

# Distributed Motion Tomography for Time-Varying Flow Fields

Dongsik Chang and Fumin Zhang  
School of Electrical and  
Computer Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332  
Email: {dsfrancis3,fumin}@gatech.edu

**Abstract**—Knowledge of a flow field is crucial to guide oceanic mobile sensing platforms; yet, existing regional ocean models provide insufficient spatial and temporal resolutions for precise guidance of mobile platforms. By combining computational ocean models with real-time data streams collected from mobile platforms, generic environmental models (GEMs) provide high-resolution predictions of ocean currents near the mobile platforms. Motion tomography (MT), which is a novel method for constructing GEMs, employs trajectory information of multiple mobile platforms to create a high-resolution spatial map of ocean flow in the region traversed by the mobile platforms. This paper extends the MT method to resolve the coupling between temporal variations and spatial variations in flow modeling. To incorporate temporal variability of flow into MT mapping, MT employs a parametric flow model and constructs a time-varying flow field by estimating the parameters of the flow model. The original MT problem deals with collective information of all the mobile platform trajectories. To address the communication limitation for data collection, distributed MT is developed in which mobile platforms share their estimated model parameters with other mobile platforms nearby. The paper demonstrates that the proposed methods in both non-distributed and distributed fashion successfully construct an underlying time-varying flow field that affects the trajectories of mobile platforms.

## I. INTRODUCTION

Ocean flow modeling is an active research area in oceanography. Following considerable achievements in ocean flow modeling, ocean models have found applications in various areas such as oceanic weather forecast and marine robotics. Because of the significance of ocean predictions in these areas, extensive studies have been conducted to develop methods for improving the accuracy of ocean models, including classical methods involving Lagrangian drifters and data assimilation schemes incorporating various ocean observing technologies. Furthermore, the recent introduction of mobile sensing platforms such as autonomous underwater vehicles (AUVs) in oceanographic and marine robotics research brings new techniques for ocean model error analysis (e.g., [1]) and ocean flow modeling (e.g., [2]).

Geophysical flow dynamics are typically modeled by spatio-temporal partial differential equations (PDEs). Physics-based flow models numerically solve these PDEs under known initial and boundary conditions. Existing physics-based regional flow models (e.g., [3], [4], [5], [6]) are typically formulated with the spatial resolution of a few kilometers and the temporal

resolution of a few hours, and because of the high dimension of state variables, the implementation of these models is computationally expensive. However, for increased navigation and sampling performance of AUVs, computationally efficient and higher resolution flow models are required. This paper presents a novel method for constructing generic environmental models (GEMs) that are data-driven computational ocean models combined with real-time data streams collected from mobile sensing platforms and provide high-resolution flow predictions in the vicinity of the mobile platforms.

To construct GEMs, an ocean model developed in [7] approximates ocean flow in a small area near AUVs using spatial and temporal basis functions. Once initialized using a historic data set, the model assimilates flow estimates that are computed along the trajectories of AUVs to update model parameters in real time. In addition to this parametric flow model, our previous studies [8], [9], [10] developed a novel method referred to as *motion tomography* (MT) that creates a spatial map of a flow field through multiple AUVs scanning a region of interest. MT models the influence of flow on the vehicle trajectory to formulate an “inverse problem” that has been the core problem underlying medical CT. By solving this inverse problem, MT constructs a high-resolution spatial map of flow in the region traversed by AUVs.

This paper extends our previous work by combining the MT method with a parametric flow model to resolve temporal variations for MT. To account for the coupling between temporal and spatial variations in flow modeling, we formulate an inverse problem associated with the influence of time-varying flow on the vehicle trajectory in a spatially and temporally discretized domain. By solving this inverse problem, we obtain an estimate of an underlying time-varying flow field. To deal with the high dimension of the solution space resulting from the both spatially and temporally discretized flow field, we incorporate a parametric flow model. The MT method is then used to estimate parameters of the flow model. To consider the case in which trajectory information of multiple vehicles cannot be obtained collectively, distributed implementation of the MT method, referred to as distributed MT, is developed. The paper validates the proposed approaches through simulations and the simulation results suggest successful mapping of a time-varying flow field.

In the rest of the paper, Section II introduces modeling a flow field and the influence of flow on vehicle trajectories. Section III presents the formulation of the MT method for time-varying flow fields. Then, distributed MT is presented in Section IV. Section V demonstrates the proposed methods through simulations and Section VI concludes the paper.

## II. BACKGROUND INFORMATION

### A. Parametric Flow Field Modeling

For fast computation and high resolution, we introduce a data-driven flow model in which ocean flow is approximated by using basis functions such as sinusoids in Fourier series, wavelets, piecewise polynomials, and splines [11]. Each of such basis functions is associated with a parameter (also known as a coefficient or weight). These parameters are estimated from data collected from various ocean observing technologies such as HF-radar systems or AUVs [7].

Let us denote the time by  $t \in \mathbb{R}$  and the position by  $\mathbf{r} \in \mathbb{R}^2$ . For position  $\mathbf{r}$ , we define a series of spatial basis functions indexed by  $i$  as  $\phi_i(\mathbf{r}) \in \mathbb{R}$  to approximate spatial variations of flow. For time  $t$ , a series of temporal basis functions indexed by  $j$  is given by  $\psi_j(t) \in \mathbb{R}$ , which is used to approximate temporal variations of flow. We denote constant parameters coupled with the spatial basis functions for the  $x$  and  $y$  components of flow by  $\eta_{x,i} \in \mathbb{R}$  and  $\eta_{y,i} \in \mathbb{R}$ , respectively. We also denote constant parameters coupled with the temporal basis functions for the  $x$  and  $y$  components of flow by  $\rho_{x,j} \in \mathbb{R}$  and  $\rho_{y,j} \in \mathbb{R}$ , respectively.

With  $M$  spatial basis functions and  $N$  temporal basis functions, the  $x$  and  $y$  components of flow velocity  $\mathbf{f}$  at position  $\mathbf{r}$  and time  $t$  is represented by

$$\begin{aligned} f_x(\mathbf{r}, t; \Theta_x) &= \left( \sum_{i=1}^M \eta_{x,i} \phi_i(\mathbf{r}) \right) \left( \sum_{j=1}^N \rho_{x,j} \psi_j(t) \right) \\ f_y(\mathbf{r}, t; \Theta_y) &= \left( \sum_{i=1}^M \eta_{y,i} \phi_i(\mathbf{r}) \right) \left( \sum_{j=1}^N \rho_{y,j} \psi_j(t) \right), \end{aligned} \quad (1)$$

where  $\Theta_x$  and  $\Theta_y$  denote all the parameters for the  $x$  and  $y$  components of flow, respectively (i.e.,  $\Theta_x = [\eta_x^T, \rho_x^T]^T$  and  $\Theta_y = [\eta_y^T, \rho_y^T]^T$  for  $\eta_x = [\dots, \eta_{x,i}, \dots]^T$ ,  $\eta_y = [\dots, \eta_{y,i}, \dots]^T$ ,  $\rho_x = [\dots, \rho_{x,i}, \dots]^T$ , and  $\rho_y = [\dots, \rho_{y,i}, \dots]^T$ ). Without loss of generality, we can express (1) as

$$\begin{aligned} f_x(\mathbf{r}, t) &= h(\mathbf{r}, t; \Theta_x) \\ f_y(\mathbf{r}, t) &= h(\mathbf{r}, t; \Theta_y), \end{aligned} \quad (2)$$

where  $h(\mathbf{r}, t; \cdot)$  is a nonlinear mapping from  $\Theta_x \in \mathbb{R}^{N+M}$  to  $f_x \in \mathbb{R}$  or from  $\Theta_y \in \mathbb{R}^{N+M}$  to  $f_y \in \mathbb{R}$ .

### B. Horizontal Motion of AUVs Under Flow

To model the influence of flow on the vehicle trajectory, we consider the horizontal motion of AUVs in the presence of flow. The horizontal vehicle motion is modeled using a first-order particle model with constant speed,  $s_h$ . Let us denote the time by  $t \in \mathbb{R}$  and the position by  $\mathbf{r} \in \mathbb{R}^2$ . Let  $\mathcal{T} = [t^0, t^f]$ , with observation time horizon  $T_{\text{obs}} = t^f - t^0 > 0$ , be

a bounded time interval called observation interval. We denote the vehicle heading by  $\theta$  and suppose a vehicle travels using a constant flow estimate,  $\tilde{\mathbf{f}}$ , during the observation interval.

The predicted vehicle position,  $\tilde{\mathbf{r}}(t)$ , during the observation interval can be obtained by integrating

$$\dot{\tilde{\mathbf{r}}}(t) = s_h \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} + \tilde{\mathbf{f}}. \quad (3)$$

The estimated flow may differ from the real flow experienced by a vehicle, and the actual vehicle position,  $\mathbf{r}(t)$ , can be computed by using the following equation:

$$\dot{\mathbf{r}}(t) = s_h \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} + \mathbf{f}(\mathbf{r}, t) = \dot{\tilde{\mathbf{r}}}(t) + \mathbf{f}(\mathbf{r}, t) - \tilde{\mathbf{f}}, \quad (4)$$

which is usually unknown because of the unknown flow velocity,  $\mathbf{f}(\mathbf{r}, t)$ . Since  $\tilde{\mathbf{f}}$  is known along the vehicle trajectory, we let  $\tilde{\mathbf{f}} = 0$  for simplicity throughout the paper.

Suppose during the observation interval, the actual position,  $\mathbf{r}$ , of an AUV is available only at  $t = t^0$  and  $t = t^f$ . Because of the unknown flow  $\mathbf{f}$ , the actual trajectory differs from the predicted trajectory. This difference is referred to as the *motion-integration error*. Let us define  $\gamma = \{\mathbf{r}(t)\}_{t \in \mathcal{T}} : \mathcal{T} \rightarrow \mathbb{R}^2$  as the trajectory of a vehicle. From (3) and (4), the motion-integration error over one observation interval is given by

$$\mathbf{d}(\gamma, \mathcal{T}) = \int_{t^0}^{t^f} (\dot{\mathbf{r}}(\tau) - \dot{\tilde{\mathbf{r}}}(\tau)) d\tau = \int_{t^0}^{t^f} \mathbf{f}(\mathbf{r}, \tau) d\tau. \quad (5)$$

## III. MOTION TOMOGRAPHY FOR TIME-VARYING FLOW

The MT method estimates a flow field based on the motion-integration error and trajectory information of AUVs. Suppose we deploy vehicle  $i$  with  $\mathbf{r}_i$  denoting its position in domain  $\mathcal{D}$  as illustrated in Fig. 1. After observation interval  $\mathcal{T}$ , we obtain the motion-integration error of the vehicle according to (5). Let us discretize  $\mathcal{T} = [t^0, t^f]$  into  $T$  sub-time intervals such that  $\mathcal{T}_1 = [t^0, t^1]$ ,  $\mathcal{T}_2 = [t^1, t^2]$ ,  $\dots$ ,  $\mathcal{T}_T = [t^{T-1}, t^f]$ , which satisfy  $\mathcal{T} = \cup_{m=1}^T \mathcal{T}_m$  and  $\cap_{m=1}^T \mathcal{T}_m = \emptyset$ . Let us define flow velocity during sub-time interval  $\mathcal{T}_m$  as  $\mathbf{f}_m(\mathbf{r}_i) = \mathbf{f}(\mathbf{r}_i, t)$ ,  $t \in \mathcal{T}_m$  and assume  $\mathbf{f}_m(\mathbf{r}_i)$  is constant (e.g.,  $\mathbf{f}(\mathbf{r}_i, t)$  is evaluated at  $t = t_m$  for  $t \in \mathcal{T}_m$ ) so  $\mathbf{f}_m(\mathbf{r}_i)$  is piecewise constant in time. Then, for the  $i$ th vehicle trajectory  $\gamma_i$ , (5) can be rewritten as

$$\mathbf{d}(\gamma_i, \mathcal{T}) = \sum_{m=1}^T \int_{\mathcal{T}_m} \mathbf{f}_m(\mathbf{r}_i) d\tau. \quad (6)$$

We use  $\mathbf{d}_i = \mathbf{d}(\gamma_i, \mathcal{T})$  hereafter for simplicity.

Let us discretize the domain  $\mathcal{D}$  into  $P = R \times S$  grid cells with  $D_{(r,s)}$  denoting the  $(r, s)$ th cell. We define index  $j = (r-1)S + s$  such that  $D_j \equiv D_{(r,s)}$ ,  $j = \{1, \dots, P\}$ . We denote the flow velocity in grid cell  $D_j$  for sub-time interval  $\mathcal{T}_m$  by  $\mathbf{f}_{(m,j)}$  and assume flow within a cell for one sub-time interval is constant. For each trajectory  $\gamma_i$  within grid cell  $D_j$  for sub-time interval  $\mathcal{T}_m$ , we introduce arc-length parameter  $l_i$ , given by

$$dl_i = s_{\text{tr}}^i(\mathbf{f}_{(m,j)}) dt, \quad (7)$$

in which  $s_{\text{tr}}^i$  is the speed of the  $i$ th vehicle along its actual trajectory. We assume that vehicle heading is constant within

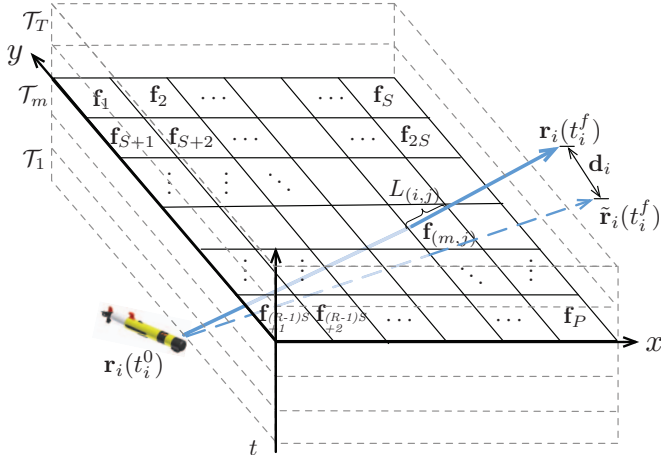


Fig. 1. Illustration of MT mapping formulation for a time-varying flow field. Actual (the blue solid line) and predicted (the blue dashed line) vehicle trajectories are displayed in a discretized domain.

a cell for one sub-time interval. Then, the speed of the  $i$ th vehicle along the trajectory is given by

$$s_{\text{tr}}^i(\mathbf{f}_{(m,j)}) = \left\| s_{\text{h}} \begin{bmatrix} \cos \theta_{(m,j)}^i \\ \sin \theta_{(m,j)}^i \end{bmatrix} + \mathbf{f}_{(m,j)} \right\|, \quad (8)$$

in which  $\theta_{(m,j)}^i$  is the  $i$ th vehicle heading in the  $j$ th grid cell for the  $m$ th sub-time interval. Equation (8) leads to the linear trajectory within a cell for one sub-time interval and the piecewise linear trajectory over the domain  $\mathcal{D}$ .

For the  $i$ th vehicle passing through the  $j$ th cell for the  $m$ th sub-time interval, the length of the vehicle trajectory,  $L_{(m,j)}^i$ , can be obtained by

$$L_{(m,j)}^i = \int_{\gamma_i[\mathcal{T}_m, D_j]} dl_i, \quad (9)$$

in which  $\gamma_i[\mathcal{T}_m, D_j]$  represents curve  $\gamma_i$  within spatial interval  $D_j$  and temporal interval  $\mathcal{T}_m$ . A vehicle trajectory within these spatial and temporal intervals depends on flow in the current cell during the current sub-time interval as well as flow in the previous cells that the trajectory passes through for the previous sub-time intervals. Therefore, for the  $i$ th vehicle trajectory  $\gamma_i$  from  $t = t^0$  to  $t = t_{(m,j)}^f$ , in which  $t_{(m,j)}^f$  is the time that the vehicle leaves the spatio-temporal interval  $[\mathcal{T}_m, D_j]$ , we define a set of indices for cells that the vehicle trajectory visits as  $\kappa_{(m,j)}^i = \{(\alpha - 1)P + \beta \mid [\mathcal{T}_\alpha, D_\beta] \cap \{\mathbf{r}_i(t)\}_{t \in [t^0, t_{(m,j)}^f]} \neq \emptyset, \alpha = \{1, \dots, T\}, \beta = \{1, \dots, P\}\}$ . Then, the length of a vehicle trajectory within a cell is defined as  $L_{(m,j)}^i = L_{(m,j)}^i(\mathbf{f}_{\kappa_{(m,j)}^i})$ .

By substituting (7) into (6) and following the discretization setting designed above, (6) can be discretized as

$$\mathbf{d}_i = \sum_{m=1}^T \sum_{j=1}^P \frac{L_{(m,j)}^i(\mathbf{f}_{\kappa_{(m,j)}^i})}{s_{\text{tr}}^i(\mathbf{f}_{(m,j)})} \mathbf{f}_{(m,j)}, \quad i = \{1, \dots, K\}. \quad (10)$$

Considering the flow velocity along the  $x$  and  $y$  directions separately, we have

$$d_{x,i} = \sum_{m=1}^T \sum_{j=1}^P \frac{L_{(m,j)}^i(\mathbf{f}_{\kappa_{(m,j)}^i})}{s_{\text{tr}}^i(\mathbf{f}_{(m,j)})} f_{x,(m,j)} \quad (11)$$

$$d_{y,i} = \sum_{m=1}^T \sum_{j=1}^P \frac{L_{(m,j)}^i(\mathbf{f}_{\kappa_{(m,j)}^i})}{s_{\text{tr}}^i(\mathbf{f}_{(m,j)})} f_{y,(m,j)}.$$

By constructing vectors  $\mathbf{d}_x = [d_{x,1}, d_{x,2}, \dots, d_{x,K}]^T$ ,  $\mathbf{d}_y = [d_{y,1}, d_{y,2}, \dots, d_{y,K}]^T$ ,  $\mathbf{f}_x = [\dots, f_{x,(m,1)}, f_{x,(m,2)}, \dots, f_{x,(m,P)}, \dots]^T$ , and  $\mathbf{f}_y = [\dots, f_{y,(m,1)}, f_{y,(m,2)}, \dots, f_{y,(m,P)}, \dots]^T$ , we can rewrite (11) as

$$\mathbf{d}_x = \mathbf{L}(\mathbf{f})\mathbf{f}_x \quad (12)$$

$$\mathbf{d}_y = \mathbf{L}(\mathbf{f})\mathbf{f}_y,$$

where  $\mathbf{f} = [\mathbf{f}_x, \mathbf{f}_y]$  and

$$\mathbf{L}(\mathbf{f}) = \begin{bmatrix} \dots & \frac{L_{(m,1)}^1(\mathbf{f}_{\kappa_{(m,1)}^1})}{s_{\text{tr}}^1(\mathbf{f}_{(m,1)})} & \dots & \frac{L_{(m,P)}^1(\mathbf{f}_{\kappa_{(m,P)}^1})}{s_{\text{tr}}^1(\mathbf{f}_{(m,P)})} & \dots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \dots & \frac{L_{(m,1)}^K(\mathbf{f}_{\kappa_{(m,1)}^K})}{s_{\text{tr}}^K(\mathbf{f}_{(m,1)})} & \dots & \frac{L_{(m,P)}^K(\mathbf{f}_{\kappa_{(m,P)}^K})}{s_{\text{tr}}^K(\mathbf{f}_{(m,P)})} & \dots \end{bmatrix}.$$

Equation (12) is nonlinear and underdetermined ( $K < T \times P$ ).

MT creates a map of an underlying time-varying flow field  $\mathbf{f}$  by solving (12) through an iterative process that alternates trajectory tracing and flow field estimation [8], [9], [10]. Let us use  $k$  to denote the index for iterations. Since vehicle trajectories are typically unknown, starting from an initial guess of estimates for a flow field, trajectory tracing computes vehicle trajectories using the flow estimates and updates matrix  $\mathbf{L}$ . Then, estimates for a flow field is updated via a row-action approach by iterating the following optimization process with linearized constraints for (12):

$$\mathbf{f}_x^{k+1} = \underset{\mathbf{f}_x}{\text{argmin}} \frac{1}{2} \|\mathbf{f}_x - \mathbf{f}_x^k\|^2 \quad (13)$$

$$\mathbf{f}_y^{k+1} = \underset{\mathbf{f}_y}{\text{argmin}} \frac{1}{2} \|\mathbf{f}_y - \mathbf{f}_y^k\|^2$$

subject to

$$d_{x,i} = \mathbf{L}_i(\mathbf{f}^k)\mathbf{f}_x \quad (14)$$

$$d_{y,i} = \mathbf{L}_i(\mathbf{f}^k)\mathbf{f}_y,$$

in which  $\mathbf{L}_i$  is the  $i$  row of matrix  $\mathbf{L}$  and  $i = \text{mod}(k, K) + 1$ .

*Remark 1.* Let us consider either the  $x$  or  $y$  component only. To solve an inverse problem for (13), we may consider an optimization problem

$$\text{minimize} \quad \frac{1}{2} \|\mathbf{f}\|^2 \quad (15)$$

$$\text{subject to} \quad \mathbf{d} - \mathbf{L}(\mathbf{f})\mathbf{f} = 0, \quad (16)$$

which can be viewed as a nonlinear least squares problem. Note that  $\mathbf{L}(\mathbf{f})$  is constructed by nonlinear vehicle trajectories. Because of the high dependency of vehicle trajectories on flow along the trajectories, the gradient of  $\mathbf{L}(\mathbf{f})$  is not easy to compute and the convexity of the constraint is not guaranteed.

In other words, (15) is a non-convex optimization problem. To deal with this problem, we employ an iterative row-action approach that does not require the computation of the gradient of  $\mathbf{L}(\mathbf{f})$  and linearize the constraint at every iteration, turning (15) into a convex optimization problem.

Since we obtain (12) by discretizing (5) in both space and time, the dimension of the solution space for  $\mathbf{f}_x$  and  $\mathbf{f}_y$  may be very high. In addition, we may not have enough trajectory information to estimate full solution space. To resolve this dimension issue, we incorporate a parametric flow model in (2). Following the discretization setting for MT, (2) can be discretized as

$$\begin{aligned} f_{x,(m,j)} &= h(\mathbf{r}_j, t_m; \Theta_x) = h_{(m,j)}(\Theta_x) \\ f_{y,(m,j)} &= h(\mathbf{r}_j, t_m; \Theta_y) = h_{(m,j)}(\Theta_y), \end{aligned} \quad (17)$$

in which  $m = \{1, \dots, T\}$  and  $j = \{1, \dots, P\}$  are indices for time and position, respectively,  $\mathbf{r}_j$  represents the position for grid cell  $D_j$ , and  $t_m$  represents the time for sub-time interval  $\mathcal{T}_m$ . By defining matrix  $\mathbf{H}(\cdot)$  such that the element of its  $(m, j)$ th row is  $[h_{(m,j)}(\cdot)]$ ,  $\mathbf{f}_x$  and  $\mathbf{f}_y$  in (12) can be expressed as

$$\begin{aligned} \mathbf{f}_x &= \mathbf{H}(\Theta_x) \\ \mathbf{f}_y &= \mathbf{H}(\Theta_y). \end{aligned} \quad (18)$$

By employing a parametric flow model, we have  $\mathbf{L}(\mathbf{f}) = \mathbf{L}(\Theta)$ , in which  $\Theta = [\Theta_x, \Theta_y]$ . By substituting (18) into (12), the motion-integration error equations become

$$\begin{aligned} \mathbf{d}_x &= \mathbf{L}(\Theta)\mathbf{H}(\Theta_x) \\ \mathbf{d}_y &= \mathbf{L}(\Theta)\mathbf{H}(\Theta_y). \end{aligned} \quad (19)$$

Now, to estimate a flow field, we can solve the equations in (19) for  $\Theta_x$  and  $\Theta_y$  instead of solving the equations in (12) for  $\mathbf{f}_x$  and  $\mathbf{f}_y$ , reducing the dimension of the solution space. In other words, instead of (13), we iteratively solve the following optimization problems:

$$\begin{aligned} \Theta_x^{k+1} &= \operatorname{argmin}_{\Theta_x} \frac{1}{2} \|\Theta_x - \Theta_x^k\|^2 \\ \Theta_y^{k+1} &= \operatorname{argmin}_{\Theta_y} \frac{1}{2} \|\Theta_y - \Theta_y^k\|^2 \end{aligned} \quad (20)$$

subject to

$$\begin{aligned} d_{x,i} &= \mathbf{L}_i(\Theta^k)\mathbf{H}(\Theta_x) \\ d_{y,i} &= \mathbf{L}_i(\Theta^k)\mathbf{H}(\Theta_y), \end{aligned} \quad (21)$$

where  $i = \operatorname{mod}(k, K) + 1$ .

Notice that because of the nonlinearity of  $\mathbf{H}(\cdot)$ , solving (20) for  $\Theta_x$  and  $\Theta_y$  is difficult. As a workaround, we fix spatial or temporal parameters one by one using the previous estimates and solve (20) for the other parameter. To express the flow model in (1) in a matrix notation, let us define  $\Phi(\mathbf{r}) = [\dots, \phi_i(\mathbf{r}), \dots]^T$ ,  $i = \{1, \dots, M\}$  and  $\Psi(t) = [\dots, \psi_j(t), \dots]^T$ ,  $j = \{1, \dots, N\}$ . Suppose at the  $k$ th iteration, we have estimates for parameters,  $\rho_{x|y}^k$  and  $\eta_{x|y}^k$ . By fixing temporal parameters  $\rho_{x|y}$  using  $\rho_{x|y}^k$ , (17) becomes

$$\begin{aligned} f_{x,(m,j)} &= (\rho_x^k)^T \Psi(t_m) (\Phi(\mathbf{r}_j))^T \eta_x \\ f_{y,(m,j)} &= (\rho_y^k)^T \Psi(t_m) (\Phi(\mathbf{r}_j))^T \eta_y. \end{aligned} \quad (22)$$

By constructing matrix  $\mathbf{H}^{\rho_{x|y}^k}$  such that the element of its  $(m, j)$ th row is  $[(\rho_{x|y}^k)^T \Psi(t_m) (\Phi(\mathbf{r}_j))^T]$ , (18) becomes

$$\begin{aligned} \mathbf{f}_x &= \mathbf{H}^{\rho_x^k} \eta_x \\ \mathbf{f}_y &= \mathbf{H}^{\rho_y^k} \eta_y. \end{aligned} \quad (23)$$

Let us define  $\mathbf{G}^{\rho_{x|y}^k}(\eta) = \mathbf{L}(\eta, \rho^k) \mathbf{H}^{\rho_{x|y}^k}$ . Then, we estimate spatial parameters  $\eta_{x|y}$  by the following optimization problem:

$$\begin{aligned} \eta_x^{k+1} &= \operatorname{argmin}_{\eta_x} \frac{1}{2} \|\eta_x - \eta_x^k\|^2 \\ \eta_y^{k+1} &= \operatorname{argmin}_{\eta_y} \frac{1}{2} \|\eta_y - \eta_y^k\|^2 \end{aligned} \quad (24)$$

subject to

$$\begin{aligned} d_{x,i} &= \mathbf{G}_i^{\rho_x^k}(\eta^k) \eta_x \\ d_{y,i} &= \mathbf{G}_i^{\rho_y^k}(\eta^k) \eta_y. \end{aligned} \quad (25)$$

Following a similar process, by fixing  $\eta_{x|y}$ , we estimate  $\rho_{x|y}$  by the following optimization problem:

$$\begin{aligned} \rho_x^{k+1} &= \operatorname{argmin}_{\rho_x} \frac{1}{2} \|\rho_x - \rho_x^k\|^2 \\ \rho_y^{k+1} &= \operatorname{argmin}_{\rho_y} \frac{1}{2} \|\rho_y - \rho_y^k\|^2 \end{aligned} \quad (26)$$

subject to

$$\begin{aligned} d_{x,i} &= \mathbf{G}_i^{\eta_x^k}(\rho^k) \rho_x \\ d_{y,i} &= \mathbf{G}_i^{\eta_y^k}(\rho^k) \rho_y, \end{aligned} \quad (27)$$

where we define  $\mathbf{G}^{\eta_{x|y}^k}(\rho) = \mathbf{L}(\eta^k, \rho) \mathbf{H}^{\eta_{x|y}^k}$  and  $\mathbf{H}^{\eta_{x|y}^k}$  such that  $[(\eta_{x|y}^k)^T \Phi(\mathbf{r}_j) (\Psi(t_m))^T]$  is the element of its  $(m, j)$ th row.

Based on the above two sub-optimization problems, we obtain a flow model parameter estimation algorithm (Algorithm 1). In the algorithm, the updating equations include relaxation parameter  $\lambda$  which affects the convergence rate. Please note that once we update estimates for either one of the spatial and temporal parameters, we use updated estimates to update estimates for the other parameters. To check the convergence, for  $k$  that satisfies  $\operatorname{mod}(k, K) + 1 = K$ , we define residuals  $\mathbf{r}_{\mathbf{f},x}^k = \mathbf{L}(\mathbf{f}^k) \mathbf{f}_x^k - \mathbf{d}_x$  and  $\mathbf{r}_{\mathbf{f},y}^k = \mathbf{L}(\mathbf{f}^k) \mathbf{f}_y^k - \mathbf{d}_y$ . We continue updating the solutions until the Euclidean norms of residuals,  $\|\mathbf{r}_{\mathbf{f},x}^k\|$  and  $\|\mathbf{r}_{\mathbf{f},y}^k\|$ , are sufficiently small (i.e., below a threshold  $\epsilon_f$ ). The MT method then create a map of a flow field by iterating a process that alternates flow model parameter estimation and trajectory tracing. (For more details, please refer to [10].)

#### IV. DISTRIBUTED MOTION TOMOGRAPHY

Suppose we navigate  $K$  AUVs and create a map of flow fields from data collected from the vehicles through MT. To obtain the best possible results, MT should use trajectory information available from all the vehicles. However, because of the limited communication capabilities of the vehicles or the large scale of applications, data may be collected or shared among vehicles in a distributed manner. In this case, only

---

**Algorithm 1: MT flow model parameter estimation**

---

**Data:** Motion-integration errors  $\mathbf{d}_i, i = \{1, \dots, K\}$

```
1 Set  $k = 0$ .
2 Make an initial guess of the solutions,  $\eta_x^0, \eta_y^0, \rho_x^0$ , and  $\rho_y^0$ .
3 repeat
4   for  $i = 1$  to  $K$  do
5     Update the solutions by
6
7      $\eta_x^{k+1} = \eta_x^k + \lambda^k \frac{d_{x,i} - \mathbf{G}^{\rho_x^k}(\eta^k)\eta_x^k}{\|\mathbf{G}^{\rho_x^k}(\eta^k)\|^2} \left(\mathbf{G}^{\rho_x^k}(\eta^k)\right)^T$ 
8      $\eta_y^{k+1} = \eta_y^k + \lambda^k \frac{d_{y,i} - \mathbf{G}^{\rho_y^k}(\eta^k)\eta_y^k}{\|\mathbf{G}^{\rho_y^k}(\eta^k)\|^2} \left(\mathbf{G}^{\rho_y^k}(\eta^k)\right)^T$ 
9      $\rho_x^{k+1} = \rho_x^k + \lambda^k \frac{d_{x,i} - \mathbf{G}^{\eta_x^{k+1}}(\rho^k)\rho_x^k}{\|\mathbf{G}^{\eta_x^{k+1}}(\rho^k)\|^2} \left(\mathbf{G}^{\eta_x^{k+1}}(\rho^k)\right)^T$ 
10     $\rho_y^{k+1} = \rho_y^k + \lambda^k \frac{d_{y,i} - \mathbf{G}^{\eta_y^{k+1}}(\rho^k)\rho_y^k}{\|\mathbf{G}^{\eta_y^{k+1}}(\rho^k)\|^2} \left(\mathbf{G}^{\eta_y^{k+1}}(\rho^k)\right)^T$ .
11    Let  $k = k + 1$ .
12  end
13  Construct  $\mathbf{f}_x$  and  $\mathbf{f}_y$  using updated estimates for the
14  parameters.
15   $\mathbf{r}_{\mathbf{f},x}^k = \mathbf{L}(\mathbf{f}^k)\mathbf{f}_x^k - \mathbf{d}_x$ .
16   $\mathbf{r}_{\mathbf{f},y}^k = \mathbf{L}(\mathbf{f}^k)\mathbf{f}_y^k - \mathbf{d}_y$ .
17 until a stopping condition is met (e.g.,
18  $\|\mathbf{r}_{\mathbf{f},x}^k\|, \|\mathbf{r}_{\mathbf{f},y}^k\| \leq \epsilon_f$ )
```

---

partial information may be available for the MT mapping. In this section, we present distributed MT.

Suppose vehicle  $i$  has the knowledge of its own  $\mathbf{d}_i$  and  $\mathbf{L}_i$  only. We can view the problem in (20) as a joint optimization problem in which vehicle  $i$  solves the  $i$ th optimization problem and computes a common estimate of flow model parameters  $\Theta$  that is shared among all the vehicles. Suppose the solution is not shared and vehicle  $i$  computes its own estimate of model parameters denoted by  $\Theta_i$  based on only its own knowledge of  $\mathbf{d}_i$  and  $\mathbf{L}_i$  by solving the  $i$ th optimization problem separately. Then, the solution is clearly sub-optimal and inevitably biased.

To overcome this issue, we consider that vehicle  $i$  shares its estimate of flow model parameters locally (i.e., with a subset of the remaining vehicles). Let us use  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  to denote an undirected graph comprising a set of (vehicle) nodes  $\mathcal{V}$  with a set of edges  $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ . We use  $e_{(i,j)} = e_{(j,i)} \in \mathcal{E}$  to denote the edge between an unordered pair of vehicles  $v_i$  and  $v_j$ . We assume a connected graph and define a group of vehicles who can communicate with vehicle  $i$ , including itself, as its neighbor  $\mathcal{N}_i = \{i\} \cup \{j | e_{(i,j)} \in \mathcal{E}\}$ . We consider a sparse network  $\mathcal{G}$  of AUVs, each sharing data with its neighbors only.

Suppose the  $i$ th vehicle iteratively updates its own estimate of flow model parameters along with parameters shared by its neighbors. As a preliminary step, we assume that communication among neighbors is always available. In other words, at each iteration  $k$ , vehicle  $i$  has  $\Theta_i^k$  and  $\Theta_j^k, j \in \mathcal{N}_i$ . For

distributed MT to be implemented on vehicle  $i$ , we modify the sub-optimization problem in (24) and (26) and formulate the following optimization problems:

$$\eta_{x,i}^{k+1} = \underset{\eta_{x,i}}{\operatorname{argmin}} \frac{1}{2} \left( \sum_{j \in \mathcal{N}_i} \|\eta_{x,i} - \eta_{x,j}^k\|^2 \right) \quad (28)$$

$$\eta_{y,i}^{k+1} = \underset{\eta_{y,i}}{\operatorname{argmin}} \frac{1}{2} \left( \sum_{j \in \mathcal{N}_i} \|\eta_{y,i} - \eta_{y,j}^k\|^2 \right)$$

subject to

$$\begin{aligned} d_{x,i} &= \mathbf{G}_i^{\rho_{x,i}^k}(\eta_i^k)\eta_{x,i} \\ d_{y,i} &= \mathbf{G}_i^{\rho_{y,i}^k}(\eta_i^k)\eta_{y,i}, \end{aligned} \quad (29)$$

and

$$\rho_{x,i}^{k+1} = \underset{\rho_{x,i}}{\operatorname{argmin}} \frac{1}{2} \left( \sum_{j \in \mathcal{N}_i} \|\rho_{x,i} - \rho_{x,j}^k\|^2 \right) \quad (30)$$

$$\rho_{y,i}^{k+1} = \underset{\rho_{y,i}}{\operatorname{argmin}} \frac{1}{2} \left( \sum_{j \in \mathcal{N}_i} \|\rho_{y,i} - \rho_{y,j}^k\|^2 \right)$$

subject to

$$\begin{aligned} d_{x,i} &= \mathbf{G}_i^{\eta_{x,i}^k}(\rho_i^k)\rho_{x,i} \\ d_{y,i} &= \mathbf{G}_i^{\eta_{y,i}^k}(\rho_i^k)\rho_{y,i}, \end{aligned} \quad (31)$$

respectively.

By solving the optimization problems in (28) and (30), at each iteration we update the solutions by the following equations:

$$\begin{aligned} \eta_{x,i}^{k+1} &= \frac{\sum_{j \in \mathcal{N}_i} \eta_{x,j}^k}{|\mathcal{N}_i|} \\ &+ \frac{|\mathcal{N}_i|d_{x,i} - \sum_{j \in \mathcal{N}_i} \mathbf{G}_i^{\rho_{x,i}^k}(\eta_i^k)\eta_{x,j}^k}{|\mathcal{N}_i| \|\mathbf{G}_i^{\rho_{x,i}^k}(\eta_i^k)\|^2} \left( \mathbf{G}_i^{\rho_{x,i}^k}(\eta_i^k) \right)^T \end{aligned} \quad (32)$$

$$\begin{aligned} \eta_{y,i}^{k+1} &= \frac{\sum_{j \in \mathcal{N}_i} \eta_{y,j}^k}{|\mathcal{N}_i|} \\ &+ \frac{|\mathcal{N}_i|d_{y,i} - \sum_{j \in \mathcal{N}_i} \mathbf{G}_i^{\rho_{y,i}^k}(\eta_i^k)\eta_{y,j}^k}{|\mathcal{N}_i| \|\mathbf{G}_i^{\rho_{y,i}^k}(\eta_i^k)\|^2} \left( \mathbf{G}_i^{\rho_{y,i}^k}(\eta_i^k) \right)^T, \end{aligned} \quad (33)$$

and

$$\begin{aligned} \rho_{x,i}^{k+1} &= \frac{\sum_{j \in \mathcal{N}_i} \rho_{x,j}^k}{|\mathcal{N}_i|} \\ &+ \frac{|\mathcal{N}_i|d_{x,i} - \sum_{j \in \mathcal{N}_i} \mathbf{G}_i^{\eta_{x,i}^k}(\rho_i^k)\rho_{x,j}^k}{|\mathcal{N}_i| \|\mathbf{G}_i^{\eta_{x,i}^k}(\rho_i^k)\|^2} \left( \mathbf{G}_i^{\eta_{x,i}^k}(\rho_i^k) \right)^T \end{aligned} \quad (34)$$

$$\begin{aligned} \rho_{y,i}^{k+1} &= \frac{\sum_{j \in \mathcal{N}_i} \rho_{y,j}^k}{|\mathcal{N}_i|} \\ &+ \frac{|\mathcal{N}_i|d_{y,i} - \sum_{j \in \mathcal{N}_i} \mathbf{G}_i^{\eta_{y,i}^k}(\rho_i^k)\rho_{y,j}^k}{|\mathcal{N}_i| \|\mathbf{G}_i^{\eta_{y,i}^k}(\rho_i^k)\|^2} \left( \mathbf{G}_i^{\eta_{y,i}^k}(\rho_i^k) \right)^T, \end{aligned} \quad (35)$$

where the first term on the right side of each equation is the average of the estimates of the vehicles in the neighborhood set and the second term is the adjustment of the estimate with respect to all the available estimates.

Based on the distributed optimization problems designed above, we obtain a distributed parameter estimation algorithm (Algorithm 2). At the beginning, each vehicle computes initial guesses of estimates for flow model parameters independently. Then, at each iteration, vehicle  $i$  shares its estimate  $\Theta_i^k$  with its neighbors  $j \in \mathcal{N}_i$ . Then, each vehicle updates their estimates using (32)–(35). The algorithm repeats this process until a stopping condition is met.

---

**Algorithm 2:** Distributed MT flow model parameter estimation

---

**Data:** Motion-integration errors  $\mathbf{d}_i, i = \{1, \dots, K\}$

- 1 Set  $k = 0$ .
  - 2 Make an initial guess of the solutions,  $\eta_{x,i}^0, \eta_{y,i}^0, \rho_{x,i}^0,$  and  $\rho_{y,i}^0$ .
  - 3 **repeat**
  - 4     Vehicle  $i$  shares its estimates  $\eta_{x,i}^k, \eta_{y,i}^k, \rho_{x,i}^k,$  and  $\rho_{y,i}^k$  with its neighbors  $j \in \mathcal{N}_i$ .
  - 5     Each vehicle updates  $\eta_{x|y}$  and  $\rho_{x|y}$  by (32)–(35).
  - 6     Let  $k = k + 1$ .
  - 7 **until** a stopping condition is met.
- 

## V. SIMULATION RESULTS

This section validates our proposed methods through simulations in which we navigate multiple vehicles under flow in a domain of interest. To simulate “true” flow, we use post-processed historic HF-radar data which provide approximately  $3 \times 3 \text{ km}^2$  spatial resolution and 30-minute temporal resolution. During vehicle navigation, the data are spatially and temporally interpolated along vehicle trajectories. After vehicles travel under this simulated flow, the trajectory information and motion-integration errors of vehicles are used to estimate flow model parameters through our proposed methods.

*Remark 2.* Typical HF-radar data are hourly data with  $6 \times 6 \text{ km}^2$  spatial resolution (available at <http://hfradar.ndbc.noaa.gov>) with a three hour processing delay, which may be considered low spatial and temporal resolutions for guiding mobile sensing platforms.

For a GEM that parameterizes the motion-integration errors in (12), we use a data-driven computational flow model in (2). To initialize basis functions for (2), we select 30-day historical HF-radar data from the time when vehicles start traveling. By decomposing the tidal component of flow from the data using the T\_Tide MATLAB<sup>®</sup> toolbox [12], we construct temporal basis functions for the tidal component of flow. For the data of the selected period and region, T\_Tide extracts total 29 tidal constituents and 3 major constituents ( $M2, N2,$  and  $S2$ ) are used. Considering the tidal residual and pairs of sine and cosine functions,  $N = 3 \times 2 + 1$ . For the non-tidal component, we use 0th to 2nd order Laguerre polynomials as temporal

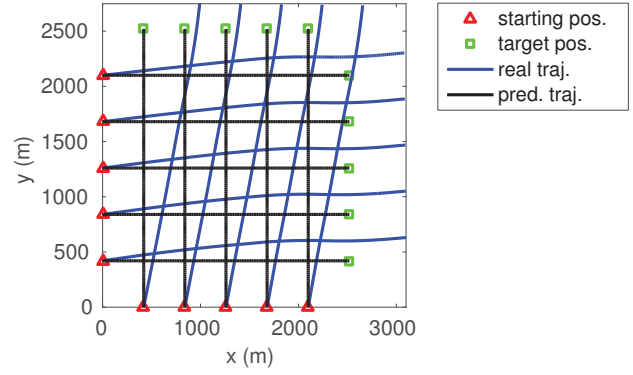


Fig. 2. Vehicle trajectories in a domain of interest. The predicted trajectory of a vehicle, connecting its starting position (a red triangle) and target position (a cyan rectangle), is displayed in a black line along with its real trajectory in a blue line.

basis functions (i.e.,  $L = 3$ ). To account for spatial variations of flow,  $M = 5$  radial basis functions are used.

*Remark 3.* Although we use the historic HF-radar data from the time period starting from the beginning time of vehicle navigation, please note that these data are used only for the initialization of basis functions. For parameter estimation, we use the proposed methods.

After the basis functions for (2) is initialized, we navigate multiple vehicles in a domain. In our simulations, total  $K = 10$  vehicles navigate through the domain for 2 hours: 5 from left to right and the other 5 from bottom to top. Figure 2 shows simulated trajectories of the vehicles. Each vehicle travels at  $0.35 \text{ m/s}$  towards its target position and we assume that the positions of the vehicle are known only at the starting and ending times of its navigation. After vehicles complete their navigation, the trajectory information and motion-integration errors are collected from the vehicles to construct a map of an underlying flow field through MT.

We first demonstrate the effectiveness of resolving temporal variations for MT. For 10 vehicles traveling for 2 hours, the domain is discretized into  $P = 5 \times 5$  grid cells and  $T = 5$  sub-time intervals. Figure 3 compares the  $x$  and  $y$  components of real flow and those of estimated flow constructed using parameters that are computed through MT. The figure illustrates that the estimated flow closely follows the trend of the real flow. The root-mean-square (rms) error for the  $x$  component of the estimated flow is  $0.0072 \text{ m/s}$  and that for the  $y$  component of the estimated flow is  $0.0166 \text{ m/s}$ . To reduce the error, we can increase the number of spatial and temporal basis functions, but we used a minimal number of basis functions ( $N = 7, L = 3,$  and  $M = 5$ ) to balance the dimension of the parameters and the flow mapping accuracy.

To implement distributed MT, vehicles exchange their estimated parameters for the flow model in (2) with their

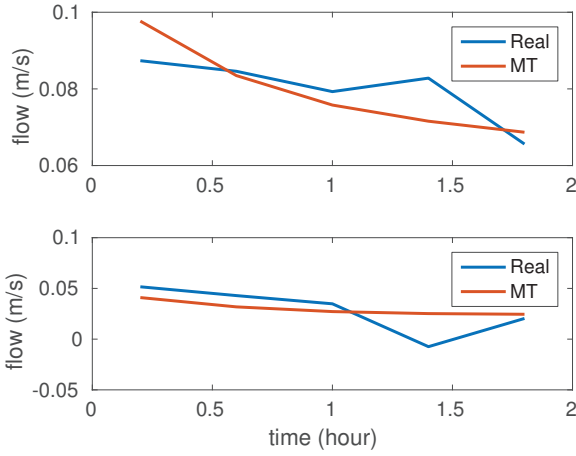


Fig. 3. Comparison between real flow and estimated flow constructed using parameters computed through non-distributed MT at the center of the domain where vehicles navigate.

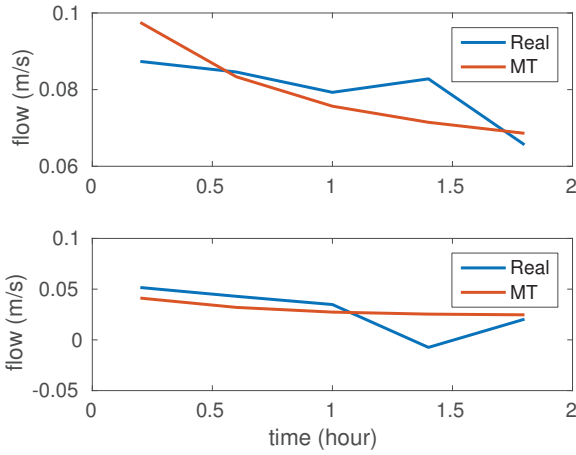


Fig. 4. Comparison between real flow and estimated flow constructed using parameters computed through distributed MT at the center of the domain where vehicles navigate.

neighbors within 1 kilometer. Then, each vehicle updates its own parameter estimates through the distributed MT method. Figure 4 compares the  $x$  and  $y$  components of real flow and those of estimated flow constructed using parameters that are computed through distributed MT. It is straightforward to see that estimates of each vehicle updated using Eqs. (32)–(35) will reach the average of all the estimates and this average will converge to estimates computed through non-distributed MT. Therefore, the rms errors for the  $x$  and  $y$  components of the estimated flow obtained through distributed MT are very close to those for the  $x$  and  $y$  components of the estimated flow obtained through non-distributed MT.

## VI. CONCLUSION

This paper presents the motion tomography (MT) method for a time-varying flow field. To incorporate temporal vari-

ability of flow into MT, temporal domain of a flow field is discretized along with discretized spatial domain. To reduce the dimension of variables to be estimated through MT, a parametric flow model is employed. Then, MT constructs a time-varying flow field by estimating parameters for the model. To account for the case in which vehicle trajectory information cannot be obtained collectively, distributed MT is developed so that individual vehicles can estimate flow model parameters with knowledge of parameter estimates shared from their neighbors only. Using parameters estimated through both non-distributed and distributed MT, a time-varying flow field is successfully constructed.

## ACKNOWLEDGMENT

The research work is supported by ONR grants N00014-10-10712 (YIP) and N00014-14-1-0635; and NSF grants OCE-1032285, IIS-1319874, and CMMI-1436284.

## REFERENCES

- [1] K. Szwaykowska and F. Zhang, “Trend and bounds for error growth in controlled Lagrangian particle tracking,” *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 10–25, 2014.
- [2] L. Kuznetsov, K. Ide, and C. K. R. T. Jones, “A method for assimilation of Lagrangian data,” *Monthly Weather Review*, vol. 131, no. 10, pp. 2247–2260, 2003.
- [3] R. A. Luettich, J. J. Westerink, and N. W. Scheffner, “ADCIRC: An advanced three-dimensional circulation model for shelves, coasts, and estuaries. Report 1. Theory and methodology of ADCIRC-2DDI and ADCIRC-3DL,” Coastal Engineering Research Center, Vicksburg, Mississippi (U.S.), Tech. Rep., 1992.
- [4] R. Bleck, “An oceanic general circulation model framed in hybrid isopycnic-Cartesian coordinates,” *Ocean Modelling*, vol. 37, pp. 55–88, 2002.
- [5] A. F. Shchepetkin and J. C. McWilliams, “The regional oceanic modeling system (ROMS): A split-explicit, free-surface, topography-following-coordinate oceanic model,” *Ocean Modelling*, vol. 9, no. 4, pp. 347–404, 2005.
- [6] D. B. Haidvogel, H. Arango, W. P. Budgell, B. D. Cornuelle, E. Curchitser, E. Di Lorenzo, K. Fennel, W. R. Geyer, A. J. Hermann, L. Lanerolle, J. Levin, J. C. McWilliams, A. J. Miller, A. M. Moore, T. M. Powell, A. F. Shchepetkin, C. R. Sherwood, R. P. Signell, J. C. Warner, and J. Wilkin, “Ocean forecasting in terrain-following coordinates: Formulation and skill assessment of the Regional Ocean Modeling System,” *Journal of Computational Physics*, vol. 227, no. 7, pp. 3595–3624, 2008.
- [7] D. Chang, X. Liang, W. Wu, C. R. Edwards, and F. Zhang, “Real-time modeling of ocean currents for navigating underwater glider sensing networks,” in *Cooperative Robots and Sensor Networks*, ser. Studies in Computational Intelligence, A. Koubãa and A. Khelil, Eds. Springer Berlin Heidelberg, 2014, vol. 507, pp. 61–75.
- [8] W. Wu, D. Chang, and F. Zhang, “Glider CT: Reconstructing flow fields from predicted motion of underwater gliders,” in *Proceedings of the Eighth ACM International Conference on Underwater Networks and Systems*, 2013, p. 47.
- [9] D. Chang, W. Wu, and F. Zhang, “Glider CT: Analysis and experimental validation,” in *Distributed Autonomous Robotic Systems*, ser. Springer Tracts in Advanced Robotics, N.-Y. Chong and Y.-J. Cho, Eds. Springer Japan, 2016, vol. 112, pp. 285–298.
- [10] D. Chang, W. Wu, C. R. Edwards, and F. Zhang, “Motion tomography: Mapping flow fields using autonomous underwater vehicles,” *International Journal of Robotics Research (under review)*, 2015.
- [11] K. M. Lynch, I. B. Schwartz, P. Yang, and R. A. Freeman, “Decentralized environmental modeling by mobile sensor networks,” *IEEE Transactions on Robotics*, vol. 24, no. 3, pp. 710–724, 2008.
- [12] R. Pawlowicz, B. Beardsley, and S. Lentz, “Classical tidal harmonic analysis including error estimates in MATLAB using T\_TIDE,” *Computers and Geosciences*, vol. 28, pp. 929–937, 2002.