{H}_k = \left[\nabla_{x_k} h_k^T(x_k)\right]^T \Big|_{x_k=\hat{x}_{k|k-1}} \tag{53}$$

The EKF has been heavily used and has become the de facto standard for nonlinear filtering/estimation. Navigation and in particular vehicular navigation has been the most defining application for the EKF. The EKF despite its proliferation is not an optimal filter (Ristic et al., 2004) due to the accuracy of $(\bar{x}, P)$ being dependent on the error between (44)-(45) and (52)-(53). When this error is zero, the equations breakdown to (36)-(37).
The EKF has proliferated in navigation applications as it provides reasonable estimates despite its non-optimality, and it is relatively simple to implement as well as computationally cheap to support.

5.2.2 UKF

The Unscented Kalman Filter (UKF) is a modification of the standard Kalman Filter. Unlike the EKF, which makes a linear approximation of the nonlinear system (52)-(53), the UKF makes no approximation or assumption of the system or measurement model. Rather, the UKF approximates the Gaussian distribution via a set of appropriately placed sample points, sigma points. This method is motivated by the logic that it is possible to make a more accurate approximation to a Gaussian distribution using well placed sample points, creating a sample mean and covariance, rather than a linear approximation of a nonlinear function using a Jacobian operator (Ristic et al., 2004). These points are placed according to the unscented transform (Ristic et al., 2004), giving the Kalman Filter variant its name. Replacing $(\bar{x}, P)$ with a set χ which are related by (55)-(56), and using the unscented transform (54), produce (57)-(66). The unscented transform places sigma points, χ^i, according to the scaled (by κ) matrix square root of P_a. P_a is the covariance matrix P augmented with the control or measurement covariance matrices Q and R respectively. P is augmented such that the sigma points are placed as to capture the most information about the effect of the control or measurements. n_a is the size of the augmented state vector, x_a, which is the x augmented with zeros to accommodate the size P_a. This augmenting can be performed in two different ways, either Q augments P for (57)-(59) and R augments P for (60)-(66), or Q and R all augment P for (57)-(66).

$$\chi^0 = x_a = \begin{bmatrix} \bar{x} \\ 0_{Q\times 1} \\ 0_{R\times 1} \end{bmatrix} \qquad W^0 = \frac{\kappa}{n_a + \kappa}$$
$$\chi^i = x_a \pm \left(\sqrt{(n_a + \kappa)P_a}\right)_i \quad W^i = \frac{1}{2(n_a + \kappa)} \quad i = 1,..,2n_a \tag{54}$$

$$\bar{x} = \sum_{i=0}^{2n_a} W^i \chi^i \tag{55}$$

$$P = \sum_{i=0}^{2n_a} W^i \left[\chi^i - \bar{x}\right]\left[\chi^i - \bar{x}\right]^T \tag{56}$$

$$\chi^i_{k|k-1} = f_{k-1}\left(\chi^i_{k|k-1}\right) \tag{57}$$

$$\hat{x}_{k|k-1} = \sum_{i=0}^{N-1} W^i_{k-1} \chi^i_{k|k-1} \tag{58}$$

$$P_{k|k-1} = Q_{k-1} + \sum_{i=0}^{2n_a} W^i_{k|k-1}\left[\chi^i_{k|k-1} - \hat{x}_{k|k-1}\right]\left[\chi^i_{k|k-1} - \hat{x}_{k|k-1}\right]^T \tag{59}$$

$$\hat{z}_{k|k-1} = \sum_{i=0}^{2n_a} W^i_{k-1} h_k\left(\chi^i_{k|k-1}\right) \tag{60}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\left(z_k - \hat{z}_{k|k-1}\right) \tag{61}$$

$$P_{k|k} = P_{k|k-1} - K_k S_k K_k^T \tag{62}$$

where

$$K_k = P_{xz} S_k^{-1} \tag{63}$$

$$S_k = R_k + P_{zz} \tag{64}$$

$$P_{xz} = \sum_{i=0}^{2n_a} W^i_{k-1}\left[\chi^i_{k|k-1} - \hat{x}_{k|k-1}\right]\left[h_k\left(\chi^i_{k|k-1}\right) - \hat{z}_{k|k-1}\right]^T \tag{65}$$

$$P_{zz} = \sum_{i=0}^{2n_a} W^i_{k-1}\left[h_k\left(\chi^i_{k|k-1}\right) - \hat{z}_{k|k-1}\right]\left[h_k\left(\chi^i_{k|k-1}\right) - \hat{z}_{k|k-1}\right]^T \tag{66}$$

The UKF has one drawback in that, in the navigational estimation context, it is not as computationally efficient as the EKF, due to the requirement for multiple evaluations of the state dynamic equation (57) and a large matrix squareroot operation (54) which, however, can be offset somewhat with Cholesky decomposition (Ristic et al., 2004). The UKF is not affected by the nonlinearities in the dynamics, and as such produces an estimate that captures the 'true' mean and covariance more accurately than that of the EKF. The unscented transform used for defining the relationship between sample points and $(\bar{x}, P)$ is accurate to the third order Taylor series expansion for Gaussian distributions, as opposed to only first-order for the Jacobians in the EKF.

5.3 Particle Filtering

The particle filter is an alternative means of implementing the Bayesian filtering algorithm. The hypothesis of the particle filter is that an arbitrary distribution can be approximated by a multinomial distribution with a sufficiently large number of sample points in the solution space. Alternatively, a particle filter represents the pdf of the state estimate as a weighted set of sample points, i.e., 'particles' in the solution space (67). Particle filtering is also referred to as the Sequential Monte Carlo method as it is a recursive sequential analogue to the Markov Chain Monte Carlo (MCMC) batch methods (Doucet et al., 2001). A particle filter uses Sequential Importance Sampling (SIS), which is a recursive version of importance sampling. Equations (32)-(35) are applied to the multinomial representation (67) to produce (68)-(69). In the particle filter, a sample, x_k^i, is drawn from the proposal density (68) whose weight is modified based on the support from the proposal, likelihood, and transitional densities (69).

$$p(x_k|Z_{1:k}) \approx \sum_{i=1}^{N} \omega_k^i \delta(x_k - x_k^i) \tag{67}$$

so

$$x_k^i \sim q(x_k|x_{k-1}^i, z_k) \tag{68}$$

$$\omega_k^i \propto \omega_{k-1}^i \frac{p(z_k|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i, z_k)} \tag{69}$$

As a consequence of the finite number of particles being evaluated, the drawback of the basic algorithm becomes evident due to a shortcoming of the SIS algorithm. If allowed to run unattended, the weights will converge to a point where all but one of the weights will have a mass of zero with the remaining particle having all the mass, a weight of one. This condition is defined as particle degradation and has the consequence that only one particle is contributing to the solution voiding the principle of the particle filter hypothesis. All particle filters employ a re-sampling stage (Douc et al., 2005) to redistribute the particles so as to avoid degradation, and produce a proposal density $q(x_k|x_{k-1}, z_k)$ that ensures x is drawn from an appropriate subset of the solution space. Two methodologies are available for resampling in a particle filter. The first is to only resample when it is needed via some fitness measure of the particle weights (70), and this methodology produces an architecture that conforms to the SIS with Resampling (SISw/R) frame work. Alternatively resampling can occur at every epoch, regardless of particle fitness, fitting under the definition of Sequential Importance Resampling (SIR).

$$\widehat{N}_{eff} = \frac{1}{\sum_{i=1}^{N} (\omega_k^i)^2} \tag{70}$$

An appropriate choice of proposal density is crucial as it can, if optimal, minimize the variance of the sample weights introduced in the sampling stage. An example of this is the first practical particle filter implementation, the 'Bootstrap' particle filter (Ristic et al., 2004), whose SIR architecture is one of two essential factors for the filter's successful operation. The other factor is that the bootstrap filter makes convenient use of (71) for the proposal

distribution, which can easily be sampled from and which simplifies (69) to produce (72) but is not optimal.

$$q\left(x_k \middle| x_{k-1}^i, z_k\right) \triangleq p\left(x_k \middle| x_{k-1}^i\right) \tag{71}$$

$$\omega_k^i \propto \omega_{k-1}^i p\left(z_k \middle| x_k^i\right) \tag{72}$$

Sequential Importance Resamping (SIR) is the principal architecture employed in effectively all particle filter implementations. Applications related to mobile robots include target identification and tracking both airborne and marine (Ristic et al., 2004), terrain navigation , airborne navigation , underwater navigation , and Simultaneous Localization And Mapping (SLAM) . The bootstrap architecture however has a drawback, that is when the system noise is small or the likelihoods are peaked, the particles within the bootstrap filter can still degenerate due to lack of diversity and poor proposal choice, respectively. There are numerous variations of the SIR architecture, as used in the original bootstrap particle filter, that are designed to address this problem. Most of the variants consist of modifying two stages of the particle filter: the proposal stage, thus changing the proposal density used, and the resampling stage, thus changing the resampling densities used.

5.3.1 Resampling

Resampling consists of creating a new unweighted set **X**, from a weighted set **X***, whose distribution reflects **X*** resulting in the weights being normalized and avoiding degradation. All current resampling algorithms use the same technique to sample from the posterior. They all create a new set **X** by applying uniformly generated random numbers to an inverse of the cumulative distribution function (cdf) of the state estimate (73) (Doucet et al., 2001). This is not a cdf in the conventional sense in that it does not accumulate over the solution space but over the particle indices indicated in Fig. 5.

A uniform number applied to (73) produces a value defining the index of a particle. The indicated particle's state then constitutes a sample in the new distribution. There are four common resampling algorithms: Multinomial, Stratified, Systematic, and Residual. These four algorithms are differentiated by how the uniformly distributed numbers are generated.

$$\text{for } i = 1,..,N, \text{set } I^i = D_{\bar{\omega}}^{-1}\left(u^i\right) \text{ and } \tilde{\xi}^i = \xi^{I^i} \tag{73}$$

Multinomial -	for i = 1,..,N,	$u_i \sim u((0,1])$	
Stratified -	for i = 1,..,N,	$u_i \sim u((N^{-1}(i-1), N^{-1}i])$	
Systematic -	for i = 1,..,N,	$u_i = N^{-1}(i-1) + u_1$	where $u_1 \sim u(0,N^{-1}]$
Residual -	for i = 1,..,m	$N^i = \lvert N\omega^i \rvert + N'^i$	where $\lvert N\omega^i \rvert$ is the integer part and denotes the number of samples from X^i
then			
for i = 1,..,(N-R),	$u_i \sim u((0,1])$		where R = nm$\lvert N\omega^i \rvert$

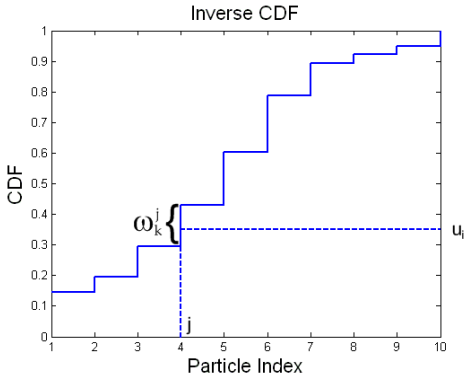


Fig. 5. CDF Example

5.3.2 Regularized Particle Filter

The Regularized particle filter uses regularization of the empirical distribution by means of a kernel method (74) in the resampling stage. Regularization avoids particle degradation by generating diversity through exploiting the smoothness that is contained in the distribution in a continuous space. This diversity makes the filter more robust at the expense of the theoretical disadvantage that the samples are no longer asymptotically approximate to those of the posterior.

$$x_k^{i\,*} = x_k^i + h_{opt} D_k \epsilon^i \tag{74}$$

5.3.3 MCL Particle Filter

The Monte-Carlo Localizer was developed specifically for robots that operate in buildings (Doucet et al., 2001) to offset the drawbacks of the case dependent quality of the bootstrap filter's proposal choice. The MCL employs a modified resampling stage that uses the observations in the generation of the resampled distribution. The resampling stage of the MCL includes the observation by drawing samples from not only the prior, as in the bootstrap filter, but a fixed percentage from the likelihood as well (75). This indirectly improves the proposal distribution when the likelihoods are peaked, by placing particles at suitable points in the solution space according to the likelihood before applying to (71), thus indirectly producing (68).

$$\begin{aligned} &\text{for } i = 1,..,N-R, && \text{set } I^i = D_\omega^{-1}(u^i) \text{ and } \tilde{\xi}^i = \xi^{I^i} \\ &\text{for } i = 1,..,R, && \text{set } \tilde{\varsigma}^i = z_k^i \\ & && \text{where } \quad z_k^i \sim p(z_k) \end{aligned} \tag{75}$$

5.3.4 Auxiliary Particle Filter

The Auxiliary particle filter uses a simulation based method to define the proposal distribution. The proposal density is generated in a three stage process. First, a deterministic projection of the set x_k from the set x_{k-1} is generated, as shown in (76), based on some characteristic of x. Second, the weights are modified in (77) based on this prediction and

current observations. Last, the resultant multinomial distribution is resampled according to (73). This resampling creates a set of indices that denote the collection of particles at k-1 which if propagated according to the bootstrap filter (71)-(72) create a distribution equivalent to $q(x_k^i|x_{k-1}^i, z_k)$.

$$q(x_k, i|Z_k) \propto p(z_k|\mu_k^i)p(x_k|x_{k-1}^i)\omega_{k-1}^i \tag{76}$$

$$\omega_k^j \propto \omega_{k-1}^{i^j} \frac{p(z_k|x_k^j)p\left(x_k^j|x_{k-1}^{i^j}\right)}{q(x_k^j, i^j|Z_k)} = \frac{p(z_k|x_k^j)}{p(z_k|\mu_k^{ij})} \tag{77}$$

5.3.5 Rao-Blackwellised Particle Filter (RBPF)

Rao-Blackwellisation is a method of marginalizing states with Gaussian error distributions resulting in two systems: a linear Gaussian system (78) which can be processed by Kalman filter algorithms (36)-(43), and a nonlinear non-Gaussian system (79) which can be estimated by particle filter algorithms (67)-(69). This split approach avoids the 'curse of dimensionality' by lowering the number of dimensions to be estimated by the particle filter. With less dimensions, the particle filter will require significantly less particles to obtain a given accuracy. However, this comes at the expense of significantly increased computational cost per particle, since each particle will need to propagate a Kalman representation of the marginalized states. The RBPF has been studied for navigational purposes but has received most attention in tracking applications (Ristic et al., 2004).

$$x_k^a = F_{k-1}x_{k-1}^a + v_{k-1} \tag{78}$$

$$B_k^m = B_k^m + G_{k-1}x_{k-1}^a + w_{k-1} \tag{79}$$

5.3.6 Hybrids

As can be seen, the above set of variants is all concerned with either indirectly or directly defining the proposal density for the given problem. This has led to the proliferation of hybrid filters in which a Kalman Filter variant is used to estimate (80) for a particle filter. Examples of this include the Extended Particle Filter and the Unscented Particle Filter.

In these filters, the EKF (44)-(53) or UKF (54)-(66) equations are used to generate a mean and covariance estimate, $\hat{x}_k^i$ P_k^i, for each particle, x_k^i, such that they can be used as the proposal density (80).

$$q(x_k|x_{k-1}^i, z_k) \triangleq N(\hat{x}_k^i, P_k^i) \tag{80}$$

As revealed in (80) N particles, i.e., N Kalman filter representations need to be calculated at each time step k and as such present a significant computational burden but with the benefit of significantly increased accuracy. This trade off is much like the RBPF in that although more computationally intensive per particle, it requires fewer particles to achieve a given accuracy.

5.3.7 MAPR Particle Filter

The Measurement Assisted Partial Resampling (MAPR) particle filter is a novel new particle filter algorithm which is based on the bootstrap algorithm (71)-(72) with a modified resampling stage (Lammas, 2008). In place of the resampling scheme described in (73), MAPR uses a heuristic resampling scheme which nominates a proportion of the particles for resampling. This heuristic resampling scheme draws on influences from the Regularised (74) and MCL (75) particle filters and is designed to minimise variance in the particle weights introduced due to the resampling process.

6. Implementation and Analysis

To compare the algorithms detailed in the previous section, the algorithms are implemented in a navigational system and tested in the scenario described in the introduction and illustrated in Fig. 1. To implement this navigational system, the state (25) and measurement (26) models must be defined. In this scenario, the vehicle dynamics (22) defines the state models (44), (57), and (69), and the measurement equations (23)-(31) define the observation models (45), (58), and (69). The filters are run synchronously at 100 Hz in order to mitigate some of the nonlinearity effects of the Kalman Filter variants. This sampling rate is chosen under the consideration that, within a reasonable range, the smaller the sample period, the better the linear approximation to a nonlinear function. The particle filters are run using a particle population of 1000, i.e., N=1000, which is generally considered as insufficient, for all but the MAPR filter, to estimate the given scenario which has a state vector with 10 states.
For the sake of clarity and conciseness, the results presented will be restricted to four of the aforementioned filters, namely the EKF, UKF, Bootstrap, and MAPR filters. The measurements available in this scenario are taken from the IMU, DVL, Compass, and GPS modules described in Table 1. In the scenario, it is assumed that the seafloor is visible to the DVL transponders from the surface, and as such provides a continuous stream of velocity measurements. This premise means that only the position states needed to be estimated via dead reckoning, i.e., for the period of time the vehicle is submerged.

Part No.	Sensors					Biases	
	Vendor	Description	Data	Hz	RMS	Case 1	Case 2
MMA7260Q	Freescale	3 axis accelerometer*	a_x, a_y, a_z	100	0.0063	0	0.01
CRS03-04	Silicon Sensing	3x 1 axis Gyroscopes*	$\omega_x, \omega_y, \omega_z$	100	0.0045	0	0.01
HMR3000	Honeywell	Compass	ϕ, θ, T_x, T_y, T_z	20	0.002 0.001	N/A	
LEA-4T	uBlox	GPS	$PR_{1:N}$ $\delta_{1:N}$	10	20 0.03		
Explorer	Teledyne	DVL	$V_{1:4}$	7	0.0055		
* the 3 axis accelerometer and the 3 single axis gyroscopes aligned orthogonally form an Inertial Measurement Unit (IMU)							

Table 1. Sensors and Intrinsic Errors

Two variants of this scenario are analysed based on Table 1. The first variant (Case 1) assumes a perfectly calibrated IMU, with only Gaussian noise left as the source of errors. The second variant (Case 2) assumes a more realistic IMU which, apart from Gaussian noise, has a bias of 0.01 rad/s for the gyroscopes and 0.01 m/s² for the accelerometers. This arrangement allows analysis of the robustness of the four filters to 'non-ideal' measurements in comparison to 'ideal' measurements.

A Monte Carlo trial of the scenario is performed for each of the four filters to compare the systematic characteristics. Each trial consists of 400 runs, following the same trajectory with independently identically distributed (i.i.d.) measurements. The results of these trials for unbiased (ideal) IMU and the biased (non-ideal) IMU are presented in Fig. 6 and Fig. 7, respectively. Figs. 8a and 8b show the cross-track errors of the estimates presented in Fig. 6a and Fig. 7a, respectively.

Figs. 6a and 6b show the top down view of the estimated raster trajectory and the estimate and the residual of the surge, u, respectively. As can be seen in Fig. 6b, the EKF and UKF produce a smaller error compared to the particle filter variants. This behaviour is predictable since the errors in this scenario assume only Gaussian noise and for the given sampling rate, the EKF and UKF with their associated limited nonlinearities in (22) and (23)-(32) should theoretically produce an optimal solution. It should also be noted that although the Kalman variants are optimal in this case, the MAPR's position estimate is still comparable to those of the Kalman Filters, as shown in Fig. 8a, and its surge estimate though degraded is still less than 5mm/s.

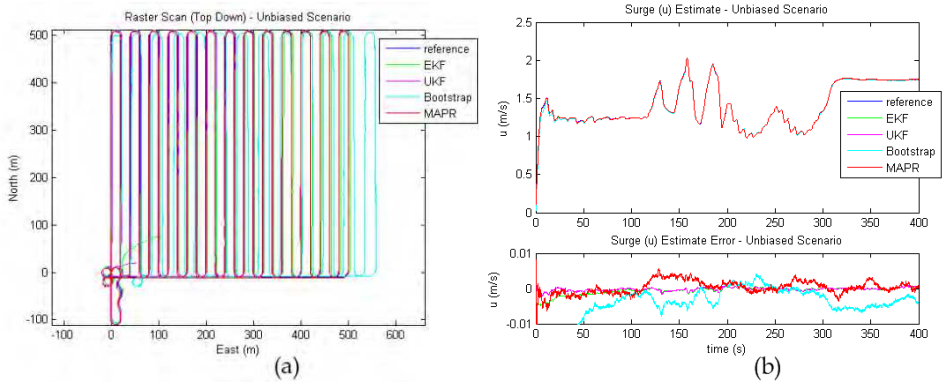
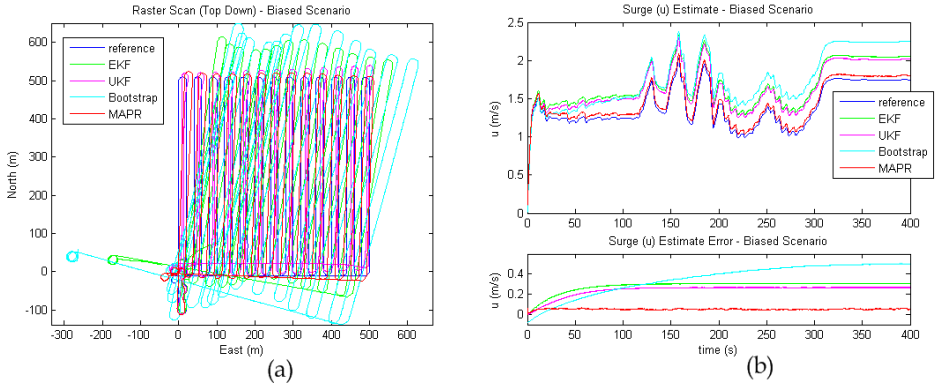


Fig. 6. Raster Scan (a) and Surge Estimates (b) for the Unbiased Case

Fig. 7 shows the result of the more realistic scenario which includes residual bias errors in the IMU. In Fig. 7a, it can be seen that the EKF and Bootstrap filters suffer significantly in their position estimates from these biases when compared with Fig. 6a. Similarly, significant distortions in the velocity estimate due to the biases are also evident in Fig. 7b. Though the UKF suffers in a similar way to the EKF and Bootstrap filters in estimating the velocity, the UKF can still estimate the position accurately. The reason for the deterioration of the EKF estimates is because, despite there being minimal nonlinearities in the measurements, GPS still contains non-negligible nonlinearities which is compounded by its limited availability and by the critical impact of poor initial estimates on the dead reckoning process. Likewise,

for the Bootstrap filter, limited GPS availability and the impact of initial estimates are also compounding factors, however, in this case the initial error is due to inadequacies in the bootstrap filter for small particle population, i.e., N=1000. As evident in Fig. 7b, the best performance in term of estimating velocity in this case is the MAPR filter which has an error six times less than that of the next accurate estimate, UKF.



(a) (b)

Fig. 7. Raster Scan (a) and Surge Estimates (b) for the Biased Case

In Fig. 7a, as in Fig. 6a, the position error of the UKF and MAPR filter appear to be comparable. However, in Fig. 8, the difference between the unbiased and biased cases becomes more evident. In Fig. 8a, with no biases and only Gaussian noise, it can be seen that, apart from the Bootstrap filter, all the filters have a comparable cross-track error, with the MAPR filter being marginally less accurate than the Kalman filters. In Fig. 8b, with the inclusion of biases, the MAPR clearly has a smaller cross-track error than that of the UKF in almost all parts of the trajectory except for a small section at the start which exhibits a larger error.

In conclusion, although some navigation filters may work more accurately than others with 'ideal' sensors, the performance degradation of these filters in the presence of increasing sensor bias is evident as demonstrated in the test scenario presented here. This has the consequence that filters that may appear not to be desirable in the ideal test may actually perform better than the perceived optimal choice under realistic conditions. As such, the use of high-fidelity simulation and field testing of proposed navigation algorithms is essential to identify the optimal selection. As shown in this section, although the merits of the MAPR filter may not seem apparent for ideal or very expensive sensor suites, the MAPR filter performs better than the UKF which is considered to be the best of the filters in the ideal situation. Moreover, the MAPR filter retains many of the desirable advantages of particle filters, such as multimodal distributions which are commonly encountered when using SLAM-based positioning. The multimodal distributions result from multiple-position hypothesises due to the similarity of different features or different angles of the same feature, as viewed by the sensors. This multimodal support is increasingly desirable as the use of SLAM for positioning becomes more prolific in mobile robotic applications.

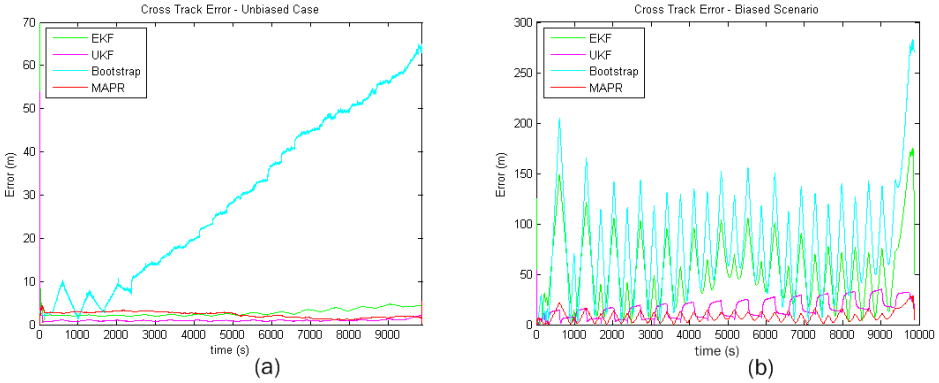


Fig. 8. Cross-Track Error for Unbiased Case (a) and Biased Case (b)

7. Conclusion

The chapter has provided a review of various sensing modalities and filtering techniques used in AUV navigation. Specifically, a range of suitable navigation sensors as carried by modern AUVs is discussed, as a basis for selecting a sensor platform that is appropriate for the intended navigation algorithm and application. A set of algorithms, based on the Bayesian filter family, is reviewed. These filters include various versions of Kalman and particle filters, as well as the new MAPR particle filter developed by the authors. Implementations and analyses of these filters, in the context of a navigation estimation system for an AUV, are provided. These implementations have been tested based on the growth in the filters' state estimate errors, particularly with respect to position estimates, during a raster scan mission, typical for survey class AUVs. The information provided will allow vehicle designers to consider the use of these concepts for the purpose of improving performance and efficiency, and reducing vehicle instrumentation costs.

Acknowledgements

This work is supported by the CSIRO Wealth from Oceans Flagship program through the Subsea Pipelines Cluster project.

8. References

Arulampalam, S.; Maskell, S.; Gordon, N. & Clapp, T. (2001). A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 50. 1. (February 2002), page numbers 15)

Caiti, A.; Garulli, A.; Livide, F. & Prattichizzo, D. (2004). Localization of Autonomous Underwater Vehicles by Floating Accoustic Bouys: A Set Membership Approach. *IEEE Journal of Oceanic Engineering*, 30. 1. (January 2005), page numbers 13)

Choset, H. & Nagatani, K. (2001). Topological simultaneous localization and mapping (SLAM): towardexact localization without explicit localization. *IEEE Transactions on Robotics and Automation*, 17. 2. numbers 125-137)

Dissanayake, M.; Newman, P.; Clark, S.; Durrant-Whyte, H. F. & Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17. 3. numbers 229-241)

Douc, R.; Cappe, O.; Polytech, E. & Palaiseau, F. (2005). Comparison of resampling schemes for particle filtering.

Doucet, A.; Freitas, N. d.; Gordon, N.; Smith, A.; Crisan, D.; Moral;, P. D. & Jacob, J. (2001). *Sequential Monte Carlo Methods in Practice*, Springer, 0-387-95146-6, New York

Fossen, T. I. (2002). *Marine Control Systems: Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles*, Marine Cybernetics, 82-92356-00-2, Trondheim, Norway

Gebre-Egziabher, D.; Elkaim, G. H.; Powell, J. D. & Parkinson, B. W. (2001). A nonlinear, two-step estimation algorithm for calibrating solid-state strapdown magnetometers.

Lammas, A. (2008). Improving Navigational Accuracy for AUVs using the MAPR Particle Filter. Oceans'08. Quebec, Canada.

Ristic, B.; Arulampalam, S. & Gordon, N. (2004). *Beyond the Kalman Filter; Particle Filters for Tracking Applications*, Artech House, 1-58053-631-x, Boston

Titterton, D. H. & Weston, J. L. (2004). *Strapdown inertial navigation technology*, Peter Peregrinus Ltd,

Tsui, J. B. Y. (2005). *Fundamentals of global positioning system receivers: a software approach*, Wiley-Interscience

Mobile Robots Navigation

Edited by Alejandra Barrera

ISBN 978-953-307-076-6
Hard cover, 666 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

Mobile robots navigation includes different interrelated activities: (i) perception, as obtaining and interpreting sensory information; (ii) exploration, as the strategy that guides the robot to select the next direction to go; (iii) mapping, involving the construction of a spatial representation by using the sensory information perceived; (iv) localization, as the strategy to estimate the robot position within the spatial map; (v) path planning, as the strategy to find a path towards a goal location being optimal or not; and (vi) path execution, where motor actions are determined and adapted to environmental changes. The book addresses those activities by integrating results from the research work of several authors all over the world. Research cases are documented in 32 chapters organized within 7 categories next described.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821