Lagrangian Data Assimilation at Depth: Assimilating Glider Data into a Two-Layered Coastal Upwelling Model

Adam Mallen
Adviser: Elaine Spiller
Department of Mathematics, Statistics, and Computer Science, Marquette University
July 20, 2016

1 Introduction

The advent of ocean drifters, floats, and gliders has dramatically changed how we observe the ocean. All of these instruments are Lagrangian because they are advected by the ocean, and data gathered from their sensor measurements are taken along a trajectory influenced by the forcing from the flow field instead of fixed locations in space. However, gliders are unique in that they also can maneuver relative to the flow according to a prescribed “flight” plan rendering their trajectories quasi-Lagrangian. Assimilating Lagrangian data into Eulerian models presents a host of challenges which can be overcome by including the instrument coordinate as part of the state variable. This approach is known as Lagrangian Data Assimilation (LADA) and has been widely studied in the case of fully Lagrangian instruments [15, 26, 25, 3, 40, 32]. In this work, we will adapt LADA to quasi-Lagrangian data as well as depth-dependent data and test this approach on a depth-dependent flow model.

Ocean gliders are autonomous underwater vehicles (AUVs) which are fitted with sensors for targeted, subsurface data collection [41]. Gliders provide a relatively low-cost means of sampling fine horizontal structures in the ocean and have proven to be particularly valuable in observing boundary currents [34]. They propel themselves by manipulating their buoyancy while lift and drag acting on fixed wings and rudders translate some of the vertical buoyancy force into forward motion. The nature of this propulsion creates a series of saw-tooth shaped dives which together comprise a mission. Glider missions are programmed to traverse targeted paths while collecting pressure, salinity, temperature, and potentially various biological or chemical data [23]. GPS sensors provide measurements of the gliders’ locations while on the surface, but not at depth. The other sensors take their measurements en-route during the glider’s dives, and if the background flow is mischaracterized, uncertainty in the location of the subsurface measurements can be large.

In practice, gliders often employ dead reckoning integration of their prescribed glide velocity (velocity relative to the background flow) to compute a bearing and range intended to surface the glider at a prescribed waypoint. Upon surfacing, the glider uses GPS to acquire its position and
compute the error between the way point and the actual surfacing location. This error is attributed
to the surrounding ocean flow and is used to compute a spatially (and temporally) averaged flow
velocity. This single average flow velocity is used to describe the background flow for the entire
region of the previous dive and also used to estimate the flow when aiming the glider for the next
waypoint [31, 18]. For coastal missions, these flows can be quite complex and can vary strongly
over the spatial regions sampled by a glider’s mission. In such a situation, the average flow estimate
computed from dead reckoning is often not a good description of the background flow.

Previous studies on Lagrangian data assimilation focused on models of flows which only vary
in latitude and longitude. This is natural for studies which involve Lagrangian instruments such as
drifters—which float near the surface—or floats—which move along on a surface of equal pressure.
Gliders, on the other hand, must dive in saw-tooth patterns to propel themselves and will encounter
advecting flow along an interval of depths ranging from the surface to the maximum depth of each
dive. Therefore, we must adapt the Lagrangian data assimilation framework to handle 3-D nature
of glider trajectories in a depth-varying flow model. To this end, this work explores the assimilation
of simulated glider trajectories in a model with stratified flow.

In several recent studies, glider data has been assimilated into to regional ocean/sea models
(usually constructed on spatial grids) with the focus of improving model output [31, 6, 13, 29, 16].
These studies average glider observations spatially and temporally and use interpolation to perform
the Eulerian assimilation with this data onto the grid model. For large-scale ocean models, the
uncertainty bounds on the glider mission path would likely be sub grid-scale. In this case, Eulerian
assimilation with interpolated data is a sensible approach. However, in coastal regions or free jets,
spatial variations in the flow can be significant over the few hundred meters traveled by a glider in
a single dive. This is an issue when researchers are interested in accurately aiming gliders on this
scale (e.g. feature tracking, etc. [34]) or estimating and quantifying uncertainty of precise spatial
fields. In the later case, that is, Lagrangian data assimilation offers a mechanism to reconstruct
estimates of gliders’ underwater trajectories and to describe attendant uncertainties in those paths.
This is particularly important to give an accurate spatial footprint to sub-surface data collected by
the glider that is otherwise only reported as time-series.

As an initial investigation into Lagrangian glider data assimilation, we consider a complex, but
low dimensional background flow given by a two-layer model of upwelling over the continental
shelf [4, 11]. The flow model is based on shallow-water dynamics, and the vertical velocity is
asymptotically small over most of the domain and will be ignored. Within each layer, the horizontal
velocity is constant in depth but varies in the horizontal coordinates. Further, the horizontal velocity
will vary from layer to layer, and in some parts of the domain alongshore flow can even move
in opposite directions between layers. The large variation in flow strength between layers and
in the horizontal directions creates challenges for estimation of the flow using only the average
flow velocity from dead reckoning. The goal of this simulation experiment is to illustrate the
ability of Lagrangian data assimilation techniques to provide superior estimates of the flow field
and underwater trajectories in a case where the assumption of horizontally and vertically uniform
flow fails.
2 Background: Data Assimilation

Data assimilation is the method of combining a predictive model of a system state with noisy partial (or indirect) observations of that system in order to estimate the full system state. Data assimilation has a history in the mathematics of geophysical systems, particularly in weather prediction and oceanographic and atmospheric modeling where chaotic or noisy systems require periodic model recalibration with sequential assimilation of observation data [17, 9, 10].

Generally, data assimilation schemes seek to describe a posterior distribution on the state of the system $x$ given available data $y$ as given by Bayes theorem

$$p(x|y) \propto p(y|x)p(x).$$

The distribution $p(x)$ describes prior knowledge of the state, and the likelihood $p(y|x)$ relates observations to possible state values. Consider a system with dynamics described by

$$\dot{x} = f(x(t), t)dt + g(x(t), t)dW \quad (1)$$

where $x$ is the model state, $f$ is the deterministic component of the dynamics (model drift), $g$ is the stochastic component of the dynamics (model diffusion), and $W$ is a noise process. Let $Y_j$ denote an observation at time $t_j$. Observations are given by $Y_j = H(x(t_j)) + \eta_j$ where $H(\cdot)$ is an observation operator and the $\eta_j$’s are assumed to be independent identically distributed (iid) random variables representing measurement errors.

2.1 Lagrangian Data Assimilation

If the state represents an Eulerian velocity field and the observations are locations of a Lagrangian instrument which is advected by the flow, then the observation operator mapping the flow to the instrument’s locations is nonlocal and nonlinear [2]. Often, the naturally Lagrangian form of the instruments’ data is instead used to derive Eulerian velocity information in order to be assimilated with established Eulerian assimilation schemes which are already associated with many advanced ocean and coastal models used in practice. However, this can require troublesome approximations and interpolations in order to fit Lagrangian location data into forms accepted by Eulerian assimilation schemes [21, 22, 19]. The approach introduced by Kuznetsov et al. [15] known as Lagrangian data assimilation (LaDA) circumvents these difficulties by appending the Lagrangian instrument coordinates to the state, and likewise an advection model to the model equations for the original state variables.

Let the parameters and/or variables that determine the flow be denoted by $\theta(t)$, and let the Lagrangian instrument’s location be denoted by, $\xi(t) = [x(t), y(t), z(t)]^T$. Then the velocity field at the coordinates $\xi(t)$ is given by

$$U(\xi(t); \theta(t)) = [u_x(\xi(t); \theta(t)), u_y(\xi(t); \theta(t)), u_z(\xi(t); \theta(t))]^T \quad (2)$$

and is parameterized by the current flow state $\theta(t)$. The augmented system state becomes $x(t) = [\xi(t)^T, \theta(t)]^T$, and the observations become simple projections onto the instrument’s coordinates at the observation times (plus measurement noise $\eta$). Thus, the $j^{th}$ observation is given by

$$Y_j = H \begin{bmatrix} x(t_j) \end{bmatrix} + \eta_j = [1 \ 0] \begin{bmatrix} x(t_j) \end{bmatrix} + \eta_j. \quad (3)$$
In this Lagrangian data assimilation framework, the system dynamics are given by

\[
\begin{align*}
    dx &= \left[ \frac{d\xi}{d\theta} \right] dt + \left[ \begin{array}{c} K \\ G(\theta(t), t) \end{array} \right] dW
\end{align*}
\]

where \( G \) describes prescribed location changes from the flight plan and \( K \) is the matrix of diffusivity constants (which will be defined in equation 34). When assimilating glider data into a flow model, LaDA will result in a posterior distribution of flow field parameters, as well as a distribution of associated glider trajectories.

### 2.2 Sequential data assimilation

Sequential data assimilation techniques work by iterating a cycle of ‘forecast’ and ‘analysis’ steps for each observation time. Given the distribution of states \( p(x(t_{j-1})|Y_{1,...,j-1}) \) after assimilating all the data up through the previous assimilation time \( t_{j-1} \), we evolve this distribution of states according to equation 4 until the next assimilation time \( t_j \). This is called the ‘forecast’ distribution of states \( p_f(x(t_j)|Y_{1,...,j-1}) \), and it represents our prior knowledge of the state at the assimilation time \( t_j \) given all the previous observations up until (but not including) the observation we are about to assimilate at time \( t_j \). Using the forecast distribution \( p_f(x(t_j)|Y_{1,...,j-1}) \) and the observation \( Y_j \), the ‘analysis’ step approximates the posterior distribution according to Bayes theorem. This is called the ‘analysis’ distribution of states \( p_a(x(t_j)|Y_{1,...,j}) \), and it describes our estimate of the state at time \( t_j \) given all the data up to and including the observation at that time. The analysis distribution at time \( t_j \) is used to compute the forecast distribution at the next assimilation time \( t_{j+1} \).

Many sequential data assimilation techniques stem from the historic Kalman filter [14]. In their seminal paper, Kalman outlines a filter for the optimal update to state estimates given a set of observations of the state. The filter involves propagating a Gaussian distribution of state estimates through time and updating this distribution based on the likelihood of the states generating the observed data given the known measurement error properties. If the dynamics of the system and the observation operator are linear, and if the observation error is Gaussian, then the Kalman filter is the optimal method of combining model forecasts with observational data to update estimates of the state. Because of these constraints, many advanced techniques based on the Kalman filter have been invented which can handle nonlinear systems and non-Gaussian distributions.

Two assimilation methods, the ensemble Kalman filter (EnKF) and the particle filter (PF), are used in this Lagrangian data assimilation study and are briefly outlined below. Both methods use an ensemble of states to approximate the forecast and analysis distributions. The EnKF uses a few ensemble members to compute a Gaussian approximation while the PF uses an ensemble of many members and associated weights to create a discrete approximation to the distributions. The analysis step of the EnKF updates the state values of the ensemble members, while the analysis step of the PF updates the ensemble members’ weights. For a relatively small dimensional system, a particle filter with a large number of particles provides an excellent description of the posterior distribution even when the system in question is strongly nonlinear. The EnKF sometimes struggles.
when faced with strong nonlinearity, but is easily applicable to large dimensional systems and does not suffer from filter divergence as a standard particle filter does in high dimensions.

2.3 The Ensemble Kalman Filter

The ensemble Kalman filter with perturbed observations (summarized here following the work by Evensen [8]) is a sequential data assimilation technique that evolves an ensemble of model states through time and performs Kalman filter style updates as new observations are incorporated.

Given an ensemble of $N_e$ model states at time $t_{j-1}$, each ensemble member is evolved to arrive at an ensemble of forecast model states at the observation time $t_j$. This forecast ensemble is used to generate a Gaussian estimate of the prior distribution at time $t_j$. We denote the forecast ensemble as $\{x_{j,f}^i | i = 1, ..., N_e\}$. The forecast ensemble mean $\bar{x}_{j,f}$ and the forecast ensemble covariance $P_{j,f}$ can be estimated as follows:

$$\bar{x}_{j,f} = \frac{1}{N_e} \sum_{i=1}^{N_e} x_{j,f}^i$$

$$P_{j,f} = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (x_{j,f}^i - \bar{x}_{j,f})(x_{j,f}^i - \bar{x}_{j,f})^T.$$  

Recall that the observations are assumed to have the form $Y_j = H(x_j) + \eta_j$. In this study, the observation errors $\eta_j$ are taken to be iid Gaussian random variables with mean 0 and known covariance $R$ (i.e., $\eta_j \sim \mathcal{N}(0, R)$). We create an ensemble of $N_e$ perturbed observations with mean equal to $Y_j$ and covariance $R$ according to $Y_j^i = Y_j + \epsilon_j^i$ where $\epsilon_j^i \sim \mathcal{N}(0, R)$. Further, we compute the covariance of the ensemble of perturbed observations by

$$R_e^j = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} \epsilon_j^i \epsilon_j^i^T.$$  

During the analysis step, each ensemble member is updated according to

$$x_{j,a}^i = x_{j,f}^i + P_{j,f} H^T (HP_{j,f}H^T + R_e^j)^{-1} (Y_j^i - Hx_{j,f}^i)$$

and the analysis mean can be found with

$$\bar{x}_{j,a} = \frac{1}{N_e} \sum_{i=1}^{N_e} x_{j,a}^i.$$  

The analysis covariance can be estimated by

$$P_{j,a} = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (x_{j,a}^i - \bar{x}_{j,a})(x_{j,a}^i - \bar{x}_{j,a})^T.$$
The analysis ensemble is used to generate a Gaussian approximation of the posterior distribution at time \( t_j \). The analysis ensemble \( \{x_{j,a}^i\} \) is then evolved to the next observation time and used as the forecast ensemble for the next assimilation step. Sequentially, this cycle of forecast/analysis can be used to assimilate each observation.

2.4 The Particle Filter

Particle filtering is a sequential data assimilation technique that uses a (typically large) set of sample states, each with an associated weight, to describe the forecast and analysis distributions ([30]). These represent discrete approximations to the prior and posterior distributions used in Bayes theorem. We start by taking \( N_p \) random samples of the initial state from a prescribed prior distribution at time \( t_0 \), \( x_0^i \sim p(x_0) \), and use \( w_0^i = \frac{1}{N_p} \). The set of particles (state values and corresponding weights) is \( \{(x_0^i, w_0^i)\} \), and these represent a discrete approximation to the starting prior.

To compute the analysis distribution at the \( j \)th observation time, start with the set of particles \( \{(x_{j-1}^i, w_{j-1}^i)\} \) which represent the analysis distribution from the previous observation time \( p_a(x_{j-1}|Y_1,...,j-1) \). An approximation to the prior distribution at the \( j \)th observation time is obtained by evolving the states forward in time to \( t_j \), yielding particles \( \{(x_j^i, w_{j-1}^i)\} \) which are samples from the forecast distribution \( p_f(x_j|Y_1,...,j-1) \). Next, we compute the likelihood of each sampled state generating the given observation \( p(Y_j|x_j^i) \). If the observation error is Gaussian with covariance matrix \( R \) then the likelihood of particle \( x_j^i \) generating the observation \( Y_j \) is given by

\[
p(Y_j|x_j^i) \propto \exp \left[-\frac{1}{2} (Y_j - H x_j^i)^T R^{-1} (Y_j - H x_j^i) \right]. \tag{11}
\]

With the forecast distribution and likelihood computed, the analysis step uses Bayes formula to reweight the particles according to

\[
w_j^i = \frac{p(Y_j|x_j^i) \cdot w_{j-1}^i}{\sum_{k=1}^{N_p} p(Y_j|x_k^j) \cdot w_{j-1}^k} . \tag{12}
\]

The reweighted particles \( \{(x_j^i, w_j^i)\} \) represent the analysis distribution for the \( j \)th observation \( p_a(x_j|Y_1,...,j) \) and form a discrete approximation to the posterior distribution at time \( t_j \).

3 Models

3.1 Two-Layer Coastal Flow Model

We consider a two-layer model which describes geostrophic flow parallel to the coast along the continental shelf ([4, 11]). For simplicity, we assume that the flow is in steady state or is changing at a rate that is slow compared to the time scales of glider missions (though in practice this may not be the case). We consider the case where the lower layer fluid extends partially onto the shelf.
Figure 1: Schematic of the coastal upwelling model and its parameters. Pictured is a vertical cross-shore section perpendicular to the coastline with a constant $y$ value. Each vertical cross-shore section is identical since we assume that there is no change in the flow, bathymetry, or layer interface in the alongshore direction. The coast is at $x = 0$, and the sea floor of the continental shelf slopes from a depth of $z = 0$ at the coast to a depth of $-D_0$ at the shelf break ($x = W$). The layer interface grounds on the continental shelf at a distance $x = B$ from the coast. The depth of the layer interface ($-d_1(x)$) changes in the offshore direction, approaching a farfield depth of $-D_1$. Three different flow regions are highlighted. Region I describes the area with only one layer between the coast and the layer interface grounding ($0 < x < B$). Region II is the two layer region over the continental shelf ($B < x < W$). Region III is the two layer region out past the shelf break ($x > W$).

and grounds at a distance $B$ from the coast creating a coastal upwelling system. Figure 1 shows a schematic depicting the two-layer model over the continental shelf, and table 1 summarizes the model variables, parameters, and dimensional scales used. The continental shelf is represented by a constantly sloping region of the ocean floor just off the coast. The depth increases from zero at the coast to $-D_0$ at the shelf break. The shelf break lies at a distance $W$ from the coast. The interface between the top and bottom layer also slopes down in the off shore direction, as required by the thermal wind relation. The velocity in the offshore direction is zero in each layer. Offshore of the shelf break the total depth is considered infinite and the lower layer alongshore velocity taken to be a known constant $V$. Far offshore past the shelf break, the top layer alongshore velocity approaches the same value $V$ and the depth of the layer interface approaches the constant value of $-D_1$.

For the purposes of this data assimilation experiment, the bathymetry is assumed to be known a priori, while the shape and location of the layer interface is considered unknown. In other words, the depth of the shelf break $-D_0$ and the distance of the shelf break from the coast $W$ are consid-
Variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x$</td>
<td>horizontal distance away from the shore (horizontal location in the offshore direction)</td>
</tr>
<tr>
<td>$y$</td>
<td>horizontal location in the alongshore direction</td>
</tr>
<tr>
<td>$z$</td>
<td>depth</td>
</tr>
<tr>
<td>$d_1(x)$</td>
<td>the thickness of the top layer</td>
</tr>
<tr>
<td>$v_1(x)$</td>
<td>the alongshore velocity of the fluid flow in the top layer</td>
</tr>
<tr>
<td>$v_2(x)$</td>
<td>the alongshore velocity of the fluid flow in the second, bottom, layer</td>
</tr>
</tbody>
</table>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B$</td>
<td>the $x$ location where the layer interface grounds</td>
</tr>
<tr>
<td>$W$</td>
<td>the $x$ location where the shelf break occurs</td>
</tr>
<tr>
<td>$D_0$</td>
<td>the thickness of the water column at the shelf break</td>
</tr>
<tr>
<td>$D_1$</td>
<td>the far-field thickness of the top layer. $-D_1$ is the depth of the top layer far from the shore.</td>
</tr>
<tr>
<td>$g'$</td>
<td>reduced gravity</td>
</tr>
<tr>
<td>$f$</td>
<td>the Coriolis parameter</td>
</tr>
<tr>
<td>$V$</td>
<td>the known constant velocity of the flow far offshore</td>
</tr>
</tbody>
</table>

Dimensional Scales and Substitutions

<table>
<thead>
<tr>
<th>Scale/Substitution</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda = \frac{(g'D_1)^{1/2}}{f}$</td>
<td>a reference scale for horizontal length</td>
</tr>
<tr>
<td>$\beta = \frac{(g'D_1)^{1/2}}{f}$</td>
<td>a reference scale for along-shore velocity</td>
</tr>
<tr>
<td>$D_T(x) = D_0 \frac{x}{W}$</td>
<td>the total thickness of the water column at $x$ (over the shelf, $0 &lt; x &lt; W$)</td>
</tr>
<tr>
<td>$D_B = D_0 \frac{B}{W}$</td>
<td>the thickness of the water column at $x = B$</td>
</tr>
</tbody>
</table>

Table 1: This table summarizes the variables, parameters, and substitutions used in the velocity and layer interface equations of the two-layer coastal upwelling flow model.
ered known, while the far-field depth of the layer interface $-D_1$ and the distance from the coast of the interface grounding $B$ are considered unknown and will be quantified by the assimilation. Other model variables are considered known a priori, such as the fluid density of the top and bottom layer ($\rho_1$ and $\rho_2$), the Coriolis parameter ($f$), and reduced gravity ($g' = (\rho_2 - \rho_1)/\rho_2$). Thus, we consider that the flow can be parameterized by $\theta = [B, D_1]^T$, and the goal of the assimilation is to estimate these parameters.

We let $x$, $y$, and $z$ denote the offshore, alongshore, and vertical directions, with the coast at $x = z = 0$. Here the coastline is considered straight since we assume changes in the topography and flow are small in the alongshore direction relative to the scale of glider missions in the offshore direction. Similarly, the distance $x = W$ from the coast to the shelf break and the distance $x = B$ from the coast to the the layer interface grounding are also both constant in the alongshore direction.

The model contains no wind forcing and no friction, and the flow is assumed to be steady. Changes in the flow in the alongshore ($y$) direction are due entirely to variations in the slope of the shelf or in the position $x = W(y)$ of the shelf break. If the scale over which these variations occur is large compared to the typical $W$, it can be shown that the alongshore velocity components $v_1$ and $v_2$ (in layers 1 and 2) are in geostrophic balance and that the offshore components, while non-geostrophic, are weak. If the alongshore extent of the total glider path is small compared to the same scale, then we can approximate $W$ and the bottom slope as constant with $y$. In this limit, the offshore velocity components in each layer are zero.

Due to the lack of wind forcing and friction, the potential vorticity within each layer is conserved following streamlines. Following Dale and Barth [4], we assume that the potential vorticity $((\partial v_i/\partial x + f)/d_i$ ($i = 1, 2$)) is uniform within each layer. If the flow velocity is assumed to approach a constant far offshore and the layer thicknesses approach values $d_1 = D_1$ and $d_2 = \infty$, then the layer potential vorticities are given by

$$f + \frac{\partial v_1}{\partial x} = \frac{f}{D_1}d_1$$  \hspace{1cm} (13)$$

$$f + \frac{\partial v_2}{\partial x} = 0.$$  \hspace{1cm} (14)

The thermal wind relation gives the following relationship between the flow velocities and the layer interface

$$f(v_1 - v_2) = g'\frac{\partial d_1}{\partial x}.$$  \hspace{1cm} (15)

There are three important quantities which vary in the offshore direction: the alongshore velocity in the top layer $v_1(x)$, the alongshore velocity in the bottom layer $v_2(x)$, and the thickness of the top layer $d_1(x)$. The differential equations given by 13, 14, and 15 provide us with model equations for alongshore velocity and top layer thickness which are described below. For clarity in the following paragraphs we use some substitutions that are outlined in table 1.

In the offshore direction, there are three distinct regions of the flow. The three regions are highlighted in the schematic in figure 1. The first region (I) lies between the coast and the layer
interface grounding \((0 < x < B)\). Here there is only one layer of fluid. Thus, the depth of the top layer is the depth of the whole water column

\[
d_1(x) = D_T(x) = D_0 \frac{x}{W}.
\]

The flow advecting a glider in this region at coordinates \((x, y, z)\) is given by

\[
U(x, y, z; \theta) = \begin{bmatrix}
U_x(x; \theta) \\
U_y(x; \theta) \\
U_z(x; \theta)
\end{bmatrix} = \begin{bmatrix}
v_1(x; \theta) \\
0
\end{bmatrix}.
\]

In region I the alongshore velocity of the second layer does not exist, and the alongshore velocity of the top layer is given by

\[
v_1(x) = \frac{f}{2} \left( \frac{D_T(x)}{D_1} \right) x - \frac{f}{2} \left( \frac{D_B}{D_1} \right) B + \beta e^{\frac{1}{x}(B-W)} - \beta \left( \frac{D_B}{D_1} \right) - fx + fW + V.
\]

The second region (II) represents the area on the continental shelf with two layers of fluid \((B < x < W)\). This region lies between the layer interface grounding \(x = B\) and the shelf break \(x = W\). In region II the depth of the top layer is

\[
d_1(x) = \frac{1}{2} D_1 e^{\frac{1}{x}(x-W)} - \frac{1}{2} D_1 e^{\frac{1}{x}(-x+2B-W)} + D_B e^{\frac{1}{x}(-x+B)}.
\]

The flow advecting a glider in this region is given by

\[
U(x, y, z; \theta) = \begin{bmatrix}
U_x(x; \theta) \\
U_y(x; \theta) \\
U_z(x; \theta)
\end{bmatrix} = \begin{bmatrix}
v_1(x; \theta) \\
0
\end{bmatrix} \quad \text{for} \quad z > -d_1(x; \theta)
\]

\[
U(x, y, z; \theta) = \begin{bmatrix}
U_x(x; \theta) \\
U_y(x; \theta) \\
U_z(x; \theta)
\end{bmatrix} = \begin{bmatrix}
v_2(x; \theta) \\
0
\end{bmatrix} \quad \text{for} \quad z < -d_1(x; \theta).
\]

Here the alongshore velocities of the top and bottom layer are

\[
v_1(x) = \frac{1}{2} \beta e^{\frac{1}{x}(x-W)} + \frac{1}{2} \beta e^{\frac{1}{x}(-x+2B-W)} - \beta \frac{D_B}{D_1} e^{\frac{1}{x}(-x+B)} - fx + fW + V
\]

\[
v_2(x) = -fx + fW + V.
\]

The third region (III) lies out past the shelf break \((x > W)\). In region III the depth of the top layer approaches the farfield value of \(-D_1\) and is given by

\[
d_1(x) = D_1 - \frac{1}{2} D_1 e^{\frac{1}{x}(x+W)} - \frac{1}{2} D_1 e^{\frac{1}{x}(-x+2B-W)} + D_B e^{\frac{1}{x}(-x+B)}.
\]
Here the flow advecting a glider is given by

\[
U(x, y, z; \theta) = \begin{bmatrix} U_x(x; \theta) \\ U_y(x; \theta) \\ U_z(x; \theta) \end{bmatrix} = \begin{bmatrix} 0 \\ v_1(x; \theta) \\ 0 \end{bmatrix} \quad \text{for} \quad z > -d_1(x; \theta) \] (25)

\[
U(x, y, z; \theta) = \begin{bmatrix} U_x(x; \theta) \\ U_y(x; \theta) \\ U_z(x; \theta) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ v_2(x; \theta) \end{bmatrix} \quad \text{for} \quad z < -d_1(x; \theta). \] (26)

In region III out past the shelf break the alongshore velocity of the bottom layer is the known constant \(V\), and the alongshore velocity of the top layer approaches \(V\) far offshore (i.e., \(\lim_{x \to \infty} v_1(x) = V\)). The alongshore velocities are

\[
v_1(x) = \frac{1}{2} \beta e^{\frac{1}{2}(-x+W)} + \frac{1}{2} \beta e^{\frac{1}{2}(-x+2B-W)} - \beta \frac{D_B}{D_1} e^{\frac{1}{2}(-x+B)} + V \] (27)

\[
v_2(x) = V. \] (28)

In order to generate the observation data used for the assimilation experiment, we simulate a ‘true’ glider trajectory advected by a flow field created using this model with a particular set of ‘true’ values for the model variables and parameters: \(B = 5 \text{ km}, W = 20 \text{ km}, D_0 = 150 \text{ m}, D_1 = 150 \text{ m}, g' = 10^{-2} \text{ m/s}^2, f = 10^{-4} \text{ s}^{-1}, \) and \(V = -0.5 \text{ m/s}\). Many of these values are taken from [4]. As stated earlier, in the assimilation experiment the layer interface grounding location \(B\) and the top layer far-field thickness \(D_1\) are considered unknown, and the goal of the assimilation is to estimate these parameters. The flow, as a function of location \(\xi\), is parameterized by the unknown parameters \(\theta = [B, D_1]^T\) and denoted as \(U(\xi; \theta)\). Figure 2 shows the realization of the flow model with the true model parameters.

We model ocean turbulence as a noise process added to the background flow when simulating the glider dynamics. The details of this process are described in section 3.4.

### 3.2 Model of Glider Motion

To propel themselves, gliders manipulate their buoyancy by changing their volume while maintaining a constant mass. Volume change is achieved by moving oil between external bladders and the gliders pressure hull (Spray [28] and Seaglider [7]) or by moving water in and out of a port in the nose (Slocum [27]). The resulting (upward or downward) buoyant force is balanced by lift and drag forces generated by a glider’s hull, wings, and tail ([28, 7, 5]) with resultant nearly-steady motion along saw-tooth-shaped paths through the water (e.g., figure 3). When a glider is actively adjusting its buoyancy, particularly during dive-to-climb or climb-to-dive transitions, the motion is no longer steady, but our model of glider flight will ignore these periods for simplicity.

We restrict our model of glider motion through the water to the \((x, z)\) plane, and in our simulation we assume that the glider has control over its glide angle \(\phi\) and its glide speed \(S\) (with some error). In this way, we are assuming that the glider can control its path through the water relative
Figure 2: The coastal flow model used for the simulated glider mission is pictured. This realization of the model uses the parameter values listed in section 3.1. The top plot shows the bathymetry of the continental shelf and the location of the layer interface. The ‘true’ trajectory of the simulated glider is also shown for reference. The bottom plot shows the upper and lower layer velocities as a function of $x$ (the offshore distance from the coast).

to the advecting flow (with some error). In practice the glider controls its pitch angle $\gamma$ and its volume $V_g$, but with an appropriate model of the lift and drag acting on the glider one can compute the pitch angle and volume that yield the desired glide angle and glide speed in a steady dive at equilibrium. The buoyancy force $F_b$ depends on the difference between the mass of the glider $m_g$ and the mass of the displaced fluid $m_f$

$$F_b = (m_g - m_f)g$$

$$= (m_g - \rho V_g)g$$

where $\rho$ is the local fluid density and $g = 9.8 \text{ m s}^{-2}$ is gravity. The angle of attack $\alpha$ is the difference between the glider’s pitch angle and its glide angle

$$\gamma = \phi + \alpha.$$  

12
Figure 3: Schematic of a single dive by an underwater glider. At the surface, the glider obtains a GPS fix and communicates via satellite to send data and receive commands. The glider then reduces its volume and sinks while maintaining a nose-down pitch to move forward horizontally and steering a specified heading. At maximum depth, the glider increases its volume to begin rising toward the surface. A nose-up pitch during ascent maintains forward motion. At the surface again, the glider obtains a GPS fix and reestablishes communications. Illustrated maximum dive depth, horizontal distance, and dive duration are for a Spray glider, but ratios of dive depth to horizontal displacement and duration are similar for each type of glider. Illustration by Jack Cook, Woods Hole Oceanographic Institution.
The angle of attack takes the opposite sign of both the pitch angle and glide angle ($\alpha$ is positive during descent and negative during ascent). In a steady glide the angle of attack and the glide speed depend on the equilibrium values of lift, drag, and buoyancy. At equilibrium the forces are balanced and we get the following relationship

$$
\begin{bmatrix}
F_b
\end{bmatrix} = \begin{bmatrix}
\cos(\phi) & \sin(\phi)
\end{bmatrix}
\begin{bmatrix}
D
\end{bmatrix} - \begin{bmatrix}
\sin(\phi) & \cos(\phi)
\end{bmatrix}
\begin{bmatrix}
L
\end{bmatrix}.
$$

(31)

The mass of the glider and the equilibrium values of lift, drag, and angle of attack are approximated experimentally by researchers before the glider is deployed ([28, 7, 5, 27]). With these one can relate the desired glide angle and desired glide speed to the glider volume and pitch angle which the glider can control. Thus, for simplicity, we assume that the glider has control over its glide angle $\phi$ and its glide speed $S$.

### 3.3 Glider Mission Plan

The glider’s mission plan prescribes desired glide angles $\phi_d$, desired glide speeds $S_d$, and desired dive depths $-D_d$ for all of the dives that are to be undertaken in the simulation experiment. However, we assume some uncontrollable variability in the glide angle and speed which creates deviations in the ‘true’ trajectory from the desired trajectory. For each dive, a simulated glider is given a small Gaussian perturbation to its glide angle with a noise level of $\sigma_\phi = 1.0^\circ$ and to its glide speed with a noise level of $\sigma_S = 0.01 \text{ m s}^{-1}$. This noise in the glider model represents the inherent uncertainty that comes from an imperfect model, imperfect estimates of model parameters, and noisy sensors. If we take into account the Gaussian noise perturbation to the glide angle $\eta_\phi \sim \mathcal{N}(0, \sigma_\phi^2)$ and the Gaussian noise perturbation to the glide speed $\eta_S \sim \mathcal{N}(0, \sigma_S^2)$ then the true trajectory of the simulated glider due to gliding is given by the true glide angle $\phi_{true} = \phi_d + \eta_\phi$ and the true glide speed $S_{true} = S_d + \eta_S$.

In our experiments we give the glider a mission plan of gliding straight in the offshore ($x$) direction performing $N_{dives} = 14$ dives starting 15 km from the shore ($\xi(t_0) = [x(t_0), y(t_0), z(t_0)]^T = [15 \text{ km}, 0 \text{ km}, 0 \text{ m}]^T$). Each dive consists of a descending portion and an ascending portion, with a desired dive depth of $-D_d = -100 \text{ m}$. The descending portion involves the glider diving with a constant desired glide angle of $\phi_d = -20^\circ$, and the ascending portion involves the glider surfacing with a constant desired glide angle of $\phi_d = 20^\circ$. The glider mission prescribes a desired constant glide speed of $S_d = 0.27 \text{ m s}^{-1}$. Using these quantities ($\phi_d$ and $S_d$), the time it would take the glider to reach the desired depth $-D_d$ and then return to the surface (if no error was present) is $t_{descend} \approx 30 \text{ minutes}$. For simplicity we assume that the glider spends no time at the surface between dives and assume that the turnaround at the bottom of the dive is instantaneous, even though in practice gliders will typically spend about 15 minutes at the surface between dives to communicate via satellite and send data.

For each dive, the simulated glider descends for the prescribed amount of time $t_{descend}$ using the true perturbed glide angle ($\phi_{true} = \phi_d + \eta_\phi$) and glide speed ($S_{true} = S_d + \eta_S$). Thus, for
the descending portion of the dive the velocity component due to buoyancy driven gliding $G(t)$ is given by

$$G(t) = \begin{bmatrix} G_x \\ G_y \\ G_z \end{bmatrix} = \begin{bmatrix} S_{true} \cdot \cos(\phi_{true}) \\ 0 \\ S_{true} \cdot \sin(\phi_{true}) \end{bmatrix},$$  \hspace{1cm} (32)

where $\phi_{true} = -20^\circ + \eta_{\phi}$ and $S_{true} = 0.27 + \eta_S \text{ m s}^{-1}$. For the ascending portion the gliding velocity is also given by equation 32 except where $\phi_{true} = 20^\circ + \eta_{\phi}$.

3.4 Diffusivity, Glider Dynamics, and System State

A low level diffusive process ($\kappa = 1 \text{ m}^2 \text{ s}^{-1}$) is used to model ocean turbulence [20]. One could attempt to account for the presence of unresolved eddies in the model by adding noise in the form $\kappa^{1/2}dW$ to the model velocity. Here $dW$ is a Weiner process with a variance $dt$ and $\kappa$ is a horizontal eddy diffusivity. Estimates of $\kappa$ from surface drifters in the California current can exceed $100 \text{ m}^2 \text{ s}^{-1}$ [33], though these estimates include the affects of the largest eddies. Although our model does not include eddies, modern coastal models would be able to resolve eddies as large as the width of the shelf as well as scales that are considerably shorter. In such models the unresolved eddies would be characterized by a much smaller $\kappa$, and we select $1 \text{ m}^2 \text{ s}^{-1}$ as representative. Over a small timestep $dt$ the contribution to the glider’s displacement due to the noise process is $\sqrt{\kappa}dW$ in the $x$ and $y$ dimensions.

The glider moves due to a combination of advection from the flow $U(\xi; \theta)$, velocity from buoyancy driven gliding $G(t)$, and the diffusion noise process. Section 3.2 outlines how we model the gliding velocity $G(t)$ for a given desired glide angle and glide speed prescribed by the glide mission a priori. The glider’s velocity is also affected by advection from the background flow $U(\xi(t); \theta)$ described in section 3.1, where $\xi(t) = [x(t), y(t), z(t)]^T$ is the glider’s location at time $t$. The glider’s displacement $d\xi$ over a small timestep $dt$ will be the sum of these two velocities integrated over the timestep along with the diffusivity noise contribution. Thus,

$$d\xi = [U(\xi(t); \theta) + G(t)]dt + KdW,$$  \hspace{1cm} (33)

$$K = \begin{bmatrix} \sqrt{\kappa} & 0 & 0 \\ 0 & \sqrt{\kappa} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$  \hspace{1cm} (34)

where $dW = [dW_x, dW_y, dW_z]^T$ is a vector of Weiner processes for each spatial dimension.

As outlined in section 2.1, the system state $x = [\xi, \theta]^T$ contains both the flow model parameters $\theta$ and the glider’s location $\xi$, and the system dynamics describe the change of the flow model parameters $d\theta$ and the movement of the glider $d\xi$ over a small time step $dt$. For the flow model considered here, the unknown model parameters $\theta = [B, D_1]^T$ are constant in time (i.e., $d\theta = 0dt + 0dW$). The glider’s displacement over a small timestep $dt$ is given by equation 33. Thus,
the system dynamics are given by,

\[
\frac{dx}{d\theta} = \begin{bmatrix} \frac{d\xi}{d\theta} \\ U(\xi(t) ; \theta) + G(t) \end{bmatrix} dt + \begin{bmatrix} K \\ 0 \end{bmatrix} dW
\]  

(35)

### 3.5 Glider Data

Gliders use GPS receivers to determine their latitude and longitude with minimal measurement error while at the surface. However, GPS signals do not penetrate below the sea surface, so precise position information is not available while a glider is submerged. We use the subscript \( s \) to denote variables associated with surface location data. To generate the observations, we simulate the true glider trajectory–using the true model parameters to dictate the flow field–and then observe the glider’s location (with some measurement error) at the observation times. For each dive, \( j = 1, \ldots, N_{\text{dives}} \), we simulate the glider trajectory from the previous surfacing time \( t_{j-1,s} \) to the next surfacing time \( t_{j,s} \) by integrating

\[
\xi(t) = \xi(t_{j-1,s}) + \int_{t_{j-1,s}}^{t} (U(\xi(\tau) ; \theta) + G(\tau))d\tau + \int_{t_{j-1,s}}^{t} KdW
\]  

(36)

to obtain \( \xi(t) \) for \( t \) from \( t_{j-1,s} \) to \( t_{j,s} \).

In the case of the glider’s surfacing location, our observation data consist of the \((x, y)\) coordinates of the glider upon surfacing after completing a dive. The observations have the form,

\[
Y_{j,s} = H_s \begin{bmatrix} x(t_{j,s}) \\ y(t_{j,s}) \end{bmatrix} + \eta_{j,s}
\]  

(37)

\[
H_s = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.
\]  

(38)

The additive measurement noise for these observations is mean zero Gaussian \( \eta_{j,s} \sim \mathcal{N}(0, R_s) \) with a measurement error of \( \sigma_s = 10 \) meters and covariance \( R_s = \sigma_s^2 I \). The observation operator \( H_s \) is a projection matrix which selects the \((x, y)\) coordinates of the glider at the surfacing time \( t_{j,s} \).

The glider’s other data sensors (temperature, pressure, salinity, etc.) take measurements with high frequency throughout the entire dive, including the time when the glider is submerged. These other data are typically ignored in Lagrangian data assimilation, which often only uses periodic GPS observations of an instrument’s latitudinal and longitudinal location for assimilation. From measurements of temperature, salinity, and pressure, estimates of seawater density are obtained from the equation of state of seawater [12]. Operating in the two-layer flow model described in section 3.1, a strong and abrupt change in density signifies that the glider has crossed the layer.
boundary. Since the glider can also observe its depth at high frequency by using its pressure sensor, it has the ability to observe the depth at which it has crossed the layer boundary while descending and ascending during a dive. We use the $D$ subscript to denote variables associated with the glider crossing the layer interface while descending and the $A$ subscript to denote variables associated with the glider crossing the layer interface while ascending. For the $j^{\text{th}}$ dive, the depth of the glider when crossing the layer interface while descending and ascending is denoted by $Y_{j,D}$ and $Y_{j,A}$ respectively. These observations have the form,

\[
Y_{j,D} = H_D \times (t_{j,D}) + \eta_{j,D} \\
= [z(t_{j,D})] + \eta_{j,D} \\
Y_{j,A} = H_A \times (t_{j,A}) + \eta_{j,A} \\
= [z(t_{j,A})] + \eta_{j,A}
\]

where $t_{j,D}$ is the time when the glider crosses the layer interface while descending during its $j^{\text{th}}$ dive and $t_{j,A}$ is the time when the glider crosses the layer interface while ascending during its $j^{\text{th}}$ dive. The additive measurement noise for these observations is mean zero Gaussian $\eta_{j,D} \sim \mathcal{N}(0, R_D)$ and $\eta_{j,A} \sim \mathcal{N}(0, R_A)$ with a measurement error of $\sigma_L = 1$ meter and known covariances $R_D = R_A = \sigma_L^2$. The observation operators $H_D$ and $H_A$ are projection matrices which select the $z$ coordinate of the glider at the observation times $t_{j,D}$ and $t_{j,A}$. Once we have simulated the true glider trajectory using equation 36, we can generate our set of observation data for the $j^{\text{th}}$ dive by applying equations 37, 39, and 40 to the simulated glider trajectory.

4 Lagrangian Assimilation of Glider Data into a Layered Coastal Model

Our objective is to show how Lagrangian data assimilation techniques can be used on glider-type data to describe the surrounding flow field as well as forecast glider trajectories. To this end, we perform identical twin simulation experiments. The “true” glider trajectory is the result of simulating a glider mission in a particular realization of our flow field model (described in section 3.1) that is parameterized by “true” values of the flow model parameters. For each dive, the observation data is created by “observing” the true glider trajectory and adding measurement noise (described in section 3.5). We then, with the LADA framework, assimilate this data into the coastal flow model. This section outlines the specific extended Kalman filter and particle filter implementations used for our assimilation experiment. For comparison we also compute a spatially and temporally averaged estimate of the flow constructed through dead-reckoning integration of the glider’s velocity through the water (a common practice used by operational gliders) which is also described below.
The output of our assimilation is a series of forecast and analysis probability distributions of the flow model parameters \( \theta = [B, D_1]^T \) as well as the glider’s location at each of the surfacing times. Some central tendency of these distributions (mean, median, etc.) can be used as an estimate of the flow for that dive, while the shape of the distributions (variance and higher moments) provide a description of the uncertainty of the estimate. Samples from the forecast or analysis distributions can be used to simulate glider trajectories that together comprise a distribution of trajectories associated with the distribution of flow parameters. So, the assimilation not only describes the flow field but also provides estimates of the glider’s trajectory and a description of the uncertainty in those estimates. Furthermore, sample trajectories from an analysis distribution can be used to estimate the glider’s location while subsurface and hence provide spatial footprints for measurements collected during the past dive. Likewise, sample trajectories from the forecast distributions can be used to forecast the glider’s path in future dives and aid in aiming the glider to desired locations.

4.1 EnKF

We use the ensemble Kalman filter with perturbed observations described in section 2.3 to assimilate observations of the glider’s depth when crossing the layer interface as well as observations of the glider’s location upon surfacing. The assimilation estimates the layer interface grounding location \( B \) and the farfield upper layer thickness \( D_1 \) which parameterize the flow.

Following the LaDA framework outlined in section 2.1, the \( N_e = 100 \) ensemble member states have the form \( x^i(t) = [\xi^i(t), \theta^i]^T = [x^i(t), y^i(t), z^i(t), B^i, D_1^i]^T \), for \( i = 1, \ldots, N_e \). Each member of the initial ensemble \( x^i_0 \) is given a perturbation of the glider’s starting location

\[
\xi^i_0 = \xi(t_0) + \begin{bmatrix} \xi^i_x \\ \xi^i_y \\ 0 \end{bmatrix}
\]

where the perturbations \( \epsilon^i = [\epsilon^i_x, \epsilon^i_y]^T \) share the characteristics of the surfacing observation noise \( (\epsilon^i \sim \mathcal{N}(0, R_s)) \). The initial flow model parameters \( \theta^i \) are drawn from a given initial distribution \( p(\theta) \) yielding the initial ensemble:

\[
\{x^i_0\} = \left\{ \begin{bmatrix} \xi^i_0 \\ \theta^i \end{bmatrix} \bigg| i = 1, \ldots, N_e \right\}
\]

The starting joint distribution \( p(\theta = [B, D_1]^T) \) on \( B \) and \( D_1 \) is a truncated Gaussian with mean \([10 \text{ km}, 100 \text{ m}]^T\) and a diagonal covariance matrix with entries \( \sigma_B^2 = 5^2 \text{ km}^2 \) and \( \sigma_{D_1}^2 = 50^2 \text{ m}^2 \). Values of \( B \) are restricted to those lying on the continental shelf \( (B \in [0, W]) \), and values of \( D_1 \) must be positive.

Since we are assimilating surfacing observations as well as observations of the glider’s depth when crossing the layer interface, we perform three separate assimilation passes for each dive—one for each of the observations acquired during that dive (the depth where the glider crosses the layer interface while descending and ascending and the location where the glider surfaces). For each
pass, we evolve each member of the ensemble from its starting surface location until the time of the observation and use that collection of states as the forecast ensemble for the assimilation of that observation. The ENKF updates each ensemble member’s estimate of the flow parameters $\theta_i$. The updated parameters are then used with the same starting surface locations to compute the forecast ensemble for the next pass. A detailed description of each of the three passes is outlined below.

**Pass 1: Descending Layer Depth Observation**

For each dive, we start with the analysis ensemble from the previous dive and denote it

$$\{x_{i\text{init}}^i\} = \left\{\begin{bmatrix} \xi_{\text{init}}^i \\ \theta_{\text{init}}^i \end{bmatrix}\right\},$$

(45)

where $\xi_{\text{init}}^i$ denotes the starting surface location of the simulated glider for the $i^{th}$ ensemble member and $\theta_{\text{init}}^i$ denotes the estimate of the flow model parameters for that ensemble member. For the first dive we use the initial ensemble described above in equation 44 (i.e., $\{x_{\text{init}}^i\} = \{x_0^i\}$).

For each member of the ensemble, the state $x_{\text{init}}^i$ is evolved (according to the dynamics prescribed by $\theta_{\text{init}}^i$) until the simulated glider crosses the layer interface while descending. This ensemble of states represents the forecast ensemble for the first pass of assimilation, denoted as $\{x_{fD}^i\} = \{[\xi_{fD}^i, \theta_{fD}^i]^T\}$. From this ensemble we can compute the forecast ensemble mean $\bar{x}_{fD}$ and forecast ensemble covariance $P_{fD}$ according to equations 5 and 6. Further, we create an ensemble of perturbed observations $\{Y_D^i\}$ and compute its covariance $R_D$ according to 7 and 7. In the first pass assimilation we update the flow model parameters with a modified version of equation 8

$$\theta_{aD}^i = \theta_{fD}^i + \Phi P_{fD} H_D^T (H_D P_{fD} H_D^T + R_D)^{-1} (Y_D^i - H_D x_{fD}^i)$$

(46)

$$\Phi = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$  

(47)

The update equation is modified from equation 8 to include the projection matrix $\Phi$ so that only the flow model parameters $\theta$ are updated.

**Pass 2: Ascending Layer Depth Observation**

For the next pass of assimilation we start with the ensemble

$$\{x_{\text{init}}^i\} = \left\{\begin{bmatrix} \xi_{\text{init}}^i \\ \theta_{aD}^i \end{bmatrix}\right\},$$

(48)

which contains the original glider starting surface locations $\xi_{\text{init}}^i$ from equation 45 but uses the updated model parameters $\theta_{aD}^i$ from the first pass assimilation (from equation 46). For each member of the ensemble, the state $x_{\text{init}}^i$ is evolved according to the dynamics prescribed by $\theta_{aD}^i$ until the simulated glider crosses the layer interface while ascending. This ensemble of states represents
the forecast ensemble for the second pass of assimilation, denoted as \( \{ x^i_{fa} \} = \{ [ \xi^i_{fa}, \theta^i_{fa} ]^T \} \). The second pass assimilation updates the flow model parameters according to
\[
\theta^i_{fa} = \theta^i_{fa} + \Phi P^f_a H^T_a (H_A P^f_a H^T_A + R^e_A)^{-1} (Y^i_A - H_A x^i_{fa})
\] (49)
where \( \Phi \) is the same projection matrix from the first pass assimilation (from equation 47).

**Pass 3: Surfacing Location Observation**

Similarly, for the third and final pass of assimilation we start with the ensemble
\[
\{ x^i_{init} \} = \left\{ \begin{bmatrix} \xi^i_{init} \\ \theta^i_{aA} \end{bmatrix} \right\}
\] (50)
which contains the original glider starting surface locations \( \xi^i_{init} \) from equation 45 but uses the updated model parameters \( \theta^i_{aA} \) from the second pass of assimilation. For each member of the ensemble, the state \( x^i_{init} \) is evolved (according to the dynamics prescribed by \( \theta^i_{aA} \)) until the simulated glider surfaces. This ensemble of states represents the forecast ensemble for the third pass of assimilation, denoted as \( \{ x^i_s \} = \{ [ \xi^i_s, \theta^i_s ]^T \} \). The third pass analysis ensemble is computed by updating each member of the forecast ensemble according to
\[
x^i_a = x^i_s + P^f_s H^T_s (H_s P^f_s H^T_s + R^e_s)^{-1} (Y^i_s - H_s x^i_s).
\] (51)
In this final pass we update the entire system state instead of just the flow model parameters, and this ensemble of updated states \( \{ x^i_a \} \) is the analysis ensemble for the entire dive. For the next dive, this analysis ensemble is used as the initial ensemble in equation 45 for the first pass of assimilation, and the process repeats.

### 4.2 Particle Filter

We use the particle filter described in section 2.4 to assimilate observations of the glider’s depth when crossing the layer interface as well as observations of the glider’s location upon surfacing. The assimilation estimates the layer interface grounding location \( B \) and the farfield upper layer thickness \( D_1 \) which parameterize the flow. Following the LaDA framework outlined in section 2.1, the particles have the same form as the ensemble members for the EnKF, \( \mathbf{x}^i(t) = [ \xi^i(t), \theta^i ]^T = [ x^i(t), y^i(t), z^i(t), B^i, D_1^i ]^T \). At the start of the experiment, we create a "cloud" of \( N_p = 10,000 \) particles, in the same way we create the initial ensemble for the EnKF. The perturbations to the starting glider location \( \epsilon^i \) and the initial flow model parameters \( \theta^i \) are drawn from the same distributions as the initial ensemble of the EnKF, and the initial cloud of particles has a similar form to the ensemble of model states from the EnKF,
\[
\{( x^i_0, w^i_0 ) \} = \left\{ \begin{bmatrix} \xi^i_0 \\ \theta^i_0 \end{bmatrix}, w^i_0 \right\} \mid i = 1, ..., N_p \right\}.
\] (52)
Each particle represents an initial estimate of the unknown model parameters and begins with an initial weight of $w_0^i = \frac{1}{N_p}$.

For the assimilation of the $j$th dive, we construct an initial cloud of particles for this particular assimilation cycle (similar to the ensemble in equation 45). We denote this initial cloud as

$$\{ (x_{i,\text{init}}^i, w_{j-1}^i) \} = \left\{ \left[ \begin{array}{c} \xi_{i,\text{init}}^i \\ \theta^i \end{array} \right], w_{j-1}^i \right\} | i = 1, ..., N_p \}.$$  (53)

This is the cloud that will be evolved through time to yield forecast distributions for the observations of this dive, and its construction is described below.

For the first dive ($j = 1$), we use the initial cloud $\{ x_{i,\text{init}}^i, w_0^i \} = \{ (x_0^i, w_0^i) \}$ described above in equation 52. For all further dives ($j > 1$), we will have a resulting analysis distribution from completion of the assimilation of the first $j-1$ dives. We denote this as

$$\{ (x_{j-1}^i, w_{j-1}^i) \} = \left\{ \left[ \begin{array}{c} \xi_{j-1}^i \\ \theta^i \end{array} \right], w_{j-1}^i \right\} | i = 1, ..., N_p \}.$$  (54)

Using this set of particles, we resample the glider’s surfacing location for each particle in the cloud. Since we were able to observe the glider’s true surfacing location after the $j$th dive, we resample the particles’ glider surface locations centered around that observed location and with the known measurement error statistics. Let the observed surface location from the $j-1$th dive be denoted as $\xi_{j-1,s}^{\text{obs}} = [Y_{j-1,s}, 0]^T = [x_{j-1,s}^{\text{obs}}, y_{j-1,s}^{\text{obs}}, 0]^T$. Then, the set of resampled glider locations is

$$\{ \xi_{i,\text{init}}^i \} = \left\{ \xi_{j-1,s}^{\text{obs}} + \left[ \begin{array}{c} \epsilon_x^i \\ \epsilon_y^i \\ 0 \end{array} \right] \right\},$$  (55)

where the perturbations $\epsilon^i = [\epsilon_x^i, \epsilon_y^i]^T$ share the characteristics of the surfacing observation noise ($\epsilon^i \sim \mathcal{N}(0, R_s)$). The initial cloud (equation 53) is constructed by combining these resampled surface locations $\{ \xi_{i,\text{init}}^i \}$ with the model parameters $\{ \theta^i \}$ from the particles in the analysis distribution of the $j$th – 1 dive. Thus, the initial cloud is

$$\{ (x_{i,\text{init}}^i, w_{j-1}^i) \} = \left\{ \left[ \begin{array}{c} \xi_{i,\text{init}}^i \\ \theta^i \end{array} \right], w_{j-1}^i \right\},$$  (56)

where $\xi_{i,\text{init}}$ is described in equation 55 and $\theta^i$ comes from equation 54. This resampling helps to avoid numerical instability and filter divergence caused by effectively useless particles with near zero weight traveling far away from the realistic spatial domain of the glider’s mission because of the particle’s incredibly poor estimate of the flow field. It also helps to avoid issues where a particle with a reasonable estimate of the flow field is punished with a low weight for the entire rest of the mission when it only had few surfacings far from the observed locations due to unusually strong realizations of system and model noise for those particular dives.
We compute the forecast distribution for the $j$th dive by evolving this initial cloud forward in time until each particle’s simulated glider surfaces and completes the $j$th dive. We denote these states as $x^i_{fs}$, and denote the forecast distribution as $\{(x^i_{fs}, w^i_{j-1})\}$. Since we integrated the states for the entire dive, we can similarly compute the forecast distribution’s states when they crossed the layer interface while descending and ascending. These are given by $\{(x^i_{fD}, w^i_{j-1})\}$ and $\{(x^i_{fA}, w^i_{j-1})\}$, where $x_{fD}$ and $x_{fA}$ are the state values when the particle’s simulated glider crosses the layer interface while descending and ascending respectively.

The individual likelihoods for each observation during the $j$th dive are given by equation 11 with the specific observation ($Y_{j,D}$, $Y_{j,A}$, and $Y_{j,s}$), the specific observation operator ($H_D$, $H_A$, and $H_s$), the specific observation error covariance ($R_D$, $R_A$, and $R_s$), and the specific particle states at the observation times ($x^i_{fD}$, $x^i_{fA}$, and $x^i_{fs}$). We compute the overall likelihood for all of the observations during the $j$th dive by multiplying the individual likelihoods together. Let $Y_{j,*} = [Y_{j,D}, Y_{j,A}, Y_{j,s}]^T$ be a composite of all three of the observations during the $j$th dive, and let $x^i_{f,*} = [x^i_{fD}, x^i_{fA}, x^i_{fs}]^T$ be a composite of all the particle states at their respective observation times. Similarly, we create a composite observation operator that maps the particle states at each of the observation times to the composite observation

$$H_* = \begin{bmatrix} H_D & 0 & 0 \\ 0 & H_A & 0 \\ 0 & 0 & H_s \end{bmatrix}.$$  

The matrices $H_D$, $H_A$, and $H_s$ in equation 57 are the same as those defined in equations 41, 42, and 38. Further, we denote the composite observation covariance as

$$R_* = \sigma_a \begin{bmatrix} R_D & 0 & 0 \\ 0 & R_A & 0 \\ 0 & 0 & R_s \end{bmatrix}.$$  

Since the observation noise is considered Gaussian, we can use this notation to write the overall likelihood of all the observations during the $j$th dive as

$$p(Y_{j,*}|x^i_{f,*}) \propto \exp \left[ -\frac{1}{2} [Y_{j,*} - H_* x^i_{f,*}]^T R_*^{-1} [Y_{j,*} - H_* x^i_{f,*}] \right].$$  

During the update step, we use this likelihood to reweight the particles by computing $\{w^i_j\}$ as described in equation 12. The analysis distribution will be $\{(x^i_a, w^i_j)\}$, which contains the states after the simulated gliders have completed the $j$th dive ($x^i_a = x^i_f$), along with their associated updated weights $w^i_j$. This analysis distribution will be used to create the initial cloud to be used for the assimilation of the $j$th + 1 dive.

In some applications the likelihood is very peaked causing numerical instability when the weights are computed. In these cases, one can artificially inflate the variance with an inflation term $\sigma_a$ by using $\sigma_a R_*$ instead of $R_s$ when computing the likelihood in equation 59 ([11]). We use a value of $\sigma_a = 25$ in our simulations.
4.3 Glider-Based Current Estimates

In the absence of position fixes underwater, gliders can use dead reckoning to estimate their position underwater. Measurements of pitch, roll, heading, and depth at regular intervals, are combined with established models of glider flight to estimate a glider’s horizontal velocity through the water. Integration of this estimate over the course of a dive yields a dead-reckoned estimate of displacement during the dive. The difference between this estimated surfacing position and the position fix obtained by GPS following the dive is due to the net effects of currents on the glider during the dive. Divided by the duration of the dive, this difference gives an estimate of the ocean currents averaged along the path of the glider (e.g., [35, 24, 36]). Figure 4 shows the difference between the observed surfacing location and the dead reckoning estimate of the surfacing location with no background flow.

We take \( t_{j-1,s} \) and \( t_{j,s} \) to be the starting and ending times of the \( j \)th dive, and \( \xi_{j-1,s}^{obs} = [Y_{j-1,s}, 0]^T = [x_{j-1,s}^{obs}, y_{j-1,s}^{obs}, 0]^T \) and \( \xi_{j,s}^{obs} = [Y_{j,s}, 0]^T = [x_{j,s}^{obs}, y_{j,s}^{obs}, 0]^T \) to be the observed starting and ending surface locations of the glider for the \( j \)th dive. The dead-reckoned estimate of the glider’s surfacing location is denoted as \( \xi_{j}^{DR} \) and computed by

\[
\xi_{j}^{DR} = \xi_{j-1,s}^{obs} + \int_{t_{j-1,s}}^{t_{j,s}} G(\tau) \, d\tau
\]  

(60)

where \( G(\cdot) \) is the glider’s through-water velocity given by the model of glider motion described in section 3.2. We denote the difference between the dead-reckoned estimate of the glider’s surfacing location and the observed surfacing location as \( \Delta_j = \xi_{j,s}^{obs} - \xi_{j}^{DR} \) and the total time of the dive as \( T_j = t_{j,s} - t_{j-1,s} \). The dead reckoning estimate of averaged current for the \( j \)th dive is \( \bar{U}_j = \frac{\Delta_j}{T_j} \). Using this estimate of averaged flow, we can forecast the next dive by integrating

\[
\xi_{j}^{DRF}(t) = \xi_{j,s}^{obs} + \int_{t_{j,s}}^{t} (\bar{U}_j + G(\tau)) \, d\tau.
\]  

(61)

5 Simulation Experiments

5.1 Experimental Results

This section outlines the assimilation experiment described in the previous sections and shows the results. We begin by simulating a “true” glider mission according to the mission guidelines laid out in section 3.3. The entire true trajectory is the result of integrating the glider dynamics described by equation 4 for the entire duration of the mission (all \( N_{dives} = 14 \) dives). In this integration, each component of the dynamics is computed with “true” values of model parameters and “true” realizations of the stochastic variables, although these are considered unknown to the assimilation
processes. At the observation times of each dive (the times when the glider crosses the layer interface and the times when the glider surfaces), the observation data is created according to section 3.5 by adding realizations of measurement noise to the true glider trajectory at the observation times.

The flow field $U(\cdot)$ advecting the glider when simulating the true mission is the two-layer coastal upwelling system outlined in section 3.1 along with the parameter values described in that section (including the parameters $\theta = [B, D]^T = [5 \text{ km}, 150 \text{ m}]^T$ which are unknown to the assimilation processes and will be estimated). The velocity component due to buoyancy driven gliding $G(t)$ is prescribed by the mission plan outlined in section 3.3 ($N_{\text{dives}} = 14$ identical dives in the offshore direction aimed at a target depth of $-D_d = -100$ m with a desired glide angle of $\phi_d = \pm 20^\circ$ and desired glide speed of $S_d = 0.27$ m s$^{-1}$). However, recall that the true $G(t)$ component will also include a true realization of the glide perturbations ($\eta_\phi$ and $\eta_S$) described in section 3.3 and equation 32. For each dive, the perturbations are sampled randomly according to the statistics described in section 3.3. The true trajectory also depends on a true realization of the diffusivity noise process which is sampled according to the statistics described in section 3.4. Figure 5 shows the path of the true glider over the course of its mission.

With the observations computed from the simulated true trajectory, we use the ensemble Kalman filter (EnKF) and particle filter (PF) separately to assimilate the observations and compare the out-

---

**Figure 4:** A schematic depicting how the glider-based average flow velocity is computed through dead-reckoning integration of its gliding velocity. The distance marked by $\Delta$ in the figure represents the total horizontal displacement of the glider due to the flow. This distance divided by the total time of the dive yields the average flow velocity advecting the glider.
puts. The assimilation is done sequentially for each dive, assimilating all three observations of the
dive together as described in sections 4.1 and 4.2. When integrating the glider dynamics to evolve
the ensembles’ states for both the EnKF and the PF, each ensemble member uses its own realiza-
tion of the diffusivity noise process as well as its own realizations of the perturbations to the glide
angle and glide speed for each dive. Evolving the ensemble members is necessary to compute the
forecast distributions of both the EnKF and the PF for each dive. However, with the PF, the glider
trajectories created by these evolutions also serve as forecast trajectories associated with the flow
field described by each particle. Taken as a whole, these trajectories and their associated particles’
weights form a distribution of forecast glider trajectories for the upcoming dive. These distributions
of forecast trajectories are highlighted in figures 5 and 6. We also perform glider-based estimates
of the average flow for each dive as described in section 4.3. Forecast trajectories based on these
average flow estimates as well as the EnKF assimilation are also shown in figures 5 and 6.

Figure 5 shows the entire simulated glider mission, while figure 6 highlights the 5th dive of
the mission for detail. The “DR forecast” comes from the glider-based dead-reckoning average
flow estimate described in section 4.3. The forecast trajectory is computed by performing dead-
reckoning integration of the glider’s buoyancy driven glide velocity \( G(\cdot) \) plus the glider-based
average flow velocity computed from the previous dive. The EnKF estimate shown in the plots is
the trajectory using the forecast mean of the flow parameters. No perturbations to the glider model
are applied to this trajectory (\( \eta_\phi = 0 \) and \( \eta_S = 0 \) in equation 32), nor is any diffusivity noise
applied to the trajectory (\( K = 0 \) in equation 35). The PF estimates show some of the top weighted
particle’s forecast trajectories.

Figure 7 shows the posterior distributions of \( B \) and \( D_1 \) from the ENKF and PF. The posteriors
from a few different dives are highlighted to show the change in the posterior distributions as more
data is assimilated. As expected, the posterior distributions’ centers move towards the true model
parameters as more data is assimilated. Further, the posterior’s become more peaked around those
centers because the assimilation techniques become more confident in their descriptions of the flow
field as more data is assimilated.

Figure 8 presents various error statistics between the “truth” and the different flow estimating
techniques. The top plot shows the average and maximum error between the forecasted trajectories
and the truth for each dive of the mission. The bottom two plots show errors between the assimila-
tion techniques’ descriptions of the flow model parameters and the true values of the parameters.
The EnKF errors were computed by measuring the distance between the mean of the EnKF fore-
cast distribution and the truth at each dive. The PF errors were computed by taking the weighted
mean distance between all particles and the truth. The “DR” errors were computed by measuring
the distance between the true trajectory and the forecast trajectory computed by performing dead-
reckoning integration of the glider’s velocity plus the glider-based average flow velocity computed
from the previous dive (as described in section 4.3).
Figure 5: An overhead view of the simulated glider’s true trajectory through the entire mission is shown along with different forecast estimates of the glider’s trajectory. The true glider’s location is shown in red. Shown in cyan is the glider’s trajectory forecasted using the glider-based average flow computed from the previous dive. The glider’s trajectory advected by the flow using the mean parameters of the ENKF’s forecast distribution is shown in magenta. The forecasted glider trajectory from 30 high weighted particles is shown in green to illustrate what some of the PF estimated trajectories look like. In the large plot, surfacing locations of the simulated gliders are marked with a square. A zoomed in plot of the 5th dive is also shown for detail. A more detailed figure depicting the 5th dive of the glider’s mission is shown in figure 6.
Figure 6: The simulated glider’s true trajectory for the 5th dive of the mission is shown along with different forecast estimates of the glider’s trajectory. The true glider’s location is shown in red. Shown in cyan is the glider’s trajectory forecasted using the glider-based average flow computed from the previous dive. The glider’s trajectory advected by the flow using the mean parameters of the ENKF’s forecast distribution is shown in magenta. The forecasted glider trajectory from 30 high weighted particles is shown in green to illustrate what some of the PF estimated trajectories look like. Locations at special times are denoted with special markers. Diamonds mark the location of simulated gliders at the time when the true glider crossed the layer boundary while descending. Squares mark the location of simulated gliders at the time when the true glider crossed the layer boundary while ascending. Surfacing locations of the simulated gliders are marked with an x.
Figure 7: The posterior distributions of the model parameters $B$ and $D_1$ are shown after 4 different dives. The assimilation is performed on both surfacing location observations as well as layer interface crossing depth observations while descending and ascending as described in section 3.5. The approximations to the posterior distribution are computed using the assimilation techniques outlined in section 4.1 and section 4.2. For each of the 4 dives, we show the particle filter and EnKF approximations of the joint posterior distribution. We plot each particle’s values of $B$ and $D_1$ colored by the particle’s weight. The EnKF estimate shows the Gaussian approximation to the posterior derived from the analysis mean and covariance computed from the ensemble. The ellipses show 1 and 2 standard deviations from the mean.
Figure 8: Error statistics between the truth and different flow estimation techniques is shown. The top plot shows the average and maximum error between the forecasted trajectories and the truth for each dive of the mission. The bottom two plots show errors between the assimilation techniques descriptions of the flow model parameters and the true values of the parameters. EnKF errors were computed using the mean of the forecast distribution at each dive. PF errors were computed using a weighted mean sum of the individual errors from each particle.
6 Conclusion and Future Work

In this work, we construct a framework for assimilating quasi-Lagrangian ocean glider data in order to update descriptions of the advecting flow. The Lagrangian data assimilation (LaDA) framework introduced by Kuznetsov et al. [15] is adapted to be used on observation data collected by ocean gliders while both on the surface and at depth. As a proof of concept, we simulate a glider mission advected by a two-layer coastal upwelling flow model and use the observational data collected from this simulated glider mission to perform data assimilation according to the updated LaDA framework outlined in this work. The data assimilation is performed using both an ensemble Kalman filter approach and a particle filter approach, both of which are standard data assimilation techniques used across a variety of applications.

From the results of this experiment, it is clear that both the EnKF and the PF are able to qualitatively capture the shape of the glider’s trajectory. Figures 5 and 6 highlight this agreement between the assimilations’ estimated trajectories and the glider’s true trajectory underwater. Also, these figures show the failure of the traditional glider-based average flow estimate to reconstruct qualitatively similar subsurface trajectories. The errors in figure 8 reinforce these conclusions by showing quantitatively that the LaDA techniques’ estimated trajectories agree more with the truth than the trajectory estimates based on the glider-based average flow.

The average error of the glider-based average flow forecasts are not much worse than the average error for the EnKF mean forecasts, but the maximum error for each dive is worse. This happens because the glider-based average flow estimate assumes that the flow is constant for the entire dive, and this assumption yields large errors between the forecasted location and the truth when spatial variance in the flow velocity is high. In particular, since the glider spends a significant portion of each dive in both layers and since the flow velocity is much different between layers, any forecasted trajectory which assumes the flow is constant will mischaracterize the shape of the trajectory.

Furthermore, after only a few dives and assimilation cycles, the posteriors of both the EnKF and the PF have begun to close in on the true flow model parameters even though we choose a broad initial prior distribution on the flow parameters—which represents a large amount of initial uncertainty in the flow field before any assimilation is performed. Figure 7 shows that both techniques yield posteriors centered near the true model parameter values and becoming more peaked as more data observations are assimilated. The posteriors that result from the data assimilation are important for both providing estimates of the flow field and also characterizing our uncertainty in those estimates. Posteriors centered around the true model parameters will give good descriptions of the entire flow field, and posteriors that are narrowly peaked tell us that the uncertainty or variance in those descriptions is low.

6.1 Future Work

One glaring simplification made in this work is the assumption that the flow field is time invariant over the course of the glider mission. In a sense, our simulation experiment supposes that the flow field does change over time otherwise we would not require glider data assimilation to update our
description of the flow. However, our assumption that the flow model parameters are time invariant in our system dynamics \( d\theta = 0dt + 0dW \) in equation 33) means that we are assuming that the change in the flow happens at a rate which is very slow compared to the length of time of the glider’s mission. Though many oceanic flow structures do change slowly, tides, eddy propagation, and other faster scale coastal phenomena omitted from our experiments can affect operational gliders on timescales which are relevant to their missions. What if one wanted to model these changes to the flow field state that occur on faster time scales? One of the strengths of the LaDA technique is that the ability to handle a time-varying system is already naturally built into the assimilation framework, even if this particular aspect was not exploited in the experiments presented in this work. If one had a model for the change in the flow field parameters \( \theta \) over time, a description of these dynamics simply needs to be substituted into the \( f_\theta(\cdot) \) and \( g_\theta(\cdot) \) terms in equation 4 (which would subsequently replace the zero terms in the bottom rows of equation 33). Then, the rest of the steps of our experiments can be repeated without any major changes as long as the additions of \( f_\theta(\cdot) \) and \( g_\theta(\cdot) \) to the state dynamics are kept consistent throughout all of the simulation and assimilation steps that involve integrating the system state through time. We believe one promising direction for the future work of this project involves exploring the application of LaDA to glider data in time-varying flow field models.

Another simplification made in this work is the low dimensionality of the flow model. Many operational coastal models compute velocity fields on large spatial grids which leads to hundreds or thousands of state dimensions. Historically, the particle filter struggles with large dimensional systems because of the unwieldy amount of particles and computational complexity necessary to sample enough of the state space. One of the reasons why the particle filter was successful in our experiment was because our model only consisted of two dimensions. Work by Van Leeuwen and others ([38, 37, 39]) has investigated modifications to typical particle filters in order to use them on high dimensional geophysical systems. Moving forward, assimilation of operational glider data into operational coastal models will likely require either abandoning the particle filter for other techniques or intelligently modifying the basic PF approach to handle the large dimensions of traditional coastal models. Ensemble Kalman filters and other variations on the Kalman filter, offer alternatives to the particle filter which typically perform well in the face of increasing state dimensionality. However, operational coastal models are also traditionally nonlinear which present their own problems for Kalman filter style techniques. Ocean gliders offer a new tool to help researchers describe coastal flows, but successful future integration of glider data assimilation techniques into existing coastal system frameworks will depend on overcoming these and other hurdles presented by the depth dependent, nonlinear, and quasi-Lagrangian aspects of glider data.
References


