AN EVALUATION OF BAYESIAN METHODS FOR BATHYMETRY-BASED LOCALIZATION OF AUTONOMOUS UNDERWATER ROBOTS

Jungseok Hong,* Michael Fulton, and Junaed Sattar
Department of Computer Science and Engineering
University of Minnesota Twin Cities
Minneapolis, MN, 55455, USA
{[1]jungseok, [2]fulto081, [3]junaed} at umn.edu

ABSTRACT

This paper presents an evaluation of a number of probabilistic algorithms for localization of autonomous underwater vehicles (AUVs) using bathymetry data. The algorithms, based on the principles of the Bayes filter, work by fusing bathymetry information with depth and altitude data from an AUV. Four different Bayes filter-based algorithms are used to design the localization algorithms: the Extended Kalman Filter (EKF), Unscented Kalman Filter (UKF), Particle Filter (PF), and Marginalized Particle Filter (MPF). We evaluate the performance of these four Bayesian bathymetry-based AUV localization approaches under variable conditions and available computational resources. The localization algorithms overcome unique challenges of the underwater domain, including visual distortion and radio frequency (RF) signal attenuation, which often make landmark-based localization infeasible. Evaluation results on real-world bathymetric data show the effectiveness of each algorithm under a variety of conditions, with the MPF being the most accurate.

***K*eywords** underwater localization, bathymetry-based AUV localization, low-cost localization

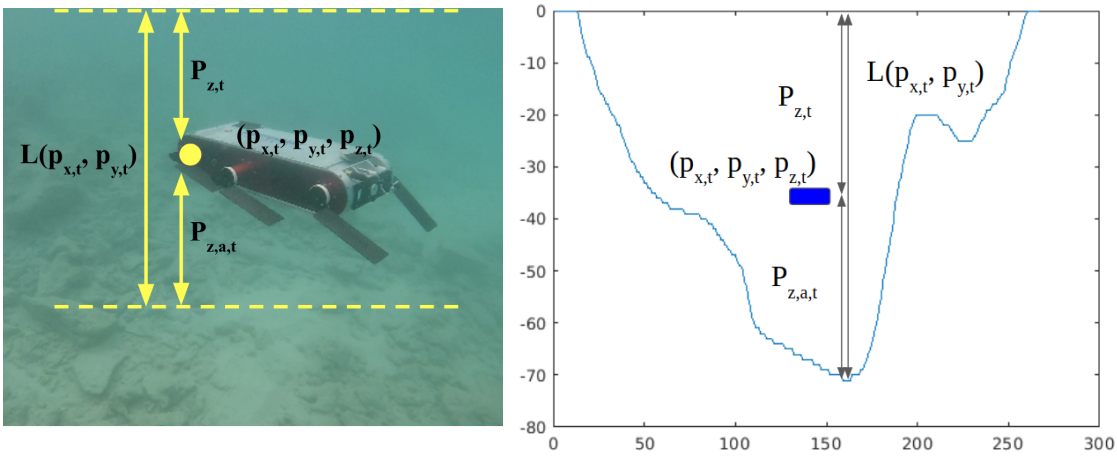
1 Introduction

The field of underwater robotics has recently been experiencing significant development, primarily driven by active research in AUVs. AUVs have seen applications in environmental monitoring (*e.g.*, [1, 2, 3]), bathymetry surveys [4], and security [5], among others. For AUVs to navigate and operate such missions successfully, the ability to *localize* accurately is essential. However, underwater localization is a challenging and open problem due to the unique circumstances AUVs face: GPS and other forms of RF-based communications are either completely unavailable or limited to extremely short ranges, and landmark-based localization using exteroceptive sensors can often be hampered by environmental factors. Our work in this paper presents a novel, low-cost approach for localizing AUVs in water bodies for which bathymetry information is available.

Robot localization problems in all environments have been studied extensively. The problem we address in this paper is underwater-specific and a subset of Terrain-based Navigation (TBN) [6], which is widely used across domains and refers to a general localization problem using prior maps. In a broad sense, there are primarily three major techniques to address the problem for underwater robots: using inertial data combined with dead reckoning [7], acoustic transponders [8], and landmark-based (also known as *geophysical* features) approaches (*e.g.*, [9, 10]).

The first of these techniques uses an inertial measurement unit (IMU) and velocity measurement (*e.g.*, from a Doppler velocity log (DVL)) to estimate the position of the robot by correcting the IMU's drift with velocity information. This approach, while widely adopted, often struggles with drifts over time; also, it requires expensive, high-accuracy IMUs (*e.g.*, [11, 12, 13]). The techniques using acoustic transponders include long baseline (LBL) [14], ultra-short

*The authors are with the Department of Computer Science and Engineering, University of Minnesota Twin Cities, 200 Union St SE, Minneapolis, MN, 55455, USA



(a) (b)

Figure 1: (1a) Visual representation of AUV location in a body of water. The top of the figure is assumed to be the surface. The surface and bottom of the body are indicated by the dashed yellow lines. (1b) Representation of the AUV (in blue) location against a lake profile.

baseline (USBL) [15], and short baseline (SBL) systems. However, most of these techniques require either a surface ship carrying a *transponder* or pre-installed beacons on the floor of the water body in question. In many applications, it is impractical to install these devices for localization purposes due to the added cost and associated overhead. Lastly, landmark-based methods use visual and acoustic sensors to detect known features in the marine environment, usually on the floor, and AUVs can localize themselves relative to such landmarks. However, optical distortions such as scattering, absorption, and attenuation (*e.g.*, arising from turbidity in the water) can be extreme which result in features only being visible up close. This vision-related issue makes it challenging to use vision-based methods for broad-area localization. Landmark-based methods with an acoustic sensor provide practical means to tackle underwater localization problems; however, these require prior information on the topology of the water body, which is often available as bathymetry data.

Bathymetry data can assist in AUV localization for each (x, y) *grid* location on the surface of the water body where height information is available. For the rest of this paper, the term *height* will refer to the total depth of the water body from the floor to the surface (shown as L in Fig. 1b). Although the depth of the robot can vary, the height of the water column is constant at that specific grid location. Therefore, an AUV would need both a depth sensor and a bottom sounder or altimeter to utilize this information.

The general goal of the AUV localization process is to infer the robot's position (x, y, z) and orientation (ϕ, θ, ξ) in 6-DOF with respect to the body of water. Localization using bathymetry data generally uses a form of the Bayes filter algorithm to estimate these *state variables* [16]. The Bayes filter [17] provides a measure of *belief* representing knowledge about the state of an AUV under a Markovian assumption. However, while the Bayes filter provides an optimal solution, it is often intractable to compute.

In this paper, we developed and implemented AUV localization algorithms for water bodies with bathymetry data, taking depth data from a pressure sensor and altitude data from a single-beam sonar as inputs, by using four Bayes filter-based methods: the EKF, UKF, PF, and MPF. The EKF and UKF are parametric implementations of the Bayes filter algorithm with Gaussian assumptions, and the PF is a nonparametric implementation. The MPF, otherwise known as the Rao-Blackwellized Particle Filter (RBPF), is a "hybrid" approach that combines the Kalman Filter (KF) and the PF [18].

The main contributions of this paper are the following:

- Propose localization algorithms to estimate the position of an AUV along all three axes (x, y, z),
- Propose low-cost underwater AUV localization algorithms which work with bathymetry data, and
- Compare and evaluate the performance of four localization algorithms with real-world bathymetric data and different motion models.

2 Related Work

Underwater localization using landmark-based methods with acoustic sensors has been widely studied. For these methods, ranging-type sonars including the single-beam, profiling, and multi-beam varieties have been used [10]. Multi-

Table 1: Selected existing localization algorithms

	Sensors	Parameters of state vector s	Algorithms
Teixeira et al. [22]	DVL, Single-beam sonar	$b = (b_x, b_y)$ $s = (x, y, b)$	PF
Fairfield and Wettergreen [20]	Multi-beam sonar	$q = (\phi, \theta, \psi, x, y, z)$ $s = (q, \dot{q}, \ddot{q})$	PF
Ura et al. [19]	Profiling sonar	$s = (x, y)$	PF
Nakatani et al. [21]	Profiling sonar	$s = (x, y)$	PF
Williams and Mahon [23]	Single-beam sonar	$q = (x, y, z)$ $s = (q, \dot{q})$	PF
Meduna et al. [24]	Single-beam sonar	$s = (x, y)$	PMF
Kim and Kim [25]	Single-beam sonar	$p = (\phi, \theta, \psi)$ $q = (u, v, w)$ $s = (x, y, z, p, q)$	MPF

beam and profiling sonars collect multiple measurements, and they can give more accurate results than single-beam sonars. Table 1 summarizes the selected existing localization algorithms. (ϕ, θ, ψ) represents the Euler angles, (u, v, w) is the AUV velocity in the body-fixed frame, and (b_x, b_y) is the velocity bias.

Although DVL, multi-beam sonar, and profiling sonar-based methods yield better results (*e.g.*, [19, 20, 21, 22]), such sensors can be prohibitively expensive. Single-beam sonars use a narrow acoustic projection to measure altitude and are thus vulnerable to noise. However, they have been widely adopted to solve localization problems since they are among the most affordable acoustic sensors [13]. Williams and Mahon [23] proposed a localization algorithm based on the PF, but the computational burden of the algorithm is heavy. Meduna et al. [24] presented a point mass filter (PMF)-based algorithm, but it is limited to the (x, y) positions of an AUV. Kim and Kim [25] used a single-beam sonar with the MPF and estimated the 6-DOF position and orientation of an AUV along with the velocity. However, the algorithm requires highly accurate IMU data which requires costly, high-end IMUs.

Several Bayes filter-based methods have been used to solve the localization problem [6] with sonar data. Among Bayes filters, the EKF and UKF have seen the most use in this domain (*e.g.*, [26, 27]). Karimi et al. [12] showed that the EKF can outperform the UKF in their particular case. However, the UKF captures nonlinearity up to the second-order term in the state transition process [10], which in theory could outperform the EKF in similar applications. We thus develop both EKF and UKF-based algorithms and compare their performance in AUV localization. Although the EKF and UKF can handle unimodal Gaussian distributions, they often fail to converge when the underlying distribution is multi-modal. The inherent nonlinearity of the underwater terrain or nonlinear AUV motions underwater make it challenging for these methods to work reliably. To address such issues, the PF has been widely used (*e.g.*, [28, 29, 23, 30, 31, 19, 32, 21, 33, 34, 22]). However, the PF is computationally expensive and thus can be prohibitive to run on-board AUVs for real-time localization. The MPF, on the other hand, has a lower computational cost and provides similar benefits to the PF, handling nonlinearity to some extent [18], thus making it potentially useful for underwater localization (*e.g.*, [35, 25]). However, localization with bathymetry data considering 3-DOF state vectors and using four Bayes filter-based algorithms (EKF, UKF, PF, and MPF) is yet to be extensively studied.

3 Problem Formulation

3.1 Motion Model

A general discrete time state-space model can be represented as Eq. 1 to formulate the localization problem where x_t is a state vector, u_t is a control input, and y_t is a measurement. As mentioned in Section 1, only the 3D position of an AUV is included in the state vector. f and h can be either linear or nonlinear functions. q_t and r_t represent the noise from motion and measurements. The model in Eq. 1 is used for the EKF, UKF, and PF-based localization algorithms. The model for the MPF-based localization algorithm is introduced in Section 4.4.

$$\begin{cases} x_t = f(x_{t-1}, u_t) + q_t \\ y_t = h(x_t) + r_t \end{cases} \tag{1}$$

The state vector and control inputs are defined as follows:

$$x_t = \begin{bmatrix} p_{x,t} & p_{y,t} & p_{z,t} \end{bmatrix}^T \tag{2}$$

$$u_t = \begin{bmatrix} v_{x,t} & v_{y,t} & v_{z,t} \end{bmatrix}^T \tag{3}$$

The AUV motion models are defined in Eqs. 4 and 5.

3.1.1 Linear motion model

All state variables are updated linearly.

$$f(x_t, u_t) = x_t + u_t * dt \tag{4}$$

3.1.2 Linear/Nonlinear mixed motion model

We propose a linear/nonlinear mixed motion model to implement a motion showing nonlinearity without using the Euler angles. Among the three state variables in the state vector, $p_{x,t}$ and $p_{y,t}$ are updated based on the height of the position $(p_{x,t}, p_{y,t})$. Therefore, the changes in $p_{x,t}$ and $p_{y,t}$ are proportional to the height of the water body at the position $(p_{x,t}, p_{y,t})$. In other words, the greater the height of the water body at the given position, the greater the change. Unlike the other two variables, the state variable $p_{z,t}$ is updated linearly as in the linear motion model.

$$f(x_t, u_t) = \begin{bmatrix} p_{x,t} + a\left[\frac{L(p_{x,t}, p_{y,t})}{a_d} + a_{off}\right]dt \\ p_{y,t} + b\left[\frac{L(p_{x,t}, p_{y,t})}{b_d} + b_{off}\right]dt \\ p_{z,t} + v_{z,t} * dt \end{bmatrix} \tag{5}$$

a, a_d, a_{off}, b, b_d, and b_{off} are constants defined for each water body. $L(p_{x,t}, p_{y,t})$ is the height of the water body at the position $(p_{x,t}, p_{y,t})$.

3.2 Measurement Model

The measurement function h is the same for both the linear model and the mixed model.

$$h(x_t) = \begin{bmatrix} p_{z,t} & p_{z,a,t} = L(p_{x,t}, p_{y,t}) - p_{z,t} \end{bmatrix}^T \tag{6}$$

L is a bathymetry map, $p_{z,t}$ represents the depth of the vehicle from the surface measured by the pressure sensor, and $p_{z,a,t}$ represents the altitude of the AUV measured by the single-beam sonar. Therefore, the sum of $p_{z,t}$ and $p_{z,a,t}$ is the height $L(p_{x,t}, p_{y,t})$ at the position $(p_{x,t}, p_{y,t})$ as shown in Fig. 1.

4 Methodology

Since the motion model in Eq. 5 and the measurement model in Eq. 6 are nonlinear, it is necessary to use nonlinear Bayes filter algorithms to solve the localization problem. The EKF and UKF are widely used to handle nonlinear state estimation with the assumption that the state variables follow a Gaussian distribution, but they could fail when the distribution is not Gaussian [36]. The PF [28] is resilient to various types of noise, but it is computationally expensive. The MPF [34] uses the PF for nonlinear state variables and the KF for linear state variables because the KF is a filter optimal for estimating linear state variables.

4.1 Extended Kalman Filter

In order to approximate a nonlinear system, the EKF takes the first-order of the Taylor series expansion [17]. Linear matrices in the KF are replaced with the Jacobians to make predictions. The Jacobians for the mixed motion model in Eq. 5 and the measurement model in Eq. 6 are shown in Eqs. 7 and 8, respectively. The EKF requires the following inputs: the state vector, state covariance, control inputs, and measurements. With these inputs, the EKF uses this well-known two-step approach:

- Prediction Step: The state vector is updated based on the motion model and control inputs. Once it is updated, the Jacobian, state covariance, and noise covariance matrices are used to find the state vector and state covariance matrix for the next time step.
- Correction Step: The Kalman gain is calculated using the Jacobian, state covariance, and measurement noise covariance matrices. After the calculation, it is used to refine the state vector from the prediction step.

The EKF assumes that state variables follow a Gaussian distribution with a single mode. Due to this assumption, the EKF is computationally efficient, and each update is $O(d^3)$ where d is the dimension of the state vector x_t [37]. The

EKF is directly applied to Eq. 1 for linear and mixed motion cases to localize AUVs using the depth and altitude measurements.

$$F_t = \begin{bmatrix} 1 + \frac{a}{a_d}\left[\frac{\partial L(x,y)}{\partial x}\right]dt & \frac{a}{a_d}\left[\frac{\partial L(x,y)}{\partial y}\right]dt & 0 \\ \frac{b}{b_d}\left[\frac{\partial L(x,y)}{\partial x}\right]dt & 1 + \frac{b}{b_d}\left[\frac{\partial L(x,y)}{\partial y}\right]dt & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

$$H_t = \begin{bmatrix} 0 & 0 & -1 \\ \frac{\partial L(x,y)}{\partial x} & \frac{\partial L(x,y)}{\partial y} & 1 \end{bmatrix} \tag{8}$$

4.2 Unscented Kalman Filter

The UKF [38] is another approach to estimate a nonlinear system using the Unscented Transform instead of the Taylor series. It samples sigma points to capture the mean and covariance of a Gaussian distribution. After that, the UKF propagates the points through the true nonlinear system. In this way, the UKF can handle higher degrees of nonlinearity than the EKF. The UKF requires the same inputs as the EKF along with some additional parameters, namely n, α, β, and κ. Like the EKF, the UKF has a two-step estimation process:

- Prediction Step: The UKF chooses $2n + 1$ sigma points from the Gaussian distribution and passes them through the motion model f where n is the number of dimensions. α, β, and κ are used to determine weights for each sample and the distribution of the sigma points.
- Correction Step: The state vector from the prediction step is used to generate sigma points. Their measurements, noise covariance, and state covariance matrices are used to calculate the Kalman gain. With the gain, the state vector is updated like the EKF. With this, the UKF approximates a Gaussian distribution with the sigma points.

In this way, the UKF can create a more accurate approximation of the Gaussian distribution than the EKF. Since the estimation is based on the Gaussian distribution assumption, it may not work for multi-modal distributions. In most cases, the UKF shows notable improvements compared to the EKF, despite possessing similar computational complexity [37]. The UKF is also directly applied to Eq. 1 for linear and mixed motion cases.

4.3 Particle Filter

The PF [31] implements the Bayes filter algorithm using sequential Monte Carlo methods. Unlike the EKF and UKF, the PF does not require any assumptions regarding the distribution. Instead, it uses N *particles* to approximate the distribution of the state vector x_t. The more particles there are, the more accurate the approximation of the distribution. Furthermore, it can handle distributions with high nonlinearity and multiple modes. However, the computational complexity of the PF for each update is $O(Nd^2)$, and it increases as N grows. When N is much larger than d, the PF can be much slower than the EKF and UKF [33]. This computational burden is often the main drawback of the PF for real-time implementations.

We proposed a depth-based PF localization (d-PFL) in Algorithm 1. **PFL_update()** takes bathymetry data, measurements, and control inputs and returns propagated particles and their weights. In order to evaluate the weights w_t, the multivariate Gaussian distribution is used as shown in Eq. 9.

$$w_t = w_{t-1} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)} \tag{9}$$

$$\hat{x} = \sum_{m=1}^{M} w^{[m]} x^{[m]} \tag{10}$$

During the update process, the algorithm only assigns weights if the particle is within the boundaries of the map. Once the weights for the particles are calculated, they are normalized to ensure that they sum to 1. Then, particles are resampled based on their weights. In order to avoid a situation where all the particles are trapped in incorrect positions in similar environments, some of the N *particles* are sampled randomly at each time step. Although this can degrade the accuracy of the algorithm, it decreases the chance of incorrect estimation occurrences. The pose of the AUV is estimated using Eq. 10 once the propagated particles and the corresponding weights are updated.

Algorithm 1 Depth-based PF Localization

D-PFL main
L = Bathymetry data of a target lake
N = The number of particles
x_{init} = Initial pose of an AUV
x_1 = **Initialize_around_pose**(L, N)
z_t = Sensor measurements
u_t = Control input
for $t = 1, ..., T$ **do**
    $x_p = x_t$
    for $m = 1, ..., N$ **do**
        $x(m, :)$ = **motion_update**$(u_t, x_p(m, :), L)$
        $w(m)$ = **sensor_update**$(z_t, x(m, :), L)$
    end for
    $w_{total} = sum(w)$
    for $m = 1, ..., N$ **do**
        $w(m) = w(m)/w_{total}$
    end for
    x_t = **resample_particles**$(x(m, :), w, L, rand)$
    $w_t = w$
    est_pose = **PFL_get_pose**(x_t, w_t)
end for
return est_pose

4.4 Marginalized Particle Filter

The MPF was proposed to reduce the computational complexity while retaining a similar performance when the model has a linear substructure [36]. The core idea of the MPF is to marginalize linear state variable(s) from the state vector and use the KF to estimate the linear state variable(s). The PF is then used to estimate the remaining nonlinear state variable(s). The computational complexity of the MPF is defined in [39] and can be simplified to $O(Nd_n^3)$ for our case where d_n is the dimension of the nonlinear state variable(s).

In order to apply the MPF [18], the model in Eq. 1 is separated into linear and nonlinear state variables as shown in Eqs. 20 and 21. The motion model noise q_t^n, q_t^l and the measurement model noise r_t are assumed to be Gaussian with zero mean. The matrices A, C, and G are determined by the motion model.

$$x_t = \begin{bmatrix} x_t^n \\ x_t^l \end{bmatrix} \tag{20}$$

$$\begin{cases} x_{t+1}^n = f_t^n(x_t^n) + A_t^n(x_t^n)x_t^l + G_t^n(x_t^n)q_t^n \\ x_{t+1}^l = f_t^l(x_t^n) + A_t^l(x_t^n)x_t^l + G_t^l(x_t^n)q_t^l \\ y_t = h_t(x_t^n) + C_t(x_t^n)x_t^l + r_t \end{cases} \tag{21}$$

In our case, the ratio $\frac{N(k)}{N_{PF}}$ is 1.1 where $N(k)$ is the number of particles that can be used for the MPF, and N_{PF} is the number of particles used for the standard PF. The ratio means that the MPF can use 10% more particles than the PF while retaining the same computational complexity as the standard PF. However, the EKF and UKF are still faster than the MPF, albeit less accurate.

The MPF localization algorithm is shown in Algorithm 2 along with selected simplified equations where Q and P are covariance matrices. The equations in detail can be found in [18].

5 Experimental Setup and Results

5.1 Bathymetry Data

The following lakes located in Minneapolis, MN, USA were chosen: Lake Bde Maka Ska, Lake Nokomis, Lake Hiawatha, and Lake Harriet. The lakes were chosen since they are large, well-studied, have a non-flat floor, and are easy to access for future field studies. The bathymetry data was acquired from the Minnesota Department of Natural Resources (MN DNR) [40] (see Fig. 2). The grid size of each lake's bathymetry data is mostly 5m but is 10m in some

Algorithm 2 MPF-based Localization

Initialize particles

Initialize nonlinear state variables

$$x_{0|-1}^{n,(i)} \sim p(x_0^n) \tag{11}$$

Initialize linear state variables

$$\{x_{0|-1}^{l,(i)}, P_{0|-1}^{(i)}\} = \{\bar{x}_0^l, \bar{P}_0\} \tag{12}$$

for $t = 1, ..., T$ **do**

PF measurement update

Evaluate the weights

$$w_t^{(i)} = p(y_t | X_t^{n,(i)}, Y_{t-1}) \tag{13}$$

Estimate nonlinear state variables

$$\hat{x}_{t|t}^n = \sum_{i=1}^{N} w_t^{(i)} x_{t|t-1}^{n,(i)} \tag{14}$$

Resample particles

$$Pr(x_{t|t}^{n,(i)} = x_{t|t-1}^{n,(j)}) = \widetilde{w}_t^{(j)} \tag{15}$$

KF measurement update

Estimate linear state variables

$$x_{t|t}^{l,(i)} = x_{t|t-1}^{l,(i)} + K_t(y_t - h_t - C_t \hat{x}_{t|t-1}^l) \tag{16}$$

$$\hat{x}_{t|t}^l = \sum_{i=1}^{N} w_t^{(i)} x_{t|t}^{l,(i)} \tag{17}$$

PF prediction

Propagate nonlinear state variables

$$x_{t+1|t}^{n,(i)} \sim p(x_{t+1|t}^n | X_t^{n,(i)}, Y_t) \tag{18}$$

KF prediction

Propagate linear state variables

$$\begin{aligned} \hat{x}_{t+1|t}^l = \bar{A}_t^l \hat{x}_{t|t}^l + G_t^l (Q_t^{ln})^T (G_t^n Q_t^n)^{-1} z_t \\ + f_t^l + L_t(z_t - A_t^n \hat{x}_{t|t}^l) \end{aligned} \tag{19}$$

end for

Table 2: Model parameters

Parameter	Value
No. of particles for the PF, N_{PF}	5000
No. of particles for the MPF, N_{NPF}	300
Motion noise cov., Q (m)	$0.01 \begin{bmatrix} v_x^2 & 0 & 0 \\ 0 & v_y^2 & 0 \\ 0 & 0 & (0.3048 v_z)^2 \end{bmatrix}$
Measurement noise cov., R (m)	$\begin{bmatrix} 0.3048^2 & 0 \\ 0 & 0.3048^2 \end{bmatrix}$
Initial uncertainty cov., P (m)	$\begin{bmatrix} 1^2 & 0 & 0 \\ 0 & 1^2 & 0 \\ 0 & 0 & 0.3048^2 \end{bmatrix}$

lakes. The lake height at each position is given in feet, but we converted it to meters for our study. In our experiments, we assumed that the grid size is 1m to simplify the calculations, and the bathymetry data was scaled accordingly.

Figure 2: Visualization of raw bathymetry data in tagged interchange file format (TIFF) of Lake Bde Maka Ska. Source: Minnesota Department of Natural Resources.

Table 3: Motion parameters

Linear Motion Parameters					
Lake	$v_x(m/s)$	$v_y(m/s)$	$v_z(m/s)$	a	b
Bde Maka Ska	1	-3	-0.1524	-0.02	0.06
Nokomis	2	-4	-0.1524	-0.1	0.2
Hiawatha	1	2	-0.3048	-0.05	-0.1
Harriet	1.5	-3	-0.3048	-0.03	0.06

Noninear Motion Parameters							
Lake	a	a_d	a_{off}	b	b_d	b_{off}	$v_z(m/s)$
Bde Maka Ska	0.6	0.2	3	2	10	23	-0.1524
Nokomis	-0.1	0.2	1	1	-1	-1	-0.1524
Hiawatha	-0.05	-0.1	1	1	0	0	-0.3048
Harriet	-0.1	0.2	3	3	1	1	-0.3048

5.2 Simulation Settings

The goal of this study is to evaluate each algorithm with real bathymetry data as a prerequisite to choosing a deployable localization algorithm. Due to the unique challenges of the underwater environment, it is extremely difficult, if not impossible, to obtain the ground truth of the AUV's positions. Thus, to quantify the accuracy and efficiency of the algorithms, we simulated the ground truth position of the AUV and evaluated the performances of each filter on the simulated AUV's motion. The linear and mixed motion models are designed to test the performance of each localization algorithm on the bathymetry data from different lake environments. Table 2 includes the model parameters for the simulation. The control inputs for each algorithm and lake were separately designed due to the lakes' different sizes and heights. In Table 3, the parameters are defined for the linear motion in Eqs. 3 and 4, and the mixed motion in Eq. 5. For the mixed model, x and y are nonlinear state variables, and z is a linear state variable.

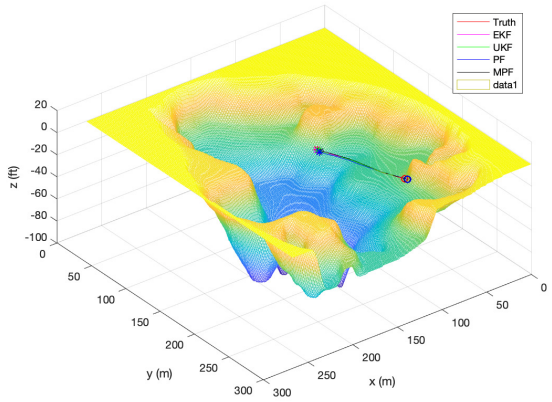
5.3 Results and Discussions

We measured the performance of our algorithms on a 4.20GHz Core i7-7700K processor running Ubuntu 18.04.2 LTS with 16GB of DDR3 memory with Matlab R2018b. 100 runs were performed with each filter for each lake and motion, and a total of $3,200$ runs were conducted.

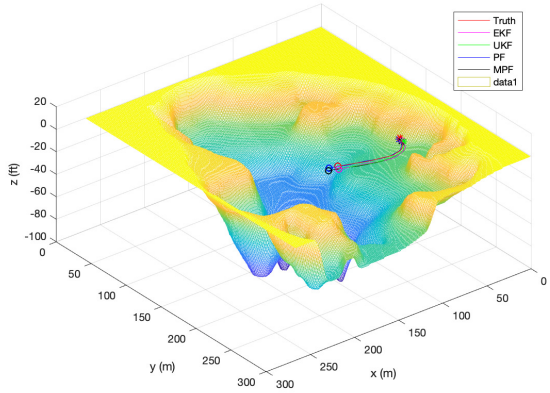
The results are summarized in Table 4. The runtime for each case was measured, and we used the root-mean-square error (RMSE) in Eq. 22 as a metric to evaluate the performance for each axis where T is the number of steps, p_g is the ground truth, and p_e is the estimated position.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T}(p_g - p_e)^2}{T}} \tag{22}$$

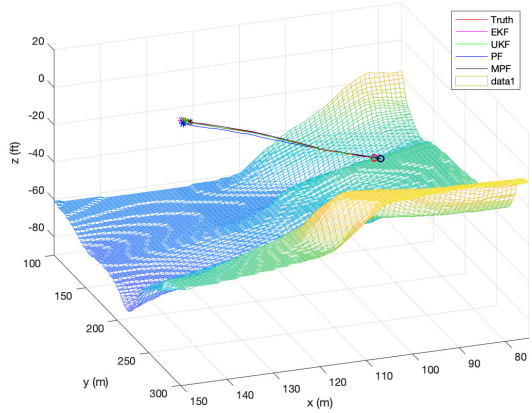
One run in Lake Bde Maka Ska is shown in Figs. 3 and 4. For both the linear and nonlinear cases, the EKF generally shows the worst performance. The UKF performed well in the linear case, but it deviated from the ground truth in the



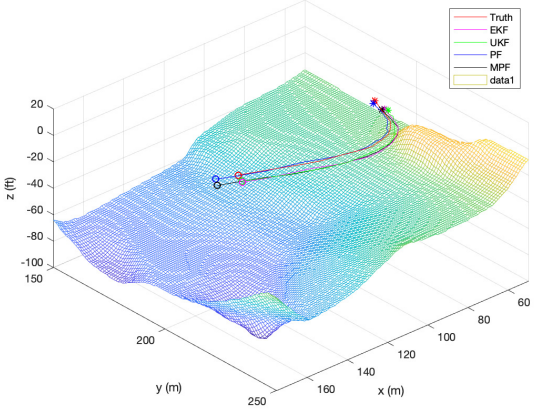
(a) Evaluation of linear motion estimations with bathymetry data of Lake Bde Maka Ska (top view).



(b) Evaluation of mixed motion estimations with bathymetry data of Lake Bde Maka Ska (top view).



(c) Evaluation of linear motion estimations with bathymetry data of Lake Bde Maka Ska (zoom-in view).



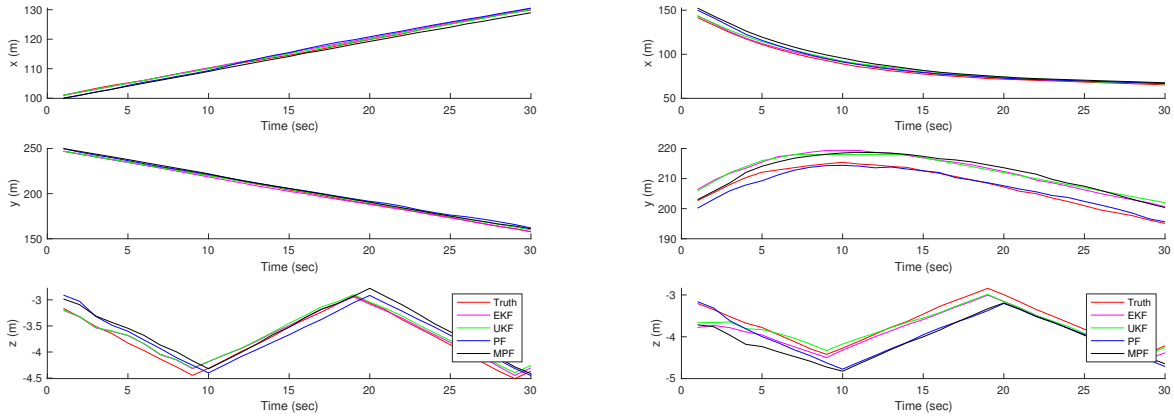
(d) Evaluation of mixed motion estimations with bathymetry data of Lake Bde Maka Ska (zoom-in view).

Figure 3: Localization performance of the four algorithms for an AUV with an altimeter and depth sensor within Lake Bde Maka Ska using bathymetry data (Start:❍ End:✳).

nonlinear case. It is noticeable that the PF and MPF mostly outperformed the EKF and UKF for both the linear and mixed motion cases. However, the PF often diverged from the ground truth and showed unstable performances. A likely cause is that the randomly selected particles can make the estimation diverge from the ground truth when a water body's bathymetry data does not have enough territorial variations.

5.3.1 Linear motion case

The EKF performs the worst for most lakes, and the results are not reliable since the performance varied between lakes. The UKF does not always perform best, but it gives reliable and relatively accurate results with lower computational complexity. The PF shows the most accurate result for Lake Hiawatha, but it performs worst for Lake Nokomis. This result is likely caused by the fact that Lake Nokomis has a symmetrical structure and fewer variations in height. The MPF generally gives the most accurate and reliable results, and it is computationally cheaper than the PF. Overall, the MPF is the most reliable and accurate filter according to the results. It is worth mentioning that the UKF is a good option if an AUV does not have much computational power and high accuracy is not required. The PF can be used for localization if the bathymetry data has enough variation in height and has an asymmetrical structure, provided that an AUV has enough computational power.



(a) Evaluation of linear motion estimations on x,y,z axis.

(b) Evaluation of mixed motion estimations on x,y,z axis.

Figure 4: Evaluation of each algorithm's performance over time within Lake Bde Maka Ska using bathymetry data for an AUV with an altimeter and depth sensor.

Table 4: Localization performance evaluation for an AUV in Lake Bde Maka Ska; bathymetry data from the MN DNR.

Lake	Number of steps	Linear motion model					Mixed motion model				
		Method	Runtime (s)	$RMSE(m)$			Method	Runtime (s)	$RMSE(m)$		
				x	y	z			x	y	z
Bde Maka Ska	50	EKF	90.15	4.09	12.43	0.42	**EKF**	90.13	**2.25**	**5.01**	**0.53**
Bde Maka Ska	50	UKF	52.33	3.07	3.26	0.40	UKF	53.92	4.07	9.02	0.55
Bde Maka Ska	50	**PF**	252.14	**1.38**	**3.57**	**0.67**	PF	279.03	4.18	9.56	1.32
Bde Maka Ska	50	MPF	94.02	2.88	2.32	0.74	MPF	103.70	4.18	3.90	1.15
Nokomis	50	EKF	57.77	3.92	5.60	0.43	EKF	59.84	4.96	7.12	0.49
Nokomis	50	UKF	29.72	3.05	3.98	0.42	UKF	29.47	12.74	12.70	0.46
Nokomis	50	PF	251.50	13.69	23.12	0.89	PF	291.11	11.90	18.60	0.98
Nokomis	50	**MPF**	79.42	**2.84**	**3.10**	**0.76**	**MPF**	79.87	**4.12**	**4.80**	**0.76**
Hiawatha	30	EKF	12.97	3.56	3.57	0.48	EKF	12.88	3.33	2.63	0.51
Hiawatha	30	UKF	6.95	3.07	2.93	0.53	UKF	7.32	3.03	3.00	0.57
Hiawatha	30	**PF**	137.52	**1.66**	**2.57**	**1.13**	**PF**	152.90	**1.15**	**2.33**	**1.15**
Hiawatha	30	MPF	37.82	3.10	3.51	0.97	MPF	42.73	3.33	3.47	0.90
Harriet	30	EKF	37.03	5.03	7.20	0.63	EKF	35.80	5.98	9.40	0.63
Harriet	30	UKF	19.43	3.09	3.09	0.59	UKF	18.01	4.08	6.03	0.54
Harriet	30	PF	149.71	1.77	3.72	1.13	**PF**	176.66	**1.62**	**2.84**	**1.16**
Harriet	30	**MPF**	51.71	**2.93**	**2.42**	**1.10**	MPF	54.39	3.13	2.75	1.09

5.3.2 Nonlinear motion case

Like the linear motion cases, the PF and MPF generally perform better than the EKF and UKF. The UKF performs poorly for some cases due to the high nonlinearity of the motion. The PF reveals the same issue that it has in the linear motion cases: the estimations diverge when there are not enough variations in bathymetry data. Similar to the linear cases, the MPF gives reliable and accurate results overall.

5.3.3 Discussions

The MPF shows the most reliable and accurate results for both the linear and mixed motion cases. If an AUV needs a well-rounded algorithm for the localization problem, the MPF is the best filter among the four filters. However, if the bathymetry data of a lake has enough variation, and the task requires high accuracy, then the PF is a better option. For an AUV with low computational power, the UKF could be the best filter for localization if the AUV's motion is mostly going to be linear.

6 Conclusion

Using bathymetry data and the measurements from a single-beam sonar altimeter and a depth sensor, we present four localization algorithms based on the EKF, UKF, PF, and MPF respectively. Also, we evaluate the performance of each filter in various aquatic environments and with multiple robot motions. The results demonstrate the feasibility of the Bayesian filter-based algorithms for localizing an AUV with bathymetric information using two low-cost sensors. The MPF-based localization generally performs best, both in terms of accuracy and computational cost. However, the UKF can be a good alternative to the PF and MPF at the expense of accuracy if an AUV mostly actuates linearly and has limited computational power. Additionally, the PF seems to be the most accurate in water bodies with sufficient terrestrial variations if an AUV possesses the necessary computational power. Future work will focus on evaluating the performance of the proposed localization algorithms when run on-board AUVs, testing the algorithms on the bathymetry data from other Minnesota lakes, and deploying active acoustic sensors to improve localization accuracy.

Acknowledgments

We are thankful to Chelsey Edge, Marc Ho, Jiawei Mo, and Julian Lagman for their assistance, the MN DNR for the bathymetric data, and the MnDRIVE Initiative for supporting this research.

References

[1] M. Moline, P. Bissett, S. Blackwell, J. Mueller, J. Sevadjian, C. Trees, and R. Zaneveld, "An autonomous vehicle approach for quantifying bioluminescence in ports and harbors," in *Photonics for Port and Harbor Security*, vol. 5780. International Society for Optics and Photonics, 2005, pp. 81–88.

[2] D. A. Fong and N. L. Jones, "Evaluation of AUV-based ADCP measurements," *Limnology and Oceanography: methods*, vol. 4, no. 3, pp. 58–67, 2006.

[3] A. Forrest, H. Bohm, B. Laval, E. Magnusson, R. Yeo, and M. Doble, "Investigation of under-ice thermal structure: small AUV deployment in Pavilion Lake, BC, Canada," in *OCEANS 2007*. IEEE, 2007, pp. 1–9.

[4] R. J. Huizinga, "Bathymetric and velocimetric surveys at highway bridges crossing the Missouri River near Kansas City, Missouri, June 2–4, 2015," US Geological Survey, Tech. Rep., 2016.

[5] S. T. Tripp, "Autonomous underwater vehicles (AUVs): a look at Coast Guard needs to close performance gaps and enhance current mission performance," COAST GUARD RESEARCH AND DEVELOPMENT CENTER GROTON CT, Tech. Rep., 2006.

[6] S. Carreno, P. Wilson, P. Ridao, and Y. Petillot, "A survey on terrain based navigation for AUVs," in *OCEANS 2010*. IEEE, 2010, pp. 1–7.

[7] P. A. Miller, J. A. Farrell, Y. Zhao, and V. Djapic, "Autonomous underwater vehicle navigation," *IEEE Journal of Oceanic Engineering*, vol. 35, no. 3, pp. 663–678, 2010.

[8] P. Batista, C. Silvestre, and P. Oliveira, "A sensor-based controller for homing of underactuated AUVs," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 701–716, 2009.

[9] R. M. Eustice, "Large-area visually augmented navigation for autonomous underwater vehicles," Ph.D. dissertation, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution, 2005.

[10] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, 2014.

[11] R. Panish and M. Taylor, "Achieving high navigation accuracy using inertial navigation systems in autonomous underwater vehicles," in *OCEANS, 2011 IEEE-Spain*. IEEE, 2011, pp. 1–7.

[12] M. Karimi, M. Bozorg, and A. Khayatian, "A comparison of DVL/INS fusion by UKF and EKF to localize an autonomous underwater vehicle," in *Robotics and Mechatronics (ICRoM), 2013 First RSI/ISM International Conference on*. IEEE, 2013, pp. 62–67.

[13] J. Melo and A. Matos, "Survey on advances on terrain based navigation for autonomous underwater vehicles," *Ocean Engineering*, vol. 139, pp. 250–264, 2017.

[14] A. P. Scherbatyuk, "The AUV positioning using ranges from one transponder LBL," in *OCEANS'95. MTS/IEEE. Challenges of Our Changing Global Environment. Conference Proceedings.*, vol. 3. IEEE, 1995, pp. 1620–1623.

[15] M. Morgado, P. Oliveira, and C. Silvestre, "Experimental evaluation of a USBL underwater positioning system," in *ELMAR, 2010 proceedings*. IEEE, 2010, pp. 485–488.

[16] T. Y. Teck, M. Chitre, and F. S. Hover, "Collaborative bathymetry-based localization of a team of autonomous underwater vehicles," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2475–2481.

[17] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.

[18] T. Schon, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2289, 2005.

[19] T. Ura, T. Nakatani, and Y. Nose, "Terrain based localization method for wreck observation auv," in *OCEANS 2006*. IEEE, 2006, pp. 1–6.

[20] N. Fairfield and D. Wettergreen, "Active localization on the ocean floor with multibeam sonar," in *OCEANS 2008*. IEEE, 2008, pp. 1–10.

[21] T. Nakatani, T. Ura, T. Sakamaki, and J. Kojima, "Terrain based localization for pinpoint observation of deep seafloors," in *OCEANS 2009-EUROPE*. IEEE, 2009, pp. 1–6.

[22] F. C. Teixeira, J. Quintas, P. Maurya, and A. Pascoal, "Robust particle filter formulations with application to terrain-aided navigation," *International Journal of Adaptive Control and Signal Processing*, vol. 31, no. 4, pp. 608–651, 2017.

[23] S. Williams and I. Mahon, "A terrain-aided tracking algorithm for marine systems," in *Field and Service Robotics*. Springer, 2003, pp. 93–102.

[24] D. K. Meduna, S. M. Rock, and R. McEwen, "Low-cost terrain relative navigation for long-range AUVs," in *OCEANS 2008*. IEEE, 2008, pp. 1–7.

[25] T. Kim and J. Kim, "Nonlinear filtering for terrain-referenced underwater navigation with an acoustic altimeter," in *OCEANS 2014-TAIPEI*. IEEE, 2014, pp. 1–6.

[26] B. He, K. Yang, S. Zhao, and Y. Wang, "Underwater simultaneous localization and mapping based on EKF and point features," in *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*. IEEE, 2009, pp. 4845–4850.

[27] B.-D. Yoon, H.-N. Yoon, S.-H. Choi, and J.-M. Lee, "UKF Applied for Position Estimation of Underwater-Beacon Precision," in *Intelligent Autonomous Systems 12*. Springer, 2013, pp. 501–508.

[28] S. Thrun, "Particle filters in robotics," in *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2002, pp. 511–518.

[29] R. Karlsson, F. Gusfafsson, and T. Karlsson, "Particle filtering and Cramer-Rao lower bound for underwater navigation," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, vol. 6. IEEE, 2003, pp. VI–65.

[30] I. Rekleitis, "A Particle Filter Tutorial for Mobile Robot Localization," Centre for Intelligent Machines, McGill University, 3480 University St., Montreal, Québec, CANADA H3A 2A7, Tech. Rep. TR-CIM-04-02, Jan. 2004.

[31] D. Salmond and N. Gordon, "An introduction to particle filters," *State space and unobserved component models theory and applications*, pp. 1–19, 2005.

[32] F. Maurelli, A. Mallios, D. Ribas, P. Ridao, and Y. Petillot, "Particle filter based auv localization using imaging sonar," *IFAC Proceedings Volumes*, vol. 42, no. 18, pp. 52–57, 2009.

[33] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.

[34] T. B. Schön, F. Gustafsson, and R. Karlsson, "The particle filter in practice," in *The Oxford Handbook of Nonlinear Filtering*, B. R. Dan Crisan, Ed. Oxford University Press, 2011, pp. 741–767.

[35] R. Karlsson and F. Gustafsson, "Bayesian surface and underwater navigation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4204–4213, 2006.

[36] T. B. Schon, R. Karlsson, and F. Gustafsson, "The marginalized particle filter in practice," in *Aerospace Conference, 2006 IEEE*. IEEE, 2006, pp. 11–pp.

[37] F. Daum, "Nonlinear filters: beyond the Kalman filter," *IEEE Aerospace and Electronic Systems Magazine*, vol. 20, no. 8, pp. 57–69, 2005.

[38] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*. IEEE, 2000, pp. 153–158.

[39] T. B. Schön, R. Karlsson, and F. Gustafsson, "The marginalized particle filter–analysis, applications and generalizations," in *ESAIM: Proceedings*, vol. 19. EDP Sciences, 2007, pp. 53–64.

[40] N. R. Department, "Lake Bathymetric Outlines, Contours, Vegetation, and DEM," June 2018, https://gisdata.mn.gov/dataset/water-lake-bathymetry.