INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL. 13, NO. 1, MARCH 2008, 67-80

Language and Logic to Enable Collaborative Behavior among Multiple Autonomous Underwater Vehicles

T. BEAN, G. BEIDLER, J. CANNING, D. ODELL, R. WALL, M. O'ROURKE, M. ANDERSON and D. EDWARDS

Abstract—**Autonomous underwater vehicles (AUVs) have successfully performed complex behaviors in the service of military objectives. These behaviors have involved individual vehicles, but multiple vehicles operating autonomously and collaboratively are required to engage in open environment missions such as mine countermeasures (MCM). This requires that vehicles communicate with one another, exchanging information that can be processed logically to ensure flexible, coordinated behavioral responses. In this paper, field experiments involving a fleet of five communicating AUVs operating collaboratively underwater are described. These experiments focus on three behaviors: vehicle replacement, leader replacement, and fleet self-organization. These behaviors are not choreographed in advance or controlled by an operator as they unfold; rather, they arise when the vehicles identify the need to respond to changing circumstances and do so in a collaborative fashion. This identification and coordinated response is dependent on a language for communicating environmental information and logics that process this information. The language and logics used to support the field experiments are also presented.**

Index Terms—**Autonomous underwater vehicle, Communication language, Logic, Collaborative behavior, Self-organization**

1. INTRODUCTION

An autonomous underwater vehicle (AUV) is an important robotic technology with valuable commercial, scientific, and military applications. In the military arena, AUVs were used to search for mines in shallow water over an area of 3.5 million square meters during key humanitarian operations in Iraq [1]. Mine countermeasures (MCM) and rapid environmental assessment (REA) represent two areas of military interest where AUV potential has been assessed [2]. In these operations and assessments, AUVs were operated independently with individual supervision. In this mode of operation, AUVs cannot scan large areas in a timely or economic manner.

Timely and economic employment of AUVs is better served by having them work together in a group, mitigating the inefficiency arising from the supervision required for independent operation. The group of vehicles must (a) operate under constraints of underwater acoustic communication and navigation, (b) have the ability to respond flexibly to uncertain, heterogeneous environments, (c) ensure complete coverage of the search area, and (d) compensate for the unforeseen loss of AUVs during the mission. Given these constraints, a collaborative approach involving teams of AUVs recommends itself: flexible response is guaranteed by autonomy, and complete, timely coverage of large areas is made more likely by cooperation. Languages [3] and control architectures [4] have been developed to support cooperation among AUVs, and simulations [5] have been performed to examine its feasibility given communicative and navigational constraints. Field experiments with groups of AUVs communicating and navigating acoustically have not been reported in the literature.

At the University of Idaho (UI), languages, logics and algorithms have been developed to enable collaborative operations among AUVs. Automatic formation control algorithms enable multiple AUVs to search cooperatively for mines in close formation [6]. Organized in a swimmer-trailer formation and programmed to conduct coordinated searches in a lawnmower pattern with one swimmer designated the leader, various autonomous behaviors associated with large area MCM have been modeled and simulated [7]. These behaviors include deployment, vehicle replacement, leader replacement, divert to point of interest, and map building. The behaviors are supported by a version of AUVish, an agent communication language designed for the vehicles [8]. Once these behaviors were simulated, testing moved to the bench and the field with a fleet of five AUVs, seen in Fig.1, capable of underwater communication and navigation. Underwater, acoustic communication facilitates collaboration among the vehicles, making it possible for them to ensure complete coverage should a vehicle be lost by reconfiguring the formation,

Fig. 1. Five UI mini AUVs being prepared for field testing

*Center for Intelligent Systems Research, University of Idaho, Moscow Idaho, 83844-0902
+ CDNSWC Acoustic Research Detachment, 33890 North Main Street, Bayview, ID 83803
† Corresponding author

and to prevent loss of data should a vehicle fail to return by sharing information, e.g., the locations of mine-like objects (MLOs). At the time of writing, formation control, deployment, vehicle replacement, leader replacement, and divert to point of interest behaviors have been field tested in an unstructured, underwater environment.

In this paper, experiments characterizing two of the collaborative behaviors, vehicle and leader replacement, are described. These were fully collaborative, autonomous behaviors that unfolded organically in the course of missions, supported by underwater acoustic signals for navigation and intervehicle communication without prior choreography or operator intervention. After detailing the language and logic underwriting these autonomous behaviors and the experimental design, results from field tests are presented and discussed. The field experiments indicate that this approach, with its language and logic, appears to be fully generalizable across a wide variety of MCM-related AUV behaviors.

2. Language and Logics Used for AUV Collaboration

Autonomous, collaborative behaviors are performed by a group of AUVs arranged in a formation that adapts to changes in the environment. This adaptation is made possible by flexible and coordinated decision making that is responsive to unpredictable changes in underwater circumstance. The leader vehicle in the formation is the primary decision maker, with follower vehicles carrying out directions that reflect a balance between pursuit of mission goals and responsiveness to current circumstances. This hierarchical control structure requires a language to serve as a medium of communicative exchange among vehicles, and on-board logics that can generate behavioral outputs given linguistic and sensory inputs under mission goal constraints. In this section, we begin by describing the basic formation geometry and foundational behavior to fix the context within which communication and decision making take place. We then describe the language and the logics utilized by the Idaho AUV fleet.

2.1 Formation-Flying Behavior

All cooperative behaviors were performed while the vehicles maintained a geometric formation composed of primary positions called swimmers (one of which is the leader) and secondary positions called trailers. The foundational behavior performed by these vehicles is formation flying, which would enable suitably equipped vehicles to sweep an area for mines. The purpose of vehicles occupying the swimmer positions is to provide robust coverage of an area by the formation in spite of the loss of vehicles from unknown causes. The purpose of vehicles occupying the trailer positions is to replace incapacitated swimmers and to perform other behaviors such as inspection of a point of interest found by the swimmers. While part of the sweep formation, trailer vehicles provide redundant coverage of the sweep area. A vehicle replacement behavior prevents multiple vehicles from occupying the same position, replaces missing swimmer vehicles with trailer vehicles if they are available, and fills the most significant formation positions with the available swimmers if no trailers are available.

Formation-flying was implemented using a leader-follower regulation algorithm [9-11] in each vehicle. In the algorithm, a single waypoint pattern in inertial coordinates is defined for a formation. Initially, each vehicle is assigned a position in the formation defined by the desired offset from the waypoint pattern, δ_i, which yields a set of virtual waypoints for the vehicle to follow as shown in Fig. 2. In our field tests, we use Long Base Line (LBL) location estimation under water (see section III.B) and GPS location estimation on the surface. One vehicle assumes the mantle of leader which by default is assigned to the Swimmer 1 position as shown in the figure. Subsequently, inputs to the formation regulation algorithm are the locations of the vehicle and the leader, P_i, with the latter transmitted by an acoustic modem. Given this information, each vehicle is regulated to its position in the formation by comparing the vehicles' locations to the appropriate follow distance, d_i. An advantage of this approach is that if a vehicle receives a new position assignment, the regulator algorithm will automatically direct the vehicle to its new formation position and waypoint path. Use of a regulator algorithm allows initiation and control of cooperative behaviors via exchange of short underwater messages.

As the formation proceeds along the waypoint path, changes occur that prompt the initiation of certain collaborative behaviors under the direction of the leader vehicle. For example, a vehicle might be lost, in which case the requirement of complete coverage will prompt the leader to replace that vehicle and adjust the search pattern so that no area is left unexamined.

2.2 Control Language for Cooperative Behaviors

Design of the communication language has been guided

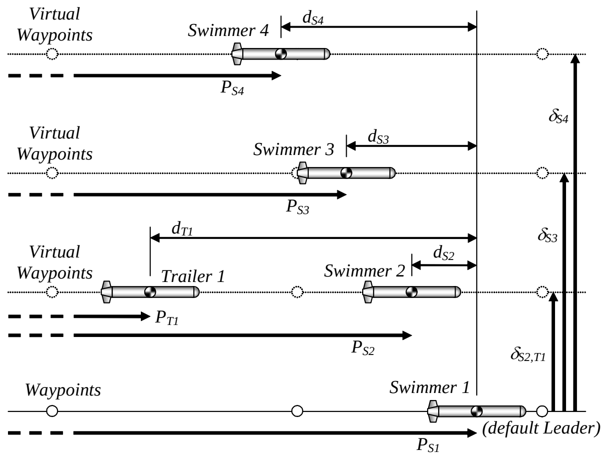

Fig. 2. Important vectors and distances for formation flying

by work done on AUVish [8], an agent communication language designed for use in AUV simulations (e.g., [7]). AUVish was modeled on several AUV languages designed for AUV-operator and AUV-AUV communication. The Marine Systems Engineering Laboratory (MSEL) was one of the first groups to design such a language, designing Conceptual Language for AUVs (COLA) for the Experimental Autonomous Vehicle (EAVE) system. COLA uses natural language concepts, such as speech acts, to deal with the limitations of underwater communication [12]. The Ocean Sampling Mobile Network (SAMON) project developed several different versions of the Generic Behavior Message Passing Language (GBML), which supplied a framework for a common language allowing cooperation of heterogeneous AUVs [13]. Versions include CCL, a tactical command and control language for multi-vehicle, semi-autonomous data collection systems [14], and Robotalk, a dynamically organized language in which the communication structure changes in response to environmental stimuli [15]. Other languages include the Compact Control Language, a low bandwidth acoustic communication protocol designed to allow multiple AUVs to communicate with a central point and each other with [16], and the Common Control Language of the Solar AUV project, which allows AUVs and the operator to communicate a common set of commands as well as information for monitoring and control [17]. Drawing influence primarily from COLA and CCL, AUVish has been developed in an *ad hoc* fashion, combining elements from these languages into a medium that supports successful achievement of mission goals.

In the transition to bench and field tests, modifications have been made to the AUVish employed in the simulations reported in previous work [7, 8], due primarily to hardware constraints that have forced the employment of significantly smaller communication packets. The result of these modifications is *AUVish-BBM*, a dialect of AUVish designed to work on the Woods Hole Oceanographic Institute (WHOI) modem. (The ‘BBM’ suffix includes the initials of the researchers directly involved in dialect development.) This modem transmits messages in varying packet sizes, including 13-bit and 32-byte packets. The changes that mark AUVish-BBM as a dialect were made to fit messages into the 13-bit packet, although the language also supports the 32-byte message. Concern about communication reliability underwater would have forced additional modification that introduced error checking, but the WHOI modem employs a cyclic redundancy check (CRC) for error detection in modem to modem underwater communication; any data packets which fail the CRC are not processed. Additional error checking at the language level is possible and is being considered in connection with new work on the formal semantics of AUVish.

The AUVish-BBM message structure differs for the 13-bit message and the 32-byte message, given the different purposes they serve. The 13-bit messages transmit the current status of the vehicles in the fleet. By default, each vehicle has a 13-bit window in the timing cycle to declare its status. All vehicles have access to a 32-byte message which can be used to make requests (Requests) or convey information (Informs). The leader uses a 32-byte message during each cycle, but must give permission to a follower vehicle to transmit its 32-byte message. This permission will be given in a 32-byte segment, with the follower vehicle sending a 32-byte message in the next cycle. Most Requests from the leader, as well as information from all vehicles (e.g., search area map information, etc.), are to be delivered in messages of this type. After describing the communication cycle, we supply syntactic and semantic specifications for the 13-bit and 32-byte messages below.

2.2.1 Acoustic Communication and Navigation Cycle

Communication among the vehicles is conducted with the 13-bit and 32-byte messages in a predetermined cycle. The communication cycle for five vehicles and an operator (i.e. User) is divided into eight communication windows as shown in Table 1. This TDMA (time division multiple access) scheme is use to avoid communications collisions that can occur when two or more nodes attempt to transmit in a shared communications medium, such as wire, RF, and acoustics. For systems that involve periodic communication, allocating bandwidth based on time slices has distinct advantages. Since each node takes its turn, as long as all nodes are synchronized, the probability of communications collisions are eliminated except for equipment malfunctions. Another alternative for these systems would be a CSMA (carrier sense–multiple access) protocol; however, the communications from one or more

Table 1: Communication Cycle

Cycle Time	Veh. No.	Content	Dur.
0	Veh1	LBL Navigation Ping	2 sec.
2		Status Mini-Packet	1 sec.
3	Veh2	LBL Navigation Ping	2 sec.
5		Status Mini-Packet	1 sec.
6	Veh3	LBL Navigation Ping	2 sec.
8		Status Mini-Packet	1 sec.
9	Veh4	LBL Navigation Ping	2 sec.
11		Status Mini-Packet	1 sec.
12	Veh5	LBL Navigation Ping	2 sec.
14		Status Mini-Packet	1 sec.
15	Veh6/ User	LBL Navigation Ping	2 sec.
17		Status Mini-Packet	1 sec.
18	Lead	Leader Position/Progress Maintenance Commands	6 sec.
24	Asgn.	Requested Information	6 sec.
		Total Time	30 sec.

nodes (and in some cases all nodes) within this protocol is always delayed, and this would be undesirable in a MCM mission where quick response might be required.

The eight window communication cycle is 30 seconds in length, as indicated in Table 1. The first 18 seconds of the broadcast cycle are divided into six windows for vehicle positioning and status reporting. Each vehicle has two seconds to perform an underwater acoustic trilateration navigation ping, then another second to send a status message to the fleet using a 13-bit message. The first five of these windows are used by the vehicles, while the sixth is available for either a future sixth vehicle or, optionally, an operator broadcast to the vehicles. The remaining 12 seconds are reserved for broadcasts using the normal, 32-byte WHOI modem messages. The first six-second window is assigned to the leader for fleet maintenance, while the second is designed to be used by any one of the vehicles as directed by the leader. The 30-second timing cycle is synchronized to seconds from midnight on GPS time, which is acquired while the vehicles are on the surface prior to starting the mission.

2.2.2 Syntax: 13-bit Message

The 13-bit messages in AUVish-BBM can be used to convey basic information about vehicle status and mission information, as well as a limited number of requests. There are six fields, but the first two of these do not count as part of the 13-bit data payload; instead, they are part of the packet header automatically included by the WHOI modem. The syntax is given as follows, displayed in Table 2.

1. From Sub #: The serial number of the vehicle sending the message.
2. To Sub #: The serial number of the intended recipient of the message. ‘0’ indicates a broadcast message to all vehicles.
3. Formation Position: (3 bits) Current position in the formation, 0 to 7.
4. Leader Indicator: (1 bit) Set to 1 if vehicle is the leader.
5. Connection Vector: (4 bits) This four-bit word expresses the sending vehicle’s communicative connectivity with other members of the fleet as of the previous communication cycle. Each bit set to 1 indicates that another vehicle has communicated with the sending vehicle in the previous cycle. When the sending vehicle has heard from all of the vehicles in the fleet, the connection vector would appear as in the first example from Table 2. When a vehicle is not heard, a 0 is introduced into the vector as shown in the remaining examples. The vehicles do not report on their own status; therefore, two different vehicles may report the same vector but their meanings will differ, as illustrated in the second and third examples from Table 2. By combining the received connection vectors with an OR operation, the connection vector becomes a report on the vehicles considered to still be a part of the formation. A vehicle is considered missing if the leader, and all the other AUVs communicating with the leader show that the vehicle is missing. The connection vector allows the formation leader to sense the presence of a vehicle even if the leader does not receive a direct communication from the missing vehicle. This information is a critical part of the vehicle replacement and leader replacement behaviors described below. (For more information about this vector, see [18].)
6. Task: (5 bits) This indicates what the vehicle is doing. There are 32 possible binary sequences associated with this field, and each sequence is used as a name of a behavior or a behavior-related piece of information to be conveyed to the formation. The categories of information include *Inactive* (Completed, Waiting, Forming, Aborted), *Formation Status* (Behind, In, Ahead, Unknown), *Point Inspection* (In Transit, Dangerous, Not Dangerous, Unknown), *Requesting 32-byte Message* (Number of Requests: 1-4), and *Leader Change* (Request, Accept, Deny, Report Multiple Leaders).

2.2.3 Syntax: 32-byte Message

The 32-byte message in AUVish-BBM is used for several purposes that cannot be accommodated in the 13-bit messages. The leader uses a 32-byte message once per communication cycle to inform the other vehicles of its position and to issue commands to the other vehicles. The leader’s position information is used by follower vehicles for formation flying, as will be illustrated. Follower

Table 2: 13-bit AUVish-BBM message structure

Message Part	**Message Header**		**Message Payload**			
Slot #	1	2	3	4	5	6
Contents	From Sub#	To Sub#	Formation Position *3 bits*	Leader Indicator *1 bit*	Connection Vector *4 bits*	Task *5 bits*
Examples	2	0	Swimmer 1	0	1111 (Heard: 1 3 4 5)	Formation Status: In
	3	0	Swimmer 2	0	1011 (Heard: 1 4 5)	Point Inspection: In Transit
	1	0	Swimmer 3	1	1011 (Heard: 2 4 5)	Formation Status: In

vehicles can use the 32-byte message to inform the leader of, for example, the location of detected mine-like objects.

As before, the 32-byte messages are divided into several fields. The “To” and “From” fields are automatically included by the modem in the message header, but each 32-byte message is a broadcast message and so the “To” field in the header is set to 0. The 32-byte message payload also contains a field that denotes intended recipients. This is included so that a single 32-byte broadcast can convey several Requests or Informs, each for a different vehicle. These contents would involve the repetition of fields 4 and 5, and possibly 6, for each distinct Request or Inform in a given payload. The syntax is described below and in Table 3.

1. From Sub #: The serial number of the vehicle sending the message.
2. To Sub #: The serial number of the intended recipient of the message. This is always set to 0, or “broadcast”.
3. Leader Progress: The total progress, in meters, made by the leader since the beginning of the mission. The other vehicles compare their own progress with that of the leader’s to determine if they need to speed up or slow down. This information is only included when the leader broadcasts a 32-byte message.
4. To Sub#: The serial number of the intended recipient of the message. This information differs from field 2 in that it introduces fields 5 and 6, making them available to a specific vehicle.
5. Content: The content being sent. This can be a request for information or action (Request), or a piece of information (Inform). As a unit with fields 4 and 6, this field can be repeated in a single 32-byte message so that the payload can carry more than one Request or Inform.
6. Arguments: Additional information required by some messages. For example, if a vehicle is being directed to assume a different position, this tells the target vehicle which position it should assume.

2.2.4 Semantics: 13-bit and 32-byte Messages

Determination of the syntax of the 13-bit message has gone hand in hand with identification of the semantic content that message will carry. The semantic content of the 32-byte AUVish-BBM messages are Requests and Informs. A summary of the message content for the leader and follower messages is given below, divided between 13-bit and 32-byte messages. (Message content types not yet field tested are in brackets.)

From Leader

- *Status*: 13-bit
- *Requests*: 32-byte
  - Leader’s progress is *X*
  - Sub #*X* needs to abort
  - Sub #*X* needs to change its position to *Y*
  - Sub #*X* needs to inspect MLO located at map coordinates *Y*, *Z*
- *Informs*: 32-byte
  - [Sub #*X* has found a MLO at map coordinate *Y*, *Z*]
  - [Sub #*X* can use 32-byte message in next cycle]

From Follower

- *Status: 13-bit*
  - Sub #*X* acknowledges leader’s request (implicitly encoded)
  - Sub #*X* is currently performing task *Y*
  - Sub #*X* is currently acting in formation position *Y*
  - Sub #*X* has connection vector *Y* (has heard from subs *wxyz*)
- *Requests: 13-bit*
  - [Sub #*X* requests to use 32-byte message]
- *Informs: 32-byte*
  - [Sub #*X* has found a MLO at map coordinate *Y*, *Z*]

2.3. Logics for Information Processing and Communication

We take a logic to be a set of constraints that ensure positively valued information flow from system inputs to system outputs. What counts as “positively valued” will vary depending on the nature of the system. So understood, there are logics at two levels that bear on AUV collaboration. At the higher level, there are the logics associated with the specific collaborative behaviors; at the lower level, there are logics associated with the management of information conveyed in the AUVish-BBM messages.

Logics for collaborative behaviors concern the coordinated modification of formation geometry necessary for successful performance of specific behaviors. For autonomous vehicles in an unstructured environment, the geometry of the formation at the moment when a behavior is initiated will vary from circumstance to circumstance. As such, the logic for that behavior will need to be responsive to a wide variety of opening orientations. With one of these orientations as inputs, the logic will specify the coordinated modifications to formation geometry required to accomplish the specific behaviors [19].

Higher-level logics for collaborative behaviors directly

Table 3: 32-byte AUVish-BBM message structure

Message Part	Message Header		Message Payload			
Slot #	1	2	3	4	5	6
Contents	From Sub #	To Sub #	Leader progress	To Sub #	Content	Arguments
Bytes (max)			4	1	1	8

address the formation, and only indirectly address each vehicle insofar as it is part of the formation; by contrast, lower-level logics associated with information management directly concern what transpires within each individual vehicle. Each vehicle sends and receives messages that communicate mission-relevant information. This information is processed and can affect vehicle behavior by prompting modification of navigational trajectories or transmission of new information via communication. The AUVs embed logics that govern the transmission and interpretation of these messages. Of special importance are the 13-bit messages, which are central to the implementation of the collaborative behaviors reported below. Each 13-bit message conveys three pieces of information about a vehicle: its task, its role, and a connection vector. Thus, the internal communication logics must ensure that this information is reflected in the 13-bit message conveyed each communication cycle.

In addition to sending messages, communication requires semantic uptake on the side of the receiver. Receiving a command changes the state of the vehicle, which is reflected in the vehicle's behavior and 13-bit status message. Each vehicle derives two status vectors from the exchanged 13-bit messages. The first status vector, called the *composite connection vector*, summarizes the vehicles known to be present in the fleet by comparing the connection vectors of the vehicles from which it has heard. The second status vector, called the *formation vector*, describes the status of the formation, including holes and duplicates. From the current formation geometry and the status of each vehicle as deduced from the status vectors, the leader composes a message that instructs certain vehicles to perform tasks, e.g., take a certain position in the formation. When the leader detects a change in state in one of these vehicles, it knows that the message has gotten through.

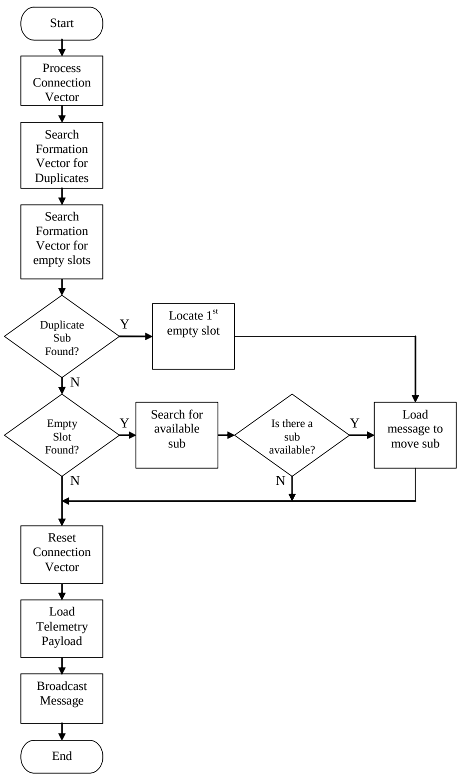

Fig. 3. Vehicle replacement logic

The logics presented below reveal how information is managed in ways that result in vehicle replacement and leader replacement. These have been designed to ensure that the proper changes in formation geometry are effected by appropriate and corresponding changes in message content, and so are lower level logics than those which operate at the level of formation geometry. While these control and communication logics have been formed and coded with the specific behavioral patterns in mind, they can generalize to support additional behaviors, such as waypoint divert [18].

2.3.1 Vehicle Replacement Logic

The goal for the vehicle replacement behavior is to ensure that swimmer locations are occupied in the formation while performing a MCM sweep. Vehicle replacement is controlled by the leader with a 32-byte message broadcast. A flow chart describing the logic used by the leader vehicle to compose the 32-byte message for vehicle replacement is shown in Fig. 3. The first step is to process the connection vectors and compute the composite connection vector. In this step, the accumulated 13-bit messages received from all of the vehicles are combined with an OR logic operation. This operation determines if any vehicles are missing from the formation. The next step is to search the formation vector for duplicates to determine if multiple AUVs report the same formation position. The formation vector is then searched to locate empty slots in the formation. The first decision branch is found at this point: "is there a duplicate vehicle?" If there is a duplicate vehicle, a 32-byte message is composed to tell the duplicate vehicle to move to the first empty slot in the formation. The second decision branch is this: "is there an empty slot?" If there is an empty slot, the logic will attempt to find an available vehicle, swimmer or trailer, to fill the slot. Any vehicle assigned to a formation position of less significance is considered available for movement. For example it is more important to fill the Swimmer 2 position than to keep a vehicle in the Swimmer 3 or Trailer positions. Once a vehicle-move message is composed, it is loaded into the 32-byte package along with the leader telemetry information and broadcasted to the formation.

2.3.2 Leader Replacement Logic

In the UI approach, the leader is not a specific position in the formation; rather, *leader* is a role that can be assumed by any one of several vehicles, so long as it meets certain conditions. The goal for leader replacement is to ensure that the formation has one and only one leader at all times. All the information required to accomplish this goal is available in the connection and formation vectors. A

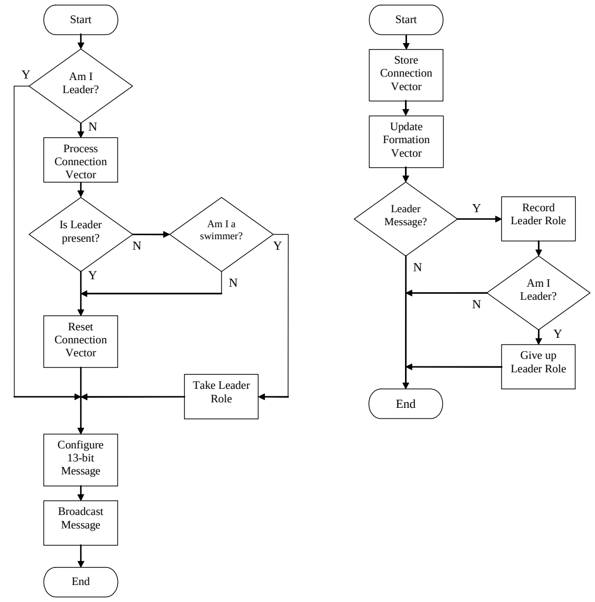

Fig. 4*:* Leader replacement logic, 13-bit broadcast (left), 13-bit receive (right)

flow chart of the leader replacement logic used to compose the 13-bit messages for broadcast is shown in Fig. 4. A vehicle must meet three conditions before assuming leadership: first, it must lose contact with the previous leader; second, all other vehicles must indicate that they have lost contact with the previous leader via their connection vectors; third, the vehicle must be in a swimmer position. After a vehicle assumes leadership it will command trailers to replace any vacant swimmer positions in subsequent 32-byte messages. The potential problem of multiple leaders is resolved by allowing the first vehicle that claims leadership to be the leader. In the case of multiple leaders, there will be multiple, overlapping 32-byte command messages which will be garbled, so all leader conflicts must be resolved with the 13-bit messages.

The leader replacement logic used to process a 13-bit message upon reception is shown in Fig. 4. In each communication cycle, each vehicle will broadcast one 13-bit message and receive up to four 13-bit messages from other vehicles in the formation. A vehicle claims leadership by setting the leader flag in its 13-bit status message to 1. As the leader, it will broadcast a 32-byte command message.

3. AUVs, Communication Hardware, Test Range, and Deployment

3.1. Autonomous Underwater Vehicles

The UI AUV fleet consists of five miniature submarines built at the UI based on a design by Stillwell et al. [20] of Virginia Tech. Each AUV is 1 meter in length and 10 cm in diameter. These vehicles have a fixed displacement and are trimmed to 2% positive buoyancy. Dynamic diving is required to submerge. Once submerged, these AUVs can travel at a maximum speed of 1.3 m/s and must maintain a minimum speed of 0.6 m/s in order to maintain depth. Vehicle speed is estimated from propeller rpm. The typical formation speed is 1 m/s (1000 rpm) for the leader, with the other AUVs adjusting their velocity to maintain their position in the formation as previously described. The two ventral fins at the rear provide pitch control. Heading control is provided by the rudder fin, and the ventral fins provide a counter torque to minimize rolling which results in flatter and faster turns. Each AUV has a 25 volt battery pack consisting of two parallel banks of seven lithium polymer battery cell stacks. The battery pack provides approximately three hours of operation under typical testing conditions.

The electronic control system on the AUV uses a distributed processing approach. A diagram of the system architecture is shown in Fig. 5. Four custom designed boards with an 8-bit model 3000 Rabbit Core Module microprocessor or a 3360 Rabbit Core Module combined with a Rabbit 3000 microprocessor are used for each functional unit. The individual boards are connected using a four-port Ethernet hub.

The instrumentation unit collects and distributes information from all of the sensors on the AUV. Internal sensors include a battery monitor, a water detector, and a thermometer; external sensors include a GPS unit, an electronic compass, a pressure sensor to determine the depth, and an accelerometer for pitch and roll. All sensors are polled at 4 Hz, with the exception of the GPS unit which was polled at 1 Hz when on the surface.

The mission control unit performs the control calculations necessary for navigation of the AUV. The unit records all incoming and outgoing Ethernet packets. A data storage capacity of 30 hours is provided by a 128Mb XDRAM card. Each 4 Hz sensor packet is used to generate a control packet which is sent to the locomotion unit. The locomotion unit controls the speed of the propulsion motor and sets the position of the three control fins.

The communications unit provides for all the external communications. When located on a dock, a wireless 802.11b Ethernet card connection is used to adjust various settings on the AUV via a webpage interface. The webpage

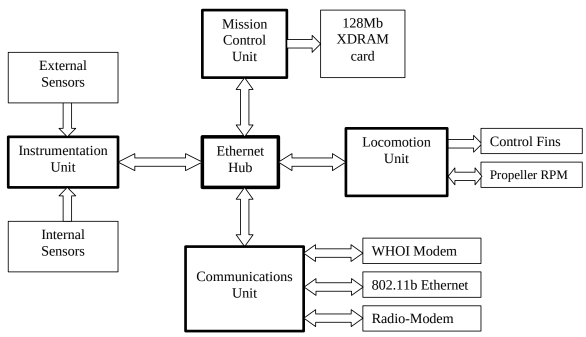

Fig.5. System architecture for AUVs

allows adjustment of the fin angles, fin mixing ratio, mission logs, formation position, AUV identification, and acoustic modem settings. When the AUV is mobile and on the water surface, a 900MHz radio modem provides communication to a base station. The base station is used to select pre-programmed missions and to provide manual control of the AUV using a joystick. The base station is configured to handle control and telemetry data from up to eight AUVs simultaneously. The communications unit also connects to the onboard acoustic modem for underwater navigation and communication.

3.2. Underwater Acoustic Navigation and Communication

A WHOI acoustic modem was used for underwater acoustic navigation and communication [21]. Underwater position is determined by acoustic Long Base Line (LBL) measurements using the WHOI acoustic modem and fixed transponder buoys. During the period between LBL measurements, the location was estimated by a dead-reckoning procedure. The dead-reckoning procedure entailed integration of vehicle heading and speed to determine location relative to an initial condition. Speed was estimated from propeller RPM measurements. The initial condition used for dead reckoning integration was the last valid position LBL location measurement. There was no independent measure of location, other than the periodic LBL measurements, to corroborate the location reported by the dead-reckoning estimate. The criterion used to judge validity of LBL location measurements was to compare the LBL and dead reckoned location estimate. If the difference between the LBL and dead-reckoned location exceeded a threshold, the LBL measurement was classified as invalid and the initial condition used for the dead-reckoning procedure was not reset. Significant error could accumulate in the dead-reckoned location if a series of LBL measurements were classified as invalid. Examples of this circumstance are often seen in the plots of the dead-reckoned location when the initial conditions were reset to a new valid LBL location measurement. (See Figs. 7-9.)

3.3. Test Range

Bayview, Idaho is the home of the Office of Naval Research Acoustic Research Detachment (ARD), a low noise, freshwater research facility. This test range was used for the AUV collaborative behavior experiments and is located at the southern end of Lake Pend Oreille. The facility is equipped with a state-of-the-art acoustic tracking system in deep water, a shallow water location with portable acoustic navigation beacons, underwater acoustic communication systems, and surface craft to support tests with miniature submarines. Field testing is conducted in a shallow-water portion of the test facility with the depth of the water ranging from ≈9-18m.

Figure 6 is a representation of the courses that the vehicles were directed to follow. The course is approximately 200 meters long, with two 85 m sweep legs and a 30 m base leg. The courses are rotated to a 45 degree angle to efficiently utilize the available space. This would represent the first leg loop of a lawnmower sweep pattern. The base course is identified by the bold white line with the waypoints marked by larger circles. The Swimmer 1 (S1) formation position has an offset of zero so this is the course it will follow. The two other primary positions, Swimmer 2 (S2) and Swimmer 3 (S3), follow imaginary waypoint courses that are offset to the northeast by 5 m and 10 m, respectively, as indicated in the figure. The edge of the protected waters available at the Bayview, Idaho ARD facility is marked in Fig. 6 with a dotted line and stars. Each star marks the location of a large steel piling that marks the edge of the base. To the northeast of these pilings are public waters that have a significant amount of recreational boat traffic during the summer months. The base station and researchers are located on the covered dock labeled command station. This test area provides a small, protected environment to perform testing.

3.4. Deployment

When a field testing begins, the AUVs are powered up and pass an initial system check. Each sub is then trimmed to run flat and straight. This is accomplished by setting all the fins to zero offset and the propeller to 1000 rpm. Each AUV is then released underwater to see if it is climbing/diving or turning left/right. The fin's trim settings are adjusted via the web page hosted on the AUV and the test is repeated until the AUV is running relatively straight and true.

Once the AUVs are ready, they are maneuvered on the

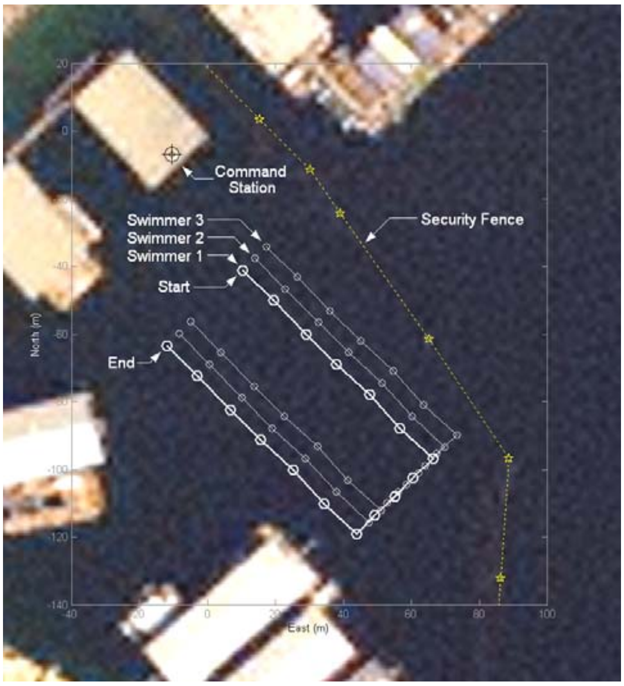

Fig. 6. Courses shown in the protected area inside of the naval base at the ARD in Bayview, Idaho.

surface of the water from the base station using radio control to a position where they can start their missions. The primary consideration is whether the AUVs have sufficient clearance to begin the mission without colliding with an obstruction such as the dock or with each other. Once in position, a radio frequency (RF) broadcast from the base station commands the AUV formation to begin one of the preprogrammed missions. The AUVs then submerge and the deployment procedure is performed, with the vehicles forming up in the offset pattern required for formation flying. The deployment procedure is entirely autonomous, relying on underwater acoustic communication among the vehicles without any intervention from an external operator. The first underwater message for each vehicle is initialized with the connection vector indicating that all vehicles are present, thereby insuring that the leader would not commence replacing vehicles simply because others had not yet had the opportunity to report their presence.

Tests are performed both on the surface and while submerged with all formation communication taking place underwater through the WHOI modem. During the surface tests, the AUVs can navigate with either GPS or LBL navigation. Surface testing also has the advantage that the formation is visible to the researchers and problems can be quickly detected. Underwater testing can only operate using LBL and dead-reckoning for navigation. Although some information can be gleaned from the time and location of the AUV surfacing at the end of the mission, in most instances the data from the test must be downloaded from all the AUVs in the formation and then processed to determine if the formation behavior is correct.

4. Field Results: Vehicle and Leader Replacement

Three experiments are described in this section that demonstrate the vehicle replacement and leader replacement behaviors. Each experiment focuses on three of the five UI AUVs, as indicated by their vehicle identification number (e.g., V1). In the first experiment, we force a missing vehicle situation to test the vehicle replacement behavior. The second experiment forces the leader to abort in order to demonstrate the leader replacement behavior. The third experiment is a demonstration of a behavior that emerges from the two previous behaviors: the ability for the fleet to self-organize. Each experiment is accompanied by a communication table and a figure that shows the vehicle tracks.

In each communication table, the first column is the cycle number, the second column is the 32-byte command, and the remaining columns are for the 13-bit status messages. In the 32-byte commands, only the information that is pertinent to the experiment is shown. If no command is present, it will look like this: *V1:* —. A vehicle replacement message will appear this way: *V1: V5 goto S2,* which can be translated as, "Vehicle 1 (*V1:*) requests vehicle 5 (*V5*) to assume (*goto*) the swimmer 2 position (*S2*)." A 13-bit message appears in this form: *L,S1,135,Form*. In the connection vector portion of the 13-bit message, a grey number denotes the transmitting vehicle. A vehicle that failed to communicate is marked with: (-) .Therefore, the above message would be translated into English as "Vehicle 1(denoted by column and grey number) is the leader (*L*), is in the swimmer 1 role (*S1*), has heard from vehicles 3 and 5 (*135*), and is currently swimming in formation (*Form*)." Where vehicles are present that do not contribute to a behavior, their communications have been omitted from the table. The first 13-bit status broadcast from each vehicle automatically reports a full connection vector, even though this cannot be true. This is necessary to avoid false replacements during the initialization of the formation.

The plots contain the vehicle tracks, waypoint paths, and positioning data. The vehicle tracks are denoted by a solid line. The waypoints that the vehicles followed are marked by circles connected with a thin line. These experiments were conducted underwater using LBL navigation techniques. The solutions of the LBL system are marked on the plot with squares. The error in the LBL navigation system yielded discontinuities in the vehicle tracks. Due to the constricted course and tight formation, the error in the LBL solutions may exceed the distance between the vehicles and cause the vehicle tracks to cross. To clarify the plots, the tracks of those vehicles that were not involved with the described behavior have been muted. This allows the reader to see more clearly the formation changes that occur during the behavior.

4.1 Vehicle Replacement Results

4.1.1 Experiment Conditions

The vehicle replacement behavior was tested by forcing vehicles in the swimmer positions to drop out of the formation. The vehicles were initialized separately such that the fleet had three swimmers and two trailers. In the test, three vehicles were instructed to perform the formation flying behavior around the entire course. Two of the swimmer vehicles were given short courses to follow so that they would disappear from the fleet, simulating lost vehicles. The communication results are tabulated in Table 4 and the tracks of the vehicles can be seen in Fig. 7. The discussion will concentrate on the communications and vehicle tracks of vehicles 1, 3, and 5.

4.1.2 Communications Results

The details of the intervehicle communication for the vehicle replacement experiment are found in Table 4. As the vehicles were initialized, vehicle 2 was assigned the swimmer 1 position, which includes the leader assignment by default. For the first three cycles, all of the vehicles were performing their tasks and were able to communicate with each other. It can be seen that among the three vehicles discussed, there was perfect communication as indicated by the connection vectors during the first three

Table 4: Communication details from vehicle replacement experiment

Cycle	32-byte - Cmd	V1-13-bit	V3 -13-bit	V5 -13-bit
1				F2,135,Form
2	V1: —	L,S1*,135,Form	S2,135,Form	F2,135,Form
3	V1: —	L,S1,135,Form	S2,135,Form	F2,135,Form
4	V1: —	L,S1,135,Form	-missing-	F2,1-5,Form
5	V1: V5 goto S2	L,S1,1-5,Form	-missing-	S2,1-5,Form
6	V1: —	L,S1,1-5,Form	-missing-	S2,1-5,Form
…	…	…	…	…

*S1 is the default leader.

cycles. During cycle 4, however, vehicle 3 completed its short course and dropped out of the formation. All of the connection vectors that occurred after vehicle 3 terminated reported that vehicle 3 was not present. When the leader composed the 32-byte command message for cycle 5, it reviewed all of the connection vectors that it received in cycle 4 and discovered that vehicle 3 was no longer present. Next the leader examined the reported positions of the vehicles and determined that vehicle 3 left an open space in a swimmer position. At this point, if more than one vehicle was assigned to another swimmer position (i.e., a duplicate vehicle), the leader would assign one of these vehicles to the position vacated by vehicle 3; however, the leader found no duplicate vehicles and so it pulled a vehicle from a trailer position, vehicle 5, and issued the vehicle replacement command. Vehicle 5 immediately reacted to this command by changing its role from a trailer to swimmer 2 and reported this change in its 13-bit message during cycle 5.

4.1.3 Behavior Results

The tracks of the vehicles participating in the experiment are shown in Fig. 7. As discussed in the previous paragraph, all of the vehicles started their courses and attempted to pursue their given waypoints. At point A,

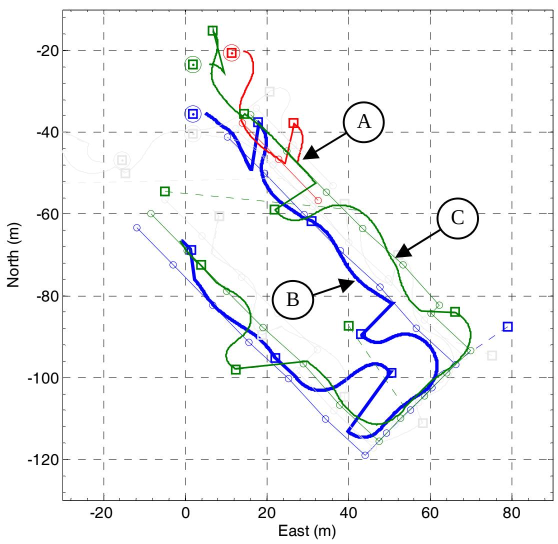

Fig. 7. Vehicle track chart for vehicle replacement experiment

Vehicle 3 completed the short course and dropped out of the formation. The leader was at point B when it realized that there was a problem in the formation and made the command for vehicle 5 to assume the swimmer 2 position. Vehicle 5 was at point C when it received the command to change positions. As discussed previously, vehicle 5 immediately changed course to follow the new set of waypoints as evidenced by its vehicle track. Vehicles 1 and 5 followed their courses to completion, where they each terminated operations. The experiment demonstrates that the language and logic previously presented does allow the leader to replace a vehicle that has stopped functioning with another vehicle.

4.2 Leader Replacement Results

4.2.1 Experiment Conditions

The second experiment is similar to the first in that vehicles were forced to disappear from the fleet. In this experiment, one of the aborted vehicles was the leader, and its loss triggered the leader replacement behavior. Like the previous experiment, three vehicles were instructed to complete the entire course. The vehicle in the default leader position and one of the other swimmers were given short courses so as to simulate vehicle loss. The details of the communication for the second experiment are found in Table 5, and the vehicle tracks are shown in Fig. 8. Again, the discussion is focused on vehicles 1, 3, and 5.

4.2.2 Communications Results

The communication details for the leader replacement experiment are outlined in Table 5. In cycle 1, we see that vehicle 3 had either not started or failed to make its first communication. Vehicle 5 reported full presence nevertheless to avoid initialization problems, as discussed in the previous section. In cycle 2, all vehicles were present and detected. In cycle 3, the leader, vehicle 1, terminated operation after sending out a 32-byte command message. Because the leader did send a 32-byte message, the connection vectors did not report the leader missing until the next cycle. As indicated previously, a vehicle must fulfill three requirements to claim the leader role: first, it must lose contact with the previous leader; second, all other vehicles must indicate that they have lost contact with the previous leader via their connection vectors; third,

Table 5: Communication details from leader replacement experiment

Cycle	32-byte - Cmd	V1-13-bit	V3 -13-bit	V5 -13-bit
1		L,S1,135,Form	-missing-	F2,135,Form
2	V1: —	L,S1,1-5,Form	S2,135,Form	F2,135,Form
3	V1: —	-missing-	S2,135,Form	F2,135,Form
4		-missing-	S2,-35,Form	F2,-35,Form
5		-missing-	L,S2,-35,Form	F2,-35,Form
6	V3: —	-missing-	L,S2,-35,Form	F2,-35,Form
7	V3: V5 goto S1	-missing-	L,S2,-35,Form	S1,-35,Form
…	…	…	…	…

the vehicle must be in a swimmer position. In cycle 4, vehicle 3 reported the leader missing, but could not claim the leader role because the last message from vehicle 5 reported the leader as present. Vehicle 5 could not claim the leader role in cycle 5 because it was not in a swimmer position. Therefore, it wasn't until cycle 6 that vehicle 3 fulfilled all of the requirements and claimed the leader role. Following this role transferral, vehicle 3 requested that vehicle 5 perform a vehicle replacement to fill the empty swimmer 1 position.

4.2.3 Behavior Results

The vehicle tracks shown in Fig. 8 represent the results of the leader replacement behavior. All of the vehicles started their courses, as indicated at the top of the figure. At point A, the leader dropped out of the formation. As discussed in the previous paragraph, the other vehicles took two full communication cycles to realize that the leader was missing. This is visible in the figure by the long distance between the leader dropping out at point A and vehicle 3 assuming the leader role at point B. Finally, the formation was completed again when the new leader requested that vehicle 5 move to the swimmer 1 position at point C.

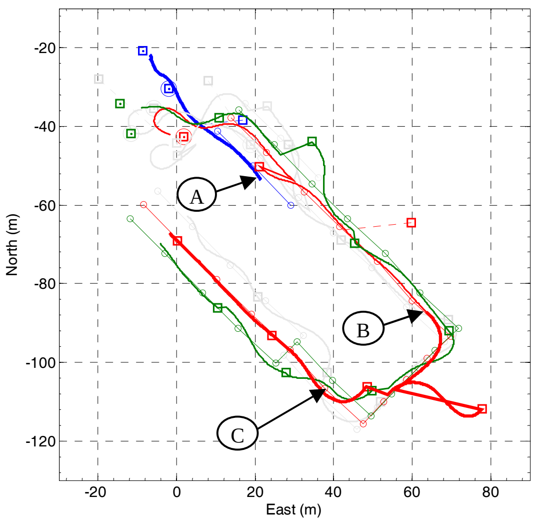

Fig. 8. Track chart for leader replacement experiment

4.3 Self-organization Results

4.3.1 Experiment Conditions

The third experiment demonstrates an emergent behavior, namely, fleet self-organization that arises when both of the replacement behaviors are implemented. The fleet of three vehicles was directed to follow the full course and correct problems as they arise. The intent was to simulate what happens if the vehicles were simply dropped in the water without carefully programming each vehicle beforehand with separate roles. Therefore, all three vehicles were intentionally initialized in the swimmer 1 (S1) position. This caused a formation conflict, since all vehicles considered themselves to be the leader and all occupied the same formation position. Successful accomplishment of the mission required that the vehicles resolve this conflict. The results from this experiment are found in Table 6 and Fig. 9. Vehicles 2, 3 and 4 were used during this experiment.

4.3.2 Communications Results

As seen in Table 6, vehicle 4 was the first vehicle to initiate a 13-bit broadcast, doing so in cycle 1 and consequently becoming the first to declare leadership. Vehicle 2 missed both of the communications from the leader due to a late start which occurred during the 32-byte broadcast. As a result, vehicle 2 also claimed leadership in cycle 2. At this time, there were two vehicles claiming leadership at once. Vehicle 4 did not receive messages from either vehicle 2 or 3 during cycle 2 as indicated in the connection vector. As a result, it did not hear vehicle 2 claim leadership and continued to claim the role in cycle 2. At this point, vehicle 2 received the message from vehicle 4 in which it claimed leadership, and following the logic discussed earlier, removed its own leader designation as indicated in cycle 3. Afterward, there was only one leader but all of the vehicles were using the swimmer 1 position. The leader recognized the duplicate vehicles and reassigned them one at a time. In cycle 4, vehicle 2 was assigned the swimmer 2 (S2) position, and in cycle 5, vehicle 3 was assigned the swimmer 3 (S3) position.

Table 6: Communication details from self-organization experiment

Cycle	32-byte	V2-13-bit	V3-13-bit	V4-13-bit
1				L,S1,234,Form
2	V4: —	L,S1,234,Form	S1,234,Form	L,S1,--4,Form
3	V4: —	S1,234,Form	S1,234,Form	L,S1,234,Form
4	V4: V3 goto S2	S1,234,Form	S2,234,Form	L,S1,234,Form
5	V4: V2 goto S3	S3,234,Form	S2,234,Form	L,S1,234,Form
6	V4: —	S3,234,Form	S2,234,Form	L,S1,234,Form
…	…	…	…	…

4.3.3 Behavior Results

In Fig. 9, it is observed that implementing both vehicle replacement and leader replacement created a self-organizing fleet. From the beginning of their tracks, vehicles 2 and 4 claimed to be the leader. Once vehicle 2 relinquished the leader role at point A, vehicle 4 is the lone leader. At point B, vehicle 3 switches to swimmer 2, and at point C, vehicle 2 moves to swimmer 3. This test showed that the logics underwriting vehicle and leader replacement enabled the fleet to resolve conflicts in which there were multiple leaders or multiple duplicate vehicles; thus, the fleet can self-organize in even these difficult circumstances and arrive at a well-formed group. One can also deduce that a fleet with this combination of behaviors will self-organize when starting without a leader.

5. Conclusions

A fleet of five autonomous underwater vehicles were developed by researchers at the University of Idaho and equipped with acoustic modems for communication and navigation. The language, AUVish-BBM, and behavioral logics were designed and implemented to enable communication and cooperation among the vehicles performing behaviors related to MCM in heterogeneous, open water environments where communication was uncertain and errors could result in incomplete or compromised information. Three of the behaviors needed to perform MCM have been field-tested in an open, unstructured environment at ARD: replacement of lost vehicles, replacement of the leader, and fleet self-organization when multiple vehicles occupied the leader role and the same formation position.

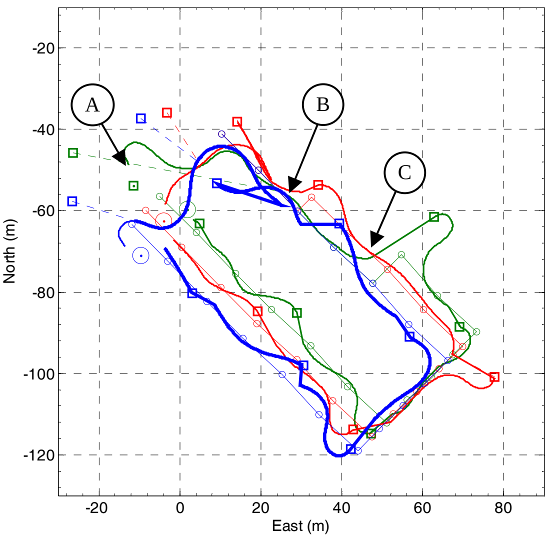

Fig. 9. Track chart for self-organization experiment

Additional behaviors have been simulated and have been the subject of preliminary field tests. These include divert to waypoint, necessary for mine inspection, and map building. AUVish-BBM is fully general and is not tied to the specific behaviors tested to date. The logics described above are associated with the specific behaviors, but as fleet self-organization evinces, they are robust enough to generate emergent behaviors when necessary to solve novel and unexpected problems. As the library of behavioral logics available to the AUVs increases in size, the number and range of emergent behaviors should increase as well. In light of the results collected to date, we believe that the approach we have adopted will generalize over a much broader range of collaborative behavior.

In addition to further simulation and field testing of MCM-related behaviors, effort is underway to formalize a semantics for AUVish-BBM, ensuring even greater control over interpretation when messages are incomplete or incorrect. Also, early work is being done on designing a compositional language that borrows from the literatures in formal computer science [22] and work on agent communication languages [23]. Once developed, a compositional language could replace AUVish-BBM and provide the vehicles with a more robust autonomy.

Acknowledgement

The authors acknowledge the Office of Naval Research, which has supported the work under the award "Developing Fleets of Autonomous Underwater Vehicles" No. N00014-05-1-0674.

References

[1] Thompson, G., "Operations mission and planning workshop", *AUV Fest 2003*, Keyport WA. 11-15 August 2003.

[2] Bovio, E., Checchi, D., Baralli, F., "Autonomous underwater vehicles for scientific and naval operations", *Annual Reviews in Control*, Vol. 20, pp. 117-130, 2006.

[3] Duarte, C., Martel, G., Buzzel, C., Crimmins, D., Komerska, R,

Mupparapu, S., Chappell, S., Blidberg, D., Nitzel, R., "A Common Control Language to Support Multiple Cooperating AUVs", *Proceedings of UUST05 Conference*, Aug 21-24, 2005.

[4] Benjamin, M.R., Battle, D., Eikstedt, D., Schmidt, H., Balasuriya, "Autonomous Control of an Autonomous Underwater Vehicle Towing a Vector Sensor Array", *Proceedings of 2007 IEEE International Conference on Robotics and Automation*, pp. 4562-4569, 2007.

[5] An, E., Beaujean, P-P., Baud, B., Carlson, T., Folledco, A., Tarn, T.J., "Multiple communicating autonomous underwater vehicles", *Proceedings of the 2004 IEEE International Conference on Robotics and Automation, ICRA '04*, Vol. 5, pp.4461- 4464, 2004.

[6] Edwards, D.B., Bean, T., Odell, D., Anderson, M.J., "A Leader-Follower Algorithm for Multiple AUV Formations", *Proceedings of the 2004 IEEE/OES Autonomous Underwater Vehicles*, Sebasco Estates, Maine, June 17-18, 2004.

[7] Rajala, A.G., Edwards, D., O'Rourke, M., "Collaborative Behavior for Vehicle Replacement in AUV Formations", *Proceedings of the 2005 ASME International Mechanical Engineering Congress and Exposition*, Orlando, Florida, November 5-11, 2005.

[8] Rajala, A.G., O'Rourke, M., Edwards, D., "*AUVish*: An Application-Based Language for Cooperating AUVs", *Proceedings of Oceans '06 MTS/IEEE-Boston*, Boston, Massachusetts, September 18-21, 2006.

[9] Edwards, D.B., Bean, T., Odell, D., Anderson, M.J., "A Leader-Follower Algorithm for Multiple AUV Formations", *2004 IEEE/OES Autonomous Underwater Vehicles*, Sebasco Estates, Maine, June 17-18, 2004.

[10] Okamoto, A., Edwards, D.B., "Regulating a Formation of a Large Number of Vehicles," *Proceedings of the 2005 ASME International Mechanical Engineering Congress and Exposition*, Orlando, Florida, November 5-11, 2005.

[11] Okamoto, A., Feeley, J., Edwards, D., and Wall, R., "Robust Control of a Platoon of Underwater Autonomous Vehicles," *Oceans'04 Techno-Ocean'04 (OTO'04) Conference Proceedings*, pp. 505-510, 2004.

[12] Turner, E.H., Chappell, S.G., Valcourt, S.A., "COLA: a Language to Support Communication between Multiple Cooperating Vehicles", *Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology*, 1994, pp. 309-316.

[13] Phoha, S., Eugene, E., "SAMON: Communicarion, Cooperation and Learning of Mobile Autonomous Robotic Agents", *Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence*, 1999. Paper No. 0-7695-0456-6/99.

[14] Phoha, S., Ortego, T., Fulbright, R., Schmiedekamp, M., "CCL: A Common Control Language for Autonomous Networks of Unmanned Underwater Vehicles", *Autonomous Robots*, 2003, pp. 1-8.

[15] Phoha, S., Schmiedekamp, M., "Robotalk: A Common Control Language for Distributed Control of Dynamically Networked Autonmous Devices." *IEEE Transactions on Parallel and Distributed Systems*, 2004.

[16] Stokey, R., "A Compact Control Language for Autonomous Underwater Vehicles", Public Release from WHOI.

[17] Duarte, C., Martel, G., Buzzel, C., Crimmins, D., Komerska, R., Mupparapu, S., Chappell, S., Blidberg, D., Nitzel, R., "A Common Control Language to Support Multiple Cooperating AUVs." *Proc. Of UUST05 Conference*, Aug 21-24, 2005.

[18] Bean, T., Canning, J., Beidler, G., O'Rourke, M., Edwards, D.B., "Designing and Implementing Collaborative Behaviors for Autonomous Underwater Vehicles", *Proceedings of UUST07*, Lee, NH, August 19-22, 2007.

[19] Healey, J. A., "Application of Formation Control for Multi-Vehicle Robotic Minesweeping," *IEEE CDC Conference Paper*, 2001, Paper No. CDC01-INV3103.

[20] Stillwell, D.J., Gadre, A.S., Sylvester, C.A., Cannell, C.J., "Design elements of a small low-cost autonomous underwater vehicle for field experiments in multi-vehicle coordination", Autonomous Underwater Vehicles 2004 IEEE/OES, pp. 1-6, 2004.

[21] Freitag, L., Johnson, M., Grund, M., Singh, S., Preisig, J., "Integrated Acoustic Communication and Navigation for Multiple UUVs", Oceans Conference Record (IEEE), Vol. 4, pp. 2065-2070, 2001.

[22] Ranta, A., *Type-Theoretical Grammar*, Oxford, Oxford University Press, 2002.

[23] Dignum, F., Greaves, M., eds., *Issues in Agent Communication*, Berlin, Springer-Verlag, 2000.

Thomas Bean is a research engineer at the University of Idaho working for the Center for Intelligent Systems Research, specializing in autonomous underwater vehicle control theory and simulation. Bean received his bachelors and masters degrees in mechanical engineering from the University of Idaho in 2003 and 2005.

Dr. John Canning is a post-doctoral researcher with the Microelectronics Research and Communications Institute at the University of Idaho. He received his PhD from the University of Idaho in 1999. Dr. Canning's areas of research include autonomous vehicles, hybrid electric vehicles and advanced lead-acid batteries.

Doug Odell is an electronics engineer at the NSWCCD Acoustic Research Detachment located at Lake Pend Oreille, in Bayview, Idaho. He obtained BSEE and MSEE degrees from Washington State University in 1885 and 1990. Odell specializes in algorithms and software for underwater tracking systems. He has supported the AUV research project since 2003.

Dr. Richard Wall is an Associate Professor of Electrical Engineering at the University of Idaho, teaching courses in embedded system design and digital systems. His research focuses on distributed embedded control for robotics, precision agricultural controls, and smart signals traffic control systems. He was the chief architect of the Ethernet based distributed control system for the fleet of autonomous underwater vehicles.

Dr. Michael O'Rourke is an Associate Professor of Philosophy at the University of Idaho and a fellow in the Center for Intelligent Systems Research. He is a three-year participant in the Autonomous Underwater Research project, focusing on the development of language and logic to support collaborative behavior.

Dr. Michael J. Anderson received his BSME degree from Oregon State University in 1983, and his MS and PhD in Mechanical Engineering from Washington State University in 1987 and 1989 respectively. Anderson is currently a Professor of Mechanical Engineering at the University of Idaho. His research interests are acoustics in fluids and solids, and transducer design.

Dr. Dean B. Edwards is a professor of chemical engineering and Director of the Center for Intelligent Systems Research and the Autonomous Underwater Research Project at the University of Idaho. Edwards' current research interests include fuzzy logic systems, autonomous vehicles, control systems, and hybrid electric vehicles.

Biographical information and photograph for **Geoffrey Beidler** are not available at the time of publication.