2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Tracking Ocean Fronts with Multiple Vehicles and Mixed Communication Losses

Brooks L. Reed and Franz S. Hover

***Abstract*— We make two contributions toward integrated monitoring over large spatial scales, with multiple collaborating vehicles. Our focus is dynamic ocean features such as fronts and plumes. To support strong networked-control designs, we first develop a clean linear time-invariant framework for tracking features, that directly couples the global structure of the process to vehicle positioning. To address the packet loss inherent in underwater acoustic communications, we then extend the synthesis technique of Imer et al. [1] to the case where measurements and control commands suffer loss with differing statistics among the multiple channels. Simulations show that the integrated feedback system achieves good performance in front tracking.**

I. INTRODUCTION

Networks of distributed mobile agents are an attractive means for tracking and pursuit of dynamic features over large spatial scales, although wireless communication brings challenges for control [1]. Dynamic missions of interest in the ocean include monitoring and following a quickly evolving plume or other process [2]–[4], but underwater, wireless communication over distances beyond about one hundred meters is almost exclusively accomplished via acoustics. Acoustic communication suffers from frequent packet loss caused by ambient noise, multipath, and changing environmental conditions, and from long delays and low data rates [5]. In this paper, we make two connected contributions towards joint estimation and pursuit of dynamic ocean features: *a linear formulation of the integrated observation problem, and a control design technique that rigorously handles multiple channels with mixed packet loss probabilities*.

Prior work on observation of ocean features tends to emphasize single vehicles, or slower timescales when multiple vehicles are in use. Path-planning with single vehicles is studied in [6], and experimental approaches specific to front detection are demonstrated in [7]. Multi-vehicle adaptive sampling is studied in [8], [9], while coordinated sampling of ocean features using drifters and vehicles is studied via an experiment in [10]. A distributed approach for plume and thermocline tracking is presented in [11]. In the above works, collaboration between vehicles is limited and communications occur at low frequencies. In contrast, dynamic feedback control with acoustic communications has been studied with formation control problems, *e.g.*, [12].

For tracking truly dynamic features with multiple vehicles, we propose an integrated closed-loop control method that provides a decomposition of spatial and temporal variations.

B. Reed is in the MIT/WHOI Joint Program in Oceanographic Engineering. F. Hover is with MIT Department of Mechanical Engineering. {brooksr8,hover} at mit.edu

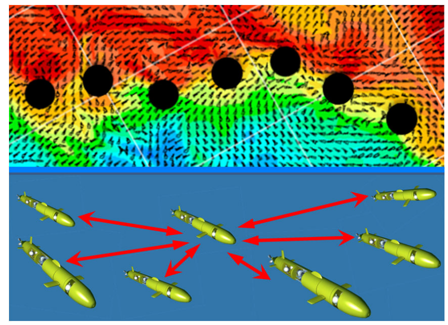

Fig. 1. Illustration of vehicles positioned to track an ocean front (above), collaborating via acoustic communications (below).

A front is modeled as a coupled LTI system representing the short-term evolution of a discrete set of points, each to be tracked by a vehicle. We use ocean model forecasts and a system identification procedure to describe the locally linear behavior of the front. The key idea is that the physics behind ocean processes introduces global spatial and dynamic structure to the system, which can be exploited by a centralized controller—as opposed to vehicles that make decisions based only on local information. Our LTI formulation is fundamentally different than most nonlinear or mode-based models for ocean processes, which have obvious advantages in descriptive capability but are not suitable for direct use with dynamic control design.

To handle the stochastic packet loss inherent in acoustic communications with multiple vehicles, we extend the optimal control technique of Imer, Yüksel & Başar [13] to a multi-channel case. The original paper considers an all-or-none lossy channel on the controller and sensor sides of a control loop, which (by the authors' admission) is a special case. The approach is inherently constructive, however, providing a complete dynamic programming (DP) recursion for control actions based on an estimated state. Schenato *et al.* [14] considered a similar problem and showed that the optimal control action is linear in the estimate and that the separation principle holds only when acknowledgments of control packets are available or there is no sensor noise and the observation matrix is invertible. Other related works include Garone *et al.* [15] and Gupta, Hassibi & Murray [16] – all of whom allow acknowledgments in their treatment of mixed-loss channels. We view the case of no acknowledgments to be valuable in underwater acoustic networks, where propagation delay and interference considerations might make them very costly in time. An additional advantage of the explicit DP computation is that it accommodates time-varying systems, relevant to operations with vehicles moving through space.

978-1-4673-6357-0/13/$31.00 ©2013 IEEE

We develop the linearized ocean front model in Section II. A system identification process is used to determine accurate low-order linear models for the dynamics of the nominal front. Initial work with a double gyre model and subspace system identification is promising; we give a brief description in Section II-C.1. The mixed-loss control design derivation is given in Section III. We present simulation results with a linear demonstration system in Section IV and conclude the paper in Section V. This work is aimed towards future experiments with our fielded system [17].

II. LINEARIZED OCEAN FRONT MODEL

A. Overview and variable definitions

The problem we consider involves a group of mobile agents with scalar measurements following a dynamic field. The scalar field of interest is denoted by $\phi(\boldsymbol{r},t)$, where $\boldsymbol{r} \in \mathbb{R}^2$ or $\mathbb{R}^3$. Predictions of the field can be obtained using numerical ocean models; however, these estimates are subject to uncertainty, which is decreased when measurements of the true field are available, as in adaptive sampling. It is desired to track the position of a small set of discrete points along an important set of features, *e.g.*, defining an ocean front, using one vehicle assigned to each point. We focus on dynamic perturbations from a nominal frontal contour, defined as the expected location of the front as predicted by the model.

The key assumptions of our front model are as follows:

- A major computational capability exists at the network center that can generate the expected evolution of the front position and gradient, as well as the associated uncertainty [18]. This expected evolution defines the "nominal" field $\bar{\phi}(\boldsymbol{r},t)$.
- Points along a nominal frontal contour are picked *a priori*. The gradient of the field at the front relates the measurement of the field of interest to the true position of the front. A good strategy for high SNR is to choose points (and place the vehicles) at locations with large magnitude gradients; little information about the field's evolution is obtained by measuring flat areas.
- The position of a given point on the front can be predicted a short time into the future, and the main deviations from this trajectory are primarily along the gradient direction. The gradient is assumed to be slowly varying; as with all other adaptive observation techniques, frequent corrections to the estimate will help.
- Vehicles drive along a line in the direction of the nominal gradient at their specified frontal point. The design variable is a scalar velocity command.

The nominal front, a contour of the nominal scalar field, is denoted as $\Theta_{\phi_0}(t) = \{\boldsymbol{r}|\bar{\phi}(\boldsymbol{r},t) = \phi_0(t)\}$.[1] We discretize the front into n frontal points, picked by the user in order to make useful measurements while satisfying vehicle dynamic constraints. In general, points are spaced along the front. Thus we have for the nominal trajectory of each point:

$$\bar{\boldsymbol{p}}^i(t) = \{\boldsymbol{r}|\bar{\phi}(\boldsymbol{r},t) = \phi_0(t), \text{ and } \mathbb{C}\}, \quad i = 1,\ldots,n, \quad (1)$$

[1]For simplicity, the frontal contour is defined here as a specific level set, however, other criteria could be used to represent the nominal front.

where $\mathbb{C}$ represents some rule set shared by all points, to be clarified later. The nominal gradient at each frontal point is $\boldsymbol{g}_0^i(t) = \nabla\bar{\phi}(\boldsymbol{p}^i(t),t)$. The gradients are obtained from the ocean model estimate and represent a local linearization of the scalar field in the neighborhood of the front.

From here on, with a slight abuse of notation, we define all variables as scalar perturbations from the nominal front in the direction of the nominal gradient $\boldsymbol{g}_0^i(t)$, and drop the dependence on t for clarity. The position of vehicle i relative to the nominal front is q^i, the position of the true perturbed frontal point is p^i, and the nominal gradient magnitude is g^i. The true measurement is of the scalar field at the vehicle location. However, a first-order Taylor expansion gives a linearized measurement equation based on the gradient:

$$z^i_{field} = \phi(q^i) \approx (q^i - p^i)g^i, \quad (2)$$

where we have set the nominal frontal contour value $\phi_0(t)$ to zero without loss of generality. Positioning vehicles such that they remain close to the front improves estimation as the linearization accuracy is good, with relative importance proportional to the gradient. Thus, the control objective is to minimize z^i_{field} over the decision horizon.

B. General LTI system formulation

The aggregate state variables for the frontal point perturbations are $\boldsymbol{x}_p(k)$, and the state variables for vehicle perturbations are $\boldsymbol{x}_q(k)$. We define n decoupled systems for vehicle dynamics, modeling low-level closed-loop control. The dynamics matrix A_p is block diagonal, the control input matrix is B_q, and the output matrix is C_q.

Models for ocean processes can be obtained via system identification applied to ensembles of ocean model simulations, as we describe in Section II-C.1, or alternatively through linearization of a PDE [19]. The positions of the n frontal points are described by a coupled system with dynamics A_q and output matrix C_q. A key aspect of this system is that the spatial coupling is represented implicitly in the structure of the A_p; the eigenvectors will not be sparse.

We use a discrete-time description so that the system is suitable for packet loss robust control design. The combined state space system in general form is:

$$\begin{aligned}\begin{Bmatrix}\boldsymbol{x}_p(k+1)\\ \boldsymbol{x}_q(k+1)\end{Bmatrix} &= \left[\begin{array}{c|c} A_p(k) & 0 \\ \hline 0 & A_q(k)\end{array}\right]\begin{Bmatrix}\boldsymbol{x}_p(k)\\ \boldsymbol{x}_q(k)\end{Bmatrix} \\ &\quad + \left[\begin{array}{c} 0 \\ \hline B_q(k)\end{array}\right]\boldsymbol{u}(k) + \begin{Bmatrix}\boldsymbol{w}_p(k)\\ \boldsymbol{w}_q(k)\end{Bmatrix}, \end{aligned} \quad (3)$$

where the vehicle process noise $\boldsymbol{w}_q$ has covariance Q_q, and the ocean model process noise $\boldsymbol{w}_p$ has covariance Q_p.

The dynamics of the frontal perturbations and vehicles are decoupled, giving the block-diagonal structure in the A matrix. However, since the frontal positions p^i cannot be observed directly, the output equation consists of the scalar field measurements as well as vehicle positions: $\boldsymbol{z}(k) = [\boldsymbol{z}_{field}(k)^T,\ \boldsymbol{z}_q(k)^T]^T$. This couples vehicle and process dynamics. Various choices exist for navigation measurements; we choose vehicle position as a common example. Thus, expressed in terms of state variables and with noise defined below, the output equation in vector form is:

$$\boldsymbol{z}(k) = \left[\begin{array}{c|c} -GC_p & GC_q \\ \hline 0 & C_q \end{array}\right] \begin{Bmatrix} \boldsymbol{x}_p(k) \\ \boldsymbol{x}_q(k) \end{Bmatrix} + \left[\begin{array}{c|c} I & G \\ \hline 0 & I \end{array}\right] \begin{Bmatrix} \boldsymbol{\nu}_\phi(k) \\ \boldsymbol{\nu}_q(k) \end{Bmatrix}$$

where $G(k) = \mathrm{diag}(\bar{g}^1(k), \ldots, \bar{g}^n(k))$ and I represents a suitably sized identity matrix. The scalar field measurement noise is $\boldsymbol{\nu}_\phi(k)$, with covariance Re_ϕ. Representing navigational uncertainty, vehicle measurement noise is ν_q, with covariance Re_q.[2]

C. Specific ocean process and vehicle models

First, we briefly present a case that directly involves stochastic fluid mechanics, and then we show a more instructive example of a coupled mass-spring system.

1) Double Gyre Ocean Model: We consider a stochastic double gyre model, simulated using a finite-volume Navier-Stokes solver [21]. This canonical fluid mechanics problem is highly nonlinear and can go unstable. That makes it a challenging test case for our methodology. On the other hand, it is a generic scenario with few physical parameters that may be useful for benchmarking.

The nominal front is taken as a section of a given vorticity contour of the mean field of 25 stochastic ensemble members. Seven frontal points are initially picked with equal spacing along this contour, and we track the evolution of these points by finding the intersection of the nominal gradient at a point with the nominal contour at the next time step. Perturbations for each realization in the ensemble are determined by finding the intersection of the nominal gradient at each point with the contour in the specific realization at a given time step. We then pass the perturbations into a MIMO subspace system identification procedure, `N4SID` [22], which outputs the frontal system matrices A_p and C_p, along with the identified noise model.

Comparative results with independent data and simulated packet losses are given in Figure 2, along with a brief description of the controllers (discussed in more detail in Section IV). The "Loners" method, that uses local information at each vehicle, gives the best vehicle positioning (upper plot). However, by not communicating, the global estimate of the front suffers. In the lower plot, we see that the "Mixed loss" and "All or none" methods reduce the estimation error relative to the lower bound by about 75%. Better positioning results in more accurate estimation for the methods that communicate, as they use the same coupled model. These initial results with a physics-based model are encouraging for our approach.

2) "Chained-mass" Linear Demonstration System: In this example, the perturbation of the front is approximated as a chained mass-spring-damper construction with n masses. The inertia of point i is m_i, $k_{i,i}$ and $b_{i,i}$ are the stiffness and damping respectively between realized frontal point i and the nominal front, and $k_{i,j}$ and $b_{i,j}$ are the stiffness and damping

[2]We assume relatively accurate navigation, see [20] for a recent discussion of underwater vehicle navigation.

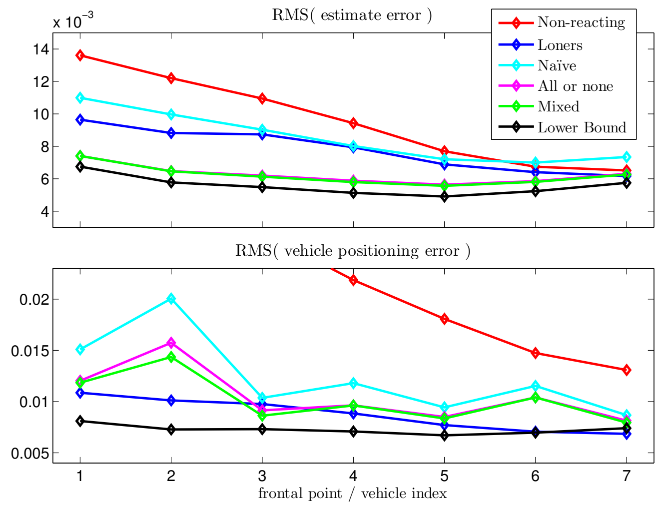

Fig. 2. Control results for the double gyre example. The "Non-reacting" controller places vehicles at the nominal front, while the "Loners" use local models to act on local information only; vehicles do not communicate. The "Lower Bound" uses the coupled model with perfect communication. The rest of the methods use the coupled model with lossy communication between vehicles. "Naïve" applies standard LQR gains to the case of lossy communications, the "All or none" method applies the algorithm of Imer *et al.* [13], and "Mixed" uses our mixed-loss algorithm developed in Section III. The latter four methods are discussed in more detail in Section IV.

respectively between points i and j. Thus, the equation of motion for frontal point i is given as:

$$\begin{aligned} \ddot{p}^{\,i} = w_p^i + \frac{1}{m_i}\Big(& -k_{i,i}p^i - b_{i,i}\dot{p}^i \\ & + k_{i,i-1}(p^{i-1} - p^i) + b_{i,i-1}(\dot{p}^{i-1} - \dot{p}^i) \\ & + k_{i,i+1}(p^{i+1} - p^i) + b_{i,i+1}(\dot{p}^{i+1} - \dot{p}^i)\Big), \end{aligned} \quad (4)$$

where p^i is the deflection of point i from the straight-chain configuration at the origin of the coordinate system.

For $i = 1$ and $i = n$, the springs and dampers that do not connect to a neighbor are set to zero (other boundary conditions could be modeled as well). In our application $k_{i,i}$ equals zero, since the position of the true front is not coupled to the nominal (expected) front.

3) Vehicle model: Conventionally, vehicles are operated with waypoint control using low-level onboard autonomy. We use velocity as the control input here, however, because the control design assumes zero control when a command packet is lost (zero-output decoder on the control channel). If position control were used, the vehicles would attempt to drive back to the nominal front whenever a command packet is lost; intuitively, since the front has a random walk mode, it makes more sense for the vehicles to stay where they are if the command is not received.

For the purposes of the chained-mass example in this paper, each vehicle is modeled with a first order system relating the velocity command to vehicle velocity, plus an integrator to give vehicle position: $x_v/u = 1/s(\tau s + 1)$. Heterogeneous vehicles could be easily modeled, although within a linear framework speed saturation and waypoint control cannot.

III. Mixed loss control design

A. Problem Statement

Building on the approach of Imer *et al.* [13], we consider LQR-type optimal control in both the TCP-like (acknowledgments) and UDP-like (no acknowledgments) cases. We extend the previous work to the case with multiple channels on each side, each with unique statistics. We first list the major assumptions:

- There is a centralized architecture for estimation and control, as the coupled models described in Section II-C leverage the global structure of ocean processes. Additionally, data assimilation often makes use of outside information such as remote sensing and involves considerable computation [23].
- The discrete-time system operates at a single rate. This scenario is justified by multiple-access methods such as frequency- or code-division multiplexing that are common in wireless RF and possible for low numbers of vehicles with acoustics [24]. There are limits, however, and we recognize that multirate control schemes [25] are also needed.
- Packet loss statistics are stationary and uncorrelated with state. In principle it would be possible to allow them to vary with the time step.

The linear time-invariant, stochastic system that we will use in our derivations is $x_{k+1} = Ax_k + B\alpha_k u_k + w_k$, where x_k is the state at time step k, u_k is the control action, and w_k is a stochastic noise source. The size n diagonal matrix α_k describes control packet success and is taken as the realization at step k of either a Bernoulli random scalar or a diagonal matrix with independent Bernoulli entries taking values zero or one. Scalar α_k is one with probability $\bar{\alpha}$, so that $E\{\alpha_k\} = \bar{\alpha}$. The matrix generalization is direct, where we consider all α_k and $\bar{\alpha}$ diagonal matrices.

The quadratic cost function is

$$E\left\{x_N^T Q_N x_N + \sum_{k=0}^{N-1}\left[x_k^T Q x_k + u_k^T \alpha_k R \alpha_k u_k\right]\right\}, \quad (5)$$

where N is the finite-time decision horizon.

B. First Backward Step

With a slight abuse of notation, we assume u is the optimal control based on I_{N-1}, the information available to the controller at step $N-1$. Using the standard DP equation [26], the cost-to-go is

$$\begin{aligned} J_{N-1}(I_{N-1}) = \min_{u_{N-1}} & E\big\{x_N^T Q_N x_N + \\ & x_{N-1}^T Q x_{N-1} + u_{N-1}^T \alpha_{N-1} R \alpha_{N-1} u_{N-1}\big\} \\ = \min_{u_{N-1}} & E\big\{x'_{N-1}(A' Q_N A + Q)x_{N-1} + \\ & u_{N-1}^T \alpha_{N-1}(R + B^T Q_N B)\alpha_{N-1} u_{N-1} + \\ & 2u_{N-1}^T \alpha_{N-1} B^T Q_N A x_{N-1} + \\ & w_{N-1}^T Q_N w_{N-1} \mid I_{N-1}\big\}. \end{aligned} \quad (6)$$

We set the derivative with respect to u_{N-1} to zero, with expected values taken for appropriate terms. Letting $\hat{x}_k = E\{x_k \mid I_k\}$ we have

$$\begin{aligned} u_{N-1}^* = & -\left[E\{\alpha_{N-1}(R + B^T Q_N B)\alpha_{N-1} \mid I_{N-1}\}\right]^{-1} \\ & \times \bar{\alpha} B^T Q_N A \hat{x}_{N-1} \\ \doteq & - M_{N-1}^{-1} \times \bar{\alpha} B^T Q_N A \hat{x}_{N-1} \end{aligned} \quad (7)$$

Inserting this control back into J_{N-1} gives

$$\begin{aligned} J_{N-1}(I_{N-1}) = & E\big\{x_{N-1}^T(A^T Q_N A + Q)x_{N-1} + \\ & w_{N-1}^T Q_N w_{N-1} \mid I_{N-1}\big\} - \\ & \hat{x}_{N-1}^T P_{N-1} \hat{x}_{N-1} \end{aligned} \quad (8)$$

with $P_{N-1} \doteq A^T Q_N B \bar{\alpha} M_{N-1}^{-1} \bar{\alpha} B^T Q_N A$.

Define $e_k = x_k - \hat{x}_k$. Since

$$\begin{aligned} & -\hat{x}_{N-1}^T(\cdot)\hat{x}_{N-1} = \\ & \qquad E\{e_{N-1}^T(\cdot)e_{N-1} - x_{N-1}^T(\cdot)x_{N-1} \mid I_{N-1}\}, \end{aligned}$$

we have

$$\begin{aligned} J_{N-1}(I_{N-1}) = & E\big\{x_{N-1}^T K_{N-1} x_{N-1} + e_{N-1}^T P_{N-1} e_{N-1} + \\ & w_{N-1}^T Q_N w_{N-1} \mid I_{N-1}\big\} \end{aligned} \quad (9)$$

with $K_{N-1} \doteq A^T Q_N A + Q - P_{N-1}$.

We go to the next backward step in the DP, again assuming the information up to step k is available to design u_k:

$$\begin{aligned} J_{N-2}(I_{N-2}) = \min_{u_{N-2}} & E\big\{x_{N-2}^T(Q + A^T K_{N-1} A)x_{N-2} + \\ & u_{N-2}^T \alpha_{N-2}(R + B^T K_{N-1} B)\alpha_{N-2} u_{N-2} + \\ & 2u_{N-2}^T \alpha_{N-2} B^T K_{N-1} A x_{N-2} + \\ & e_{N-1}^T P_{N-1} e_{N-1} + \\ & w_{N-2}^T K_{N-1} w_{N-2} + \\ & w_{N-1}^T Q_N w_{N-1} \mid I_{N-2}\big\} \end{aligned} \quad (10)$$

C. Second Backward Step for TCP

For TCP, e_{N-1} is not dependent on u_{N-2}. Straightforward manipulations give

$$u_{N-2}^* = -M_{N-2}^{-1} \bar{\alpha} B^T K_{N-1} A \hat{x}_{N-2}$$

with $M_{N-2} = E\{\alpha_{N-2}(R + B^T K_{N-1} B)\alpha_{N-2} \mid I_{N-2}\}$. The resulting cost-to-go is

$$\begin{aligned} J_{N-2}(I_{N-2}) = E\big\{ & x_{N-2}^T K_{N-2} x_{N-2} + \\ & e_{N-2}^T P_{N-2} e_{N-2} + \\ & e_{N-1}^T P_{N-1} e_{N-1} + \\ & w_{N-2}^T K_{N-1} w_{N-2} + \\ & w_{N-1}^T Q_N w_{N-1} \mid I_{N-2}\big\} \end{aligned} \quad (11)$$

with $K_{N-2} = Q + A^T K_{N-1} A - P_{N-2}$ and $P_{N-2} = A^T K_{N-1} B \bar{\alpha} M_{N-2}^{-1} \bar{\alpha} B^T K_{N-1} A$.

The only difference between this recursion and the scalar form is that the control and P_{N-2} use $\bar{\alpha} M_{N-2}^{-1} \bar{\alpha}$ instead of $\bar{\alpha}(R + B^T K_{N-1} B)^{-1}$. As noted in the Introduction, our focus is on the UDP-like case, so we will not pursue the TCP-like case further.

D. Second Backward Step for UDP

e_{N-1} above is going to be a function of u_{N-2} if there are no control acknowledgments, because when the sensor packet comes through, the estimator can only use a $\bar{\alpha}$-scaled version of the command, i.e., the expected value of α_{N-2}. This is the so-called dual effect, and e_{N-1} has to be expanded out. The estimator model from Imer *et al.* is:

$$\hat{x}_k = \beta_k x_k + (1-\beta_k)(A\hat{x}_{k-1} + B\bar{\alpha}u_{k-1}) \tag{12}$$

where β_k is a Bernoulli random variable with $P(\beta_k = 0) = \beta$ and $P(\beta_k = 1) = \bar{\beta}$. As with α, we take on a diagonal matrix β_k to allow for losses in different sensor channels.

Expanding the term from J_{N-2} above involving $e_{N-1}^T P_{N-1} e_{N-1}$ and taking expectations implicitly across the u terms, we have

$$\begin{aligned}
&\hat{x}_{N-1}^T P_{N-1}\hat{x}_{N-1} = \\
&\quad E\{x_{N-2}^T A^T Y_{N-1}^a A x_{N-2} + \hat{x}_{N-2}^T A^T Y_{N-1}^b A\hat{x}_{N-2} + \\
&\quad 2x_{N-2}^T A^T Y_{N-1}^c A\hat{x}_{N-2} \mid I_{N-2}\} + \\
&\quad u_{N-2}^T\big[\, E\{\alpha_{N-2}B^T Y_{N-1}^a B\alpha_{N-2} \mid I_{N-2}\} + \\
&\quad \bar{\alpha}B^T(\hat{Y}_{N-1}^b + \hat{Y}_{N-1}^c + \hat{Y}_{N-1}^d)B\bar{\alpha}\,\big]u_{N-2} + \\
&\quad 2u_{N-2}\bar{\alpha}B^T(\hat{Y}_{N-1}^a + \hat{Y}_{N-1}^b + \hat{Y}_{N-1}^c + \hat{Y}_{N-1}^d)A\hat{x}_{N-2}
\end{aligned} \tag{13}$$

$$\text{where } Y_k^a = \beta_k P_k \beta_k, \quad Y_k^b = (I-\beta_k)P_k(I-\beta_k)$$
$$Y_k^c = \beta_k P_k (I-\beta_k), \quad Y_k^d = (Y_k^c)^{\,T}$$

and $\hat{Y}_k^{(\cdot)}$ indicates the expected value across β_k. Note that $P_k = Y_k^a + Y_k^b + Y_k^c + Y_k^d$. The summed first three terms above factor into

$$\begin{aligned}
E\{&e_{N-2}^T A^T (Y_{N-1}^a - P_{N-1})Ae_{N-2} + \\
&x_{N-2}^T A^T P_{N-1} A x_{N-2} \mid I_{N-2}\},
\end{aligned}$$

and we can now assemble the recursion. Define

$$\begin{aligned}
T_{N-2} &= E\{\alpha_{N-2}(R + B^T[K_{N-1} + P_{N-1} - Y_{N-1}^a] \\
&\quad B)\alpha_{N-2} \quad \mid I_{N-2}\} \ - \bar{\alpha}B^T(P_{N-1} - \hat{Y}_{N-1}^a)B\bar{\alpha} \\
S_{N-2} &= \bar{\alpha}B^T K_{N-1} A.
\end{aligned}$$

It follows that

$$u_{N-2}^* = -T_{N-2}^{-1} S_{N-2}\hat{x}_{N-2}, \tag{14}$$

$$\begin{aligned}
\text{and} \quad J_{N-2} = E\{&x_{N-2}^T K_{N-2} x_{N-2} + \\
&e_{N-2}^T P_{N-2} e_{N-2} + \\
&w_{N-2}^T K_{N-1} w_{N-2} + \\
&w_{N-1}^T Q_N w_{N-1} \mid I_{N-2}\}.
\end{aligned} \tag{15}$$

$$\begin{aligned}
\text{with} \quad P_{N-2} &= S_{N-2}^T T_{N-2}^{-1} S_{N-2} + \\
&\quad A^T(P_{N-1} - \hat{Y}_{N-1}^a)A \\
K_{N-2} &= Q + A^T(K_{N-1} + P_{N-1} \\
&\quad - \hat{Y}_{N-1}^a)A - P_{N-2}.
\end{aligned}$$

From this point on, one uses the usual DP recursion [26].

E. Correlations

Correlations between channels within α and β, and indeed between α and β, are critical for wireless applications since both the control and the sensor messages use the same medium and often the same hardware. There are three distinct cases for the expectations we have employed.

1. All channels independent. For the TCP-like model, the major new requirement is calculating M_k; it will be different at each time step (since it has K_{k+1} inside), but is easy:

$$E\{\alpha Z\alpha\} = \begin{cases} \bar{\alpha}Z\bar{\alpha} & \text{for the off-diagonal elements} \\ \bar{\alpha}Z & \text{for the diagonal elements,} \end{cases} \tag{16}$$

where Z represents a symmetric matrix.

For the UDP-like model, the $\hat{Y}^a$ terms present a similar computation since we have Bernoulli β. T_{N-2} involves an expectation over both $\alpha_{N-2}^T(\cdot)\alpha_{N-2}$ and Y_{N-1}^a, but if α and β are independent this can be computed in sequence, i.e., the inner part and then the outer part.

2. α's correlated with each other and β's correlated with each other. Let $\Sigma_\beta = E\{(\vec{\beta} - \bar{\vec{\beta}})(\vec{\beta} - \bar{\vec{\beta}})\}^T$ be the covariance matrix of the vector $\vec{\beta}$. It can be easily shown that $E\{\beta Z\beta\} = Z * \Sigma_\beta$, where $*$ indicates pointwise matrix multiplication. Again an inner and an outer part can be computed separately for T_{N-2}.

3. α's and β's correlated (general case). The first term in T_{N-2} now has terms quartic in $[\alpha, \alpha, \beta, \beta]$. For this, define the four-dimensional array $L_{ijkl} = E\{\alpha_i\alpha_j\beta_k\beta_l\}$. Working through the algebra we obtain the $m \times m$ symmetric matrix

$$E\{\alpha B^T\beta P\beta B\alpha\}(i,j) = \sum_p B_{pi}\sum_q B_{qi}P_{qp}L_{ijqp} \tag{17}$$

F. Kalman Filter

Although a standard Kalman filter (KF) – which would accommodate sensor noise and incomplete measurement – has no guarantees for the lossy channel problem, several simple modifications pointed out in prior papers, *e.g.* Garone *et al*. [15], make the KF very reasonable. First, the state estimate prior reflects the fact that the control is uncertain:

$$\hat{x}_{k+1} = A\hat{x}_k + B\bar{\alpha}_k u_k.$$

The covariance prior has an added component for the same reason: with $q = u_k u_k^T$, we have

$$\begin{aligned}
P_{k+1}^{\,e} = &AP_k^{\,e}A^T + Q^{\,e} + \\
&B(E\{\alpha_k q\alpha_k\} + q - \bar{\alpha}_k q - q\bar{\alpha}_k)B^T,
\end{aligned} \tag{18}$$

where $Q^{\,e}$ is the process noise covariance. The Kalman Gain depends on β_k (known at the estimator) and is updated using the methods in [27].

IV. Simulation Results

We show simulation results for the chained-mass example developed in Section II-C.2. While many parameters affect performance, we analyze one representative test case in detail for a clean comparison of methods. We choose $n = 15$ frontal points equally spaced along the nominal front; snapshots of the front during a brief segment of one trial are shown in Figure 3. The parameters describing the ocean process are: $dt = 1$, $m_i = 1$, $k_{i,i+1} = k_{i,i-1} = 0.3$, $b_{i,i+1} = b_{i,i-1} = 0.2$, and $b_{i,i} = 0.05$. All gradients g_i are set to one. The initial condition for p is one period of a sine wave of amplitude 10, and all vehicles (and state estimates) start at zero.[3] The simulation length is 300 steps, and vehicles have a time constant τ equal to $1/10$ of the time step.

We implement a KF in place of the simple noiseless estimator, as described above. Each vehicle measures the true scalar field at its location, $\phi(q,t)$, plus noise. Vehicle position measurements come from onboard navigation, also subject to noise. Correlations may exist between all combinations of measurement and process noises. For example, process noise for an ocean model describing diffusion (*e.g.* chemical, biological) may only be loosely correlated with currents (vehicle process noise), while advecting processes (salinity, temperature) are obviously strongly correlated with currents. The shared physical domain between vehicles can introduce correlations as well. All of these models, however, can be handled within the KF framework. In this example we consider all noise to be independent, with $Q_p = 0.25I$, $Q_q = 0.01I$, $Re_q = 0.01I$ and $Re_\phi = 0.25I$.

We show comparisons of four different scenarios; these methods are the latter four described in Figure 2. For each, the true frontal evolution is the same, the same process and measurement noise realizations are used, and the same packet loss realizations are used (for Cases 1-3).

1) "Lower Bound": Communication is assumed ideal (no packet losses) for both controller design as well as simulation. The standard LQG solution is applied, with LQR controller gains and a conventional KF for estimation. Noisy measurements of the scalar field and vehicle position are available; p and $\dot{p}$ are still not directly observed.
2) "Naïve": The standard LQR controller gains (same as above) are used. True (mixed-loss) realizations for control and sensor packet losses are used in simulation. The KF is the standard missed-measurement form (with no adjustment to priors), using β_k.
3) "All or none": The control design procedure of Imer *et al.* is used, where all control packets have the same success probability, and all sensor packets have the same success probability. The scalar values of $\bar{\alpha}$ and $\bar{\beta}$ are taken as the *mean values across all channels*. In simulation, realizations of the packet loss process use the true (mixed) probabilities. The KF detailed above is used for estimation, with the modification that the mean value of $\bar{\alpha}$ is used for the adjustment to the priors. The KF uses β_k, the true (mixed-loss) sensor packet successes.
4) "Mixed loss": The mixed-loss control design detailed above is used to generate feedback gains. The KF detailed above is used for estimation (the full vector $\bar{\alpha}$ is used to adjust priors).

[3] The estimator converges quickly at startup with this scenario; practically, transients from initial conditions are an important robustness aspect to consider.

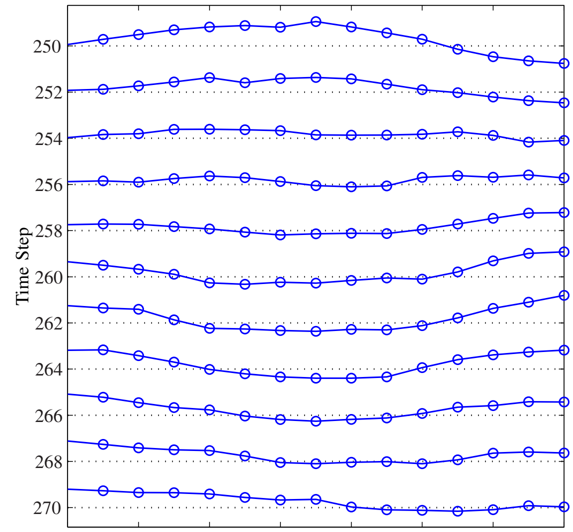

Fig. 3. Snapshots of front evolution (all sites) every two time steps near the end of the trial shown in Figure 4. The nominal front at each step is the thin horizontal dotted black line and the true front is the blue line. Frontal points p_i are open blue circles.

LQR with output weighting is used for controller design. The LQR weighting parameters for states, $\bar{Q}_{lqr}$, $\bar{R}_{lqr}$ and $\bar{N}_{lqr}$, are set following:

$$\begin{bmatrix} \bar{Q}_{lqr} & \bar{N}_{lqr} \\ \bar{N}_{lqr}^T & \bar{R}_{lqr} \end{bmatrix} = \begin{bmatrix} C^T & 0 \\ D^T & I \end{bmatrix} \begin{bmatrix} Q_{lqr} & N_{lqr} \\ N_{lqr}^T & R_{lqr} \end{bmatrix} \begin{bmatrix} C & D \\ 0 & I \end{bmatrix},$$

where Q_{lqr}, N_{lqr} and R_{lqr} are the weighting matrices for the output. In our case, $C = [-GC_p,\ GC_q]$, and $D = 0$ (N_{lqr} is infrequently used). The parameters used for controller design are: $Q_{lqr}^{z_{field}} = 100I$, $Q_{lqr}^{q} = 0.01I$, $R_{lqr} = I$, and a horizon of sixty steps. For the mixed-loss and all-or-none methods, we implement static gains taken as the first step of the finite-horizon gains (the final state penalty matrix is set as $Q_f = 1000Q$ in order to ensure convergence of the backwards gain recursion). The controller gains at each time step are a $n \times 4n$ matrix. With $n = 15$, the mixed-loss recursion for the sixty-step horizon takes roughly 0.05 seconds in Matlab, making this design method suitable for real-time implementation as time-varying parameters change.

Packet losses are simulated using independent success probabilities $\bar{\beta}$ and $\bar{\alpha}$. The packet success probability vectors $\bar{\alpha}$ and $\bar{\beta}$ are randomly generated for each trial based on a uniform distribution between 0.25 and one. In many operational scenarios, all measurements from a single vehicle are packaged into a single packet, that is successful with probability $\bar{\beta}_v^i$. Separate navigation and scalar field measurement packets could alternatively be modeled. For purposes of the "Mixed loss" control algorithm (formulated based on full state observations), we consider the former case and construct the full matrix of measurement success probabilities $\bar{\beta}$ with $\bar{\beta}_v^i$ repeated on the diagonal in appropriate order (measurements from vehicle i matched to the corresponding vehicle and process states at point i).

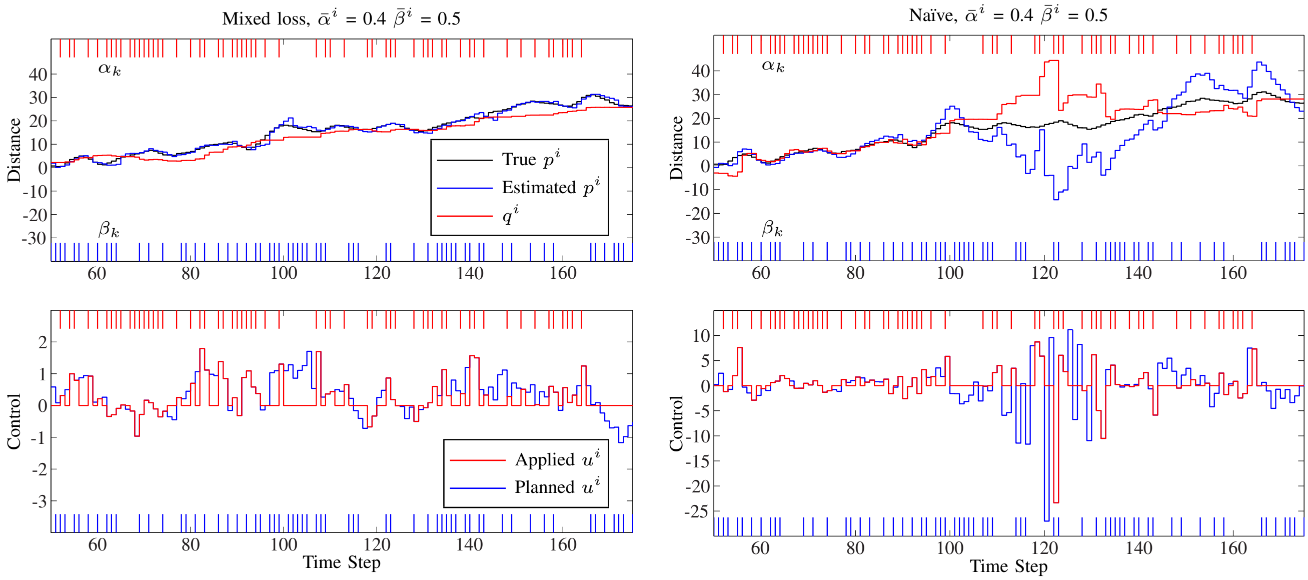

Fig. 4. A segment of simulation results for a single site/vehicle in one realization with different control methods, $\bar{\alpha}^i = 0.4$ and $\bar{\beta}^i = 0.5$ in this case. The left plots show the "Mixed loss" control design (Case 4). The RMS of z^i_{field} is 2.5 and the RMS error of $(\hat{p} - p)$ is 1.3. The right plots show the "Naïve" method with standard LQR control design under simulated packet loss (Case 2). The RMS of z^i_{field} is 5.0 and the RMS error of $(\hat{p} - p)$ is 7.3. The lower plots show the control (commanded vehicle velocity); when $\alpha_k = 0$ (lost control packet), the control applied on the vehicle (red trace) is zero. Packet loss and noise sequences are the same for both simulations; the red and blue vertical bars at the top and bottom of the plots show successful control and measurement packets, respectively.

Results from computational experiments are given in Table I. We run ensembles of 100 trials and present the minimum, mean and maximum values of $\overline{RMS(z_{field})}$ for each method. For each method, the RMS of z^i_{field} for each vehicle is computed over time, and then the average is taken across vehicles to give $\overline{RMS(z_{field})}$. We note that since the packet success probabilities are randomly generated for each trial, certain combinations of $\bar{\alpha}$ and $\bar{\beta}$ result in a system that cannot be stabilized; we have observed trials where Cases 2-4 are all unable to stabilize the system.

TABLE I

SIMULATION RESULTS (ONLY STABLE TRIALS ARE REPORTED).

$\overline{RMS(z_{field})}$	MIN	MEAN	MAX
1) Lower bound	0.74	0.76	0.78
2) Naïve	2.4	18	270
3) All or none	6.3	48	450
4) Mixed loss	2.1	3.6	11

The results from the chained-mass example highlight the value of the mixed-loss design, which consistently gave a lower $\overline{RMS(z_{field})}$, and went unstable less frequently than the naïve and all-or-none methods. In particular, we have noticed that when there is a large spread in the packet loss probabilities of the different communication channels, the all-or-none approach performs badly as some gains are not matched appropriately to the losses experienced on those channels. Additionally, the control effort required with the mixed-loss gains is significantly lower than that required by the naïve or all-or-none gains, even though the Q and R penalties are the same. It is interesting that the naïve method often outperforms the all-or-none design, which is in contrast to the preliminary double gyre results shown in Figure 2.

Figure 4 shows an example of the evolution of one site in one realization using the mixed-loss design and the same site in the same realization using the naïve LQR design. The naïve controller performs comparably to or slightly better than the mixed-loss design at times (for example, near step 75), although as shown near steps 100-150, when there are big bursts of α and β losses close in time, the naïve method suffers dramatically. This behavior has also been observed in trials with the all-or-none method. While this plot only shows performance of a single site, the coupling in the model means that the estimation difficulties of this vehicle hurt the estimation of the other points as well (and vice-versa). As the mixed-loss design accounts for the probability of packet success in each individual channel, it does not suffer from periods of extremely poor performance.

Finally, we note that in these simulations, the true simulated system is perfectly linear, and the gradients are perfectly known. This is in contrast to the stochastic, nonlinear fluid dynamics in the double gyre example described in Section II-C.1. We have run preliminary tests examining the robustness to imperfectly known gradients in the chained-mass example. When actual gradients are smaller than expected, the performance of all methods becomes sluggish due to gain reduction. However, when actual gradients are larger, the mixed-loss control scheme is able to maintain an upwards gain margin of rougly two, while the all-or-none and naïve methods can be very sensitive and often go unstable. The upwards gain margin of two is consistent with LQR/KF controllers, for which the mixed-loss method is making the best approximation under packet loss conditions.

V. CONCLUSION

Our new concept is that locally linear behavior of an ocean process admits powerful network-based control techniques on short time scales. Using a coupled model of the process, multiple cooperating vehicles can decompose spatial and temporal variations to track a dynamic feature of interest. The controllers can incorporate in a rigorous way the limitations of acoustic communications. This integrated approach to dynamic ocean monitoring appears to be successful in two preliminary examples.

Practical implementation of these techniques will require careful attention in picking the locations of each nominal p^i, in both two and three dimensions; one can think of various optimization approaches for adaptively designing trajectories as the front evolves. A related goal will be to minimize the field estimation error directly, instead of the proxy z_{field}. More detailed performance analysis can investigate factors such as the dynamics of the specific ocean process studied, packet loss distributions, and noise compared to gradient magnitudes. As our preliminary tests show, robustness to unavoidable errors in gradient prediction is important and can be analyzed further using techniques such as gain margins. Finally, our approach relies on a centralized controller that leverages the global structure of the ocean process. This limits development towards decentralized schemes, making efficient network scheduling and multirate control design especially important.

ACKNOWLEDGMENTS

Work is supported by the Office of Naval Research, Grant N00014-09-1-0700, the National Science Foundation, Contract CNS-1212597, and the Finmeccanica Career Development Professorship. We thank Prof. P. Lermusiaux for ocean model insight and the model data used in this paper.

REFERENCES

[1] J. Baillieul and P. Antsaklis, "Control and communication challenges in networked real-time systems," *Proc. of the IEEE*, vol. 95, no. 1, pp. 9–28, 2007.

[2] R. Ferrari, "A frontal challenge for climate models," *Science*, vol. 332, no. 6027, pp. 316–317, 2011.

[3] R. Camilli, C. Reddy, D. Yoerger, B. Van Mooy, M. Jakuba, J. Kinsey, C. McIntyre, S. Sylva, and J. Maloney, "Tracking hydrocarbon plume transport and biodegradation at Deepwater Horizon," *Science*, vol. 330, no. 6001, p. 201, 2010.

[4] J. Farrell, S. Pang, W. Li, and R. Arrieta, "Chemical plume tracing experimental results with a REMUS AUV," in *Proc. MTS/IEEE OCEANS*, vol. 2. IEEE, 2003, pp. 962–968.

[5] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: applications, advances and challenges," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 370, no. 1958, pp. 158–175, 2012.

[6] R. Smith, Y. Chao, P. Li, D. Caron, B. Jones, and G. Sukhatme, "Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model," *The International Journal of Robotics Research*, vol. 29, no. 12, p. 1475, 2010.

[7] Y. Zhang, J. G. Bellingham, M. A. Godin, and J. P. Ryan, "Using an autonomous underwater vehicle to track the thermocline based on peak-gradient detection," *Oceanic Engineering, IEEE Journal of*, vol. 37, no. 3, pp. 544–553, 2012.

[8] F. S. Hover, "Path planning for data assimilation in mobile environmental monitoring systems," in *IEEE/RSJ International Conf. on Intelligent Robots and Systems*. IEEE, 2009, pp. 213–218.

[9] N. K. Yilmaz, C. Evangelinos, P. Lermusiaux, and N. M. Patrikalakis, "Path planning of autonomous underwater vehicles for adaptive sampling using mixed integer linear programming," *IEEE J. Oceanic Engineering*, vol. 33, no. 4, pp. 522–537, 2008.

[10] J. Das, F. Py, T. Maughan, T. O'Reilly, M. Messié, J. Ryan, G. S. Sukhatme, and K. Rajan, "Coordinated sampling of dynamic oceanographic features with underwater vehicles and drifters," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 626–646, 2012.

[11] S. Petillo, H. Schmidt, and A. Balasuriya, "Constructing a distributed AUV network for underwater plume-tracking operations," *International Journal of Distributed Sensor Networks*, vol. 2012, 2011.

[12] T. Bean, J. Canning, G. Beidler, M. ORourke, and D. Edwards, "Designing and implementing collaborative behaviors for autonomous underwater vehicles," in *Proc. International Symposium on Unmanned Untethered SubmersibleTechnology*, 2007.

[13] O. Imer, S. Yüksel, and T. Başar, "Optimal control of LTI systems over unreliable communication links," *Automatica*, vol. 42, no. 9, pp. 1429–1439, 2006.

[14] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. Sastry, "Foundations of control and estimation over lossy networks," *Proc. IEEE*, vol. 95, no. 1, pp. 163–187, 2007.

[15] E. Garone, B. Sinopoli, and A. Casavola, "LQG control over lossy TCP-like networks with probabilistic packet acknowledgements," *International Journal of Systems, Control and Communications*, vol. 2, no. 1, pp. 55–81, 2010.

[16] V. Gupta, B. Hassibi, and R. Murray, "Optimal LQG control across packet-dropping links," *Systems & Control Letters*, vol. 56, no. 6, pp. 439–446, 2007.

[17] E. Gilbertson, B. Reed, J. Leighton, M. Cheung, and F. Hover, "Experiments in dynamic control of autonomous marine vehicles using acoustic modems," in *Proc. IEEE International Conf. on Robotics and Automation*, 2013.

[18] P. F. Lermusiaux, "Adaptive modeling, adaptive data assimilation and adaptive sampling," *Physica D: Nonlinear Phenomena*, vol. 230, no. 1, pp. 172–196, 2007.

[19] H. Zhang, J. Moura, and B. Krogh, "Dynamic field estimation using wireless sensor networks: Tradeoffs between estimation error and communication cost," *Signal Processing, IEEE Transactions on*, vol. 57, no. 6, pp. 2383–2395, 2009.

[20] J. Kinsey, R. Eustice, and L. Whitcomb, "A Survey of Underwater Vehicle Navigation: Recent Advances and New Challenges," in *IFAC Conference of Manoeuvering and Control of Marine Craft*, Lisbon, Portugal, September 2006, invited paper.

[21] T. P. Sapsis and P. F. Lermusiaux, "Dynamical criteria for the evolution of the stochastic dimensionality in flows with uncertainty," *Physica D: Nonlinear Phenomena*, vol. 241, no. 1, pp. 60–76, 2012.

[22] P. Van Overschee and B. De Moor, "N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems," *Automatica*, vol. 30, no. 1, pp. 75–93, 1994.

[23] E. Fiorelli, N. E. Leonard, P. Bhatta, D. A. Paley, R. Bachmayer, and D. M. Fratantoni, "Multi-AUV control and adaptive sampling in Monterey Bay," *IEEE J. Oceanic Engineering*, vol. 31, pp. 935 –948, 2006.

[24] P. Casari and M. Zorzi, "Protocol design issues in underwater acoustic networks," *Computer Communications*, 2011.

[25] S. Lall and G. Dullerud, "An LMI solution to the robust synthesis problem for multi-rate sampled-data systems," *Automatica*, vol. 37, no. 12, pp. 1909–1922, 2001.

[26] D. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995.

[27] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry, "Kalman filtering with intermittent observations," *IEEE Trans. Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.