Optimal Control of an Autonomous Surface Vehicle to Improve Connectivity in an Underwater Vehicle Network

Camila C. Françolin∗
Anil V. Rao†
University of Florida
Gainesville, FL 32611

Christiane Duarte‡
Gerald Martel§
Naval Undersea Warfare Center
Newport, RI 02841

Abstract
The use of an autonomous surface vehicle as an auxiliary agent to improve connectivity in a network of autonomous underwater vehicles is investigated. An algorithm is developed that consists of an optimal waypoint generator and a minimum-time guidance law that is used to steer the vehicle to the waypoint. This algorithm is used together with a communication architecture to improve underwater vehicle connectivity in a region of interest. The approach is simulated using various underwater vehicle configurations both with and without the autonomous surface vehicle, where it is found that network connectivity is improved significantly via the inclusion of the autonomous surface vehicle. An in-water hardware test is then performed and is shown to be consistent with the simulation results. The results of this study show that adding an autonomous surface vehicle to an underwater vehicle network can greatly improve the connectivity of the network.

1 Introduction

In recent years the problem of cooperative control of multiple-agent systems has received a great deal of attention. Cooperative control falls into two broad categories: centralized control and decentralized control. In centralized cooperative control, each agent is controlled through a single master process. In decentralized control, each agent controls its own motion, but does so by using information from the other agents in the system.

An important problem in cooperative control of multiple-agent air, ground, and underwater vehicle systems is maximizing the connectivity of the mobile sensor network.1–6 Several different

∗Ph.D. Student, Department of Mechanical and Aerospace Engineering. Email: francoli@ufl.edu.
†Assistant Professor, Department of Mechanical and Aerospace Engineering. E-mail: anilrao@ufl.edu. Corresponding Author. Associate Fellow AIAA.
‡Computer Scientist, Autonomous and Defensive Systems Department. Email: christiane.duarte@navy.mil.
§Computer Scientist, Autonomous and Defensive Systems Department. Email: gerald.martel@navy.mil.
definitions of network connectivity have been previously proposed. Global Message or Worst-Case Connectivity is used in systems where it is important to keep all of the nodes in a network connected. In the case of global messaging, the concept of a minimal spanning tree provides a measure of the connectivity of the network. Network Bisection Connectivity is based on the eigenvalues of the graph’s modified Laplacian matrix and comes from the minimum cut tree (MCT) problem. Finally, k-Connectivity is used in situations where it is desired to guarantee connectivity in the event that some subsets of the network are disabled.

In the case of underwater vehicles, applications of cooperative control where network connectivity is important include ocean floor mapping, mine sweeping, and area patrolling. Such tasks are usually accomplished through coordinated control of a network of vehicles, making it necessary to incorporate communication and connectivity constraints to the motion of each vehicle in the network. Previous research on this topic has focused on guaranteeing global network connectedness by restricting the motion of the underwater vehicles so that each vehicle is required to stay within communication range of one another. Ref. 3 relied on maintaining vehicle distances and line-of-sight restrictions, while Ref. 5 devised a plan to improve connectivity in a network by a method of reactive behavior coupled with planning. Ref. 6 devised a reactive control strategy in which the controller is a weighted sum of two goals: the first term guides each robot to its goal position and the second term maintains the constraints that need to be satisfied to maintain network connectivity. While these previously developed algorithms have been successful in keeping the network globally connected, requiring the vehicles to be within a specified separation distance of one another can often prove to be restrictive because the vehicles then must choose between accomplishing the mission and maintaining connectivity. In this research we focus on the case where, having no control over the network itself, we attempt to locally improve the network connectivity by adding an agent to the network and controlling the motion of the additional agent so that it acts as a communications bridge.

In this paper, we consider a decentralized network of autonomous underwater vehicles (AUVs) where each vehicle patrols a particular zone of an area of interest and transmits a status message to the network at prescribed points in time. Each vehicle in the network performs its mission requirements and has no network connectivity restrictions on its motion. Also, we assume that we have no control over the motion of these AUVs. The status message that is broadcast by each vehicle contains relevant information about that particular vehicle as well as about the network in general. Because proper operation of the network requires information contained within these status messages, it is essential that the vehicles stay connected in order to maintain a cohesive...
network and achieve the mission requirements.

We explore the possibility of improving network connectivity via the inclusion of an autonomous surface vehicle (ASV) to the network, where the ASV relays status messages between AUVs that might otherwise be disconnected. We propose a controller for the ASV that seeks to maximize the number of possible connections in the network of AUVs. Thus, the ASV acts as a bridge that relays information to the vehicles in the network. Because the motion of the underwater agents is unrestricted, it is not possible to guarantee globally connectivity of the network. Instead, in this research we seek to locally improve the connectivity of a subgroup of agents in the network. In order to ensure that no part of the network is disconnected for extended periods of time, we develop a behavioral outer-loop controller for the ASV. This outer-loop controller decides at the start of each outer-loop control cycle whether to choose the solution of a continuous-time optimal control problem or to position the ASV in a neighborhood of a vehicle that has been outside of communication range for the longest duration. The performance index of the optimal control problem is designed to maximize the transmission of relay messages from the underwater vehicles to the ASV. If the ASV chooses the solution to the optimal control problem, then the vehicle is steered to a waypoint that predicts the maximize relay transmissions at the start of the next outer-loop control cycle. This process of generating the outer-loop control command and steering the vehicle is repeated at the start of each outer-loop control cycle.

Improvements in the network connectivity were quantified using two metrics referred to as the connectivity number, $N_C$, and the propagation number, $N_P$, where $N_C$ and $N_P$ provide measures of the number of direct and indirect connections, respectively, in the network. Previous research on network connectivity has often employed different metrics from those that we propose in this research. Such metrics include the algebraic connectivity of the network, denoted as $\lambda_2$, which is the second smallest eigenvalue of the network Laplacian, $\lambda_2$ is a measure of the global connectedness of the network. Specifically, a nonzero value of $\lambda_2$ implies that the network is globally connected. In this research we are not interested in a measure of the global connectedness of the network, but are instead interested in the average number of connections (or average valency) in the network as a measure of local improvement in network connectivity.

The contributions of this paper are as follows: (1) a new concept to improve local network connectivity through the inclusion of an additional bridge agent without having to alter the motion of the agents in the network; (2) application of an integrated communication and guidance framework to control the motion of a network of AUVs; (3) the development of a guidance law that uses a behavioral controller to steer the ASV in an optimal manner to improve network connectivity.
over the duration of the mission; (4) testing of the framework and the guidance law in simulation; (5) a hardware implementation of the approach in an actual in-water test.

This paper is organized as follows. In Section 2 we provide an overview of the network connectivity problem of interest in this research. In Section 3 we describe the metrics that are used to assess the connectivity of the underwater vehicle network. In Section 4 we provide details on the algorithm used for guidance and control of the autonomous surface vehicle. In Section 5 we describe the computational approach used to implement the guidance algorithm of Section 4. In Section 6 we show simulation results for different configurations of the AUV network. In Section 7 we show the results of our approach that were obtained from an in-water hardware test. In Section 8 we provide a discussion of the approach. Finally, in Section 9 we provide conclusions.

2 Overview of Motivating Problem

Consider the problem of maintaining communication in a decentralized network of \( n \) autonomous underwater vehicles (AUVs) that move in a region of interest in a body of water. The position of the \( i^{th} \) vehicle in the network, \( A_i, \ (i \in 1, \ldots, n) \) in three-dimensional Euclidean space, \( \mathbb{E}^3 \), at time \( t \) is denoted \( p_i(t) = (x_i(t), y_i(t), z_i(t)), \ (i = 1, \ldots, n) \). The network of AUVs is dynamic in size and is reconfigurable, that is, any agent in the network can change its pattern of movement or can join or leave the network based on changing mission requirements. Furthermore, each AUV has no restriction on its motion based on network connectivity requirements. The AUVs communicate with one another through a status message, where the status message includes both information about the location and velocity of that particular agent and information about the awareness that a particular agent has about the overall state of the network. The status messages that are broadcast throughout the network are essential in order to successfully complete the mission because the behaviors built into each AUV use the information contained in these status messages to make mission-level decisions.

In the mission of interest in this paper, each underwater vehicle \( A_i, \ (i = 1, \ldots, n) \) patrols a specified zone and moves in a fixed pattern that is confined to that zone. The motion of each underwater vehicle is controlled through behaviors called \textit{waypoint} and \textit{loiter}. The \textit{waypoint} behavior guides the underwater vehicle sequentially through a series of predetermined waypoints which will compose its patrol mission. For the purposes of this paper we will consider a series of waypoints such that each AUV moves in a “bowtie” or “figure-eight” pattern in its zone. \textit{Loiter} is a behavior that controls each AUV so that it remains within a specified distance of a particular
location in the water.

Each vehicle in the network communicates through an underwater acoustic modem, where each modem transmits its status message sequentially. The use of sequential transmission prevents an overrun in communication, ensuring every communication is heard throughout the network. Each AUV broadcasts its message once before reaching and after leaving a waypoint, thus creating a pattern that discretizes the real-time mission into communication cycles of length $T_c$. Specifically, at a particular time $t$ each agent $A_i$ ($i \in 1, \ldots, n$) broadcasts to the network its estimated position $p_i(t) = (x_i(t), y_i(t), z_i(t))$ at time $t + T_c$, or at the beginning of the next communication cycle, where the estimate is obtained via an interpolation of a pre-determined trajectory that passes through the waypoints. Using the estimated position, the distance between agents $A_i$ and $A_j$ at time $t + T_c$, denoted $\Delta_{i/j}(t + T_c)$, is computed as

$$\Delta_{i/j}(t + T_c) = \|p_i(t + T_c) - p_j(t + T_c)\|_2$$

where $\| \cdot \|_2$ is the standard Euclidean norm. Table 1 describes each field in the status message and gives the number of bits each field is allotted.

Consider now a configuration of the network where the connectivity is either poor or completely lost. The goal of this research is to study the ability to restore or improve network connectivity via inclusion of an auxiliary agent in the form of an Autonomous Surface Vehicle (ASV). Because we have no control over the motion of the AUVs, it is impossible to guarantee global connectivity of the network. However, we seek to improve local connectivity at each guidance cycle through the addition of the ASV acting as a bridge between some vehicles in the network that would otherwise be out of communication range. We do this by designing a computational framework that determines the control for the ASV that optimizes a performance index, where the performance index is a direct measure of the improvement in the connectivity of the network.

In this research we determine a control that is determined using a two-stage process. First, a waypoint for the ASV is generated by solving a continuous-time optimal control problem on the duration of the communication cycle, $T_c$. The objective functional for this optimal control problem is a function of the position of the ASV and the AUVs in the network at the end of the communication cycle. Using the generated waypoint as a target, a second optimal control problem is solved where the objective function is to minimize the time required for the ASV to reach the waypoint. The control is solved for every $T_g$ time units, where $T_g$ is the guidance cycle. We note for completeness that $T_g < T_c$.

Fig. 1 gives a graphical representation of the mission of interest for this research. The region
to be patrolled is divided into concentric rings (or tiers) that encircle the region of interest. The importance of any tier is directly proportional to its proximity to the center of the ring. Furthermore, each tier is divided into zones, where the number of zones increases according to the size of the surface area of the tier. A tasking behavior assigns each agent in the simulation a tier and a position at initialization; this position will be the patrol zone \(i\) of the vehicle and will define that agent's task until further reconfiguration is needed due to the addition or removal of agents.

We focus our attention to one quarter of the patrolled circumference, assuming cylindrical symmetry to the remaining areas. Assuming three surrounding tiers, the patrolled area is broken down into six distinct zones, each patrolled by an agent, giving a maximum number of six patrolling agents. As agents must leave their patrol zones to refuel, it becomes necessary to reconfigure the patrolling zones to account for the decreased number of agents. This effort can be coordinated between the agents through communication of power resources in their respective status messages along with the refuel and retask behaviors. Refuel is a behavior which compares the current battery life of agent \(A_i\), \((i = 1, \ldots, n)\) with the battery life of other agents in the mission and assesses when an agent is ready to go back to its home station to recharge. Retask is another built-in behavior that reorganizes active agents in the network when any agent has joined or left the network.

3 Communication Model and Network Connectivity Metrics

In order to develop the algorithm described in this paper, we assume a simplified distance-based model for the acoustic modems. Specifically, we use a binary connectivity model, where the strength of the connectivity signal decays to zero when two vehicles are specified distance \(\alpha\) from one another. Although underwater communications can vary as a function of many factors, including water depth and quality, we have found through simulation and in-water tests that this model has proved to be sufficiently good as a starting point in this analysis.

As discussed Section 1, a common metric used to quantify the connectivity of a network is the algebraic connectivity. The algebraic connectivity is a measure of global connectivity and is related to the eigenvalues of the network Laplacian. However, it provides no insight into local improvements in connectivity. Because we have chosen to explore a decentralized problem where we have no control over the motion of the agents in the network, it is common for the network to be globally disconnected at various times during the mission. Therefore we choose to use a different set of metrics, as will be described below, to evaluate the improvement in connectivity in
It is possible to represent the network of interest as an undirected dynamic graph $G(x(t)) = (A, C(x(t)))$ where $A = \{A_1, \ldots, A_n\}$ represents the vertices, or agents, in the network and $C(x(t)) = \{(i, j)|\Delta_{ij}(x(t)) < \alpha\}$ represents the connections between the agents. We can then define an $n \times n$ adjacency matrix $K$ where each entry $(k_{ij})$ represents the connection between agents $A_i$ and $A_j$ such that $k_{ij} = 1$ when $(A_i, A_j) \in C(x(t))$ and $k_{ij} = 0$ otherwise. We note that since $\Delta_{ij}(x(t)) = \Delta_{ji}(x(t))$ then $a_{ij} = a_{ji}$ and $K$ is symmetric. Another important measure of the network connectivity is the matrix of node degrees (or the valency matrix), defined as a diagonal matrix that describes the number of connections of each vertex in the graph; $D(x(t)) = \text{diag}(\sum_{j=1}^{n} k_{ij}(x(t)))$.

In terms of either the adjacency or node degree matrix, we define the connectivity number, $N_C$, as the average node degree of the network at each cycle, which can give insight into improvements of local “pockets” of connectivity within the global network

$$N_C \equiv \frac{\text{trace}(D(x(t)))}{n(n-1)} = \frac{\sum_{i=1}^{n} \sum_{j>i}^{n} k_{ij}}{2n(n-1)}$$

A second important measure of network connectivity is the manner in which information is propagated from one vehicle to another indirectly through intermediate vehicles. Specifically, indirect information propagation relates how information acquired by one agent will be made available to other agents through secondary connections. In order to quantify secondary connections, we define the propagation number, $N_P$, as

$$N_P \equiv N_C + \frac{\sum_{i=1}^{n} \sum_{j>i}^{n} S_{ij}}{2n(n-1)}$$

where $S$ is a $n \times n$ matrix defined as

$$S_{ik} = \begin{cases} 1, & K_{ij} = 1 \cup K_{jk} = 1, \quad i = 1, \ldots, n, \quad j = i + 1, \ldots, n, \\ 0, & \text{otherwise} \end{cases} \quad k = 1, \ldots, n, \quad k \neq i, \quad k \neq j$$

and determines if a secondary connection exists between two agents $A_i$ and $A_k$ that have no direct connection. We note that a propagation number of one implies a strongly connected network, where no agents are excluded.

4 Autonomous Surface Vehicle Dynamics, Guidance, and Control

The autonomous surface vehicle (ASV), denoted as agent $A_0$, is controlled over equally-spaced time intervals that follow the communication cycles of duration $T_c$ time units. The input to the
ASV is chosen in a manner that maximizes the improvement in the connectivity of the network over each communication cycle. Because the speed of the vehicles as viewed by an observer in an inertial (Newtonian) reference frame is sufficiently small, it is reasonable to model the dynamics by including only the kinematics. Since the ASV is constrained to remain on the surface of the water, the two-dimensional kinematic equations that govern the motion of the ASV are given in Universal Transversal Mercator (UTM) inertial Cartesian coordinates as

\[
\begin{align*}
\dot{x} &= v \cos(\theta), \\
\dot{y} &= v \sin(\theta), \\
\dot{\theta} &= u
\end{align*}
\]  

(5)

where \(x\) and \(y\) are the two components of position, \(v\) is the inertial speed, \(\theta\) is the heading angle (and defines the orientation of the vehicle relative to East), and \(u\) is the heading angle rate. In order to effectively improve the connectivity of the network, it is assumed in this research that the ASV moves with a larger speed than each of the AUVs and that the position of the ASV is unconstrained.

Autonomous motion of the ASV is accomplished through a \textit{waypoint generator} in conjunction with a \textit{guidance algorithm} and an \textit{on-board controller}. As stated in Section 2, the ASV waypoint is generated every \(T_c\) time units, where \(T_c\) is the communication cycle. The waypoint is obtained by solving a continuous-time optimal control problem, where the objective functional of the optimal control problem is a direct measure of the improvement in the connectivity of the network. Next, the guidance command used to steer the ASV is obtained by solving a second optimal control problem where it is desired to minimize the time required to attain the waypoint obtained by the waypoint generator. The guidance command is generated every \(T_g\) time units, where \(T_g\) is the guidance cycle and \(T_g < T_c\). The on-board controller then attempts to achieve the guidance command. Effectively, the aforementioned process first provides a target for the ASV, then controls the vehicle to attain the target as quickly as possible. The process of waypoint generation, guidance, and control is depicted in Fig. 2.

4.1 Optimal Waypoint Generator

Using the estimated position of the underwater vehicles at time \(t + T_c\), the next waypoint for the ASV is obtained by solving the following continuous-time optimal control problem. Maximize the cost functional

\[
J = \sum_{i=1}^{n} W_i \arctan(\alpha - \Delta_{0/i}(t_0 + T_c))
\]  

(6)
subject to the dynamic constraints of Eq. (5), the initial conditions

\[ x(t_0) = x_0, \quad y(t_0) = y_0, \quad \theta(t_0) = \theta_0, \]  

(7)

the control inequality constraint

\[ |u| \leq u_{\text{max}}, \]  

(8)

and the terminal inequality time constraint

\[ t_0 + T_c \leq \lambda, \]  

(9)

where \( t_0 \) is the start time of the communication cycle and \( \Delta_{0/i}(t + T_c), \) \((i = 1, \ldots, n)\) is the predicted distance between the ASV and each of the agents \( A_i, \) \((i = 1, \ldots, n)\) at the beginning of the next communication cycle. Furthermore, the weights, \( W_i, \) \((i = 1, \ldots, n)\), in the cost functional of Eq. (6) are defined as

\[ W_i = \frac{t - t_{i_{\text{last}}}}{\lambda}, \quad (i = 1, \ldots, n), \]  

(10)

where \( t_{i_{\text{last}}} \) is the time at which agent \( A_i \) last transmitted its status message. It is important to note that the cost functional of Eq. (6) is a measure of the number of direct connections between agents, thus making it an appropriate measure of the improvement in the connectivity of the network at the end of the next communication cycle. Furthermore, the weights \( W_i, \) \((i = 1, \ldots, n)\) in the cost functional of Eq. (6) are chosen so that the ASV moves closer to the vehicles that have been out of communication range for the longest duration. Consequently, Eq. (6) effectively maximizes the number of AUVs within communication range of the ASV with a bias toward those agents that have been excluded from the network. Finally, it is noted that the differential equation \( \dot{\theta} = u \) is included in order to constrain the maximum and minimum allowable turn rate on the vehicle, and together with the final time constraint of Eq. (9), ensures that the ASV can reach its optimal point before the end of the guidance cycle; thus providing a more realistic waypoint.

In order to solve the optimal control problem given in Eqs. (6)–(9), the necessary information from the status message of each AUV must be supplied to the waypoint generator. Specifically, it is necessary for the ASV to know the location of each AUV at the beginning of the next communication cycle in order to position itself optimally for the beginning of the next set of underwater transmissions. Also important in the status message are the current plan that the network is performing, which indicates how many agents are on task, and the current task each vehicle is performing, which indicates what zone that particular vehicle is patrolling. All status messages are received through a database called the MOOSDB (see Section 5 below) that simulates all communication processes.
The optimal control problem of Eqs. (6)–(9) is solved twice at the start of each communication cycle. The first solution, labeled, $S_1$, is obtained with all weights $W_i$, $(i = 1, \ldots, n)$ set to unity. The second solution, labeled $S_2$, is obtained using the weights specified in Eq. 10. Solving the optimal control problem of Eqs. (6)–(9) twice ensures that no vehicle that may be within communication range is overlooked due to unequal weights. The decision to choose the solution $S_1$ or $S_2$ is made according to a set of pre-specified conditions as found in Table 2. The conditions are structured in an if-then-else construct, such that priority is given to the conditions from the top down. The waypoint obtained from the solution to this problem is denoted $(x^*_f, y^*_f)$.

Because the optimal control problem of Eqs. (6)–(9) does not have an analytic solution, it is solved numerically. In this research we choose to solve the optimal control problem of Eqs. (6)–(9) using the open-source pseudospectral optimal control software GPOPS\textsuperscript{13} together with the nonlinear programming solver SNOPT.\textsuperscript{14, 15} In order to obtain a solution to the optimal control problem of Eqs. (6)–(9) as efficiently as possible while still obtaining an accurate solution, ten Legendre-Gauss collocation points are chosen. Although choosing such a small number of collocation may seem to be insufficient, GPOPS uses Legendre-Gauss quadrature approximation, thus making it possible to obtain an accurate waypoint in a computationally efficient manner using a small number of collocation points. While it is beyond the scope of this paper to provide a detailed explanation of pseudospectral methods and their accuracy, details about various pseudospectral methods for solving optimal control problems can be found in Refs. 16–20.

As an example of the solution obtained from GPOPS to the optimal control problem of Eqs. (6)–(9), consider the case $n = 6$ (that is, five AUVs plus the ASV), $\lambda = 150$ s, and $\alpha = 750$ m. In this example the AUVs were placed in the following positions that make it impossible for all AUVs to be within communication range of the ASV:

<table>
<thead>
<tr>
<th></th>
<th>$A_0$</th>
<th>$A_1$</th>
<th>$A_2$</th>
<th>$A_3$</th>
<th>$A_4$</th>
<th>$A_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x$ (m)</td>
<td>750</td>
<td>50</td>
<td>1450</td>
<td>1450</td>
<td>750</td>
<td>50</td>
</tr>
<tr>
<td>$y$ (m)</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>850</td>
<td>850</td>
<td>850</td>
</tr>
</tbody>
</table>

The initial position of the ASV is denoted by a “\(\bigtriangledown\),” and its overall path during the time interval is shown as the blue line with triangle markers. Each agent is denoted with a “\(\bigcirc\)” if it is within communication range of the ASV range at the end of the guidance cycle and is denoted with a “\(\square\)” if it is outside communication range of the ASV at the end of the guidance cycle. Fig. 3 shows the test cases results. First, it is seen for the case where all weights are unity that the ASV attempts to position itself in the middle of the five underwater agents, thus ensuring communication with
a maximum possible number of underwater agents. Next, changing the weight of agent $A_1$ to $W_1 = 2$ results in the ASV moving closer to $A_1$ while still keeping the maximum possible number of AUVs in its communication range. Finally, the weight of agent $A_2$ was changed to $W_2 = 3$, where it is seen that the ASV moves to maintain the network connectivity of the three vehicles with the greatest weight.

Finally, as already mentioned, each AUV patrols a particular zone of the area of interest. The zone each vehicle patrols is indicated by its current task, which is broadcast in the status message. If a vehicle is outside communication range of the ASV for more than three communication cycles, and thus the ASV has not received its status message, it becomes a priority and a decision is then made to override the optimal waypoint obtained through the waypoint generator and instead go to the center of the zone of that vehicle in an attempt to re-establish a link with that vehicle. This ASV behavior is denoted as “polling” and is important in underwater applications in order to ensure an underwater vehicle has not drifted outside of the operation area. Fig. 4 summarizes the manner in which the waypoint generator computes the waypoint command.

4.2 Guidance Algorithm and On-Board Controller

The guidance algorithm is in charge of receiving navigation information from the ASV as well as sending steering commands to the on-board controller. The on-board controller runs at the same rate as the guidance cycle, receiving the guidance command and relaying back the current measured position, heading, and speed of the ASV. The guidance command used to steer the ASV consists of heading and speed. The commanded speed of the ASV is constant and preset by a configuration file. The commanded heading is found by solving the following optimal control problem. Minimize

$$J = t_f - t_0$$

subject to the dynamic constraints

$$\dot{x} = v \cos \theta \quad , \quad \dot{y} = v \sin \theta,$$

and the boundary conditions

$$x(t_0) = x_0 \quad , \quad y(t_0) = y_0 \quad , \quad x(t_f) = x^*_f \quad , \quad y(t_f) = y^*_f,$$

where $(x_0, y_0)$ is the current position of the ASV, and $(x^*_f, y^*_f)$ is the position of the waypoint generated by the optimal waypoint generator described in Section 4.1. It can be shown (see Section
2.7 of Ref. 21) that the solution to the optimal control problem of Eqs. (11)–(13) is given as

\[ \theta^* = \tan^{-1}(y_f - y_0, x_f - x_0), \]  

(14)

where the function \( \tan^{-1}(\cdot, \cdot) \) is the four quadrant inverse tangent. It is important to observe that the rate constraint on the angle \( \theta \) has been omitted from the optimal control problem of Eqs. (11)–(13) because including the rate constraint on \( \theta \) leads to an optimal control problem that does not have an analytic solution.

Fig. 5 is an example of an ASV path obtained by solving the control problem of Eqs. (6)–(9) alongside the path obtained using the guidance law of Eq. (14) [together with the vehicle rate constraint]. It can be seen that the minimum time solution provides a more direct path to the desired final waypoint, even after the rate constraints are applied.

### 4.3 Separation of Waypoint Generation and Guidance

Separating the ASV waypoint generation from the ASV guidance has several advantages. First, the waypoint generator numerically solves an optimal control problem that does not have an analytic solution. As a result, this period at which this first optimal control problem is solved (in this case, the duration of a communication cycle, \( T_c \)) must be much larger than the time required to solve the problem. In this research it was found that the CPU time required to compute optimal waypoint was between one and two seconds, much less than \( T_c \). Second, while it is possible to use as the guidance command the heading angle \( \theta \) obtained from the solution of the first optimal control problem, this angle is not desirable because the time required to steer the ASV to the optimal waypoint may be very close to the communication cycle. If a sufficiently large disturbance (e.g., a current or wind) alters the predicted path of the ASV, the ASV may not reach the next waypoint within the time constraint. Alternatively, even if the ASV completes the maneuver within the allowable time, it may reach a non-optimal waypoint, thus reducing the improvement in the connectivity. Second, the minimum-time steering angle (that is, the guidance law) can be solved for analytically, thus making it possible to perform guidance at a much higher rate than that of the communication cycle. Furthermore, the minimum-time steering angle provides time margin in the sense that the ASV will reach the target sooner than it would have had the optimal waypoint control been used.
5 Computational Approach

The aforementioned approach for improving network connectivity was implemented using the Mission Oriented Operating Suite/Interval Programming\(^{22}\) (MOOS-IvP) architecture for autonomous control. MOOS-IvP is developed and maintained by the Naval Undersea Warfare Center (NUWC) in Newport, Rhode Island, the Massachusetts Institute of Technology (MIT), and Oxford University. MOOS-IvP is a set of open-source C++ modules for controlling the operations of autonomous marine vehicles. The suite consists of a set of distinct processes communicating through a publish-subscribe database called the MOOSDB. Each variable published to MOOSDB is available to any other processes that have subscribed to that particular variable. Furthermore, variable publications and subscriptions can be performed in real time.

MOOS was chosen for this research because of its proven ability to handle communication for cooperative control of networks of underwater vehicles.\(^{23,24}\) MOOS includes the necessary control functions as well as communication modules for simulation and real-time implementation. Moreover, MOOSDB provides the unified interface standard that enables fully autonomous integration of modeling, processing, and control. This allowed for the development of one framework for both simulation and in-water testing. In simulation the framework could be linked to a vehicle model module, whereas during in-water testing the framework was incorporated directly into the underwater vehicles.

All AUVs (that is, agents $A_1, \ldots, A_n$) are guided and controlled through modules implemented in MOOS. The control of the AUVs is handled in MOOS via separation between vehicle control and vehicle autonomy. Dynamic control of the AUVs is accomplished via a MOOS process referred to as the frontseat driver, where the frontseat driver provides a navigation and control system steering the vehicle capable of streaming the vehicle’s position and trajectory information to the vehicle’s autonomy controller, called the backseat driver, and accepting back a stream of autonomy decisions that guide the vehicle such as heading, speed, and depth. Thus the vehicle is controlled through a frequent exchange of data between the autonomy controller, or the backseat driver, and the dynamic controller, or the frontseat driver; all guidance decisions being made by the autonomy controller, while all navigational information is handled by the dynamic controller.\(^{22}\) The backseat driver, or the vehicle autonomy, of all underwater agents consists of the MOOS behaviors which include waypoint, loiter, refuel and retask. For in-water tests the frontseat driver controls the vehicle directly.\(^{25}\) For the purposes of this research, the AUV dynamic model was implemented in C++ and is controlled by input of the desired speed, depth, and heading.
6 Simulation Results

Simulations were carried out using a maximum of six underwater agents, where it is noted that $C_{\text{max}} = n(n - 1)$ is the maximum number of connections for $n$ agents. Each simulation is run without the ASV (that is, $n = 6$) and with the ASV (that is $n = 7$). In all results shown the connectivity and propagation metrics are normalized by $C_{\text{max}}$. In order to show the performance of the ASV, simulations were performed with varying mean distances, $d$, between the AUVs as shown in Fig. 7. Two particular simulations, called close and spread configurations, are shown in more detail to demonstrate the effectiveness of the ASV. In the close configuration, $d = 550$ m and the AUVs are within communication range of one another. In the spread configuration $d = 750$ m and communication between adjacent vehicles was not always possible. Thus, the main role of the ASV in the close configuration is to improve the rate of information propagation in the network. On the other hand, in the spread configuration the main role of the ASV is to attempt to maintain connectivity in the network, linking as many adjacent agents as possible. Finally, in order to
test the polling behavior of the ASV, in the close configuration simulation a scenario is chosen where one of the vehicles ceased to communicate with the network for an extended period of time (specifically, a duration of six communication cycles).

The connectivity and propagation numbers, $N_C$ and $N_P$, are shown in Fig. 8 for the close configuration simulation. It is seen that $N_C$ and $N_P$ increase by including the ASV. The metrics, however, show a decrease in connectivity at the 17$^{th}$ communication cycle. This reduction in connectivity is due to the fact that the extended period of noncommunicativity of one underwater vehicle in the network leads to the ASV making the decision to attempt to regain communication with the lost vehicle as opposed to maintaining connectivity with the rest of the network. Thus we see a trade-off between overall network connectivity and vehicle polling.

Fig. 9 shows the connectivity and propagation numbers, $N_C$ and $N_P$, for the spread configuration simulation. In this case it is seen that $N_C$ and $N_P$ are greatly improved with the inclusion of an active ASV. Specifically, it is seen at the beginning of the simulation that the connectivity in the network is not improved because several communication cycles are required before the ASV is in position to assist with improving network connectivity. Once the ASV is in position, however, its inclusion in the network is seen to significantly improve network performance. Finally, Table 3 shows the results of both the close and spread configurations, where it is seen that the ASV improves the connectivity of the network in simulations.

The close and spread configurations demonstrate that including the ASV is most useful when the mean distance between the AUVs is large or when the AUV network is dynamically reconfiguring itself. In either of these cases, the AUV network requires assistance in order to maintain connectivity. The results obtained in the case where $d$ is small is promising, although the addition of the ASV may be redundant because the AUV network is already well connected. To better illustrate the improvement in network connectivity, Fig. 10 shows $N_C$ and $N_P$ as a function of $d$. These results show a large improvement in connectivity with the addition of the ASV at the upper limit of the modem communication range (that is $d = 750$ m), thus making the ASV an important asset to the network.

Fig. 11 shows the path of the ASV in the close and spread configurations. In the spread configuration it can be seen that the ASV moves throughout the network in order to propagate information and include as many AUVs as possible. As agents stay out of communication range for longer than three communication cycles, the ASV attempts to return the noncommunicating AUV to the network. In the close configuration, however, the ASV maintains a central position in order to propagate information more efficiently.
7 In-Water Experiments

The scenario described in the problem formulation section was implemented and tested at the Naval Underwater Warfare Center’s North test range in Newport, RI. The test was run using a three underwater vehicle configuration along with one surface vehicle organized into two tier. The vehicles were arranged in a close configuration with the center of their patrol set to the coordinates measured in meters from a local origin as seen in table 4.

Three of the main functions of the ASV were tested: (1) the ability of the ASV to correctly place itself in a central communication location relative to the AUVs when initialized in a non-optimal location; (2) the active repositioning of all AUVs and of the ASV when one underwater vehicle must leave its patrol area in order to refuel; and finally (3) the polling behavior of the ASV when an underwater vehicle is disconnected from the network for an extended amount of time.

7.1 In-Water Hardware

In this section we describe pertinent details of the hardware used for the in-water test conducted at NUWC North test range. Specifically, this section describes the autonomous underwater vehicles, the autonomous surface vehicle, and the underwater modem.

7.1.1 Autonomous Underwater Vehicles

The AUVs used for the in-water testing were Iver2 vehicles provided by the Maritime Autonomy Group at the Naval Undersea Warfare Center. The Iver2 AUV, pictured in Fig. 12 is a small man-portable AUV manufactured in Fall River Massachusetts by Ocean Server Technology, Inc. The Iver2 vehicles are approximately 65 inches in length and weigh approximately 50 pounds. These vehicle are five inches longer than the standard Iver vehicles and was lengthened to provide space for additional electronics including a precision clock, inertial measurement unit, and Iridium satellite transceiver. The vehicles can operate at speeds of up to 4 knots using its rechargeable batteries which have 600 Wh of capacity. Vehicle endurance is anywhere between 4 and 12 hours depending on the vehicles speed and hotel load (e.g. A/D conversion for the acoustic array). The vehicle also comes equipped with integrated depth (pressure) and altitude sensors. The Iver2 AUV is also equipped with the YSI 6600 V2 sonde Woods Hole Oceanographic Institute (WHOI) acoustic micromodem. The payload computer stack consists of a PIII-800 CPU and two 8 channel D/A boards.

Each of the Iver2 vehicles is equipped with a GPS receiver and three-axis digital compass.
providing roll, pitch, and yaw. One of the vehicles was recently equipped with a Doppler velocity
logger (DVL) to provide closed-loop speed control. The other two vehicles have open-loop speed
control using speed tables based on thruster RPM. Navigation solutions are computed using dead
reckoning and sent to the backseat over the backseat/frontseat interface. The Iver2 vehicles use
WiFi communication while on the surface and micro-modem communication while underwater.
The acoustic modem is used both for providing status messages to the topside as well as for
reception of command and control messages from the topside.

7.1.2 Autonomous Surface Vehicle

The ASV used for the in-water testing was provided by the Naval Undersea Warfare Center’s
(NUWC) Maritime Unmanned Systems Facility (MUSF). Each ASV comes with three distinct as-
sets required for operation: (1) an ASV platform; (2) a host station housed in a storage container
express (CONEX); and (3) a command and control (C2) workstation. Fig. 13 shows all three of
these components. The platform used during testing was the NUWC-4, the latest ASV developed
by the MUSF. The exterior dimensions of the NUWC-4 vehicle are consistent with the United
States Navy standard eleven meter Rigid-Hulled Inflatable Boat (RHIB). The dynamics of the
ASV, propulsion status, power, and environmental data are monitored remotely from the com-
mand and control workstation. The ASV is equipped with Global Positioning System (GPS) and
Inertial Measurement Unit (IMU) sensors to support navigation.

The ASV host platform consists of a fifteen feet CONEX box with telescoping poles to support
the host antenna array system. The CONEX box houses the command and control workstation
and must be located within line-of-sight range of the ASV operating area for transmission of navi-
gational data. The C2 operator has the capability of loading and initiating the execution of a route,
which consists of a series of waypoints to the ASV core system. The C2 operator also has the
capability to send throttle, rudder, and transmission control commands to the ASV core system
in order to perform emergency maneuvers as required. For this in-water experiment an on-board
computer running the ASV algorithm provided all guidance commands which were then sent to
the C2 operator who executed all commands remotely.

7.1.3 Underwater Modem

The underwater acoustic modem used for communication and status message relaying amongst
agents during the in-water testing was designed by the Acoustic Communications Group at the
WHOI. The WHOI Micro-Modem is a compact, low-power, underwater acoustic communication and navigation subsystem. It has the capability to perform low-rate frequency-hopping frequency-shift keying (FH-FSK), variable rate phase-coherent keying (PSK), and two different types of long base line navigation, narrowband and broadband. For the purposes of this in-water test the modem was configured to send and receive a 32 byte message and the default data rate is 80 bps. The system operates at approximately 10, 15 or 25 kHz and uses 4 kHz of bandwidth. In order to allow for the ASV to communicate with the underwater vehicles, a transducer towfish, shown in Fig. 14 was towed by the ASV.

### 7.2 In-Water Results

The connectivity and propagation numbers for the in-water tests were calculated from the log files each agent collected during the tests. Each log file consists of a text file containing all information transferred and stored through the MOOSDB during the experiment. The three ASV functions tested were divided into two main experiments, following the same format as done in simulation. In the first experiment the ASV was initialized in a position on the corner of the operation area, and as it acquires information about the underwater network it correctly positions itself in an optimal position to propagate information amongst underwater agents. In the second experiment the ASV is initialized in a central location relative to the underwater agents, and after a number of intervals one of the underwater agents ceases communication such that the ASV’s “polling” behavior is tested.

Fig. 15 shows the results for the first in-water test. We note that the connectivity and propagation numbers are worse for the network with the ASV included while the ASV is traveling to optimally position itself, as was seen in simulation. Once the ASV is correctly positioned the improvements in connectivity and propagation number are obvious with the inclusion of the ASV in the network. Fig. 16 shows the results for the second in-water test. It can be seen that the connectivity and propagation numbers are increased with the addition of the ASV. The connectivity, however, decreases at communication cycle 13 due to the fact that after one of the AUVs ceases to communicate with the network for an extended period of time. As a result, the ASV moves to attempt to restore communication with the noncommunicating vehicle. This last result shows that the ASV will trade overall network connectivity in order to attempt to regain communication with a lost vehicle.

Table 5 summarizes the mean improvements in connectivity made with the addition of the ASV to the underwater network for each in-water test. From these results it can be seen that the
improvements for the in-water tests were higher than in simulation. Simulation results assumed 
perfect communication, whereas actual in-water communication have a number of different fac-
tors which can hinder a message from reaching its target. This result shows that the ASV concep;t
is most useful when used in situations of intermittent and uncertain communication.

8 Discussion

We have seen that the addition of an ASV can improve the connectivity of a network of AUVs
while still leaving each vehicle free to perform its mission task. The simulations performed in this
research show an average improvement of five percent and 12 percent, respectively, in connectiv-
ity number and propagation number throughout the entire simulation. Note, however, that these
average numbers can be skewed due to the initial positioning and initialization of the ASV. Actual
improvement of up to 20 and 55 percent were seen in the connectivity and propagation numbers,
respectively, for particular communication cycles. The results given here indicate that including
an ASV is most beneficial in a dynamically reconfigurable network of underwater vehicles, such
that it can ensure that no vehicles will be left out of connectivity for extended periods of time.
We have also seen that the ASV is useful when used in situations of intermittent and uncertain
communication, such as what is often seen in real-world underwater applications. In a case where
the network is already connected on a known mission path, the use of an ASV may be costly and
redundant, as an extra vehicle must be added to the network for its implementation. In situations,
however, where network connectivity is unreliable or even nonexistent, the increase in connectiv-
ity provided by the ASV may outweigh the cost of adding an extra vehicle, and allow a network
to perform a mission that may otherwise not be possible to accomplish.

9 Conclusions

A new approach for improving the connectivity of an underwater vehicle network by introduc-
ing an autonomous surface vehicle into the network has been studied. An algorithm has been
developed that consists of an optimal waypoint generator and a minimum-time guidance law.
The approach is designed to improve the number of connections in the network, thus impro-
ving the overall connectivity of the network. Simulations were performed for different network
configurations, and it was found that the inclusion of the autonomous surface vehicle improves
network communication. In-water experiments were also run using a network of three under-
water vehicles and one surface vehicle. Results of the in-water experiments were consistent with the simulations, demonstrating that the inclusion of the autonomous surface vehicle improves network connectivity.

Acknowledgments

The authors gratefully acknowledge support for this research from the U.S. Office of Naval Research under Grant N00014-08-1-1173 and Contract N0001410WX20839. The authors would also like to thank Mr. Scott Sideleau, Dr. Donald Eickstedt, Mr. Gary Choplin, and Mr. Kyle DeMedeiros for their help and support during the course of this research.

References


Figure 1: Mission scenario consisting of a fleet of autonomous underwater vehicle and an autonomous surface vehicle. The autonomous underwater vehicles move in a region of interest, where the ROI is divided into tiers and zones. Each AUV is assigned to a patrol zone at initialization, and these zones are reconfigured as vehicles join or leave the network.
Figure 2: Block diagram representation of the autonomous surface vehicle system.

Figure 3: Block diagram representation of the autonomous surface vehicle system.
Figure 3: Simulations showing the motion of ASV that optimize the cost function of Eq. (6).
Check for Status Messages

Run Optimization Problem

Accept “Best” Solution

Choice 1 Choice 2 Choice 3 Choice 4 Choice 5 ... Choice 9

Have Any Vehicles Been Outside of Communication Range for Longer Than Three Guidance Cycles?

Override and Go to Center of Vehicle’s Zone

YES

NO

Go to Optimal Waypoint

Figure 4: Graphical representation of the ASV guidance algorithm.
Figure 5: ASV path obtained using heading angle from the optimal waypoint generator alongside the path obtained using the minimum-time heading angle together with the vehicle rate constraints.
Figure 6: Schematic of simulation framework. Each computer runs a simulated vehicle along with the necessary MOOS communication processes.
Figure 7: Mean distances between underwater agents were varied in simulation. Two of these simulation cases will be highlighted in which \( d = 550 \) meters (close configuration) and \( d = 750 \) meters (spread configuration).
Figure 8: Connectivity number, $N_C$, and propagation number, $N_P$, for the close configuration simulation. Adding the ASV improves both $N_C$ and $N_P$ until communication cycle 17, where the decision is made to restore connectivity with a lost vehicle.
Figure 9: Connectivity number, $N_C$, and propagation number, $N_P$, for the spread configuration simulation, where adding the ASV improves both $N_C$ and $N_P$. Improvements are not seen near the start of the simulation because the ASV requires several communication cycles before it reaches its first optimal waypoint.
Figure 10: Improvements in $N_C$ and $N_P$ as a function of mean distance, $d$, from the centers of the AUV patrol zones for the case where the ASV is added to the network. It is seen that including the ASV results in a significant improvement when the mean distance is beyond the maximum range of the acoustic underwater modem.
Figure 11: Snapshots of path taken by the ASV in the close and spread configurations.
Figure 12: Iver2 AUV used for the in-water testing.
Figure 13: Autonomous surface vehicle used for the in-water testing. Each ASV comes with a host station contained within a storage container express (left), a command and control workstation (middle), and an ASV platform (right).
Figure 14: The underwater acoustic modem used for communication and status message relaying amongst agents during the in-water testing.
Figure 15: Connectivity number, $N_C$, and propagation number, $N_P$, for the first in-water test.
Figure 16: Connectivity number, $N_C$, and propagation number, $N_P$, for the second in-water test.
Table 1: Status message format and field sizes.

<table>
<thead>
<tr>
<th>Field</th>
<th>Number of Bits</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message ID</td>
<td>4</td>
<td>1 for status message, 2 for other</td>
</tr>
<tr>
<td>Platform ID</td>
<td>5</td>
<td>Vehicle Identification No.</td>
</tr>
<tr>
<td>Destination ID</td>
<td>5</td>
<td>0 for network broadcast</td>
</tr>
<tr>
<td>Timestamp</td>
<td>32</td>
<td>In UTC seconds</td>
</tr>
<tr>
<td>Latitude</td>
<td>23</td>
<td>Current latitude in <em>decimal degrees</em></td>
</tr>
<tr>
<td>Longitude</td>
<td>24</td>
<td>Current longitude in <em>decimal degrees</em></td>
</tr>
<tr>
<td>Depth</td>
<td>10</td>
<td>Current depth in <em>meters</em></td>
</tr>
<tr>
<td>Heading</td>
<td>11</td>
<td>Current heading in <em>degrees</em> (North = $0^\circ$)</td>
</tr>
<tr>
<td>Speed</td>
<td>8</td>
<td>Current speed in <em>m/s</em></td>
</tr>
<tr>
<td>Power</td>
<td>10</td>
<td>Current vehicle energy status</td>
</tr>
<tr>
<td>Ocean Current</td>
<td>23</td>
<td>Vector indicating ocean current, if available</td>
</tr>
<tr>
<td>Current Task</td>
<td>4</td>
<td>Current vehicle patrol zone</td>
</tr>
<tr>
<td>Current Plan</td>
<td>4</td>
<td>Current network tasking plan/ No. of patrol vehicles</td>
</tr>
<tr>
<td>Next Latitude</td>
<td>23</td>
<td>Next waypoint latitude in <em>decimal degrees</em></td>
</tr>
<tr>
<td>Next Longitude</td>
<td>24</td>
<td>Next waypoint longitude in <em>decimal degrees</em></td>
</tr>
<tr>
<td>Vehicle 1 Status</td>
<td>4</td>
<td><em>Integer</em> number of time steps since last communication with $A_1$</td>
</tr>
<tr>
<td>Vehicle 2 Status</td>
<td>4</td>
<td><em>Integer</em> number of time steps since last communication with $A_2$</td>
</tr>
<tr>
<td>Vehicle 3 Status</td>
<td>4</td>
<td><em>Integer</em> number of time steps since last communication with $A_3$</td>
</tr>
<tr>
<td>Vehicle 4 Status</td>
<td>4</td>
<td><em>Integer</em> number of time steps since last communication with $A_4$</td>
</tr>
<tr>
<td>Vehicle 5 Status</td>
<td>4</td>
<td><em>Integer</em> number of time steps since last communication with $A_5$</td>
</tr>
<tr>
<td>Vehicle 6 Status</td>
<td>4</td>
<td><em>Integer</em> number of time steps since last communication with $A_6$</td>
</tr>
<tr>
<td>Bit Sum</td>
<td>234</td>
<td>30 bytes</td>
</tr>
</tbody>
</table>
Table 2: Decision hierarchy for behavioral controller to accept its next position from the guidance solutions.

<table>
<thead>
<tr>
<th>Choice</th>
<th>Condition</th>
<th>Solution Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>All agents in solution 1 are in range</td>
<td>Solution 1</td>
</tr>
<tr>
<td>2</td>
<td>All agents in solution 2 are in range</td>
<td>Solution 2</td>
</tr>
<tr>
<td>3</td>
<td>More agents in solution 1 are in range than in 2</td>
<td>Solution 1</td>
</tr>
<tr>
<td>4</td>
<td>More agents in solution 2 are in range than in 1</td>
<td>Solution 2</td>
</tr>
<tr>
<td>5</td>
<td>Solution 1 encompasses agents with weight ( \geq 2 )</td>
<td>Solution 1</td>
</tr>
<tr>
<td>6</td>
<td>Solution 2 encompasses agents with weight ( \geq 2 )</td>
<td>Solution 2</td>
</tr>
<tr>
<td>7</td>
<td>SNOPT did not reach a minimum for solution 2</td>
<td>Solution 1</td>
</tr>
<tr>
<td>8</td>
<td>SNOPT did not reach a minimum for solution 1</td>
<td>Solution 2</td>
</tr>
<tr>
<td>9</td>
<td>If none of the above conditions are satisfied</td>
<td>Solution 2</td>
</tr>
</tbody>
</table>
Table 3: Connectivity and propagation numbers for close and spread configurations with and without the ASV.

<table>
<thead>
<tr>
<th></th>
<th>$N_C$</th>
<th>$N_P$</th>
<th>Percentage Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close</td>
<td>Without ASV</td>
<td>With ASV</td>
<td>4.2</td>
</tr>
<tr>
<td></td>
<td>50.9</td>
<td>55.1</td>
<td></td>
</tr>
<tr>
<td>Spread</td>
<td>45.6</td>
<td>50.8</td>
<td>5.2</td>
</tr>
</tbody>
</table>
Table 4: Position of the center of the patrol zone for each AUV in the in-water test.

<table>
<thead>
<tr>
<th>Agent</th>
<th>$x$ (m)</th>
<th>$y$ (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>450</td>
<td>150</td>
</tr>
<tr>
<td>B</td>
<td>150</td>
<td>700</td>
</tr>
<tr>
<td>C</td>
<td>750</td>
<td>700</td>
</tr>
</tbody>
</table>
Table 5: Connectivity and propagation numbers for in-water tests. For both tests we see an improvement in connectivity with the addition of the ASV.

<table>
<thead>
<tr>
<th></th>
<th>$N_C$</th>
<th>$N_P$</th>
<th>Percentage Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>With ASV</td>
<td>Without ASV</td>
<td>With ASV</td>
<td>Without ASV</td>
</tr>
<tr>
<td>Test 1</td>
<td>55.0</td>
<td>60.8</td>
<td>5.8</td>
</tr>
<tr>
<td>Test 2</td>
<td>11.4</td>
<td>26.2</td>
<td>14.8</td>
</tr>
</tbody>
</table>