

Cooperative Control for Ocean Sampling: The Glider Coordinated Control System

Derek A. Paley, *Student Member, IEEE*, Fumin Zhang, *Member, IEEE*,

Naomi Ehrich Leonard, *Senior Member, IEEE*

Abstract

We describe the implementation of a cooperative control system for underwater gliders. Gliders are a class of autonomous vehicles that are especially well suited for large-scale oceanographic surveys because of their high reliability and endurance. For these same reasons, gliders are an excellent platform for multi-vehicle control. We present the Glider Coordinated Control System (GCCS), a software suite that automatically implements feedback control of a glider fleet to a set of coordinated trajectories. The GCCS can also serve as a simulation testbed for design and evaluation of multi-vehicle control schemes. We include a cooperative control algorithm that achieves collective motion appropriate for ocean sampling. We present experimental validation of the GCCS for both a virtual glider deployment in Monterey Bay, CA and a real glider deployment in Buzzards Bay, MA.

I. INTRODUCTION

Underwater gliders are ocean sensor platforms characterized by high reliability and endurance. In a typical glider deployment, multiple gliders survey the same region of interest in order to sample the ever-changing ocean processes with adequate frequency in space and time. Gliders can operate over periods of weeks, continuously collecting valuable oceanographic data. We use feedback controls to generate coordinated trajectories for the glider fleet that maximize the array gain of the fleet [1]. Automating the coordinated control of gliders is a key to sustained operation of a glider ocean sampling network. Building on previous experience with real-time glider coordination [2], we have designed the Glider Coordinated Control System (GCCS), a software suite that automates the multi-vehicle control loop.

The GCCS also serves as a simulation testbed for development of coordinated glider controls. The GCCS allows simulation and control of gliders under realistic operating conditions in ocean fields that are provided as input. Accordingly, it is possible to use the GCCS to explore and test solutions to many of the challenges that come with controlling a network of gliders in the ocean. A strong, variable flow field, which can at times be stronger even

Research partially supported by ONR grants N00014-02-1-0826, N00014-02-1-0861, and N00014-04-1-0534. D. Paley supported by the National Science Foundation Graduate Research Fellowship, the Princeton University Gordon Wu Graduate Fellowship, and the Pew Charitable Trust grant 2000-002558.

The authors are with the Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ 08544 (email: {dpaley}{fzhang}{naomi}@princeton.edu).

than the forward propulsion of a glider, is one such challenge. Other challenges include long delays in feedback, uncertainty in communication and asynchronicity in feedback and communication. Because a number of these challenges have not yet been fully addressed by theoretical methods, the GCCS as testbed plays an unusually important role in development.

This paper is devoted to a presentation of the GCCS and its features. We further demonstrate the practical use and merit of the GCCS by describing experimental results with the GCCS from a recent virtual deployment in Monterey Bay, CA and a real ocean deployment in Buzzards Bay, MA. The outline of the paper is as follows. In Section II we give a brief overview of underwater glider capabilities and typical ocean sampling objectives. In Section III, we describe a methodology for using cooperative control to stabilize collective motion. In Sections IV and V, we present an overview of the suite of software that has been developed to implement and test feedback control of a glider fleet. We describe in Section VI experimental results from both virtual and real glider deployments and, in Section VII, give concluding remarks.

II. OCEAN SAMPLING WITH UNDERWATER GLIDERS

A. Underwater Glider Capabilities

There are a growing number and types of underwater gliders in service today [3]. To date, the GCCS has incorporated the operational details of Spray gliders, manufactured and operated by the Scripps Institution of Oceanography (SIO) [4], and Slocum gliders, manufactured by Webb Research Corporation and operated by a growing number of customers, which includes Woods Hole Oceanographic Institution (WHOI) [5]. Underwater gliders use wings for propulsion (instead of a propeller) by changing their buoyancy. Consequently, they glide through the water, periodically changing their pitch and buoyancy to inflect or reverse their glide angle. Although they travel at low speeds (0.2 to 0.3 meters per second) relative to propeller-driven underwater vehicles (1 to 3 meters per second), they are capable of much longer deployments (2 to 10 weeks vs. 12 to 36 hours).

During deployments, gliders carry sensors that sample ocean properties such as temperature, salinity, and density at depths up to 1500 meters. Gliders periodically surface to reset their dead-reckoning position error using a Global Position System (GPS). While on the surface, the gliders can receive new instructions from the data servers on shore via Iridium satellites. The gliders can also transmit measured data to the data servers using Iridium. It is possible to equip gliders with acoustic modems so that they can communicate over short distances underwater, but this capability has not yet been utilized in any deployments involving the GCCS.

B. Ocean Sampling Objectives

Because underwater gliders are able to carry a variety of different scientific sensors, they can support a wide range of sampling objectives. Possibilities include improving our understanding of ocean dynamics by monitoring the mass and heat budget of a three-dimensional ocean volume [6] and gaining insight into the spatial and temporal dynamics of ocean biology by measuring optical backscatter and bioluminescence [7]. Due to their high endurance and (relatively) low cost, gliders are a critical component in a general methodology for collecting ocean data known

as an autonomous ocean sampling network [8]. In such a network, glider sampling trajectories can be selected to maximize a variety of metrics such as model evaluation and improvements [9] and objective analysis error [10], [11]. Independent of the metric used, coordinated control algorithms can play an integral role in achieving good sampling performance by a glider fleet [1].

The application of objective analysis (OA), also known as optimal interpolation, to deriving optimal coordinated trajectories and evaluating sampling performance is investigated in [1]. Using a linear combination of sensor measurements, the algorithm provides an estimate for the average of a scalar field and the (mapping) error variance of this estimate assuming we have an *a priori* description of the mean of the field and the covariance of fluctuations around the mean. In the case that the covariance can be suitably approximated by Gaussian-like functions in space and time, then the mapping error is parametrized by the spatial and temporal decorrelation scales. The sampling performance metric described in [1] is the integral of the mapping error over space and time. Coordinated sensor trajectories can be designed and evaluated through the use of such a metric. The GCCS automates the control of multiple gliders to coordinated trajectories and computes metrics like sampling performance in real-time.

III. COOPERATIVE CONTROL OF COLLECTIVE MOTION

To coordinate a fleet of underwater gliders, we decouple the trajectory design from the trajectory tracking control. We approach the trajectory design problem using feedback control of a simplified model of glider motion assuming that the gliders are equipped with onboard controllers capable of following a prescribed trajectory. We describe the approach below and summarize the stabilizing control laws we have derived for collective motion, which are presented in more detail in [12], [13]. We note that alternative control algorithms for glider motion can also be used with the GCCS as in the Buzzards Bay, MA deployment in Section VI-B [14], [15].

A. Particle Model

For designing feedback controls that stabilize collective motion, we use a planar model of point particles with unit mass. The configuration of each particle is its position and heading in the plane. We assume that some, but not necessarily all, pairs of particles can exchange relative position and heading measurements. The interconnection topology is modeled using algebraic graph theory. The particles can be modeled as moving in a flow field and control laws developed accordingly; the design of coordinating control laws for gliders in the presence of a strong flow field is the subject of ongoing work.

The particle dynamics are described by Newton's second law equating the (control) force to the acceleration \ddot{r}_k where $r_k = x_k + iy_k \in \mathbb{C} \equiv \mathbb{R}^2$ is the particle position. Note that we use complex notation for ease and conciseness in presentation. Let $\dot{r}_k = s_k e^{i\theta_k}$ where $s_k \in \mathbb{R}$ and $\theta_k \in \mathbb{T} \equiv S^1$ are the particle speed and heading, respectively. We have

$$\frac{d}{dt}(s_k e^{i\theta_k}) = \dot{s}_k e^{i\theta_k} + s_k \dot{\theta}_k i e^{i\theta_k} \equiv (\nu_k + s_k u_k i) e^{i\theta_k}$$

where we have introduced the thrust $\nu_k = \dot{s}_k \in \mathbb{R}$ and steering $u_k = \dot{\theta}_k \in \mathbb{R}$ control inputs. For constant speed motion with identical particles, we consider only steering control, i.e. $\nu_k = 0$ so that $s_k = s_0$ for all k . The *particle*

model is

$$\begin{aligned}\dot{r}_k &= s_0 e^{i\theta_k}, \\ \dot{\theta}_k &= u_k, \quad s_0 > 0, \quad k = 1, \dots, N.\end{aligned}\tag{1}$$

The configuration space of the particle model is N copies of the special Euclidean group in the plane $SE(2)$. We drop the subscript and use bold to represent a vector such as $\mathbf{r} = (r_1, \dots, r_N)^T$. We use the inner product defined by $\langle \mathbf{x}, \mathbf{y} \rangle = \text{Re}\{\mathbf{x}^* \mathbf{y}\}$, for $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$, $n > 0$, where $*$ is the conjugate transpose.

Let $t \geq 0$ denote time. In the case that $u_k = 0$, then $\theta_k(t) = \theta_k(0)$ and each particle moves in a straight line along its initial heading. If $u_k = \omega_0 s_0 \neq 0$ then $\theta_k(t) = \theta_k(0) + \omega_0 s_0 t$ and each particle moves around a circle with radius $|\omega_0|^{-1}$. The center of the circle is given by

$$c_k = r_k + i\omega_0^{-1} e^{i\theta_k}.$$

If $u_k = u_k(\boldsymbol{\theta})$, then the heading dynamics are independent of the particle motion in the plane and we can consider the *phase model*

$$\dot{\theta}_k = u_k, \quad k = 1, \dots, N.\tag{2}$$

The configuration space of the phase model is \mathbb{T}^N .

The interconnection topology of both the particle and phase models is defined using a fixed, undirected, and connected communication graph $G(V, E)$ with N vertices in $V = \{1, \dots, N\}$ and e edges $(k, j) \in E$ whenever there exists a communication link between particle k and particle j . Let d_k be the degree of the k th node in the interconnection graph and let \mathcal{N}_k denote the set of vertices (neighbors) connected to vertex k . The *Laplacian* matrix of the interconnection graph is $L = D - A = L^T \geq 0$, where the degree and adjacency matrices are given by $D = \text{diag}(\mathbf{d})$ and $[A]_{k,j} = 1$ if $j \in \mathcal{N}_k$ and zero otherwise [16]. The quadratic form

$$Q_L(\mathbf{z}) = \frac{1}{2} \langle \mathbf{z}, L\mathbf{z} \rangle$$

vanishes only when $\mathbf{z} = \mathbf{1}z_0$ [17]. In the case that G is a complete (all-to-all) graph, then every pair of particles can communicate and $L = P \equiv NI - \mathbf{1}\mathbf{1}^T$, where $\mathbf{1} = (1, \dots, 1)^T \in \mathbb{R}^N$ and $I = \text{diag}(\mathbf{1})$. Using $P^2 = NP$, the quadratic form becomes $Q_P(\mathbf{z}) = \frac{1}{2N} \|P\mathbf{z}\|^2$.

If the particles communicate only relative position and heading measurements, the closed-loop particle model (1) is invariant under the action of $SE(2)$. The reduced configuration space of the particle model is $SE(2)^N / SE(2)$. The only relative equilibria of the particle model under the action of $SE(2)$ are parallel motion and circular motion of all particles around the same circle [18]. The reduced configuration space of the phase model under the action of the group of rotations is $\mathbb{T}^N / \mathbb{T}$. Relative equilibria of the phase model are phase-locked, i.e. the relative phases of all pairs of particles are fixed.

B. Stabilizing Collective Motion

We stabilize collective motion of the particle model using feedback controls consistent with the particle interconnection graph after [12], [19]. Parallel motion of the particles occurs when the particle headings are *synchronized*,

i.e. all the particle headings are equal. A circular formation of the particles is stabilized by driving each particle around a circle at a common frequency and synchronizing the centers of all the circles as described below.

To synchronize the particle headings, we stabilize the relative equilibria of the phase model in which the relative phases of all pairs of particles is zero, i.e. $\boldsymbol{\theta} = \theta_0 \mathbf{1}$ where $\theta_0 \in \mathbb{T}$. We introduce the phase potential

$$W_1(\boldsymbol{\theta}) \equiv Q_L(e^{i\boldsymbol{\theta}}) = \frac{1}{2} \langle e^{i\boldsymbol{\theta}}, L e^{i\boldsymbol{\theta}} \rangle,$$

which reaches its global minimum (zero) if the particle headings are synchronized and is positive otherwise. In the following theorem, gradient control is used to synchronize the particle phases. Let L_k be the k th row of the Laplacian of a fixed, undirected, and connected interconnection graph.

Theorem 1: [19] The phase model with the gradient control

$$u_k = K_1 \frac{\partial W_1}{\partial \theta_k} = K_1 \langle i e^{i\theta_k}, L_k e^{i\boldsymbol{\theta}} \rangle, \quad K_1 \neq 0,$$

forces convergence of all solutions to the critical set of $W_1(\boldsymbol{\theta})$. If $K_1 < 0$ (resp. $K_1 > 0$) then the set of synchronized states is asymptotically stable (resp. unstable).

To obtain a circular formation, we stabilize the relative equilibria of the particle model in which each particle orbits the same circle, i.e. $\mathbf{c} = c_0 \mathbf{1}$ where $c_0 \in \mathbb{C}$. The spacing potential

$$Q_L(\mathbf{c}) = \frac{1}{2} \langle \mathbf{c}, L \mathbf{c} \rangle \quad (3)$$

is zero if the particles are in a circular formation. The time-derivative of (3) is

$$\dot{Q}_L(\mathbf{c}) = \sum_{k=1}^N \langle e^{i\theta_k}, L_k \mathbf{c} \rangle (s_0 - \omega_0^{-1} u_k).$$

Gradient-like control is used to synchronize the circle centers.

Theorem 2: The particle model with the gradient-like control

$$u_k = \omega_0 (s_0 + K_0 \langle e^{i\theta_k}, L_k \mathbf{c} \rangle), \quad K_0 > 0, \quad (4)$$

forces convergence of all solutions to $\mathbf{c} = c_0 \mathbf{1}$ where $c_0 \in \mathbb{C}$.

Proof: This is proved by a straightforward adaptation of the corresponding proof in [12] where it was applied to the particular case $L = P$ and $s_0 = 1$. The potential $Q_L(\mathbf{c})$ is positive definite in the reduced space $\mathbb{C}^N / \mathbb{C}$. By the LaSalle invariance principle, solutions of the reduced system converge to the largest invariant set where $\dot{Q}_L(\mathbf{c}) = 0$ i.e.

$$K_0 \langle e^{i\theta_k}, L_k \mathbf{c} \rangle^2 = 0. \quad (5)$$

for $k = 1, \dots, N$. In this set, $u_k = \omega_0 s_0$, each particle orbits a circle of radius $|\omega_0|^{-1}$, and (5) is satisfied only if $L_k \mathbf{c} = 0$, i.e. $\mathbf{c} = c_0 \mathbf{1}$. ■

C. Control to Symmetric Phase Patterns

In the circular control (4), the particle arrangement around the circle is arbitrary. In this section, we isolate symmetric patterns of the particles such as the *splay state*, in which the particle phases are evenly spaced around the unit circle [12]. As long as the particles are undergoing circular motion, symmetric particle formations are characterized by symmetric phase patterns. Symmetric formations on closed convex curves are described in the next subsection.

Let $1 \leq M \leq N$ be a divisor of N . An (M, N) -*pattern* is a symmetric arrangement of N phases consisting of M clusters uniformly spaced around the unit circle, each with N/M synchronized phases. For any N , there exist at least two symmetric patterns: the $(1, N)$ -pattern, which is the synchronized state, and the (N, N) -pattern, which is the splay state [12]. To stabilize (M, N) -patterns, we require that the interconnection graph is d_0 -*circulant*, i.e. L is a circulant matrix and $d_k = d_0$. Any circulant matrix can be diagonalized by the discrete Fourier transform matrix $U_{kl} = e^{i(l-1)(k-1)\frac{2\pi}{N}}$ [20]. If L is circulant, then its eigenvectors can be chosen to be the columns of U . The components of the columns of U form symmetric patterns on the unit circle in the complex plane.

In the following theorem, we refer to the phase potential

$$W^{M,N}(\boldsymbol{\theta}) = \sum_{m=1}^M K_m W_m(\boldsymbol{\theta}) \equiv \sum_{m=1}^M K_m Q_L\left(\frac{1}{m} e^{im\boldsymbol{\theta}}\right) \quad (6)$$

where $K_m \in \mathbb{R}$, $m = 1, \dots, M$.

Theorem 3: [13] In the phase model with a d_0 -circulant interconnection graph, the gradient control

$$u_k = \sum_{m=1}^M \frac{K_m}{m} \frac{\partial W_m}{\partial \theta_k} = \sum_{m=1}^M \frac{K_m}{m} \langle i e^{im\theta_k}, L_k e^{im\boldsymbol{\theta}} \rangle$$

forces convergence of all solutions to the critical set of $W^{M,N}(\boldsymbol{\theta})$. An (M, N) -pattern is asymptotically stable if $K_m > 0$ for $m = 1, \dots, M-1$ and

$$K_M < M \sum_{m=1}^{M-1} \frac{K_m}{m}. \quad (7)$$

The proof in [13] uses the fact that, if $e^{i\bar{\boldsymbol{\theta}}}$ is an eigenvector of L , then $\bar{\boldsymbol{\theta}}$ is in the critical set of $W^{M,N}(\boldsymbol{\theta})$. Stability is proven by computing the spectrum of the Hessian of $W^{M,N}(\boldsymbol{\theta})$ evaluated in an (M, N) -pattern.

D. Symmetric Formations on Closed Curves

The control methodology is extended to stabilize symmetric formations on convex, closed curves in [13]. It is, in principle, possible to track a more general family of curves [14]. Let $\phi : \mathbb{T} \rightarrow [0, 2\pi)$ be a smooth map and $\rho : [0, 2\pi) \rightarrow \mathbb{C}$, $\phi \mapsto \rho(\phi)$, be a parametrization of a smooth, closed convex curve, C , with definite curvature. The tangent vector to C is $\frac{d\rho}{d\phi} \in \mathbb{C}$. If C satisfies the velocity constraint

$$\frac{d\rho}{d\phi} = \left| \frac{d\rho}{d\phi} \right| e^{i\theta_k}, \quad (8)$$

then the center of the curve is

$$c_k = r_k - \rho_k \quad (9)$$

where $\rho_k \equiv \rho(\phi(\theta_k))$.

To obtain a formation on curve C , we drive each particle around curve C and synchronize the curve centers using gradient-like control with respect to the spacing potential (3). The arc length $\sigma : [0, 2\pi) \rightarrow \mathbb{R}^+$ is

$$\sigma(\phi) \equiv \int_0^\phi \left| \frac{d\rho}{d\bar{\phi}} \right| d\bar{\phi} \quad (10)$$

and the local curvature $\kappa : [0, 2\pi) \rightarrow \mathbb{R}$ is

$$\kappa(\phi) \equiv \frac{d\theta}{d\sigma}. \quad (11)$$

By assumption, the curvature is bounded and definite, i.e. $0 < |\kappa(\phi)| < \infty$. Using (10) and (11), we obtain

$$\kappa^{-1}(\phi) = \frac{d\sigma}{d\theta} = \frac{d\sigma}{d\phi} \frac{d\phi}{d\theta} = \left| \frac{d\rho}{d\phi} \right| \frac{d\phi}{d\theta}, \quad (12)$$

where $\kappa^{-1}(\phi) = \frac{1}{\kappa(\phi)}$. Consequently, using (8) and (12),

$$\frac{d\rho}{d\theta} = \frac{d\rho}{d\phi} \frac{d\phi}{d\theta} = e^{i\theta} \kappa^{-1}(\phi). \quad (13)$$

Using (13), the time-derivative of (3) with c_k given by (9) is

$$\dot{Q}_L(\mathbf{c}) = \sum_{k=1}^N \langle e^{i\theta_k}, L_k \mathbf{c} \rangle (s_0 - \kappa_k^{-1} u_k)$$

where $\kappa_k \equiv \kappa(\phi(\theta_k))$. Therefore, the particle model with the control (4) and ω_0 replaced by κ_k forces convergence of all solutions to curve C with $\mathbf{c} = c_0 \mathbf{1}$, $c_0 \in \mathbb{C}$.

Let Ω be the perimeter of the curve given by $\Omega = \sigma(2\pi) > 0$. The curve-phase is [13]

$$\psi(\phi) \equiv \frac{2\pi}{\Omega} \sigma(\phi). \quad (14)$$

Using (12) and (14), we obtain the *curve-phase model*,

$$\dot{\psi}_k = \frac{2\pi}{\Omega} \kappa_k^{-1} \dot{\theta}_k, \quad (15)$$

where $\psi_k \equiv \psi(\phi(\theta_k))$. Control of the particle relative curve-phases is a straightforward adaptation of the results in the previous subsections for particle phases with θ replaced by ψ [13]. Symmetric formations on convex, closed curves are characterized by symmetric curve-phase patterns. For example, the *splay state formation* is a formation of particles on curve C in which the curve-phases ψ are in a splay state. Curve-phase patterns are stabilized identically to phase patterns with the phase model (2) replaced by the curve-phase model (15).

To stabilize formations with symmetric curve-phase patterns, we consider gradient-like control with respect to the composite potential

$$V(\mathbf{c}, \boldsymbol{\psi}) = K_0 Q_L(\mathbf{c}) + \frac{\Omega}{2\pi} W^{M,N}(\boldsymbol{\psi}), \quad K_0 > 0,$$

where $Q_L(\mathbf{c})$ is given by (3) with c_k from (9) and $W^{M,N}(\boldsymbol{\psi})$ is given by (6) with θ replaced by ψ .

Theorem 4: [13] Let C be a closed convex curve with definite curvature. In the particle model with a d_0 -circulant interconnection graph, the composite control

$$u_k = \kappa_k \left(s_0 + K_0 \langle e^{i\theta_k}, L_k \mathbf{c} \rangle + \sum_{m=1}^M \frac{K_m}{m} \langle i e^{im\psi_k}, L_k e^{im\psi} \rangle \right) \quad (16)$$

with $K_m > 0$, $m = 1, \dots, M-1$, and K_M given by (7) asymptotically stabilizes the set of symmetric (M, N) -formations in which each particle orbits curve C centered at $c_0 \in \mathbb{C}$.

Simulations suggest that the condition on gain K_M given in (7) is conservative. For example, simulations show local convergence to the desired (M, N) -pattern for $|K_m| = K_0$, $m = 1, \dots, M$, which does not satisfy (7). In addition, for $M = N$, simulations show convergence to the splay state when $K_m = 0$ for $m > \lfloor \frac{N}{2} \rfloor$, the largest integer less than or equal to $\frac{N}{2}$. We observe in simulation that the desired formation has a large basin of attraction if the complete (all-to-all) interconnection graph is used.

We control the location of the center of the curve tracked by the k th particle as follows. Let $\mathbf{b} \in \mathbb{C}^N$ denote the vector of desired curve centers or beacons. To drive the particles to orbit these beacons in an (M, N) -pattern, we replace the potential $V(\mathbf{c}, \psi)$ with

$$\hat{V}(\mathbf{c}, \psi) = \frac{K_0}{2} \|\mathbf{c} - \mathbf{b}\|^2 + \frac{\Omega}{2\pi} W^{M,N}(\psi), \quad K_0 > 0.$$

In this case, if we replace the control (16) with

$$u_k = \kappa_k \left(s_0 + K_0 \langle e^{i\theta_k}, c_k - b_k \rangle + \sum_{m=1}^M \frac{K_m}{m} \langle i e^{im\psi_k}, L_k e^{im\psi} \rangle \right), \quad (17)$$

solutions converge to the (M, N) -pattern on curve C centered at $\mathbf{c} = \mathbf{b}$ [13].

For illustration of the method, we use a class of smooth, closed convex curves known as superellipses, which includes circles, ellipses, and rounded rectangles. A parametrization of a superellipse and its derivative are given by equation (18) in Appendix A. Superellipses with $p = 1$, i.e. ellipses and circles, have definite curvature. For $p \geq 3$, the curvature κ_k is zero for $\{\theta_k \mid \theta_k = \frac{\pi}{2}j, j = 0, 1, 2, 3\}$ which is a set of measure zero. Simulations suggest that Theorem 4 holds for superellipses with $p \geq 3$, provided that care is taken to avoid singularities in the control law. The six symmetric patterns for $N = 12$ on a superellipse with $p = 3$ are shown in Figure 1(a) for a 4-circulant interconnection graph. Figure 1(b) shows the steady state configuration for $M = 4$ and $\mathbf{b} = ((0, 15i, 30i) \otimes \text{diag}(1, 1, 1, 1))\mathbf{1}$, where \otimes is the Kronecker tensor product.

IV. GLIDER COORDINATED CONTROL SYSTEM

The GCCS is a cross-platform software suite written in MATLAB that automatically implements feedback control of a glider fleet. The GCCS receives position and depth-averaged flow estimates for each glider when it surfaces. Using a (coordinated) control law, the GCCS plans trajectories for each glider and transmits these plans to each glider as a list of *waypoints*. A waypoint is the latitude and longitude coordinate of a destination in space. While it

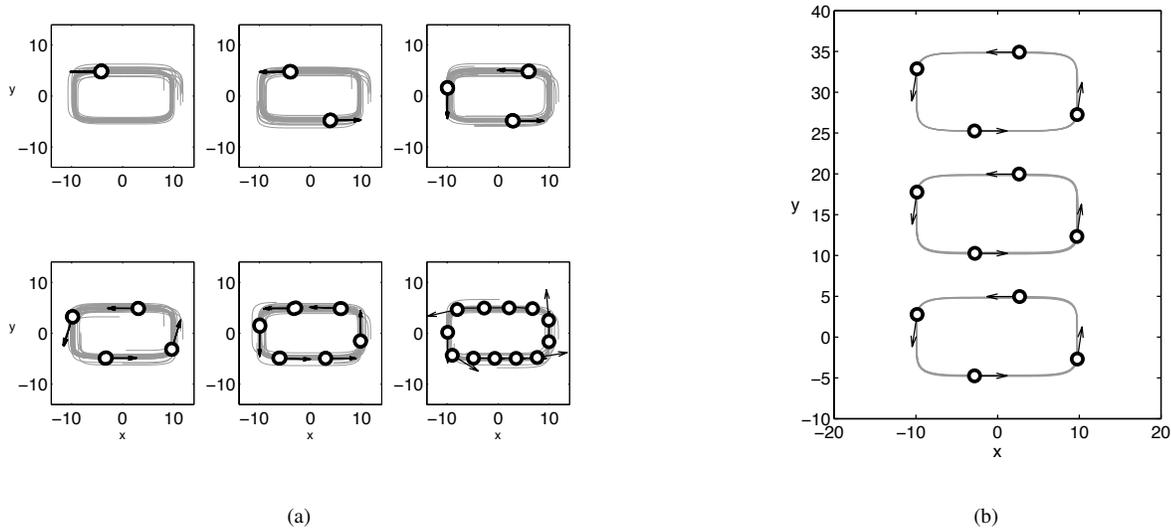


Fig. 1. (a) Numerical simulations of the control (17) with $K_0 = 0.1$ and fixed beacon locations $\mathbf{b} = \mathbf{0}$ for a 4-circulant graph and random, local initial conditions. The simulation duration is 8Ω , or approximately eight revolutions of the superellipse. The panels show the six symmetric patterns for $N = 12$ on a superellipse with $a = 10$, $b = 5$, $p = 3$ and $M = 1, 2, 3, 4, 6$, and 12 clusters. For each $M < N$, the gains are $K_m = 0.1$ for $m = 1, \dots, M - 1$ and $K_M = -0.1$. For $M = N = 12$, the gains are $K_m = 0.1$ for $m = 1, \dots, 6$ and $K_m = 0$ for $m > 6$. The steady-state curve-phase differences between the clusters in each simulation are equal to $\frac{2\pi}{M}$. (b) Steady state configuration of the control (17) with $K_0 = 0.1$, $M = 4$, and fixed beacon location $\mathbf{b} = ((0, 15i, 30i) \otimes \text{diag}(1, 1, 1, 1))\mathbf{1}$.

is underwater, each glider uses its onboard control to follow the trajectory defined by the waypoint list. To derive waypoints for the gliders, which surface asynchronously, the GCCS uses two different models: (1) a detailed glider model with flow that is integrated to estimate glider motion and (2) a particle model (with or without coordinated control) and initial conditions estimated by the detailed model that is integrated to compute desired trajectories. We refer to the software that integrates these two models as the glider integrator and particle integrator, respectively.

There are three major components in the GCCS: the glider planner, the glider simulator, and the remote input/output (IO). The simulator is used as a coordinated control testbed and is described in Section V. Both the planner and the simulator use the glider model in order to integrate the glider trajectories in geodetic coordinates. To date, detailed models of the Slocum (Webb/WHOI) and Spray (UCSD/SIO) gliders which include an implementation of the onboard navigation algorithms are available in the GCCS. The planner uses the particle model in order to generate candidate glider trajectories in a local (flat) coordinate frame. These trajectories are discretized using a waypoint generator. The waypoints are transmitted over the Internet to the glider data servers by way of the remote IO module as shown in Figure 2. Likewise, the glider positions and flow estimates are transmitted over the Internet from the glider data servers to the GCCS by way of the remote IO module. The GCCS source code is kept in a Concurrent Versioning System (CVS) repository to support ongoing development and to support archival and distribution needs.

There are two separate coordinate systems used in the GCCS. The glider integrator uses geodetic coordinates

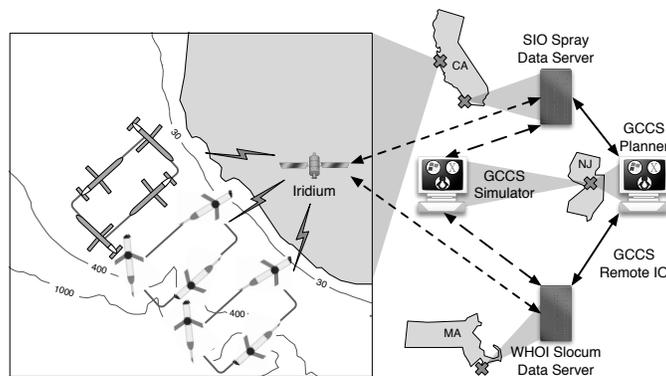


Fig. 2. The GCCS schematic diagram. The three major components of the GCCS are shown in a sample glider deployment: the glider planner, the glider simulator, and the remote IO. The glider simulator is used for virtual trials, the glider planner generates glider waypoints, and the remote IO module communicates with the SIO and WHOI glider data servers which, in turn, communicate with the gliders via Iridium satellites. The short dashed lines are used in a real deployment and the long dashed lines are used in a virtual deployment. Sample trajectories are shown in Monterey Bay, CA.

for the position of the k th glider on the earth $R_k = (\lambda_k, \phi_k) \in S^2$ where λ_k and ϕ_k are latitude and longitude, respectively. The particle integrator uses local coordinates for the position of the k th particle $r_k \in \mathbb{C} \equiv \mathbb{R}^2$. The distance $\Gamma : S^2 \times S^2 \rightarrow \mathbb{R}^+$ and azimuth $\eta : S^2 \times S^2 \rightarrow S^1$ functions for the geodetic coordinates are described in detail in Appendix B. Transformations to convert between the geodetic coordinates used for tracking the glider position and the local coordinates used for the particle positions are implemented in the GCCS but are not described here.

A. Glider Planner

The glider planner is the component of the GCCS that encapsulates the multi-vehicle control algorithm. In addition to various configuration parameters, the planner uses as input the glider surface positions, the glider depth-averaged flow estimates, the glider active waypoint lists, the parameters that define the glider coordinated trajectories (GCTs), and the control parameters. Glider position, flow estimates, and waypoint lists are obtained in real-time over the Internet from the glider data servers. The GCTs and control parameters are specified by a highly flexible input file [21] and are dictated by the objectives of the deployment. A planning cycle is initiated whenever a glider surfaces and the cycle ends when new waypoints are generated for all gliders. The waypoints are transmitted over the Internet to the glider data servers where they are queued for delivery to the gliders via satellite. Due to operational constraints, the feedback control is not computed while the gliders are on the surface. Instead, waypoints are computed and made available prior to each surfacing as shown in Figure 3.

The planning feedback loop is shown in Figure 4. At the start of each planning cycle, the planner checks to see if any gliders have surfaced since the last control cycle started. When a glider surfaces, it obtains a GPS position fix and computes a depth-averaged flow estimate for the previous dive. Then the glider transmits the position, flow estimate, and, optionally, ocean data measurements to the glider data server via Iridium satellite. The planner

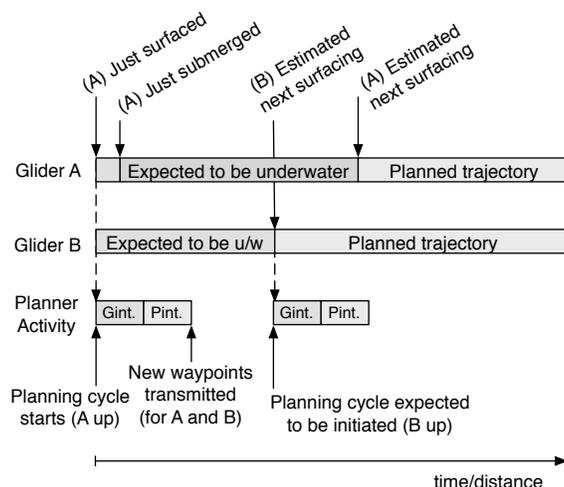


Fig. 3. The glider planner timeline for two gliders. The planning cycle is initiated when glider A surfaces. At the beginning of the planning cycle, the glider integrator (“Gint.” in the figure) predicts the next surface time and location of glider A (the next surface time and location of glider B was already computed on the previous planning cycle). Then the particle integrator (“Pint.” in the figure) generates planned trajectories using the predicted surface times and locations as initial conditions. The waypoints are generated by discretizing these trajectories. The planning cycle is complete when new waypoints for both gliders A and B are transmitted to the glider data servers. The next planning cycle is expected when glider B is due to surface.

checks the glider data servers to see if the glider position files have been updated. If a glider has surfaced, it may submerge again on its next dive before the planner detects the new surface position. For each glider that has surfaced, the planner calculates the glider’s effective horizontal speed and error in the expected surface position and time. The planner integrates the detailed glider model to estimate each glider’s underwater position and next surfacing location and time using the actual surface location, depth-averaged flow estimate, and active waypoint list. If the active waypoint file is more than one dive old, the planner estimates that the current waypoint is the next waypoint in the list after the waypoint closest to the glider. If more accurate flow estimates are not available (e.g. from an ocean model forecast), the glider’s onboard depth-averaged flow estimates are used in the glider integrator. In addition, the planner estimates the surface flow either by using multiple GPS fixes from a single surfacing or by comparing the surface and dive positions of consecutive dives. The planner stores the integrated glider trajectory for subsequent planning cycles until the glider surfaces again.

After running the glider integrator for every glider that has surfaced on the current control cycle, the planner uses the particle integrator to generate planned trajectories. Each glider’s next expected surface time and location (in local coordinates) are used as initial conditions for the particle integrator. The particle integrator uses the control module specified in the GCT file to generate the trajectories. Because the gliders are configured to communicate only with a centralized data server on shore, the time-averaged communication topology is effectively all-to-all, although the control module may not utilize communication between gliders if they are not coordinated. The generated trajectory of each glider is converted to a list of waypoints and, if the waypoint list passes the quality control filter, it is

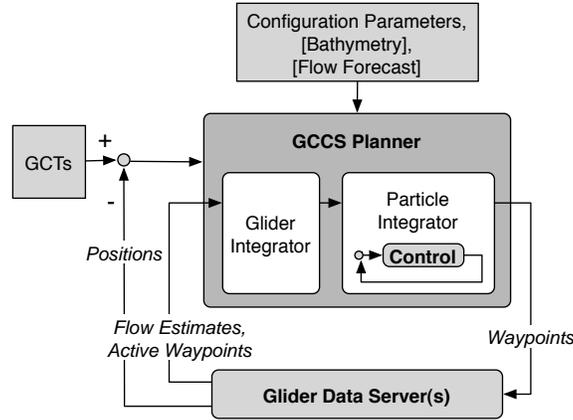


Fig. 4. The glider planner feedback loop. The planner uses as input the glider surface positions, the glider depth-averaged flow estimates, the glider active waypoint lists, the parameters that define the glider coordinated trajectories (GCTs), and various other configuration parameters. In addition, digital bathymetry data and flow forecasts can be used. The outer feedback loop drives the gliders to the GCTs. The inner feedback loop (denoted “Control” in the particle integrator) generates the planned glider positions using the control algorithm specified in the GCT file.

transmitted to the glider data server in geodetic coordinates. The glider data server transmits the waypoint list to the glider when it surfaces next. All planning steps are reported to the planner log file for troubleshooting purposes.

B. Glider Integrator

The underway glider trajectories are estimated using the detailed glider model, which is a discrete-time model. We refer to the GCCS module used in the glider planner and glider simulator that integrates the glider model as the glider integrator. The glider-specific parameters used by the glider integrator are stored in a configuration file. The glider underwater position and next surface location are estimated using a three-dimensional kinematic model which assumes a fixed vertical speed and glide angle (pitch angle plus angle of attack) for both ascent and descent. The dynamic effects of the glider onboard pitch and heading tracking controls are assumed to occur on a faster time scale than the glider model and, consequently, are ignored. Gliders move at constant speed in the direction of their desired headings and are advected by a three-dimensional flow field. In the absence of better flow predictions, the glider’s onboard estimate from the previous dive is used. If an environmental data set is available, the glider integrator will sample the temperature and salinity as a function of depth along the glider trajectory and generate *profiles*. A profile is the sampled data from a single descent/ascent cycle. If local digital bathymetry data is available, the glider integrator enforces the glider minimum altitude constraints.

Notation Like particles, gliders are indexed by $k = 1, \dots, N$. As before, let $t \geq 0$ represent absolute time in the glider integrator. The time step and discrete-time step index are denoted by $\Delta t \in \mathbb{R}^+$ and $n \in \mathbb{Z}$. The superscripts ^{sur} and ^{uw} refer to surface and underwater, respectively. The depth symbols used are glider k depth z_k , glider k minimum inflection depth z_k^{\min} , glider k maximum inflection depth z_k^{\max} , glider k minimum altitude Z_k^{\min} , and bathymetry z^{uw} . The time symbols used are glider k dive initialization time t_k^{ini} , glider k dive surface time t_k^{sur} ,

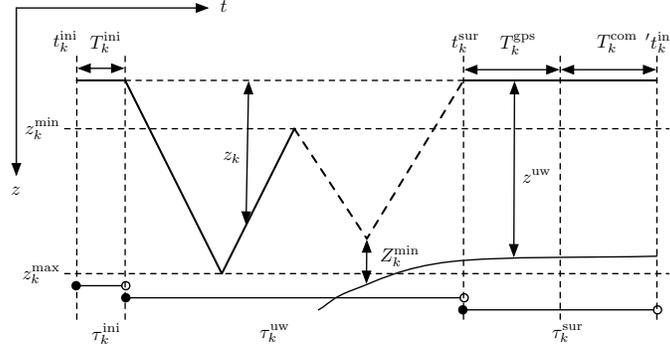


Fig. 5. Coordinates for time t and depth z calculations during a single dive interval in which the glider moves from left to right. The dive initialization time is t_k^{ini} and the surface time is t_k^{sur} . The next dive is initialized at $t_k^{\text{ini}'}$ after the glider obtains a GPS fix and performs two-way communication with the glider data server. See text for other time and depth symbol definitions.

glider k pre-dive surface duration T_k^{ini} , glider k post-dive GPS duration T_k^{gps} , and glider k communication duration T_k^{com} . The time intervals shown are, before the dive $\tau_k^{\text{ini}} = [t_k^{\text{ini}}, t_k^{\text{ini}} + T_k^{\text{ini}})$, during the dive $\tau_k^{\text{uw}} = [t_k^{\text{ini}} + T_k^{\text{ini}}, t_k^{\text{sur}})$, and after the dive $\tau_k^{\text{sur}} = [t_k^{\text{sur}}, t_k^{\text{sur}} + T_k^{\text{gps}} + T_k^{\text{com}})$. For convenience, we denote the end of interval τ by $\bar{\tau}$. The depth and time symbols are illustrated in Figure 5.

Position The k th glider estimated position $R_k \in S^2$ at time $t \geq t_k^{\text{ini}}$ is obtained by integrating the following discrete-time model, which is a function of the position R_k , depth $z_k \geq 0$, and waypoint index $p_k \in \mathbb{N}$. The p th waypoint is denoted by $\omega_k^p \in S^2$. Let $f_k \in \mathbb{R}^2$ be the horizontal component of the k th glider inertial velocity (with respect to an earth-fixed frame).

- (1) Before the dive, $t \in \tau_k^{\text{ini}} = [t_k^{\text{ini}}, t_k^{\text{ini}} + T_k^{\text{ini}})$,

$$R_k(n+1) = R_k(n) + f_k^{\text{sur}}(R_k(n))\Delta t, \\ n = 1, \dots, \lfloor \frac{t - t_k^{\text{ini}}}{\Delta t} \rfloor,$$

where $R_k(1)$ is the position of the glider at t^{ini} . Since the gliders have no propulsion on the surface, the total surface velocity f^{sur} is equal to the flow velocity on the surface.

- (2) During the dive, $t \in \tau_k^{\text{uw}} = [t_k^{\text{ini}} + T_k^{\text{ini}}, t_k^{\text{sur}})$,

$$R_k(n+1) = R_k(n) + f_k^{\text{uw}}(R_k(n), z_k(n), p_k(n))\Delta t, \\ n = \lfloor \frac{\bar{\tau}_k^{\text{ini}}}{\Delta t} \rfloor, \dots, \lfloor \frac{t - t_k^{\text{ini}}}{\Delta t} \rfloor$$

The glider waypoint number $p_k(n)$ is determined by integrating from starting waypoint number using

$$p_k(n+1) = \begin{cases} p_k(n) + 1 & \text{if } \Psi(R_k(n), p_k(n)) \\ p_k(n) & \text{otherwise} \end{cases}$$

where $\Psi(R_k(n), p_k(n))$ is a boolean waypoint completion condition. Once a waypoint is completed, then the glider heads towards the next waypoint in the list. For example, the circle condition is

$$\Psi^{\text{cir}}(R_k(n), p_k(n)) = \Gamma(R_k(n), \omega_k^{p_k(n)}) < \Gamma_0,$$

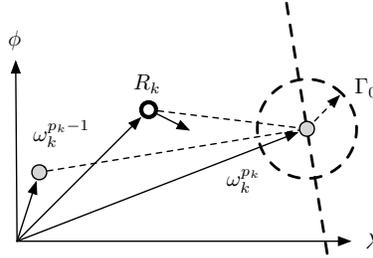


Fig. 6. The waypoint completion conditions. The glider position R_k , previous waypoint $\omega_k^{p_{k-1}}$, and current waypoint $\omega_k^{p_k}$ are depicted in geodetic coordinates where λ and ϕ represent longitude and latitude, respectively. The circle condition is satisfied if the glider enters the dashed circle of radius Γ_0 centered at the current waypoint. The finish line condition is satisfied if the glider crossed the dashed line that passes through the current waypoint and is perpendicular to the line connecting the previous and current waypoints. The glider desired heading, depicted by the arrow originating at R_k , is computed onboard the glider and estimated by the GCCS. Because the glider desired heading may compensate for the local flow, the heading vector is necessarily aimed directly at the current waypoint.

where Γ is the distance function defined in Appendix B and $\Gamma_0 \in \mathbb{R}^+$ is the radius of the waypoint circle. In the presence of strong flow, the circle condition combined with a simple heading algorithm that steers the glider toward the waypoint, can result in the glider turning directly into the flow as it approaches the waypoint. An alternate waypoint completion condition is the finish line condition, which is satisfied if the glider crosses the line that passes through the current waypoint and is perpendicular to the line passing through both the previous and current waypoints. The finish line condition combined with a heading algorithm that steers the glider toward the finish line near the waypoint can be an effective navigation strategy in the presence of strong flow. Both the circle and finish line conditions are illustrated in Figure 6.

The horizontal component of the glider total underwater velocity f^{uw} is the sum of the horizontal glider velocity relative to the flow and the estimated horizontal flow velocity. The horizontal glider speed relative to the flow is estimated from the vertical speed, pitch angle, and angle of attack. The orientation of the horizontal glider velocity is assumed to be the desired heading of the glider, which depends on the glider estimate of the flow and the current waypoint ω_k^p . The glider desired heading calculations are implemented in the GCCS but omitted here since the code is proprietary.

$$(3) \text{ After the dive, } t \in \tau_k^{\text{sur}} = [t_k^{\text{sur}}, t_k^{\text{sur}} + T_k^{\text{GPS}} + T_k^{\text{com}}),$$

$$R_k(n+1) = R_k(n) + f_k^{\text{sur}}(R_k(n))\Delta t,$$

$$n = \lfloor \frac{\bar{\tau}_k^{\text{uw}}}{\Delta t} \rfloor, \dots, \lfloor \frac{t - t_k^{\text{ini}}}{\Delta t} \rfloor.$$

Depth Let g^{uw} be the vertical component of the glider total velocity. The dive direction is given by $\zeta_k \in \{-1, 1\}$, where positive is descending. The k th glider depth at time t is estimated by integrating the following discrete-time model, which is a function of the position R_k , depth z_k , and dive direction ζ_k .

$$(1) \text{ Before the dive, } t \in \tau_k^{\text{ini}},$$

$$z_k(n+1) = 0, \quad n = 1, \dots, \lfloor \frac{t - t_k^{\text{ini}}}{\Delta t} \rfloor, \quad z_k(1) = 0$$

(2) During the dive, $t \in \tau_k^{\text{uw}}$,

$$z_k(n+1) = z_k(n) + g_k^{\text{uw}}(R_k(n), z_k(n), \zeta_k(n))\Delta t, \\ n = \lfloor \frac{\bar{\tau}_k^{\text{ini}}}{\Delta t} \rfloor, \dots, \lfloor \frac{t - t_k^{\text{ini}}}{\Delta t} \rfloor.$$

The vertical component of the glider velocity g^{uw} is the sum of the glider vertical velocity relative to the flow and the estimated vertical flow velocity (if available). The dive direction $\zeta_k \in \{-1, 1\}$ is estimated by integrating from the initial condition $\zeta_k(1) = 1$ using

$$\zeta_k(n+1) = \begin{cases} -1 & \text{if } \Phi(R_k(n), z_k(n), t) \\ 1 & \text{if } z_k(n) < -\zeta_k(n)z_k^{\text{min}} \\ \zeta_k(n) & \text{otherwise.} \end{cases}$$

where $\Phi(R_k(n), z_k(n), t)$ is a boolean condition given by

$$\Phi(z_k(n), t) = (z_k(n) > z_k^{\text{max}}) \cup (t > T_k^{\text{max}}) \\ \cup (z_k(n) > z^{\text{uw}}(R_k(n)) - Z_k^{\text{min}}).$$

In words, the glider ascends if it exceeds its maximum inflection depth, if it exceeds the maximum dive duration, or if its measured altitude is less than the minimum allowable altitude. If the glider is ascending, the dive direction switches if it is shallower than the minimum inflection depth.

(3) After the dive, $t \in \tau_k^{\text{sur}}$,

$$z_k(n+1) = 0, \quad n = \lfloor \frac{\bar{\tau}_k^{\text{uw}}}{\Delta t} \rfloor, \dots, \lfloor \frac{t - t_k^{\text{ini}}}{\Delta t} \rfloor.$$

The flow velocities used in the glider integrator are context dependent. In the glider simulator, the flow velocities are sampled from an environmental data file, e.g. a virtual ocean, or are generated by an environmental data function, e.g. a virtual tidal model. In the glider planner, the flow velocities can be sampled from an ocean forecast or, if a forecast is not available, the flow velocities can be estimated from the glider data. The estimated horizontal (depth-averaged) flow velocity is calculated by each glider when it surfaces, e.g. by obtaining the relative position of the first GPS fix obtained after surfacing to the dead-reckoned surface position, scaling this vector by the time underwater, and adding it to the previous flow estimate (if it was used in dead-reckoning). The estimated surface flow velocity is computed by the planner as described in the previous subsection. Both the estimated glider position and the estimated glider onboard (dead-reckoned) position are integrated since the dead-reckoned position is used in the desired heading and waypoint completion calculations. These positions estimates differ if different flow velocities are used.

C. Particle Integrator

The future glider trajectories are planned using a particle model under closed-loop control. We refer to the numerical tool that solves the particle model as the particle integrator. The particle model can be configured to run continuous- or discrete-time, piecewise-circular or piecewise-linear trajectories, and first-order or second-order

dynamics. The choice of MATLAB ODE solver and solver options can be set by a configuration file. The GCCS has a standard interface to accommodate coordinated control algorithms. Once the glider trajectories have been planned, they are converted to waypoint lists and transmitted to the glider data server. The waypoints can be uniformly spaced in time or non-uniformly spaced as a function of the path curvature. Each waypoint list must pass a quality control filter with a number of spacing and location conditions.

At the core of the particle integrator is the capability to apply a (possibly continuous) multi-vehicle control law to a group of particles, some of which do not initially respond to the control. When a glider surfaces, the glider integrator estimates its trajectory from the current surface location to the next surface location using the active waypoint list. While the glider is predicted to be underwater, the estimated glider trajectory can be used by the particle integrator to produce coordinated controls for the other particles but the estimated trajectory does not respond to control inputs. The glider becomes a particle at the time and location corresponding to its next surfacing. Planning continues beyond this time. Therefore, once the glider becomes a particle, the rest of its trajectory is planned by the particle integrator as shown in Figure 3.

The particle integrator uses the control module specified by the GCT file. The control interface is designed to facilitate adding new control algorithms to the GCCS. Each control algorithm must be able to extract its parameters from the GCT file. In addition, since the glider heading is not published in the position file when it surfaces, the control law is able to set the initial heading used in the particle integrator. For example, in the GCCS implementation of the coordinated control (17), the initial heading θ_k is chosen to minimize $(c_k - b_k)^2$, which is the component of the Lyapunov function which describes the error in the curve center c_k position relative to beacon b_k .

Particle Model Revisited The feedback controls from Section III are designed with respect to the continuous particle model (1). However, to support ongoing control design efforts the GCCS can be configured to use alternative particle models described below. A particle model with discrete control is

$$\begin{aligned}\dot{r}_k &= s_0 e^{i\theta_k} \\ \dot{\theta}_k &= u_k(t^u)\end{aligned}$$

where T^u is the control interval and $t^u = \lfloor \frac{t}{T^u} \rfloor T^u$. This model gives rise to piecewise circular trajectories. A particle model with discrete control and piecewise linear trajectories is

$$\begin{aligned}\dot{r}_k &= s_0 e^{i\theta_k(t^u)} \\ \theta_k(t^u) &= \theta_k(t^u - T^u) + u_k(t^u)T^u.\end{aligned}$$

A kinematic particle model is

$$\begin{aligned}\dot{r}_k &= s_0 e^{i\theta_k} \\ \dot{\theta}_k &= u_k.\end{aligned}$$

The speed s_0 used in the particle model is the effective horizontal speed of the glider, which is the horizontal speed of the glider relative to the flow scaled by $T^{\max}/(T^{\max} + T^{\text{ini}} + T^{\text{gps}} + T^{\text{com}})$. Incorporating the flow estimate

into the particle model is the subject of ongoing work.

Waypoint Generation and Quality Control Once the glider trajectories have been planned, they are converted to waypoint lists and transmitted to the glider data server only if they pass quality control. The uniform time waypoint generator plans waypoints that are equally spaced by travel time. The curvature dependent waypoint generator chooses waypoints along the particle trajectory such that portions of the trajectory with lower (resp. higher) curvature generates fewer (resp. more) waypoints subject to a maximum (resp. minimum) time condition.

To pass the quality control filter, (1) all waypoints other than those at the start of the list must be inside a prescribed bounding box; (2) the waypoints must not be shallower than the glider minimum operating depth; and (3) the waypoints must be spaced by no more (resp. less) than the maximum (resp. minimum) allowable spacing. If the waypoints meet the quality control requirements then they are converted to geodetic coordinates, written to a glider coordinated control waypoint file, and transmitted to the glider data server. Each waypoint file is given a unique message number and the period of time during which it is valid. This feature is intended to provide robustness to delays and errors incurred in satellite communication between the glider data server and the glider.

D. Remote Input/Output

The remote IO module of the GCCS has two main functions. First, it implements read and write functionality for a variety of file formats including glider-specific files. Second, it implements both standard and secure File Transmission Protocol (FTP) for communication with the glider data servers. Which protocol is used depends on the glider data server. With proper local and/or remote permissions, the remote IO module can also execute commands on a remote server and send email.

There are read and write capabilities for four main file formats implemented in the remote IO module. (1) The glider-specific position and/or profile files (ASCII format); (2) the glider coordinated trajectory file (XML format) [21]; (3) the environment and digital bathymetry data files (NetCDF, ASCII, or binary format, read only); and (4) the glider coordinated control waypoint file (ASCII format). The remote IO can also parse the planner log file for analysis purposes.

There are put, get, and get newer capabilities for four file transfer utilities implemented in the remote IO module. (1) Standard FTP; (2) secure FTP (using SSH); (3) `rsync` and `scp` (using SSH); and (4) `mail`. In order to be automated, the secure protocols require placement of the user RSA public key on the remote server. The `rsync` utilities reduce the data transmission requirements by transferring only the difference between the new and old files. The `mail` utility is used to broadcast planner error messages to the GCCS administrator.

V. GLIDER SIMULATION TESTBED

In addition to its function as a real-time automated fleet controller, the GCCS can also serve as simulation testbed for designing glider coordinated controls. As described above, a central advantage is the ability to test strategies in the presence of strong flow and communication and feedback constraints and uncertainties, challenges which are not yet fully addressed by theoretical methods. The simulated flow fields can be sampled from an virtual ocean

data file or can be generated from a tidal flow model. Simulated gliders produce measurement profile files in the same format as those produced in real deployments.

One of the main goals in developing the glider simulator was to obtain a highly configurable testbed. As a result, many parameters pertaining to the simulated gliders, virtual environment, or simulator itself can be configured by an input file. For example, the glider initial conditions and times can be specified in an input file. The glider simulator can also initialize the gliders using the most recent surface positions and times from the glider data servers. The simulator can be configured to run at, or faster than, real time and can be configured to stop simulating each glider after a prescribed number of dives, length of (glider) time, or number of repetitions of a waypoint list. The glider simulator communicates with the glider planner in exactly the same way as a real glider, i.e. by using the glider data servers.

At the start of the simulation, each simulated glider holds station until a valid, new waypoint file appears on the glider data server. The waypoint file is valid if its expiration time has not passed and is considered new if the message id number is greater than the message id number of the previous waypoint file received by the simulated glider. Once a simulated glider has received a waypoint file, its position and depth are computed using the glider integrator and virtual environment (flow and bathymetry). The onboard glider-specific software used to determine the glider desired underwater heading and the estimated depth-averaged flow are implemented in the glider simulator but not described here. After each dive, the simulator produces a glider profile file and transfers it to the glider data servers. The simulated glider profile files contain the temperature and salinity values as a function of depth that are sampled from the environment data file. The simulator also publishes glider-specific position, active waypoint, and waypoint log files to replicate the real gliders when appropriate.

VI. EXPERIMENTAL RESULTS

We describe two recent GCCS glider deployments: one virtual deployment in Monterey Bay, CA and one real deployment in Buzzards Bay, MA. In the virtual deployment, the GCCS was also used as the glider simulator. Future deployments are planned for Great South Channel, MA and Monterey Bay, CA.

A. Virtual Glider Deployment in Monterey Bay, CA

As part of the Adaptive Sampling and Prediction (ASAP) project [22], the GCCS controlled ten gliders in a virtual deployment in Monterey Bay, CA in March 2006. The simulated gliders operated in ocean conditions from August 2003. An ocean science focus of ASAP is to gain a better understanding of the three-dimensional ocean dynamics off Pt. Año Nuevo, CA by densely sampling the area with sensor-equipped underwater gliders. The virtual deployment was a pilot exercise intended, in part, to test the GCCS in preparation for the real ASAP deployment in August 2006. Four Spray and six Slocum gliders were initially deployed in the vicinity of Pt. Año Nuevo on three superelliptic tracks circumscribed by a 20 kilometer by 40 kilometer box approximately tangent to the 30 meter isobath. A fourth track was added to adapt to changing ocean conditions during the exercise. The desired and actual glider trajectories are shown in Figure 7.

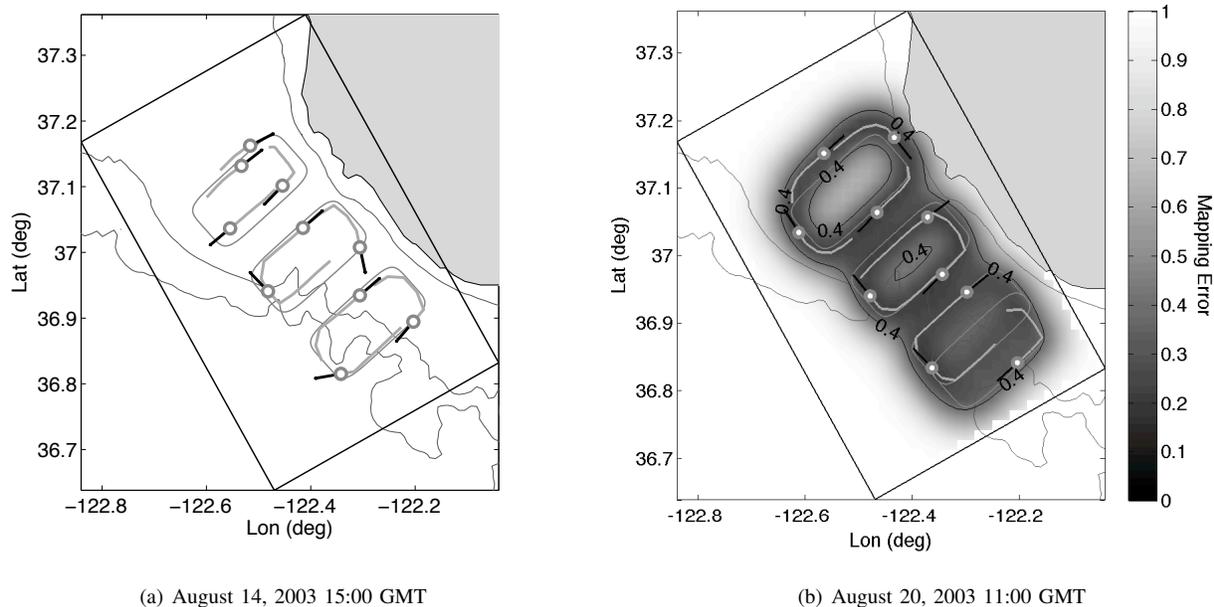


Fig. 7. GCCS planner virtual deployment results for ten underwater gliders from March, 2006 in an ocean model from Monterey Bay, CA circa August, 2003. Each glider is depicted by a grey circle and line segment, where the line segment denotes the relative direction between the two most recent surface positions. The trailing grey line shows the glider trajectory over the previous 12 hours. The superelliptic tracks have dimensions 10 kilometers by 20 kilometers. The bathymetry contour lines correspond to 30, 400, and 1000 meters. The large rectangular region denotes the area covered by the virtual ocean fields. 7(a): Four Spray gliders are controlled to a symmetric formation on the northernmost track and six Slocum gliders are controlled to a symmetric curve-phase formation with three clusters on two tracks. 7(b): The Spray coordination performance improves after the glider configuration is altered to increase their effective speed by increasing the satellite communication interval. One Slocum glider is moved to a fourth track and remains coordinated with the other Slocum gliders. The OA mapping error was computed using spatial and temporal decorrelation scales of 5 kilometers and 24 hours, respectively. The 0.4 mapping error contour line is shown.

During the deployment, the GCCS simulated the glider motion in a virtual ocean field. The virtual ocean used for this deployment was generated by the Harvard Ocean Prediction System [23] from data collected during the 2003 Autonomous Ocean Sampling Network (AOSN) experiment [24]. The virtual ocean contains temperature, salinity, and three-dimensional flow velocity at 500 meter horizontal resolution with 22 vertical levels over a 35 day period starting August 6, 2003.

The gliders were steered by the coordinated control (17). The gliders were separated into two groups, each group distinguished by its common effective glider speed. The first group contained the four Spray gliders and the second group contained the six Slocum gliders. The first group, containing the four Spray gliders, was controlled to a splay state formation on the northernmost track. Before adapting the GCTs, the Spray gliders were configured to perform (simulated) Iridium communication after every dive. The resulting frequent and lengthy surfacings, combined with surface currents of 0.15 to 0.35 m/s, resulted in a low effective speed of 0.2 m/s (median value for all four Spray gliders) and the glider coordinated performance was poor (Figure 7(a)). On the fourth day of the deployment, the Spray gliders were reconfigured to communicate on surfacing only if two or more hours had elapsed since the

last communication session. In this configuration, the glider effective speed increased by 25% to 0.25 m/s and the coordinated performance improved (Figure 7(b)).

The second group contained the six Slocum gliders, which were further divided into two subgroups of three gliders each. Each subgroup was controlled to a splay state formation and the two subgroups were synchronized with each other. Prior to adapting the GCTs, each subgroup of Slocum gliders was controlled to a splay state formation on a single track. All six gliders formed a (3,6)-pattern of curve-phases with three clusters of two gliders each, one from each subgroup (Figure 7(a)). After the fourth track was added, the southernmost subgroup of Slocum gliders was controlled to a splay state formation on two tracks and the group coordination was preserved (Figure 7(b)).

In the presence of flow, the GCCS planned to regulate the glider progress around the track to achieve the desired relative curve-phase separation between the gliders. For instance, gliders took inside (resp. outside) “lanes” on the superellipse to speed up (resp. slow down). Real-time performance metrics like the OA mapping error shown in Figure 7(b) are calculated automatically by the GCCS. By studying the 0.4 level curve in the mapping error, we observe the improvement in coverage effectiveness obtained by adding the fourth track despite keeping the number of gliders constant. For the spatial and temporal scales used to compute the mapping error, the 0.4 contour shows a gap in the sensor coverage in the center of the northern two tracks but not in the southern two tracks. Effective coverage is achieved through a combination of good track planning and good real-time coordination.

Figure 8 shows a snapshot of the GCCS planner status for the southern subgroup of Slocum gliders *we21*, *we22*, and *we23*. To illustrate the planning process, the sequence of dots on each glider trajectory can be thought of as the passenger cars in a commuter train where the locomotive of the train is the planned glider position in 12 hours (labeled with the glider name). Starting from the back of the train (the caboose) and following the directed edges towards to the front, we plot the glider surface positions over the last 12 hours, the waypoints expected to be reached on the current dive, the next expected surface position, and the sequence of new waypoints computed for the next 6 hours. The thin line starting from the next expected surface position is the planned (particle) trajectory. Since a curvature-dependent waypoint generator was used, the waypoints are spaced more closely around the corners of the track than they are along the sides. To assist in monitoring the GCCS in real-time, status images like Figure 8 are updated online after every planning cycle during a deployment.

B. Real Glider Deployment in Buzzards Bay, MA

The GCCS coordinated two Slocum gliders in Buzzards Bay, MA in March 2006; the principal investigator for the experiment was David Fratantoni of WHOI. Details of this deployment are contained in the WHOI data report [25] and details of the feedback control are described in [15]. The gliders were controlled to a single superelliptic track with a fixed, along-track separation distance in relatively strong, tidal flow conditions. The desired track and glider trajectories are shown in Figure 9 before and after being swept off course by a tidal flow that exceeded the glider effective speed of 0.2 m/s (averaged over both gliders over the whole deployment). The gliders returned to the track with the desired separation after this flow subsided. The glider depth-averaged flow measurements shown in Figure 9(b) were interpolated using OA assuming 2.5 kilometer spatial and 3 hour temporal decorrelation scales.

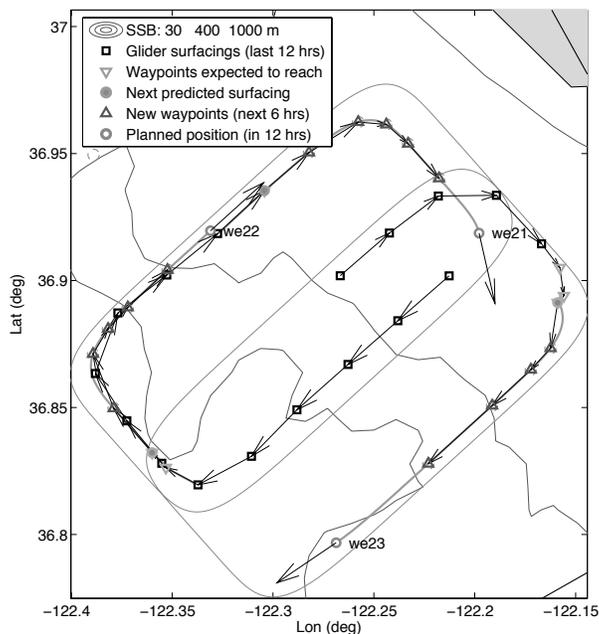


Fig. 8. GCCS planner status example. The sequence of dots on each glider trajectory can be thought of as the passenger cars in a commuter train where the locomotive of the train is the planned glider position in 12 hours (labeled with the glider name). The dotted circles around the new waypoint positions depict the size of the waypoint circle. The subsea bathymetry (SSB) contour lines correspond to 30, 400, and 1000 meters.

VII. CONCLUSION

Because of their high reliability and endurance, autonomous underwater gliders are an excellent platform for adaptive ocean sampling with multi-vehicle coordinated control. We present the GCCS, which is an automated software suite that implements real-time feedback control of a glider fleet to a set of coordinated trajectories. During each planning cycle, the glider motion is estimated using a detailed, three-dimensional glider model and the future trajectories are generated using a planar particle model. Collective motion of the particle model in symmetric formations appropriate for ocean sampling can be stabilized using gradient-like control laws. The implementation of multi-vehicle control laws in the GCCS was tested in a virtual deployment of ten gliders in Monterey Bay, CA and in a real deployment of two gliders in Buzzards Bay, CA. The Buzzards Bay deployment underscored the importance of our ongoing work on improved robustness of the coordinated control in adverse flow conditions. The GCCS will be used in upcoming glider deployments in Great South Channel, MA and Monterey Bay, CA.

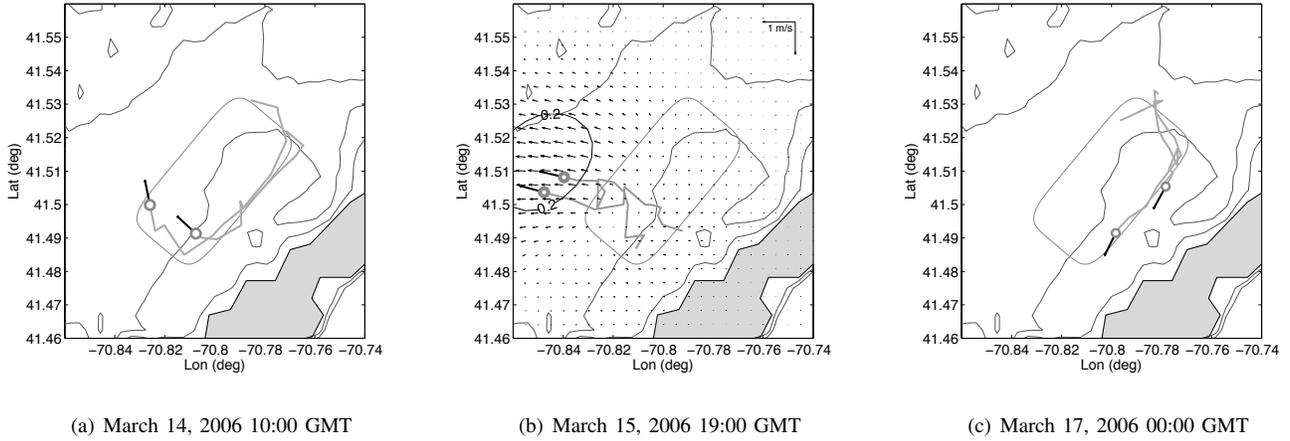


Fig. 9. GCCS planner real deployment results for two Slocum gliders from March, 2006 in Buzzards Bay, MA. The gliders were controlled to a 2.8 kilometer by 5.6 kilometer superelliptic track with a fixed, along-track separation distance of $1/6$ the curve perimeter. The bathymetry contour lines correspond to 10, 15, and 30 meters. 9(a): The gliders maintain the desired separation in the presence of benign flow conditions. 9(b): The gliders are swept off course by tidal flow that exceeded the glider effective speed. The interpolated flow velocity is depicted by small black arrows; the level curve corresponding to a flow magnitude of 0.2 m/s is shown. 9(c): The gliders return to the track with the desired separation after the strong tidal flow subsides.

APPENDIX A

SUPERELLIPSE PARAMETRIZATION

A parametrization of a superellipse and its derivative are [13]

$$\rho(\phi) = a (\cos \phi)^{\frac{1}{p}} \pm ib (\sin \phi)^{\frac{1}{p}} \quad (18)$$

$$\frac{d\rho}{d\phi} = -\frac{a}{p} (\cos \phi)^{\frac{1-p}{p}} \sin \phi \pm i \frac{b}{p} (\sin \phi)^{\frac{1-p}{p}} \cos \phi, \quad (19)$$

for $p = 1, 3, 5, \dots$ and $a, b > 0$. The plus (resp. minus) sign in (18) corresponds to rotation in the positive (resp. negative) sense; for clarity, we only give the positive case here. For $a > b$ (resp. $a = b$), $p = 1$ is an ellipse (resp. circle) and $p \geq 3$ is a rounded rectangle (resp. square).

Using (8), (10), (18), and (19), we obtain $\tan \theta = -\frac{b}{a} (\cot \phi)^{\frac{2p-1}{p}}$ and

$$\sigma(\phi) = \int_0^\phi \left[\frac{a^2}{p^2} (\cos^2 \bar{\phi})^{\frac{1-p}{p}} \sin^2 \bar{\phi} + \frac{b^2}{p^2} (\sin^2 \bar{\phi})^{\frac{1-p}{p}} \cos^2 \bar{\phi} \right]^{\frac{1}{2}} d\bar{\phi}.$$

Using the definitions of ρ_k , κ_k , and ψ_k yields

$$\rho_k = \frac{(a^{2p} \sin^2 \theta_k)^{\frac{1}{2p-1}} - i (b^{2p} \cos^2 \theta_k)^{\frac{1}{2p-1}}}{\left[(a^2 \sin^2 \theta_k)^{\frac{p}{2p-1}} + (b^2 \cos^2 \theta_k)^{\frac{p}{2p-1}} \right]^{\frac{1}{2p}}}, \quad (20)$$

$$\kappa_k = \frac{2p-1}{a} \cos^2 \theta_k \sin \theta_k \left[1 + \left(\frac{b^2}{a^2} \cot^2 \theta_k \right)^{\frac{p}{2p-1}} \right]^{\frac{1}{2p}} \times \left[1 + \left(\frac{a^2}{b^2} \tan^2 \theta_k \right)^{\frac{p}{2p-1}} \right], \quad (21)$$

and

$$\psi_k = \frac{2\pi}{\sigma(2\pi)} \int_0^{\phi(\theta_k)} \kappa^{-1}(\phi(\bar{\theta}_k)) d\bar{\theta}_k. \quad (22)$$

For $p = 1$, equations (20) and (21) give a parametrization and curvature of an ellipse,

$$\rho_k = \frac{a^2 \sin \theta_k - ib^2 \cos \theta_k}{\sqrt{a^2 \sin^2 \theta_k + b^2 \cos^2 \theta_k}} \quad (23)$$

and

$$\kappa_k = \frac{1}{a^2 b^2} (a^2 \sin^2 \theta_k + b^2 \cos^2 \theta_k)^{\frac{3}{2}}. \quad (24)$$

Setting $b = a$ in (24), one obtains the constant curvature of a circle with radius a , $\kappa_k = \frac{1}{a}$.

APPENDIX B

GEODETIC UTILITIES

Let λ and ϕ denote longitude and latitude, respectively. We use the following radii of curvature for approximating distance, azimuth, and velocity on the earth geoid: $\Gamma_\phi : [-\frac{\pi}{2}, \frac{\pi}{2}] \rightarrow \mathbb{R}^+$ and $\Gamma_\lambda : [-\frac{\pi}{2}, \frac{\pi}{2}] \rightarrow \mathbb{R}^+$ given by

$$\Gamma_\phi(\phi) = \frac{a(1-e^2)}{(1-e^2 \sin^2 \phi)^{\frac{3}{2}}} \quad (25)$$

and

$$\Gamma_\lambda(\phi) = \frac{a}{(1-e^2 \sin^2 \phi)^{\frac{1}{2}}} \quad (26)$$

where a and e are the semi-major axis and eccentricity of the earth geoid, respectively.

Using $R_k = (\lambda_k, \phi_k) \in [0, 2\pi) \times [-\frac{\pi}{2}, \frac{\pi}{2}] = S^2$, we define the distance function $\Gamma : S^2 \times S^2 \rightarrow \mathbb{R}^+$, where

$$\Gamma(R_1, R_2) = \left[\left[(\lambda_1 - \lambda_2) \Gamma_\lambda \left(\frac{\phi_1 + \phi_2}{2} \right) \cos \left(\frac{\phi_1 + \phi_2}{2} \right) \right]^2 + \left[(\phi_1 - \phi_2) \Gamma_\phi \left(\frac{\phi_1 + \phi_2}{2} \right) \right]^2 \right]^{\frac{1}{2}} \quad (27)$$

The onboard distance functions are glider-specific and not reproduced here.

The azimuth function $\eta : S^2 \times S^2 \rightarrow S^1$ is given by

$$\eta(R_1, R_2) = \tan^{-1} \left[\frac{(\lambda_2 - \lambda_1) \Gamma_\lambda \left(\frac{\phi_1 + \phi_2}{2} \right) \cos \left(\frac{\phi_1 + \phi_2}{2} \right)}{(\phi_2 - \phi_1) \Gamma_\phi \left(\frac{\phi_1 + \phi_2}{2} \right)} \right] \quad (28)$$

where we use the four quadrant inverse tangent and azimuth is measured clockwise from north. The onboard azimuth functions are glider-specific and not reproduced here.

ACKNOWLEDGMENT

The authors would like to thank Francois Lekien, Pradeep Bhatta, Dmitrijs Gurkins, and Jeff Pinner (Princeton); David Fratantoni and John Lund (WHOI); Russ Davis, Jeff Sherman, and Brent Jones (SIO); Pierre Lermusiaux, Wayne Leslie, and Pat Haley (Harvard); and the entire ASAP team.

REFERENCES

- [1] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis, "Collective motion, sensor networks and ocean sampling," *Proceedings of the IEEE*, 2006, special issue on "The Emerging Technology of Networked Control Systems," to appear.
- [2] 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*, 2006, to appear.
- [3] D. L. Rudnick, R. E. Davis, C. C. Eriksen, D. M. Fratantoni, and M. J. Perry, "Underwater gliders for ocean research," *Marine Technology Society Journal*, vol. 38, no. 1, pp. 48–59, 2004.
- [4] J. Sherman, R. E. Davis, W. B. Owens, and J. Valdes, "The autonomous underwater glider "Spray"," *IEEE J. Oceanic Engineering*, vol. 26, no. 4, pp. 437–446, 2001.
- [5] D. C. Webb, P. J. Simonetti, and C. P. Jones, "SLOCUM: An underwater glider propelled by environmental energy," *IEEE J. Oceanic Engineering*, vol. 26, no. 4, pp. 447–452, 2001.
- [6] D. L. Rudnick and R. E. Davis, "Mass and heat budgets on the northern California continental shelf," *J. of Geophysical Research*, vol. 93, no. C11, pp. 14,013–14,024, 1988.
- [7] I. Shulman, D. J. M. Jr., M. A. Moline, S. H. D. Haddock, J. C. Kindle, D. Nechaev, and M. W. Phelps, "Bioluminescence intensity modeling and sampling strategy optimization," *J. of Atmospheric and Oceanic Technology*, vol. 22, pp. 1267–1281, 2005.
- [8] T. B. Curtin, J. G. Bellingham, J. Catipovic, and D. Webb, "Autonomous oceanographic sampling networks," *Oceanography*, vol. 6, no. 3, pp. 86–94, 1993.
- [9] P. F. J. Lermusiaux, "Adaptive modeling, data assimilation and adaptive sampling," submitted to *J. of Inverse Problems*, special issue on Mathematical Issues and Challenges in Data Assimilation for Geophysical Systems: Interdisciplinary Perspectives.
- [10] L. S. Gandin, *Objective Analysis of Meteorological Fields*. Israel Program for Scientific Translations, 1965.
- [11] F. P. Bretherton, R. E. Davis, and C. B. Fandry, "A technique for objective analysis and design of oceanographic experiments applied to MODE-73," *Deep-Sea Research*, vol. 23, pp. 559–582, 1976.
- [12] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion: All-to-all communication," to appear in *IEEE Trans. Automatic Control*.
- [13] D. A. Paley, N. E. Leonard, and R. Sepulchre, "Collective motion of self-propelled particles: Stabilizing symmetric formations on closed curves," submitted to 45th IEEE Conf. Decision and Control.
- [14] F. Zhang and N. E. Leonard, "Coordinated patterns on smooth curves," in *Proc. IEEE Int. Conf. Networking, Sensing and Control*, April 2006, pp. 434–440.
- [15] F. Zhang, D. M. Fratantoni, D. A. Paley, N. E. Leonard, and J. M. Lund, "Control of coordinated patterns for ocean sampling," submitted to *Int. J. Control*, special issue on "Navigation, Guidance and Control of Uninhabited Underwater Vehicles".
- [16] C. Godsil and G. Royle, *Algebraic Graph Theory*, ser. Graduate Texts in Mathematics. Springer-Verlag, 2001, no. 207.
- [17] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.
- [18] E. W. Justh and P. S. Krishnaprasad, "A simple control law for UAV formation flying," Institute for Systems Research, Univ. Maryland, Tech. Rep. 2002-38, 2002.
- [19] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Group coordination and cooperative control of steered particles in the plane," to appear in *Proc. Workshop on Group Coordination and Cooperative Control*, Tromso, Norway, May 2006.
- [20] P. J. Davis, *Circulant Matrices*. John Wiley & Sons, Inc., 1979.
- [21] F. Lekien, D. Gurkins, N. Leonard, D. Paley, and F. Zhang, "Glider Coordinated Trajectories Markup Language user guide and reference manual," Dynamical Control Systems Laboratory, Dept. Mechanical and Aerospace Engineering, Princeton University.

- [22] "Adaptive Sampling and Prediction, MBARI website, <http://www.mbari.org/asap/>," 2006.
- [23] "Adaptive Sampling and Prediction, Harvard website, http://oceans.deas.harvard.edu/asap/index_asap.html," 2006.
- [24] "Autonomous Ocean Sampling Network, MBARI website, <http://www.mbari.org/aosn/>," 2003.
- [25] D. M. Fratantoni and J. M. Lund, "Glider operations in Buzzard's Bay, MA," Woods Hole Oceanographic Institution, Tech. Rep. BUZZ0306, 2006.