

# Deployment Protocol for Underwater Wireless Sensors Network based on Virtual Force

Abeer Almutairi and Saoucene Mahfoudh  
Faculty of Computing and Information Technology  
King Abdullziz University Jeddah, KSA

**Abstract**—Recently, Underwater Sensor Networks (UWSNs) have attracted researchers' attention due to the challenges and the peculiar characteristics of the underwater environment. The initial random deployment of UWSN where sensors are scattered over the area via planes or ships is inefficient, and it doesn't achieve full coverage nor maintain network connectivity. Moreover, energy efficiency in underwater networks is a crucial issue since nodes utilize battery power as a source of energy and it is difficult and sometimes impossible to change or replenish these batteries. Our contribution in this research is to improve the performance of UWSNs by designing UW-DVFA, an underwater 3-D self-distributed deployment algorithm based on virtual forces. The main target for this work is to stretch the randomly deployed network in the 3-D area in a way that guarantees full area coverage and network connectivity.

**Keywords**—Deployment algorithm; underwater wireless sensor network; virtual force; coverage; connectivity

## I. INTRODUCTION

Nowadays, the growing interests toward exploring and monitoring oceans due to the broad application prospects, such as natural disaster detection and prevention, oil and mineral discovery and extraction, and military surveillance, have drawn the attention toward Underwater Wireless Sensors Networks (UWSNs) [1]. Unlike terrestrial sensor networks that rely on radio signal communication channels with high bandwidth and low propagation delay, UWSNs employ acoustic signals for underwater hop by hop node communication. The reason behind the designing of this stand is the peculiar characteristic of the underwater environment which attenuates the radio signal and makes it inefficient to be directly applied in UWSN. However, employing acoustic signals into UWSN can lead to many challenges and difficulties such as the large propagation delay and limiting bandwidth with high transmission loss. Furthermore, the inevitable movement that caused by the water current induces the Doppler Effect, which affects the signal intensity [2].

Therefore, researchers have become recently interested to explore other aspects of UWSNs such as the deployment of the network, sensors allocation and synchronization, power management and routing protocols. One fundamental task that is common to all these aspects is the network deployment which needs to be urgently addressed due to its direct impact on the other aspects.

In UWSNs, sensors must be deployed properly to collaborate with each other in three-dimensional space to perform monitoring and collecting information tasks according to the application requirements. Deployment of UWSN has attracted significant attention recently due to its tangible impact on

the other aspect of UWSN. The deployment in UWSN can be classified as static and distributed deployment [3]. In static deployment, nodes are deployed manually in predefined positions and cannot move once they are deployed. However, distributed deployment considered the mobility capability of sensors and assumed the initial deployment is randomly done using planes or ships to cover the desired area.

Sparse node distribution faces many challenges such as inefficient area coverage and lack of network connectivity. Moreover, reducing the deployment energy consumption to increase network lifetime is a major issue in UWSNs [4]. There are two aspects of energy consumption: one is related to the movement and communication between nodes during the deployment phase, the other is related to the difficulties raised from recharging the power of nodes. Hence, recharging underwater sensors can be time-consuming and expensive due to the need of imposing special underwater vehicles into the network for such purpose.

Different deployment algorithms such as clustering, connected tree, Practical swarm, Voronoi, and Virtual force are proven to success in the deployment of 2-D terrestrial sensor networks. Some of them are modified to be used in UWSN as we can see in the literature. Virtual Force is one of the known algorithms that is used in deploying wireless sensor network [5], [6]. It is based on the idea from the potential field and disk packing where sensors behave as a source of forces that are exerting among other. In this work, we will consider adopting this algorithm in UWSN.

The rest of the paper is organized as follows. Section II presents the state of the art related to UWSN deployment. In Section III, we present the basic underwater network architecture along with the models we used for coverage rate, acoustic propagation, and energy consumption. In Sections IV and V, we present the underwater distributed virtual force algorithm (UW-DVFA) and the performance evaluation of it, respectively. Finally, we conclude this paper with a summary of our contributions in Section VI.

## II. RELATED WORK

The fundamental task on UWSN deployment is to correctly locate each node in order to monitor the whole area. Different studies were aiming to find the best 3-D deployment techniques for UWSN. One technique is by filling a pre-calculated space with polyhedron and manually deployed sensors in each polyhedron for achieving 100% 3-D coverage and connectivity [8], [9]. For instance, Felamban et al. [9] deployed nodes in ways that form truncated octahedron with the objective

of minimizing the transmission loss under a given monitored volume and number of sensor nodes.

Another technique is presented in [10] where the deployment algorithm partitions the monitored volume into layers and clusters and then determines and selects the best cluster shape for manual deployment.

All these approaches are considered static and manual. They are centralized and require pre-knowledge about the monitored area which is not practical in an underwater environment where the water current and wind change the shape of the topology frequently. Moreover, the static manual deployment is not feasible due to accessibility and time constraints, especially in large monitoring volume. Therefore, distributed deployment where nodes can adjust their underwater positions through mobility after the random initial deployment is highly needed for most of the underwater applications.

Different studies were focused on how to stretch the randomly deployed 2-D network topology on the surface or at the bottom of the ocean in a distributed manner to form a 3-D network. According to the mobility of sensors, node distributed deployment techniques are falling into two main categories which are movement-assisted deployment [7] and sensor-self deployment [11]. In the former, the free mobility of special kind of sensors (Robots) is used to assist other sensor nodes in fulfilling the network requirement [12], [13], [14]. Despite its success in achieving coverage and connectivity, the use of these robots leads to very high network deployment cost and energy consumption.

On the contrary, the latter exploit the mobility of the sensor and can be classified into two categories: self-depth adjustment, and freely movement deployment. In self-depth adjustment, nodes are anchored to the ground or at the surface of the oceans and can adjust their depth by controlling the anchored wire and moving vertically along the z-axis in 3-D space [15], [16]. In freely movement deployment, nodes are attached to buoy that allow them to move freely in each direction by following deployment algorithm. In the following subsections, we present some research works related to the sensor-self deployment in UWSN.

#### A. Self-depth Adjustment Technique

Regarding the depth-adjustment deployment [18], [17], Akkaya et al. [19] propose an algorithm for adjusting the position of sensor nodes after their initial deployment to reduce coverage overlaps between neighbor nodes based on clustering and Graph Colouring. However, this algorithm create small coverage holes between each two nodes which affect the whole connectivity in the topology.

Two other approaches for adjusting nodes position that are randomly deployed on the surface of the ocean are presented in [20], [21]. Both approaches implementing Voronoi Diagram to adjust the nodes position based on the density of sensor nodes. Although this algorithm is operated in a centralized manner, the authors claim that the algorithm can be applied in a distributed way. However, in this case, each node needs to know all the location of all nodes in the network which imply a very high cost in term of message control transferring.

Senel et al. [22] propose a node position adjustment algorithm based on the connected dominating set (CDS) where the nodes are deployed initially on the surface of the ocean. The algorithm first determines the CDS in the 2-D network and establishes a connected backbone of nodes. In this algorithm, the number of iterations increases when the number of nodes increases, thus the complexity of the algorithm increases considerably. Hence, the deployment duration may be significantly high.

All the above techniques restrict the movement of the sensors by adjusting the depth vertically. However, in this work, we will focus on the self-deployment strategies that take into consideration the mobility of the sensor nodes.

#### B. Freely Movement Deployment Technique

Many techniques exploit the free mobility of sensor in all direction. For instance, Xia et al. [23] present a particle swarm inspired sensor self-deployment algorithm which simulates the flying behavior of bird flocking or fish schooling. The introduced algorithm aims to cover the area with high density of events and make the distribution of sensors (particles) match that events. Indeed, a group of random particles fly across the region and search for the optimal position. In case an event is in the sensing range of a particle, the particle will obtain the event location and send its location information to the nearest particles.

Similar to this approach Feng et al. [24] presents the underwater sensor network redeployment algorithm based on wolf search (RAWS) to obtain a fair underwater network coverage. The idea of the algorithm is based on the simulation of preying and escaping from predators. Indeed, the authors invoke the wolf search algorithm where wolves exhibit three typical behaviors: preying initiatively, preying passively, and escaping.

Thus, after the initial deployment, the coverage is initially calculated. Then, the first iteration of the algorithm begins such that each node will follow one of three scenarios. The first one is Active Coverage, when node  $s(i)$  detect a target coverage point  $p(i)$ , that fall in the sensing range of that node, the node  $s(i)$  will move in the same direction of the coverage point  $p(i)$  and maintain its position. The second case is the Passive Coverage, when node  $s(i)$  does not have any coverage point within its sensing range. In such case, if the node  $s(i)$  have a one-hop neighbor  $s(j)$ , and the neighbor node  $s(j)$  have more point to cover, then the node  $s(i)$  will move to the direction of that neighbor node. Otherwise, the node  $s(i)$  will move in any direction. The last one is Escape Mechanism, where in case of the existence of obstacles, the node moves in a random direction with a distance larger than it sensing radius to avoid those obstacles. The algorithm continues the iterations until the coverage is achieved. In this algorithm, each wolf has an independent search capability, thus increasing the diversity of the search space.

These two approaches were mainly focused on achieving the needed coverage despite the connectivity. Also, it suffer from computational load.

Jiang et al. [25] were concerned about achieving best connectivity and they present a Guaranteed Full Connectivity

Node Deployment algorithm where nodes are divided into two types either coverage nodes or connectivity nodes. Consequently, the coverage nodes used first to guaranty fully area coverage using the greedy iterative strategy. Then, the connectivity nodes used to improve the network connectivity in an area with disconnection.

To summarize, the primary target of the underwater self deployment algorithms is to maximize the sensors area coverage while maintaining the connectivity given a limited number of the sensors by spreading nodes in the whole area following a specific mathematical approach.

The virtual force (VF) algorithm is one of the algorithms that is used in terrestrial sensor network that utilize the concept of Artificial Potential Field. The virtual force (VF) algorithm is proven to have a faster convergence compared to the other algorithms in WSN and node deployment. In the [27], the authors introduce the 3-D VF, an extended version of the known virtual force algorithm, to be used on 3-D areas. However, with the complication of the underwater environment, it needs to be modified in order to be effectively applied in oceans. In this work, we will adopt the existing algorithm to create underwater distributed virtual force algorithm UW-DVFA.

### III. NETWORK ARCHITECTURE AND MODELES

#### A. Network Architecture

UWSN consists of a large number of underwater sensors that are capable of forwarding and receiving acoustic waves. As shown in Fig. 1, these sensor nodes detect and collect ambient data over a 3-D space and transmitting it to surface sink nodes via a multi-hop path. The surface sink nodes offload the collected data to an on-shore station through radio communication.

The initial deployment of these sensor nodes is generally random. Our target is to cover all the monitoring area by the needed number of sensor nodes following our redeployment algorithm while maintaining the connectivity toward the sink node.

#### B. Underwater Acoustic Propagation Model

The acoustic propagation in the underwater environment affects the transmission from node to node propagated through the ocean at the sound speed (1500 m/s). In this section, we will present the underwater attenuation, noise, and the related signal to noise ratio that we utilized in our algorithm.

1) *Underwater attenuation*: The attenuation (dB/m) modeled as the signal loss over a distance ( $d$ ) in meter that associated with frequency ( $f$ ) in kilohertz as:

$$10 \log_{10} \left( \frac{A(d, f)}{A_0} \right) = k 10 \log_{10} d + \frac{d}{10^3} 10 \log_{10} \alpha(f) \quad (1)$$

Where  $A_0$  is normalizing constant,  $K$  is the spreading factor ( $k = 1$ : Cylindrical,  $k = 1.5$ : Practical,  $k = 2$ : Spherical) and  $\alpha(f)$  is the Absorption Coefficient which is derived by Thorp Formula [28] as:

$$10 \log_{10} \alpha(f) = 0.11 \frac{f^2}{1 + f^2} + 44 \frac{f^2}{4100 + f^2} + 2.75 \times 10^{-4} f^2 + 0.003 \quad (2)$$

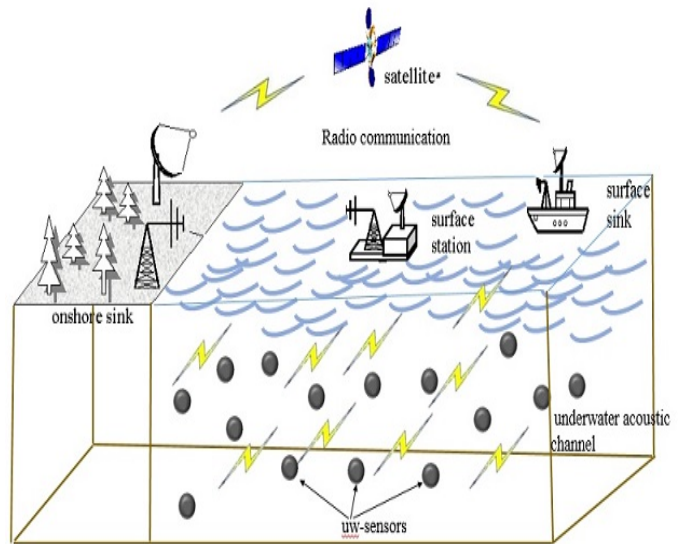


Fig. 1. UWSN architecture.

2) *Noise and signal to noise ratio*: There are four main sources of noise in the oceans which are the turbulence, shipping, wind driven waves and thermal noise that influences different frequency region. The power spectral density (p.s.d.) can be expressed as the following formula:

$$10 \log_{10} N(f) = \eta_0 - 18 \log_{10} f \quad (3)$$

Where  $f$  is the frequency in kilohertz. The constant level  $\eta_0$  is adjusting according to the specific deployment site.

The Signal to noise ratio can be expressed as a function of both attenuation and noise as:

$$SNR(d, f) = \frac{S(f)\Delta f}{N(f)\Delta f} = \frac{S(f)}{A(d, f)N(f)} \quad (4)$$

Where  $S(f)$  is the p.s.d of the transmitted signal.  $\Delta f$  is the narrow frequency around  $f$ .

#### C. Energy Model

There are many causes of energy depletion for the sensors in the UWSN such as sensing, processing, communication, overhearing, etc. Moreover, the mobility of sensors in the underwater area is one of the principal sources of energy consumption. The influence of some of these sources is considered smaller than the other. Indeed, the communication between nodes and the mobility of sensor nodes consider the primary sources of energy depletion. Thus, in this paper, we only consider studying the energy consumption that is due to data transmission and reception and the movement during the deployment phase while neglecting the others.

Therefore, the energy spent in transmitting one packet of length  $P_l$  bits over a distance  $d$  is given by

$$E_{tx} = P_{tx} \times T_{tx} \quad (5)$$

Where  $P_{tx}$  is the transmission power, and  $T_{tx}$  denotes the transmission time that given by

$$T_{tx} = \frac{P_l}{r} \quad (6)$$

Where  $r$  is the transmission rate of the information packet.

Correspondingly, the energy spent in receiving one  $P_l$  bits packet is given by

$$E_{rx} = P_{rx} \times T_{tx} \quad (7)$$

Where  $P_{rx}$  is the electronics power of the reception, and  $T_{tx}$  denotes the transmission time.

The energy consumed due to the movement by distance  $d$  is given by

$$E_m = d \times e_{mu}(d) \quad (8)$$

Where  $e_{mu}$  is the energy consumption per movement distance.

Thus, the total energy consumed can be calculated by:

$$E_{total} = E_{tx} + E_{rx} + E_m \quad (9)$$

Where  $E_{tx}$  and  $E_{rx}$  depend on the amount of data transmitted and received, respectively.  $E_m$  depends on the distance traveled by the sensor during its lifetime. Hence, the number of messages sent and received by the algorithm reflect the energy consumed by the transmission and reception of the packets. Moreover, the distance traveled during the deployment algorithm reflect the energy consumed by the movement of the sensors.

#### D. Coverage and Connectivity Rate

Considering a specific volume of water described as  $a$ ,  $b$  and  $c$  area of the interested field and  $N$  number of randomly deployed sensors in the given area. This specific area to be covered is called the region of interest. The coverage problem can be described as the positioning of nodes such that their sensing zones all together cover the region of interest.

More precisely, each sensor has a sensing range  $R_s$  and transmission range  $R_t$ . For each sensor  $S_i$  who deployed in  $(x_i, y_i, z_i)$ , and for each point  $P_i$  with the coordinate  $(x, y, z)$ , we can obtain the coverage  $C_{P_i}(S_i)$  of the  $P_i$  by the sensor  $S_i$  following the binary equation:

$$C_{P_i}(S_i) = \begin{cases} 1, & \text{if } \exists S_i \mid d(S_i, P_i) < R_s \\ 0, & \text{if otherwise} \end{cases}$$

Where  $d(S_i, P_i)$  is the Euclidean distance between  $S_i$  and  $P_i$ . Thus, when the coverage is equal to 1, it denotes a full coverage for this point.

Moreover, it is proven that when the following constraint  $R_t \geq \sqrt{3}R_s$  is applied, the connectivity will be implied if the coverage is achieved. Thus we will focus on achieving the full area coverage following the above model.

## IV. UNDERWATER DISTRIBUTED VIRTUAL FORCE ALGORITHM

### A. Assumptions

This work is based on the following assumptions:

- Sensor nodes have the ability to change their position by moving freely in each direction of the 3-D space.
- Each sensor node knows its position by using one of the localization techniques such as the one presented in [26].
- Each node has spherical sensing range ' $R_s$ ' and transmission range ' $R_t$ ' where  $R_t \geq \sqrt{3}R_s$ .
- The first deployment is randomly done on the surface of the area either by planes or ships.

### B. Virtual Force Algorithm Principle

The main principle behind this algorithm is to consider each sensor node as a source of forces that are exerted among its neighboring nodes. Hence, for each sensor  $S_i$ , the algorithm will calculate the force  $F_{ij}$  that is exerted by the neighbor sensor  $S_j$ . These forces are either attractive force or repulsive force depends on the distance between them. If the two neighbor nodes are too close to each other the repulsive force will be exerted to avoid node stacking. On the other hand, if the two neighbor nodes are far from each other, an attractive force will be exerted to maintain the complete homogeneity in the deployment and avoid coverage hole.

The main target of the algorithm is to reach the pre-determined distance threshold ( $D_{th}$ ) between all neighboring nodes after numbers of iteration where in each iteration the nodes are moving to different locations according to the applied forces thus all nodes are equidistant.

Fig. 2 illustrates an example of these forces exerted on 4 nodes  $S1$ ,  $S2$ ,  $S3$  and  $S4$ . The force  $F_{13}$ , exerted by  $S1$  on  $S3$ , is an attractive force because the distance between these two sensors is bigger than  $D_{th}$ . However, the force  $F_{12}$  is a repulsive force since the distance between sensor  $S1$  and  $S2$  is smaller than  $D_{th}$ . The distance between  $S1$  and  $S4$  is equal to  $D_{th}$  hence the force  $F_{14}$  is null.

### C. UW-DVF Algorithm

In order to extend the DVFA principle to be used in UWSN, some of the roles must be carefully tuned. First, most of the UWSN applications in the underwater environment requires the 3-D area monitoring and coverage. Hence, the distance threshold of the virtual force needed to be changed to harmonize with the area. Second, the existence of the water current force affect the deployment of the network in many aspects and need to be addressed. In the following subsections, we will explain the modification we applied regarding these issues.

1) *Distance threshold*: One of the significant roles in DVFA is choosing the  $D_{th}$ . In flat 2-D areas, the value of  $D_{th}$  depends on triangular tessellation where each neighbor is located at one of the vertexes of the triangular. Such arrangement is not accurate when it comes to 3-D underwater environment.

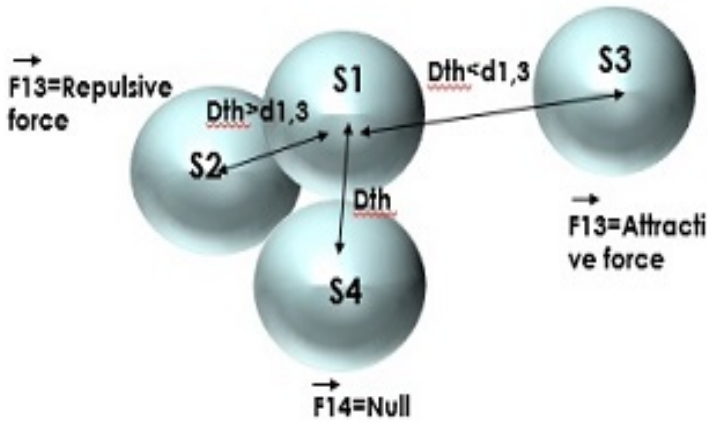


Fig. 2. An example of virtual forces exerted between nodes.

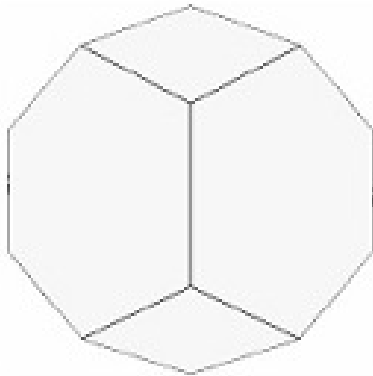


Fig. 3. A truncated octahedron.

According to [8], the best placement strategy for obtaining the optimal deployment in the 3-D environment is by filling the monitoring area with nodes to form truncated octahedron tessellation.

As illustrated in Fig. 3, the truncated octahedron has 14 faces, 8 of these faces are hexagonal, and 6 are square. Therefore, each node will have 14 neighbor nodes located at the faces of the truncated octahedron.

It means that nodes are not equidistance. Thus, if the node is located at one of the hexagonal faces, the  $D_{th}$  value will be expressed as  $\sqrt{6}a$  where  $a$  is the edge length. However, if the neighbor node is located at one of the square faces, the  $D_{th}$  value will be expressed as  $2\sqrt{2}a$ .

However, in virtual force, the distance threshold must be a unique value for all the nodes neighbor. Thus, the truncated octahedron cannot be adopted. To overcome this issue, a regular dodecahedron is introduced in [27] to optimize the deployment in 3-D area instead of the truncated octahedron. As shown in Fig. 4, the node will be located at the center of the dodecahedron, and it will have 12 neighbors with the same distance. Therefore, the  $D_{th}$  will be equal to:

$$D_{th} = a\sqrt{\frac{5}{2} + \frac{11}{5}\sqrt{5}} \quad (10)$$

Where  $a$  is the edge length of a regular dodecahedron that can be calculated depends on the sensing range ( $R_s$ ) for each node as:

$$a = \frac{4R_s}{\sqrt{3}(1 + \sqrt{5})} \quad (11)$$

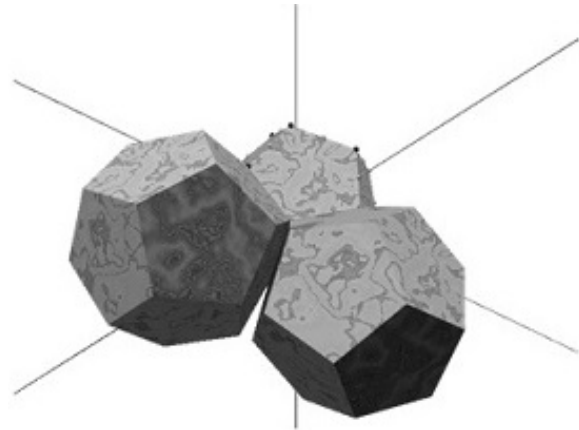


Fig. 4. Regular dodecahedron tessellation.

2) *Water current force*: The continued movement of the water current in the underwater environment have a crucial impact on the deployment process and hence the whole efficiency of the network. Indeed, the existence of water currents considers as an external force that is exerted on the sensors by the fluid media. Thus, we consider the effect of water current force in our algorithm.

According to the oceanography, oceans are stratifying with rotating characteristic. Hence the vertical movement can be negligible with respect to the horizontal one [29]. Thus, in our work, we consider the horizontal displacement of the water currents.

Moreover, the velocity of the currents is varying in each depth. Indeed, it keeps decreasing while we move down where it tends to be quiet in the deep of the oceans [30]. For instance, Fig. 5 shows the velocity profile for one of the fastest ocean currents in the world across the Gulf Stream at the Straits of Florida and Cape Hatteras. The velocity profile indicates that the maximum velocity is near to the surface around the top 200 meters. Velocity decreases with depth and the flow below 1000 meters is usually less than 10 cm/s [31]. In our work, we take into consideration the variation of water current velocity based on the depth of the area.

Consequently, the Force of the water current  $F_{wc}$  will be introduced along with the other virtual forces that are exerted by the algorithm where the total forces exerted on sensor  $S_i$  is:

$$\vec{F}_i = \sum_j \vec{F}_{ij} + \vec{F}_{wc} \quad (12)$$

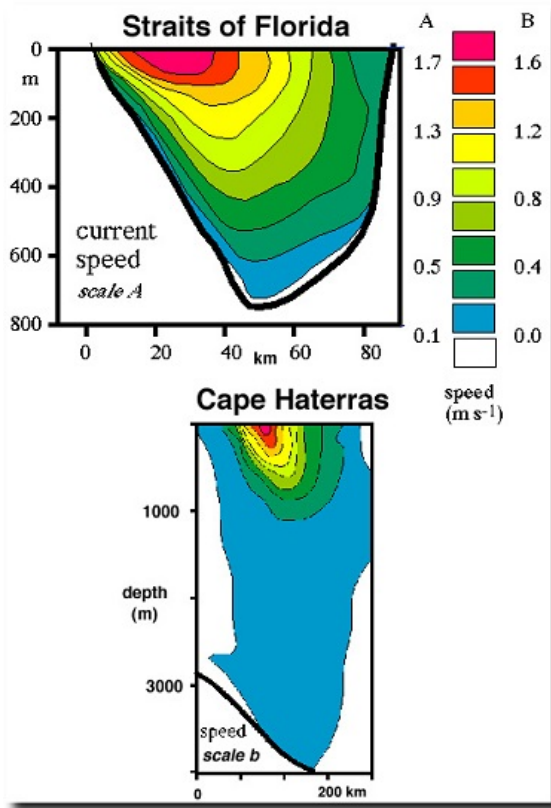


Fig. 5. The velocity of the water current in different depth.

Where,  $F_{wc}$  is the water current force and it will be varied depends on the z-axis and  $F_{ij}$  is the force exerted by the sensor  $S_j$  over the sensor  $S_i$ .

Thus, the UW-DVFA will initiate after the initial random node deployment, where each sensor node will send a Hello packet to all of its 1-hop neighbor that contain the position of the node to perform the initial neighborhood discovery. Then each node will calculate the Euclidean distance between its 1-hop neighbors such that for sensor nodes  $S_i$  and  $S_j$  the distance ( $d_{ij}$ ) will be giving by the following formula:

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \quad (13)$$

After that, each sensor will compare the  $d_{ij}$  with the  $D_{th}$  which leads to three possible results:

- $d_{ij} > D_{th}$

In this case, the node  $S_j$  will exert an attractive force on  $S_i$  to reduce the distance between them and the force will be calculated according to the following formula:

$$\vec{F}_{ij} = K_a \cdot (d_{ij} - D_{th}) \cdot \frac{(x_j - x_i), (y_j - y_i), (z_j - z_i)}{d_{ij}} \quad (14)$$

Where  $K_a$  is the attractive coefficient.

- $d_{ij} < D_{th}$

In this case, the node  $S_j$  will exert a repulsive force on  $S_i$  to increase the distance between them and the force will be calculated according to the following formula:

$$\vec{F}_{ij} = K_r \cdot (D_{th} - d_{ij}) \cdot \frac{(x_j - x_i), (y_j - y_i), (z_j - z_i)}{d_{ij}} \quad (15)$$

Where  $K_r$  is the repulsive coefficient

- $d_{ij} = D_{th}$

In this case, no force will be exerted between these two nodes. Hence there will be no movement.

Therefore, the total force on  $S_i$  will be calculated with these formulas:

$$\vec{F}_{ix} = \sum_j \vec{F}_{ijx} + F_{wcx} \quad (16)$$

$$\vec{F}_{iy} = \sum_j \vec{F}_{ijy} + F_{wcy} \quad (17)$$

$$\vec{F}_{iz} = \sum_j \vec{F}_{ijz} \quad (18)$$

Noticing that the  $F_{wc}$  on the z axes is null. However, the variation of the depth on z-axis is considered.

Thus, the new position of the node  $S_i$  that is located originally at position  $(x_i, y_i, z_i)$  will depends on the resultant force on the node  $S_i$  such that  $(x_i = x_i + F_{ix}, y_i = y_i + F_{iy}, z_i = z_i + F_{iz})$

## V. PERFORMANCE EVALUATION

In order to evaluate the performance of UW-DVFA algorithm, we implemented the algorithm in Aqua-sim network simulator. It is a simulator based on the Network simulator NS2 to simulate the underwater environment.

### A. Simulation Parameter

We conduct 20 different scenarios where nodes are randomly scattered at the surface of the monitored area. The simulation parameter we used are given in Table I.

TABLE I. SIMULATION PARAMETERS

Topology	Area size Sensor nodes Node speed	3km x 3km x 1km 300, 400, 500 10m/s
MAC Layer	protocol Transmission range $R_t$ Sensing range $R_s$	Underwater Broadcast mac 500m 250m
Simulation	Result Simulation Time	Average of 20 simulation runs 1000s 1000s
UW-DVFA	Ka Kr Hello period Dth	0.004 0.25 20s 371.7
Water Current	Velocity in depth range (0-200 m) Velocity in depth range (200-400 m) Velocity in depth range (400-600 m) Velocity in depth range (600-800 m) Velocity in depth range (800-1000 m)	2m/s 1.7m/s 1.3m/s 0.9m/s 0.5m/s



### B. Evaluation Criteria

The main goals for UW-DVFA are to obtain the full area coverage while maintaining the connectivity between nodes and to reduce the energy consumed in term of total distance travelled and number of messages exchanged.

Fig. 6 shows an example of the initial deployment. The final result of the deployment algorithm is shown in Fig. 7. The following sub-sections highlight the evaluation results of UW-DVFA.

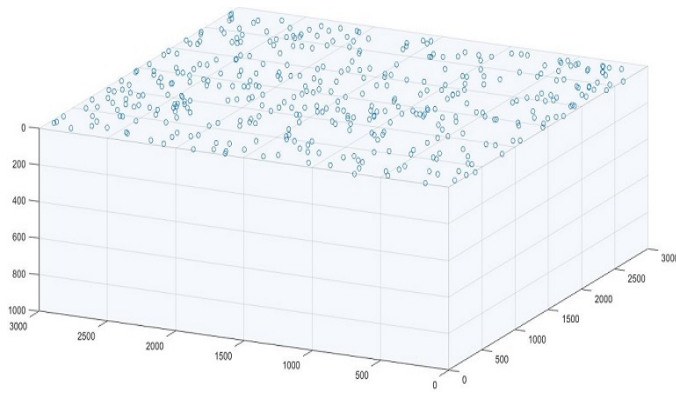


Fig. 6. Initial random deployment.

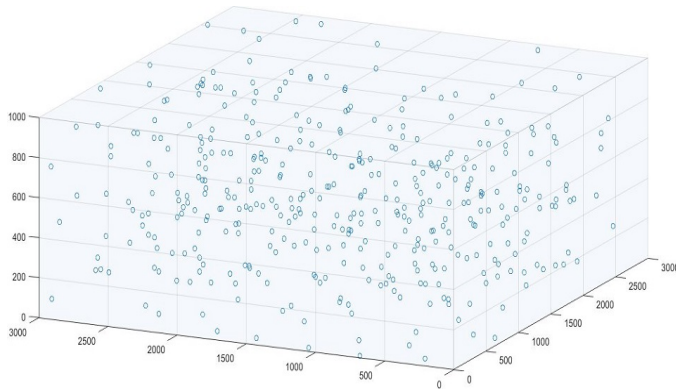


Fig. 7. Final deployment.

1) *Coverage and connectivity rate:* We conduct different simulation with a different number of nodes to evaluate the performance of the algorithm. First, we estimate the number of sensor nodes needed to cover all the area with length  $l$ , width  $w$  and height  $h$  based on the formulas given in [27]. These formulas don't provide the exact number of sensors needed to cover the monitoring area but give a lower bound and an upper bound of this number.

$$\begin{aligned} \text{Lowerbound} &= \frac{\text{Volume of the area}}{\text{Volume of the sphere}} \\ &= \frac{h \times l \times w}{\frac{4}{3}\pi(R_s)^3} \quad (19) \end{aligned}$$

Based on this formula, the lower bound for the area we were chosen is 138 nodes. Regarding the upper bound, the border of the area is taken into account: it is 6 faces and 12

edges.

$$\begin{aligned} \text{Upperbound} &= \frac{h \times l \times w}{\frac{4}{3}\pi(R_s)^3} \\ &+ 2 \frac{h \times l}{\pi(R_s)^2} + 2 \frac{h \times w}{\pi(R_s)^2} \\ &+ 2 \frac{l \times w}{\pi(R_s)^2} + 4 \frac{h}{R_s} + 4 \frac{l}{R_s} + 4 \frac{w}{R_s} \quad (20) \end{aligned}$$

Based on this formula, the upper bound for the area we were chosen is 404 nodes. In our simulation, we evaluate the network with a number of nodes around the upper bound using 300,400 and 500 nodes. Fig. 8 illustrates the coverage rate using three different number of nodes (300,400, and 500) for 20 different random deployment scenarios. As we can see, the UW-DVFA reaches 100% coverage rate in case of 400 and 500 nodes. The full coverage is reached within 250 seconds with 500 nodes for deployment while it takes 350 seconds with 400 nodes. This means that the deployment achieved the full coverage faster if the number of nodes is high. The coverage reaches 93% in case of 300 nodes. Fig. 9 illustrates a very important characteristic of UW-DVFA. In fact, the deployment algorithm maintains the initial connectivity and this connectivity between nodes is never lost during the deployment process.

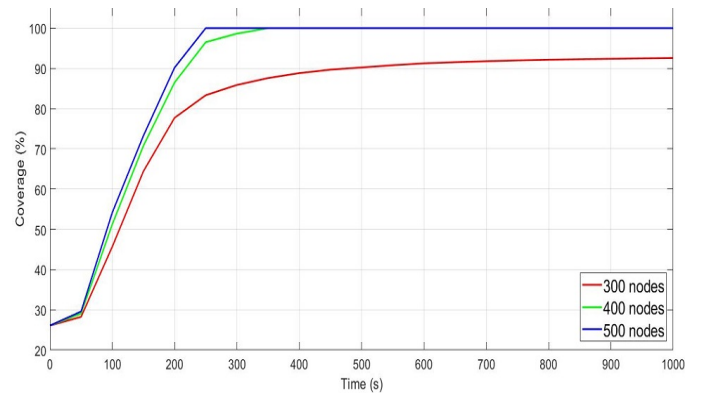


Fig. 8. Coverage rate as function of time.

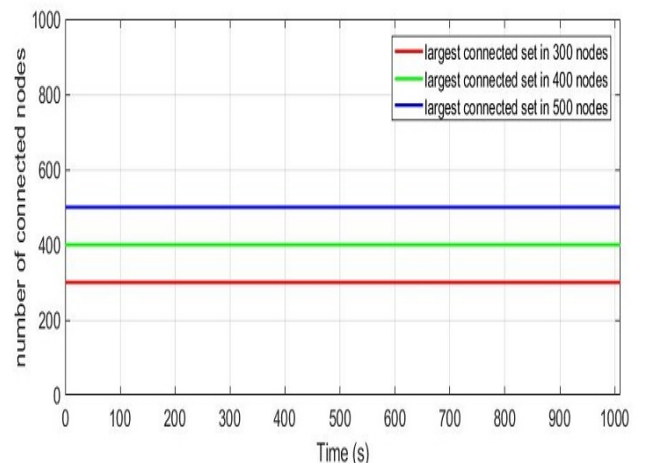


Fig. 9. Largest connected set as function of time.

2) *Distance travelled by nodes:* Fig. 10 presents the average, maximum and minimum distance traveled by nodes as a function of time. The values shown in the figure is the cumulative distance. As we can observe, the distance increased rapidly until the time around 300 seconds, where it reached the full coverage. After that, the distance continues growing slowly. We can notice that the distance traveled by nodes (maximum, minimum and average) increases if the number of the node increases. Hence, the energy consumed by node traveling to cover the area in the network with 500 nodes is higher than the energy consumed by nodes in a network with 400 nodes. As we see before, both cases reached the 100% coverage. Thus, in the rest of simulations, we only concentrate on the deployment with 400 nodes to evaluate the UW-DVFA overhead.

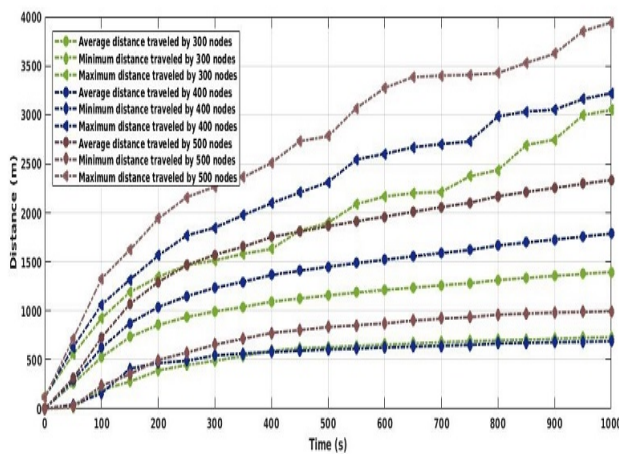


Fig. 10. The average, maximum and minimum distance traveled as function of time.

3) *Number of messages sent and received:* In order to compute the virtual forces, each node should send Hello messages periodically for neighborhood discovery. These messages permit to determine the 1-hop neighbors for each node. We notice that the performance of our deployment algorithm depends highly on the used Hello period. The reason behind this is the high propagation delay in the underwater environment due to the employment of acoustic signal. In the following simulation, we evaluate the effect of choosing different Hello period on the performance of our deployment algorithm. Fig. 11 and 12, show the number of messages sent and received with a variation of Hello period. As expected, the number of messages sent and received increase when the Hello period decreases. On the other hand, as shown in Fig. 13 the coverage rate is very low when using 5 and 10 seconds Hello period. This observation is logical due to the high propagation delay in the acoustic underwater environment. Hence, the information about the neighborhood of each node which the computation of virtual forces is not accurate. Nevertheless, in case of using 15 and 20 seconds of Hello period, the coverage rate reaches 100% in 450 and 350 seconds, respectively. Therefore, the best result is obtained when Hello period is 20 seconds since the deployment algorithm achieves the full coverage rate in shortest time with minimum overhead.

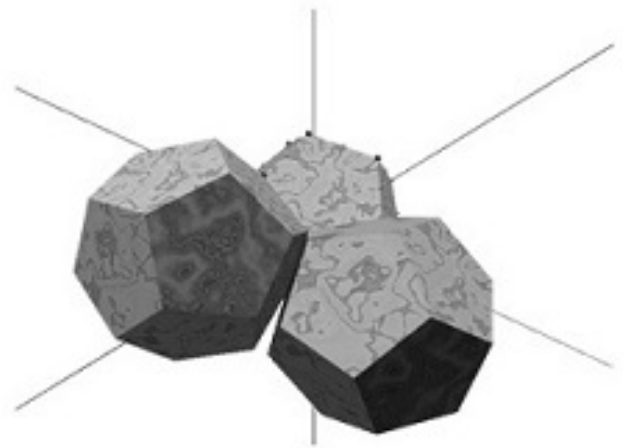


Fig. 11. Number of messages transmitted with different Hello interval.

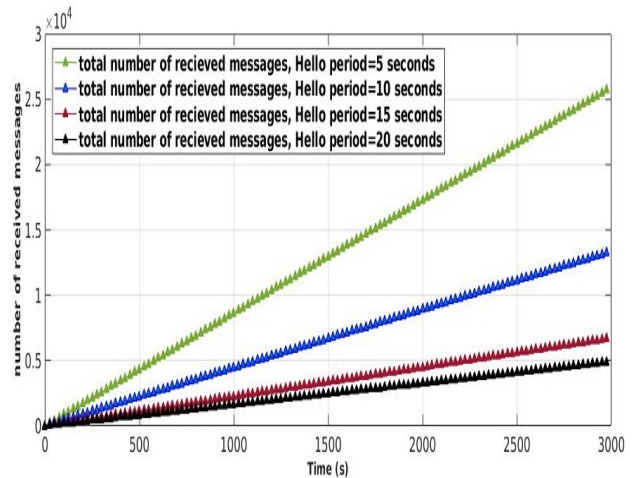


Fig. 12. Number of messages received with different Hello interval.

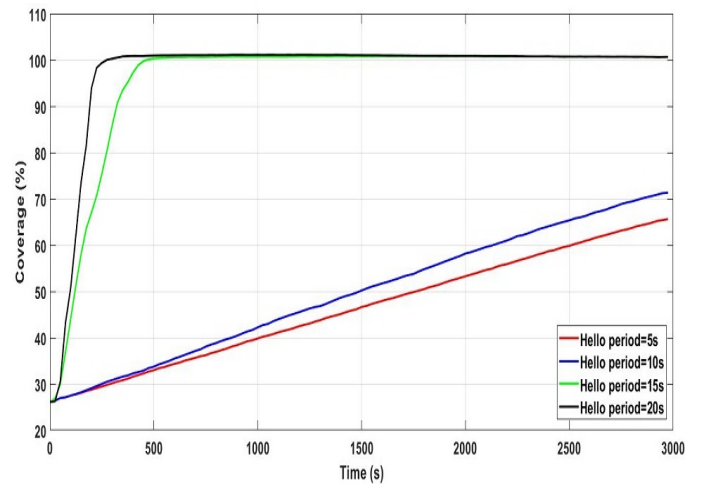


Fig. 13. Coverage rate with different hello interval.

## VI. CONCLUSION

The existence of the Underwater Sensor Networks (UWSNs) applications and the interest in exploring and moni-



toring oceans motivate us into the field of UWSN deployment. In this paper, we have adapted the distributed virtual force algorithm into the underwater environment by taking into consideration the effect of the water current force. Our main target is to reach the full coverage of the region of interest with a reasonable number of nodes while maintaining the network connectivity. We showed that the full coverage is reached faster with a number of nodes around the maximum bound. Moreover, the network connectivity is never lost the entire deployment time. We strive to investigate the energy consumed during the deployment in term of the number of messages sent and received, and the distance traveled by nodes. In future work, we will show how the deployment algorithm can increase the network lifetime by covering coverage holes by replacing battery depleted nodes.

#### ACKNOWLEDGMENT

This research was supported by King Abdulaziz City for Science and Technology (KACST). The authors would like to show gratitude to all those who support this research.

#### REFERENCES

- [1] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: applications, advances and challenges," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 370, pp. 158-175, 2012.
- [2] R. Manjula and S. S. Manvi, "Issues in underwater acoustic sensor networks," *International Journal of Computer and Electrical Engineering*, vol. 3, p. 101, 2011.
- [3] G. Han, C. Zhang, L. Shu, N. Sun, and Q. Li, "A survey on deployment algorithms in underwater acoustic sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.
- [4] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Computer Networks*, vol. 67, pp. 104-122, 2014.
- [5] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces," in *IEEE INFOCOM*, 2003, pp. 1293-1303.
- [6] K. Mougou, S. Mahfoudh, P. Minet, and A. Laouiti, "Redeployment of randomly deployed wireless mobile sensor nodes," in *Vehicular Technology Conference (VTC Fall)*, 2012 IEEE, 2012, pp. 1-5.
- [7] W. Hui, L. Meiqin, and Z. Senlin, "An efficient depth-adjustment deployment scheme for underwater wireless sensor networks," in *Control Conference (CCC)*, 2015 34th Chinese, 2015, pp. 7771-7776.
- [8] S. Alam and Z. J. Haas, "Coverage and connectivity in three-dimensional networks," in *Proceedings of the 12th annual international conference on Mobile computing and networking*, 2006, pp. 346-357.
- [9] M. Felamban, B. Shihada, and K. Jamshaid, "Optimal node placement in underwater wireless sensor networks," in *Advanced Information Networking and Applications (AINA)*, 2013 IEEE 27th International Conference on, 2013, pp. 492-499.
- [10] L. Liu, "A deployment algorithm for underwater sensor networks in ocean environment," *Journal of Circuits, Systems, and Computers*, vol. 20, pp. 1051-1066, 2011.
- [11] F. Senel, K. Akkaya, M. Erol-Kantarci, and T. Yilmaz, "Self-deployment of mobile underwater acoustic sensor networks for maximized coverage and guaranteed connectivity," *Ad Hoc Networks*, vol. 34, pp. 170-183, 2015.
- [12] B. Liu, F. Ren, C. Lin, Y. Yang, R. Zeng, and H. Wen, "The redeployment issue in underwater sensor networks," in *Global Telecommunications Conference*, 2008. IEEE GLOBECOM 2008. IEEE, 2008, pp. 1-6.
- [13] W. K. Seah, H.-X. Tan, Z. Liu, and M. H. Ang Jr, "Multiple-UUV approach for enhancing connectivity in underwater ad-hoc sensor networks," in *OCEANS*, 2005. Proceedings of MTS/IEEE, 2005, pp. 2263-2268.
- [14] G. A. Hollinger, U. Mitra, and G. S. Sukhatme, "Autonomous data collection from underwater sensor networks using acoustic communication," in *Intelligent Robots and Systems (IROS)*, 2011 IEEE/RSJ International Conference on, 2011, pp. 3564-3570.
- [15] V. Bokser, C. Oberg, G. Sukhatme, and A. Requicha, "A small submarine robot for experiments in underwater sensor networks," *Center for Embedded Network Sensing*, 2004.
- [16] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," *Network*, IEEE, vol. 20, pp. 12-18, 2006.
- [17] P. Jiang, X. Wang, and L. Jiang, "Node Deployment Algorithm Based on Connected Tree for Underwater Sensor Networks," *Sensors*, vol. 15, pp. 16763-16785, 2015.
- [18] P. Jiang, Y. Xu, and F. Wu, "Node Self-Deployment Algorithm Based on an Uneven Cluster with Radius Adjusting for Underwater Sensor Networks," *Sensors*, vol. 16, p. 98, 2016.
- [19] K. Akkaya and A. Newell, "Self-deployment of sensors for maximized coverage in underwater acoustic sensor networks," *Computer Communications*, vol. 32, pp. 1233-1244, 2009.
- [20] J. Wu, Y. Wang, and L. Liu, "A Voronoi-Based Depth-Adjustment Scheme for Underwater Wireless Sensor Networks," *International Journal on Smart Sensing And Intelligent Systems*, vol. 6, pp. 244-258, 2013.
- [21] W. Hui, L. Meiqin, and Z. Senlin, "An efficient depth-adjustment deployment scheme for underwater wireless sensor networks," in *Control Conference (CCC)*, 2015 34th Chinese, 2015, pp. 7771-7776.
- [22] F. Senel, K. Akkaya, and T. Yilmaz, "Autonomous deployment of sensors for maximized coverage and guaranteed connectivity in underwater acoustic sensor networks," in *Local Computer Networks (LCN)*, 2013 IEEE 38th Conference on, 2013, pp. 211-218.
- [23] H. Du, N. Xia, and R. Zheng, "Particle swarm inspired underwater sensor self-deployment," *Sensors*, vol. 14, pp. 15262-15281, 2014.
- [24] P. Jiang, Y. Feng, and F. Wu, "Underwater sensor network redeployment algorithm based on wolf search," *Sensors*, vol. 16, p. 1754, 2016.
- [25] P. Jiang, J. Liu, B. Ruan, L. Jiang, and F. Wu, "A new node deployment and location dispatch algorithm for underwater sensor networks," *Sensors*, vol. 16, p. 82, 2016.
- [26] H.-P. Tan, R. Diamant, W. K. Seah, and M. Waldmeyer, "A survey of techniques and challenges in underwater localization," *Ocean Engineering*, vol. 38, pp. 1663-1676, 2011.
- [27] N. Boufares, I. Khoufi, P. Minet, L. Saidane, and Y. B. Saied, "Three dimensional mobile wireless sensor networks redeployment based on virtual forces," in *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2015, pp. 563-568.
- [28] L. M. Brekhovskikh, Y. P. Lysanov, and R. T. Beyer, "Fundamentals of ocean acoustics," *The Journal of the Acoustical Society of America*, vol. 90, pp. 3382-3383, 1991.
- [29] A. Caruso, F. Paparella, L. F. M. Vieira, M. Erol, and M. Gerla, "The meandering current mobility model and its impact on underwater mobile sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications*. IEEE, 2008, pp. 221-225.
- [30] R. H. Stewart, "Introduction to physical oceanography: Robert H. Stewart", 2008.
- [31] "Gulf Stream Velocity", Gulf Stream Velocity. [Online]. Available: <https://marine.coastal.edu/gulfstream/p2a.htm>. [Accessed: 01-Nov-2017]