Simulation and Experimentation Platforms for Underwater Acoustic Sensor Networks: Advancements and Challenges

HANJIANG LUO, Zibo Vocational Institute, Shenzhen University
KAISHUN WU and RUKHSANA RUBY, Shenzhen University
FENG HONG and ZHONGWEN GUO, Ocean University of China
LIONEL M. NI, University of Macau

Ocean and water basically cover the major parts of our planet. To obtain the best utilization of the underlying resources on these parts of the Earth, people have made some research advancements. Specifically, the research on underwater wireless acoustic sensor networks (UWA-SNs) has made great progress. However, wide deployment of UWA-SNs is far from a reality due to several reasons. One important reason is that offshore deployment and field-level experiments of ocean-centric applications are both expensive and labor intensive. Other alternatives to attain this objective are to conduct simulation or experimentation that can reduce cost and accelerate the research activities and their outcomes. However, designing efficient and reliable simulation and experimentation platforms have proven to be more challenging beyond the expectation. In this article, we explore the main techniques (including their pros and cons) and components to develop simulation and experimentation platforms and provide a comprehensive survey report in this area. We classify simulation and experimentation platforms based on some typical criteria and then provide useful guidelines for researchers on choosing suitable platforms in accordance with their requirements. Finally, we address some open and un-resolved issues in this context and provide some suggestions on future research.

CCS Concepts: • **Computer systems organization** → **Embedded and cyber-physical systems**; Sensor networks; Real-time systems; • **Networks** → Network simulations; Network experimentation**;**

Additional Key Words and Phrases: Underwater acoustic sensor networks, experiment, simulators, testbeds

ACM Reference Format:
Hanjiang Luo, Kaishun Wu, Rukhsana Ruby, Feng Hong, Zhongwen Guo, and Lionel M. Ni. 2017. Simulation and experimentation platforms for underwater acoustic sensor networks: Advancements and challenges. ACM Comput. Surv. 50, 2, Article 28 (May 2017), 44 pages.
DOI: http://dx.doi.org/10.1145/3040990

1. INTRODUCTION

Unlike terrestrial wireless sensor networks (WSNs), underwater wireless acoustic sensor networks (UWA-SNs) have many unique characteristics [2, 78, 108, 114, 122].

This research was supported in part by the China NSFC Grants 61472259,61379127,61379128, Shenzhen Science and Technology Foundation (No. KQCX20150324160536457), Guangdong Talent Project 2014TQ01X238, 2015TX01X111 and GDUPS (2015), the Project of Shandong Province Higher Educational Science and Technology Program No. J11LG88, Macao FDCT Grant 149/2016/A and the University of Macau Grant SRG2015-00050-FST.
Authors' addresses: H. Luo, Zibo Vocational Institute, Shenzhen University, Liantong Rd, Zibo, ShanDong, 255314, China; email: luo.wireless@gmail.com; K. Wu (corresponding author) and R. Ruby, Shenzhen University, 3688 Nanhai Ave, Nanshan, Shenzhen, Guangdong, 518060, China; emails: {wu, ruby}@szu.edu.cn; Z. Guo and F. Hong, Ocean University of China, 238 Songling Rd, Laoshan, Qingdao, 266100, China; emails: {guozhw, hongfeng}@ouc.edu.cn; L. M. Ni, University of Macau, Avenida da Universidade, Taipa, Macau, China; email: ni@umac.mo.

DOI: http://dx.doi.org/10.1145/3040990

Consequently, the issues involved in UWA-SNs have been an active research topic since 2000 [22, 35, 58]. However, there are still many challenges in this area that need to be solved before a wide deployment of such networks. One such challenge lies in developing new and well-designed simulation or experimentation platforms while considering suitable and cost-effective software and hardware components, simulation tools, protocol frameworks, sensors, and so on [67, 82]. Experimentation platforms in particular are more crucial while conducting research in UWA-SNs. This is because by conducting simulations, emulations, or experimentations with these platforms, researchers can validate, debug, and test underwater protocols with low cost, and they can shorten the turnaround period of protocol design at the same time. Until now, researchers have developed many experimentation platforms for terrestrial WSNs. However, the entities in UWA-SNs communicate over temporally and spatially varying acoustic channels that have totally different characteristics compared with terrestrial radio communication channels in terms of path loss, multipath transmission, propagation delay, Doppler spread, noise, and so on [2, 45]. Therefore, experimentation platforms developed for terrestrial WSNs cannot be directly used for UWA-SNs.

To design suitable simulation tools and real testbeds[1] for UWA-SNs, researchers encountered more challenges than they expected at the beginning. First, modeling time varying and frequency-selective underwater acoustic channels precisely is challenging [105], which makes it even harder to design both accurate and reliable simulation tools or experimental testbeds. To deal with ever-changing underwater communication channels, acoustic modems are a critical part of UWA-SNs as these are used in designing and implementing self-adaptive protocols in accordance with time-varying environments to improve experimentation accuracy, flexibility, and efficiency. Second, it is very expensive to set up experimental testbeds for UWA-SNs compared to that for terrestrial WSNs in terms of deployment, hardware, and maintenance costs. Third, due to the difficulties and less effort made in this direction of research, there are only a limited number of practical experimental testbeds developed to evaluate communication and networking protocols. Fourth, evaluating and comparing the performance of different simulation or experimentation platforms in real-world scenarios effectively is not only difficult but also an open problem.

Even though the research on building simulation and experimentation platforms is difficult, the emerging potential applications and the increasing research interests have motivated researchers to design and develop new affordable, user-friendly simulation tools and field-level experimental testbeds. Consequently, these advancements encourage rapid progression in UWA-SNs. Over the years, many simulation platforms and field-level testbeds have been developed that either use software-based components or more sophisticated hardware-based experimentation components. The underwater research community has also put a lot of effort into reducing the gap between simulation-based and real field-level deployments to ease the transition from the laboratory-level simulation and emulation to the real field-level experimentation.

In general, to simulate, evaluate, validate, or test underwater protocols, researchers currently rely on three techniques, that is, software-based simulation, hardware-based simulation, and real field-level experiments using testbeds [128]. In order to reduce the cost, a mathematically modeled physical layer is typically used in software-based simulation. Hardware-based simulation is also called emulation, where the physical layer models are substituted by real underwater acoustic modems. Performance evaluation using software-based simulation or hardware-based simulation are much less expensive compared to that with real field-level experimental testbeds. The reason is that evaluating the performance of certain systems via simulation is more convenient, as it

[1]An experimentation platform is often named a testbed throughout this article.

allows researchers to design different system scenarios by varying network topologies, traffic patterns, channel propagation, and so on. Moreover, simulation can be conducted repeatedly for large-scale networks at relatively low cost.

However, both software-based and hardware-based simulation techniques cannot be replaced by real field-level experimental testbeds, as these techniques have their own limitations. For example, simulation tools usually take a subset of real environmental variables that lead to an approximately simplified model for channels or environmental dynamics. Furthermore, these techniques sometimes take the hardware features into account, which significantly impairs the performance of the system [86]. On the other hand, real field-level experiments are typically conducted in real environments using real sensor nodes to test or verify designed protocols. Roughly, such experiments can be conducted on a small scale (e.g., in a water pool) or in real-world scenarios by deploying the testbeds in a lake, a river, or in the ocean over a long period of time in a large scale. Although each technique has its own pros and cons, the researchers may need all these different techniques (e.g., simulation, emulation, and real field-level experiment) during different design phases before deploying an application for real underwater sensor networks.

Several survey articles summarize many aspects, such as communication and time synchronization techniques [22, 46–48], channel behavior and dynamics [105], medium access control (MAC), routing and networking [8, 19, 24, 92], deployment and localization [18, 43], architecture and testing of underwater acoustic sensor networks [26, 44, 117]. In [22], the authors provided an overview of some key techniques in this context, such as the point-to-point underwater communication techniques and networking protocols. In [105], the authors presented a tutorial-type overview on channel propagation models and their statistical characteristics of such networks. The article [44] summarized the main challenges and approaches in designing the applications of UWA-SNs. The authors also discussed about some additional issues and components of such networks, such as hardware, simulation tools, dynamics of underwater environments, development of experimental testbeds, and so on. Recently, one article [24] came out that provided an overview of current ongoing research activities in UWA-SNs and summarized the gradual development of physical, MAC, and routing protocols of such networks. Though these survey articles have given valuable advices and suggestions on simulators and testbeds, they really did not focus on underwater testbeds, and, consequently, the discussion of these survey articles were not complete.

Until now, some researchers have conducted survey on only simulation tools and testbeds developed for terrestrial WSNs [49, 52, 70]. For example, in [49], the authors presented a comprehensive survey on testbeds and experimental tools developed for WSNs and then provided an in-depth discussion on the current and future issues involved in this direction of research. Some comparison-based survey articles for UWA-SNs came out recently as well. For example, in [102], four ns-2-based simulation tools are studied and compared in terms of different properties. SUNSET and DESERT are compared in [89]. In [126], SUNSET and SeaLinx are evaluated and compared based on several performance criteria.

Although the aforementioned literature has contributed to the experimental research issues of corresponding tools and testbeds in UWA-SNs, a comprehensive review on underwater simulation platforms and field-level experimental testbeds is still missing. Therefore, in this article, we select 17 typical simulation and experimentation platforms that can conduct simulation, emulation, or even real field-level experimental testing for researchers, as shown in Table I, and aim to present a survey of the state-of-the-art testbed platforms developed since 2004. The motivation of this article is to facilitate research in UWA-SNs and assist researchers in the design of such type of platforms. More specifically, the contribution of this survey article is twofold. First, this

Table I. Selected Simulation and Experimentation Platforms

Platform	Release year	Institution	Laboratory
Seaweb	1995/2004	US Navy and Universities	The Office of Naval Research
Aqua-Lab	2007	University of Connecticut	Underwater Sensor Network Lab
UPPER	2007	National University of CES	Systems and Networking Research Lab (SysNet)
Aqua-Sim	2009/2015	University of Connecticut	Underwater Sensor Network Lab
UANT	2009	University of California, Los Angeles	Networked and Embedded Systems Lab
Aqua-Net/Mate	2009/2013	University of Connecticut	Underwater Sensor Network Lab
UW-Buffalo	2012	University of New York at Buffalo	Wireless Networks and Embedded Systems Lab
DESERT	2012	University of Padova	NAUTILUS project Lab
SUNSET	2012/2013	Sapienza University of Rome	UWSN Group Senses Lab
SeaLinx	2013	University of Connecticut	Underwater Sensor Network Lab
UnetStack	2014	National University of Singapore	Acoustic Research Laboratory
Ocean-TUNE (UCONN)	2014	University of Connecticut	Underwater Sensor Network Lab
Ocean-TUNE (UW)	2014	University of Washington	Fundamentals of Networking Lab
SUNRISE (LOON)	2014	NATO Science and Technology Organization	Centre for Maritime Research and Experimentation
SUNRISE (Porto)	2014	Porto University	Underwater Systems and Technology Laboratory
SeaNet	2015	Northeastern University, Boston	Department of Electrical and Computer Engineering
Ocean-TUNE (WaterCom)	2015	University of University of California, Los Angeles	Networked and Embedded Systems Laboratory

article presents a survey on underwater simulation and experimentation platforms and identifies research challenges, metrics, and future trends, which will help researchers to design more novel and practical testbeds. Second, we summarize 17 different platforms in terms of components, functionalities, features, and limitations as well as provide specific guidelines and recommendations that will help researchers to choose suitable research platforms according to their specific requirements.

To illustrate the advancements in underwater testbed research, in this article, we generally categorize the selected testbeds into three groups (e.g., lab-level experimental testbeds, short-term experimental testbeds, and long-term field-level experimental testbeds) based on the deployment scale, cost, and testbed characteristics, as shown in Table II. The lab-level experimental testbeds (e.g., Aqua-Lab, UPPER (Underwater Platform to Promote Experimental Research), SeaNet, UANT (Underwater Acoustic Networking plaTform), Aqua-Sim, and University of Washington- (UW) Buffalo) mainly focus on lab-level experiments and testing. Their deployment costs are relatively low and can conduct small-scale experiments. They are suitable for early-stage testing and for beginners to become familiar with underwater communication environments. The short-term experimental testbeds (e.g., Aqua-Net, SeaLinx, UnetStack, DESERT, and SUNSET) can actually conduct both lab- and field-level experiments. However, compared with those testbeds specifically designed and deployed for real field-level long-term experiments, this group of testbeds is mainly used for short-term experiments or testing. They are of medium to high cost, and the deployment scale is between medium to large. Most of them support seamless transition among simulation, emulation, or

Table II. Categorization of Simulation and Experimentation Platforms

Categorization	Main Features	Testbeds
Lab-level experimental testbeds	Low cost Small-scale Early-stage testing Suitable for beginners	Aqua-Lab UPPER SeaNet UANT Aqua-Sim UW-Buffalo
Short-term experimental testbeds	Medium to high cost Medium to large scale Short-term testing Seamless transition support	Aqua-Net/Mate SeaLinx UnetStack DESERT SUNSET
Long-term field-level experimental testbeds	High cost and large scale Long-term field testing Heterogeneous testbeds Remote access and control	Seaweb Ocean-TUNE (UCONN, UW, WaterCom) SUNRISE (LOON, Porto)

even real field-level testing. The long-term field-level experimental testbeds (e.g., Seaweb, Ocean-TUNE, and SUNRISE (Sunrise)) mainly focus on real field-level long-term large-scale deployment with high cost. They are generally equipped with heterogeneous components, such as underwater static nodes, autonomous underwater vehicles (AUVs), unmanned underwater vehicles (UUVs), or surface nodes. They also support remote access and control function, as these testbeds are usually deployed far from the shore in the ocean.

The rest of the article is organized as follows. In Section 2, the main components of simulations and experimentation platforms are introduced. We elaborately explain the metrics of testbed design in Section 3. In Section 4, Section 5, and Section 6, we discuss lab-level experimental testbeds, short-term experimental testbeds, and long-term field-level experimental testbeds, respectively, and present a detailed comparative discussion of each type. The advices and recommendations on choosing simulation and experimentation platforms are offered in Section 7. Finally, Section 8 concludes the article with some directions on future research.

2. COMPONENTS OF SIMULATION AND EXPERIMENTATION PLATFORMS

The simulation and experimentation platforms are usually composed of software-based components, underwater modems, static or mobile sensor nodes, and user interfaces [22]. Among the software-based components, simulation tools, a protocol stack, and an operating system (OS) are used to control the entire system. The acoustic modems are in charge of data transmission in the testbeds. There are usually many static or mobile sensor nodes in testbeds to sense or collect data from specific applications. On the other hand, the user interfaces are developed for researchers to communicate with or pass commands to the testbeds while conducting simulation, emulation, or field-level testing. In this section, we discuss the main components of the simulation, field, and lab-level experimentation platforms.

2.1. Software-Based Components

Among the software-based components, the simulation tools, underwater protocol stack, OS, and channel simulators are notable and important. The simulation tools include commercial, customized simulation software written by researchers or other free open source networking software. Until now, researchers have developed many well-known networking software, such as OPNET [77, 120], Omnet++ [13, 76], and ns-2 [73], which are suitable to simulate/emulate underwater sensor networks. Among these, ns-2 is the most reliable well-known open source discrete-event simulator, which provides a large volume of protocols, networks, and application models to more than 600 institutions scattered over 50 countries around the world. Ns2-Miracle is the extended version of ns-2 that has new libraries to support cross-layer message exchange facilities and allows multiple protocols to exist in the same protocol layer [11]. More importantly, it uses the Bellhop ray-tracing software [94] in the physical layer to provide more realistic underwater acoustic channels for World Ocean Simulation System (WOSS) [17, 41]. Besides the commercial network simulators, researchers have written many customized simulation software using C and C++ languages [16]. Linux is the typical OS, deployed on the hardware of testbeds and sensor nodes [71].

Since software-defined radio (SDR) has the capability to substitute conventional hardware components by software ones, it has many advantages for UWA-SNs [29, 54, 72]. For example, to deal with the harsh underwater communication channel, it is desirable to design software-defined modems (SDM) using SDR, which is adaptive with the changing communication techniques due to the complex and variable environments [20, 32]. GNU (Gnu's Not Unix) radio is one of the most popular SDR architectures, which is usually used to design advanced modulation schemes while building an early prototype of some systems. GNU radio provides many separate blocks to build prototypes rapidly, such as blocks to conduct signal processing, and so on. These blocks are connected logically with each other to perform specific functions one after another, which are the substitution of real hardware components in the design phase of early prototypes [40].

2.2. Underwater Acoustic Modems

Underwater acoustic modems are essential parts of the simulation and experimentation platforms. A more detailed comparison among different underwater acoustic modems is provided in [24, 33]. In this section, we briefly describe some commercial modems, research-oriented modems, and SDMs [20].

Some off-the-shelf commercial acoustic modems are available for researchers, and the most well-known commercial underwater acoustic modems include LinkQuest [57], Teledyne [109], EvoLogics [36], AquaSeNT [7], DSPComm [34], Tritech [112], Develogic [30], AppliedOceanSystems [6], and so on. These modems mainly ensure reliable communication with low Bit Error Rate for the applications in both shallow and deep water when the data rate is below 10Kbps. The communication distance can be from several hundred meters to several kilometers, and the maximum transmit power can be as high as 60W. Commercial modems are mainly based on reliable point-to-point long-distance communication, and hence these barely allow researchers to tune hardware parameters while testing novel protocols and applications [104].

To satisfy the requirements of the research community, a modem should be configurable while evaluating the performance of different underwater applications, protocols and algorithms. For example, researchers should have the flexibility to fully reprogram the modem (e.g., select the modulation mode) with self-adaptive networking protocols [14, 55]. Until now, several research-oriented modems have been developed for the research community, such as the WHOI (Woods Hole Oceanographic Institution) Micro-Modem, rModem [118], and so on. The WHOI Micro-Modem is one of the most well-known modems, and it was designed initially for small low-cost autonomous underwater vehicles, such as REMUS [97]. However, at the later stage, navigation equipment as well as the feature to choose phase-coherent high-data-rate communication modes and transmit power were added. The data rate of the communication modes can range from 80bps to 5400bps, and the power consumption can range from 8W to 48W [39].

rModem is also a reconfigurable modem, which is designed to simplify cross-layer optimization-based experimentation studies [9, 104]. Its field programmable gate array

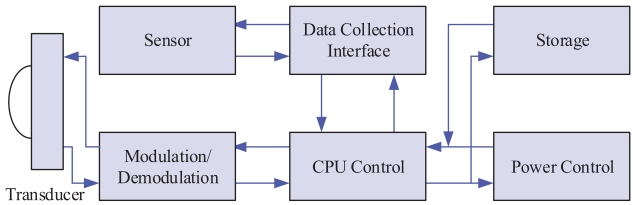

Fig. 1. The structure of an underwater sensor node.

enables researchers to choose carrier frequency from the [1−100]kHz range. With Simulink toolbox [104], researchers can reach modem hardware to handle data transmission or events recording related activities [9]. rModem can work with an omnidirectional Teledyne transducer (e.g., AT-408) and allows multiple-input-multiple-output transmissions by using its four configurable input and output channels [9]. Although rModem is reconfigurable, the protocol stacks and its Simulink tool need to be run on a personal computer (PC), which may impair its performance.

The functionalities of hardware-based modems all can be replaced by software-based implementation in SDMs, such as signal detection, error correction, modulation, and demodulation, and so on [20, 37]. Furthermore, with SDM, it is more convenient to dynamically select suitable communication techniques in accordance with ever-changing ocean environments to achieve optimal performance [31, 71, 95].

2.3. Static and Mobile Sensors

In order to set up a real field-level testbed, both static and mobile sensor nodes may be required to facilitate hybrid networking [59, 62]. Typically, the nodes are deployed on the surface of the ocean, inside the ocean, or near the sea floor [61, 63, 64, 122]. The structure of a typical static underwater acoustic sensor node is shown in Figure 1, which includes a central processing unit (CPU) to control the entire node, a transducer and a modulation/demodulation module to receive or send messages, a bundle of sensors to sense surrounding environments, a power supply module, and a data storage module.

Mobile sensor nodes are the essential parts of field-level experimentation platforms. Typical examples of such nodes are free drifting sensor nodes [53], AUVs or UUVs [65], or underwater gliders [101]. An AUV or UUV can move quietly and continuously for months under the water according to its pre-planned configuration with its solar-powered rechargeable battery [4, 60, 78]. The speed of one type of REMUS-class AUV is about 1.5m/s, which consumes around 15W amount of power. However, when the power consumption reaches about 110W, the travelling speed can reach up to 2.9m/s [106]. Underwater gliders are some other type of free drifting nodes that can freely drift throughout the ocean with little energy consumption [101].

3. DESIGN CRITERIA OF SIMULATION AND EXPERIMENTATION PLATFORMS

For any application, while designing a simulation tool or an experimentation platform, researchers encounter many challenges. The key challenge is to cope with the high temporally and spatially varying underwater communication channels and to provide real-time self-adaptive platforms based on varying environmental conditions and different application requirements. Another challenge is to design a fully configurable and flexible testbed that allows researchers to program or configure hardware or software parameters for testing different novel protocols and algorithms efficiently and accurately. Other challenges include time synchronization and event-timing; channel accuracy; integration of simulation, emulation, field-level testing with seamless transition; cost and energy efficiency; remote access and control facilities of the testbed; and so on. With the intention to overcome the aforementioned challenges, we think that the following criteria should be taken into consideration while designing a simulation tool or an experimentation platform.

—*Reconfigurability and cross-layer design support:* A typical testbed should have a flexible architecture and should be configurable at different protocol layers. This feature enables researchers to program the hardware and software components of the testbed by adjusting parameters that cover a wide range of scenarios and conditions of the applications. Basically, the success in constructing a testbed is greatly dependent on these characteristics. Self-adaptability is also an important metric that provides testbeds to have some sort of intelligence in responding to the event-related dynamic environments. On the other hand, the acoustic communication channel under the water is highly dynamic, and hence its capacity and the resultant communication delay is location dependent or spatially and/or temporally variant or can even be bursty by nature. To develop optimized protocols and algorithms with highly dynamic and ever-changing underwater acoustic channels, testbeds should provide a mechanism that allows researchers to exchange cross-layer protocol information. If the lower layer information (e.g., link or physical layer) is available to the higher layers (e.g., MAC, routing, transport, or even the application layer), then this allows researchers to reconfigure and control testbeds in an adaptive manner to verify the performance of newly designed protocols and applications.

—*Efficiency and accuracy:* The efficiency and accuracy of any arbitrary test or experiment is greatly dependent on the architecture of the corresponding simulator/testbed, its memory usage, power consumption, and so on. Among these, testbed architecture, accuracy of the simulated/emulated channel, and time synchronization are the most important factors. An accurate simulated/emulated channel ensures that simulation tools or testbeds have accurate acoustic propagation models that can capture the behaviors of real underwater acoustic channels. Consequently, the corresponding simulators/testbeds can generate reasonable evaluation results comparing with realistic application scenarios. Time synchronization is a primary basis for real-time distributed sensor network applications, and hence simulation tools and testbeds should ensure the occurrence of all events in a timely manner.

—*Seamless transition support:* Simulation, emulation, and field-level test have different roles in the lifecycle of application development. Integration of simulation, emulation, or field-level test (SEFT), and their seamless transition are important for any arbitrary tool or testbed. It is natural that researchers eliminate common errors in protocols design through preliminary simulation without involving any underwater hardware. Then, the same functionalities that were verified in the previous phase are emulated on real acoustic modems or underwater sensors to debug errors related to underwater hardware. The last step is to implement protocols or applications on field-level testbeds in the ocean for long-term testing and verification. Therefore, the testbed that integrates simulation, emulation, and field-level testing greatly facilitates application development. For such type of testbeds, seamless transition among different phases of the development without misleading information exchange is important. Efficiency of the testbed is further dependent on how the actual code is written for implementing different phases of a test or an experiment.

—*Modem support:* Underwater acoustic modem is an important component of testbeds. Testbeds should support off-the-shelf commercial and research-oriented underwater acoustic modems, and it is better to support different types of modems. A reconfigurable modem allows testbeds to change its configuration whenever it is necessary in a flexible manner, which is beneficial in both cross-layer design and handling environmental self-adaptation issues.

—*Free accessibility of the resources:* Freely accessible testbeds allow a broader group of researchers in having access on them for free and encourage collaborative underwater research. To obtain the benefit from these open access testbeds, simulation tools or software components should be downloadable from the Internet so researchers can conduct pure-software-based simulation independently. As for open access lab or field-level testbeds, the full or portions of the testbeds should be accessible to ordinary researchers for conducting real field-level experiments remotely.

—*Remote accessibility:* Field-level testbeds are normally deployed in the ocean, and hence remote access facility provides great convenience to researchers while conducting experiments via Internet or cloud-based technology at any time. Researchers should be able to access, control, and reconfigure both software and hardware-based components of the testbeds remotely. Typical tasks that researchers do frequently are hardware or network parameters configuration, functional mode selection, sensor nodes initialization, and so on. Other tasks include uploading experiment-oriented tasks, downloading experimental results, and so on.

—*Deployment cost:* Depending on different test targets and experimental requirements, researchers may set up small-, medium-, or large-scale experiments. This is because if the researchers always deploy large-scale experiments no matter the requirement is, it will incur huge expense for them. To obtain the diversity in field-level experiments, the testbed should be deployed in different geographic locations associated with different environments. To satisfy the diversity requirements of a hybrid testbed, there should be both static and mobile sensor nodes, such as AUVs, UUVs, or even underwater gliders.

4. OVERVIEW OF LAB-LEVEL EXPERIMENTATION PLATFORMS AND THEIR COMPARISON

The lab-level experimentation platforms include Aqua-Lab, UPPER, SeaNet, UANT, Aqua-Sim, and UW-Buffalo. In this section, we describe each testbed briefly that includes functional description, main features, limitations, and so on. Then, we provide a comparative study among them based on the aforementioned design criteria.

4.1. The List of Lab-Level Experimental Testbeds

4.1.1. Aqua-Lab. The Aqua-Lab (2007) is a low-cost testbed equipped with real modems, and was developed by University of Connecticut in 2007 [79]. As shown in Figure 2, the testbed is comprised of laptops, WHOI modems, a mixer, a water tank, an underwater speaker, and a hydrophone. The laptops control the acoustic modems by sending commands through the serial ports. Then, these modems drive the underwater speaker and hydrophone for acoustic communication under the water. Using a set of software application programming interfaces (APIs), researchers can control and reconfigure the modems to develop new applications or protocols. It also provides a C-language-based emulator, and hence researchers can emulate network topology, propagation delay, and attenuation to evaluate the desired algorithms and protocols in a comprehensive manner. Aqua-Lab also has a sound mixer to emulate complex underwater acoustic communication channels.

Features and limitations: Equipped with real WHOI modems, Aqua-Lab provides low-cost configurable environments. It is suitable for beginners to set up their own lab-level testbed. At the beginning, it helps them to become familiar with simple underwater acoustic communication concept. However, it lacks in providing facilities for subsequent development, and it cannot conduct complex simulations or experiments. Another limitation is that it cannot emulate concurrent data transmissions between only two modems.

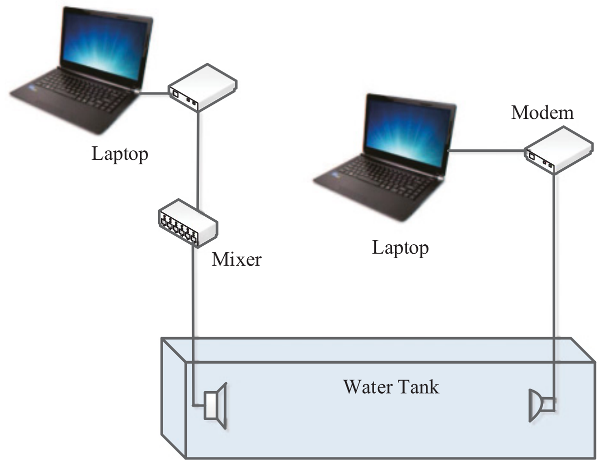

Fig. 2. A sample setup of Aqua-Lab.

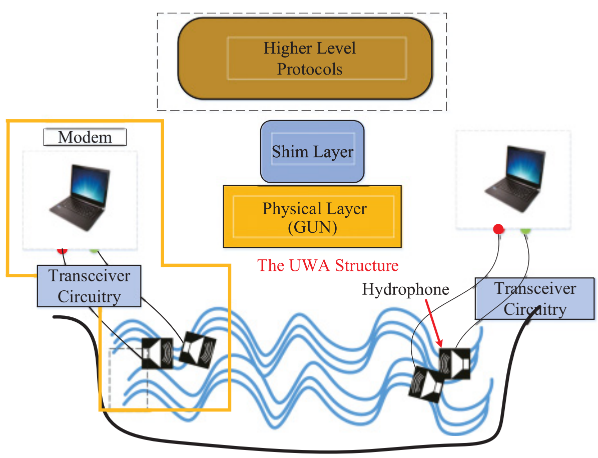

Fig. 3. A sample setup of UPPER.

4.1.2. UPPER. UPPER (2012) is an ultra-low-cost (about \$25–\$60) DIY-inspired testbed [1, 12]. To lower the cost of the testbed, the founders of this testbed use a cheaper hydrophone that is connected to a sound card, in which they implement a software-based modem using GNU radio. Therefore, the modulation and demodulation of data including the signal processing block are implemented by GNU radio, and hence it significantly reduces cost. The testbed structure is depicted in Figure 3 and composed of two main parts: a shim layer and a physical layer. The shim layer provides APIs to establish communication between remote users and the software modem for the purpose of data or tone transmission, transmitting configuration commands to the software modem, and so on. At the same time, it manages communication congestions between the shim layer and the hydrophone. There are two versions of UPPER: v1 and v2, with 6–10m and 30–60m communication ranges, respectively [12].

Features and limitations: UPPER is a DIY (Do It Yourself)-inspired cost-effective testbed, and it allows beginners to set up underwater test environments easily in their labs. Consequently, it is suitable for early underwater experiments, initial testing and validation of the protocols, and applications being designed. The limitation is that the

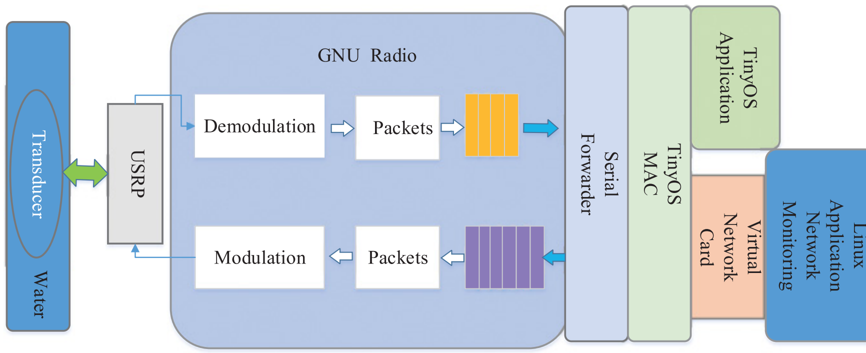

Fig. 4. The architecture of UANT.

testbed supports limited communication range, which is less than 60m due to the cost constraints. For the same reason, it is not suitable for large-scale simulation and lacks accurate underwater channel model in the simulation.

4.1.3. SeaNet. SeaNet (2015) was developed by Northeastern University in 2015 and is a low cost, fully reconfigurable software-defined networking framework based on acoustic communication media [28]. By leveraging the features of SDR (e.g., modular-based architecture, flexible design, etc.), it provides fully reconfigurable protocol stacks (e.g., physical, data-link, network, and application layers). Furthermore, it supports cross-layer design and provides real-time adaptation facility to both communication and networking protocols based on the change of environmental and application conditions. For example, the physical layer provides an adaptation mechanism to protocol designers in choosing forward error correction coding rate and modulation schemes through some user-defined decision algorithms. The preliminary prototype of SeaNet is made by using cheap commercial off-the-shelf hardware, and hence it incurs low cost.

Features and limitations: SeaNet is a low cost, fully reconfigurable underwater testbed for the next generation UWA-SNs, which is developed based on SDR. The limitation is that it supports only the construction of preliminary prototypes, such as experimentations in a small water tank. Therefore, further improvement is needed for more comprehensive experiments.

4.1.4. UANT. UANT (2009) is designed to provide a flexible approach in modifying system parameters without involving any specialized hardware components [71, 110]. With GNU radio and a well-known simulation tool TOSSIM (TinyOS Simulator) [40, 110], UANT provides configurable physical, MAC, and application layers. Consequently, researchers can rapidly implement or compare their proposed schemes (e.g., protocols, physical layer modulation schemes) with that running on a PC equipped with Linux OS or TinyOS. The sample architecture of UANT is shown in Figure 4. It includes the Universal Software Radio Peripheral transceiver [113], which is connected to a hardware transducer, the GNU radio, the TinyOS-based TOSSIM simulator, and a TinyOS-based or Linux-based application. GNU radio provides a configurable physical layer for UANT. When researchers need to design new advanced modulation schemes, they use the large library of modulation schemes and the signal processing blocks offered by GNU radio for rapid prototyping without involving any specialized hardware.

UANT also provides reconfigurable application layer by deploying TinyOS or Linux OS. UANT runs TinyOS applications on a PC equipped with TOSSIM. Since TinyOS is a well-known open source OS designed especially for energy-constrained sensor

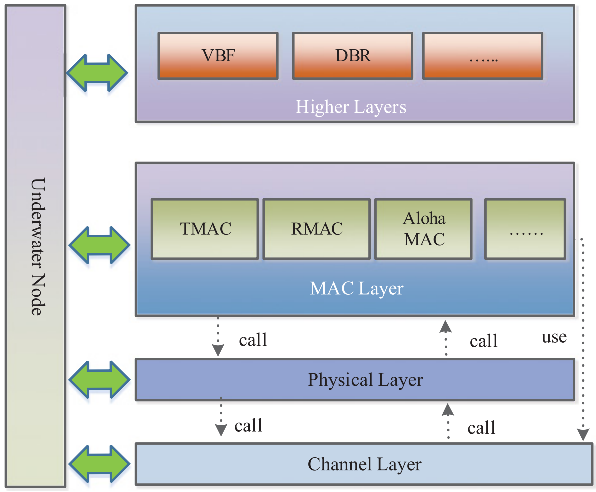

Fig. 5. The architecture of Aqua-Sim.

networks, UANT uses it in constructing real underwater environments. To emulate the exact environment, UANT changes the TinyOS configuration file to meet the specific requirements of different underwater applications. Therefore, the programs implemented for the simulation can be transferred to that to be deployed in real underwater sensor network environments with little effort because of using the same OS. For more complex applications, UANT relies on a Linux TCP/IP (Transmission Control Protocol/Internet Protocol) stack by incorporating virtual Ethernet cards that assign IP (Internet Protocol) addresses to nodes equipped with TinyOS. Therefore, when the applications are in the running mode, the nodes can communicate with each other via their IP addresses through TCP/IP protocols. Another advantage of the TCP/IP protocol is that researchers can configure MAC or physical layer to study the performance of specific applications.

Features and limitations: UANT is probably the first testbed that provides reconfiguration flexibility for both lower and higher layers of the underwater communication protocol stack (e.g., physical, MAC, and application layers). UANT adapts well-known open source tools, such as software-defined GNU radio, TinyOS, and Linux OS to provide cross-layer design flexibility to researchers. UANT is also a complete end-to-end networking-based underwater test platform. Furthermore, with TinyOS, the programs implemented for the simulation can be transferred to real underwater sensor networks with little effort. The limitation is that it is hard to run GNU radio on embedded sensors, and it is nontrivial to modify and test the original TinyOS and TOSSIM-based terrestrial sensor networks for underwater communication. TinyOS also has a rigid interface that makes the exchange of the entire physical layer in GNU radio with itself very challenging [28].

4.1.5. Aqua-Sim. Aqua-Sim (2009-2015) is developed on the top of famous simulator ns-2 [73], and perhaps the first complete packet-level simulation tool for underwater networks. Its initial version was developed in 2009 [121], and the channel and physical layer models were improved in 2013 and 2014, respectively [127, 128]. In 2015, ns-3 [74] was introduced as the extended version of ns-2 [66]. In [121], as depicted in Figure 5, Aqua-Sim includes additional underwater network protocols and can evolve independently from the old ns-2 simulation package. Furthermore, similarly to ns-2, Aqua-Sim follows the object-oriented design style, and hence additional functional or protocol modules can be introduced as new objects.

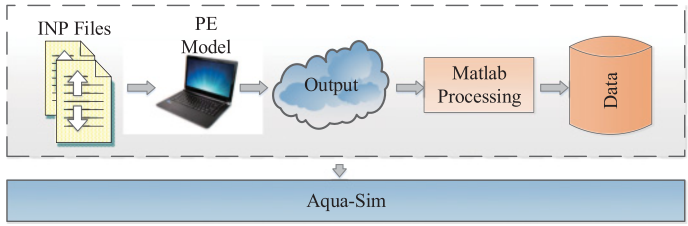

Fig. 6. The flow diagram of PE Model.

Since the initial version of Aqua-Sim used a basic acoustic channel model, the researchers found that worse overall performance of previous all field-level experiments is due to the long preamble in the data transmission period and low transmission rate. Therefore, in [127], the channel model of Aqua-Sim is improved by considering various factors (e.g., physical factors, environmental dynamics) jointly into account. This version of Aqua-Sim responds to the physical factors and effects of ocean environments by taking the refraction of acoustic signals into consideration. The refraction of acoustic signals in such environments can happen due to various types of sediment on the sea floor, such as sand, silt, clay, the depth of the water column, and the noise generated from wind and wave effects. It also integrates a shallow water attenuation model, namely Rogers Model [100], which considers the sediment type to make the cases involved with more realistic transmission loss. Furthermore, two practical system parameters that stem from the use of current commercial acoustic modems are also taken into consideration.

In [128], based on the aforementioned work, the authors further improved the accuracy of the simulation by integrating the parabolic equation (PE) model into the physical layer as an alternative to ray-based model. One type of PE model, that is, the Monterey-Miami Parabolic Equation (MMPE) [103], is selected for the simulator to capture the acoustic signal propagation that is varying with the dynamics of sea surface. However, when PE models are integrated with the physical layer, it not only incurs high computational cost but also requires a long time to finish a simulation task. Therefore, most of the PE models (e.g., MMPE) cannot be directly integrated into Aqua-Sim. In order to solve this problem, as shown in Figure 6, the MMPE model needs to be run at first, and then its output can be processed using MATLAB to produce a lookup table. This table is used to improve the simulation accuracy by providing precise information of the channel propagation.

Features and limitations: Aqua-Sim is one of the few open source simulation tools released for public use, and it can be downloaded from the Internet. Furthermore, it is a packet-level simulation platform and provides abundant underwater networks protocols for researchers. It also provides three-dimensional structure of the environment to be deployed in the simulation. Although it takes Rogers and PE models to improve the accuracy of the simulated channel, these models may be more sensitive in shallow water. Another limitation is that PE models generally incur heavy computational cost compared to most of the ray-tracing models although PE models have a better prediction on multipath transmission loss using Aqua-Sim. However, it should support ray-tracing models to provide more accurate simulation channels.

4.1.6. UW-Buffalo. UW-Buffalo (2012) is designed to provide a shared reconfigurable underwater networking platform based on Telesonar Benthos (SM-75) modem. In order to cope with the ever-changing underwater communication environment, it adopts cross-layer design by exchanging the information between the higher layers and the link or physical layer [55]. The commercial Telesonar Benthos (SM-75) modem does not

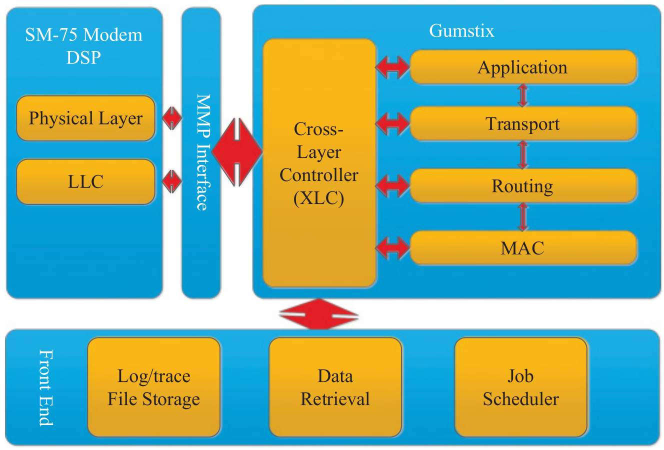

Fig. 7. The software architecture of UW-Buffalo.

provide an interface for researchers to implement self-designed MAC and networking protocols. The reason behind this is that the original code blocks of these protocols are embedded in the DSP (Digital Signal Processing) of the modem, and hence researchers are not allowed to alter or reconfigure them. UW-Buffalo modifies the implementation of the modem by integrating this with a Gumstix processor, which is a small well-known programmable system board comparable in size to a stick of gum [42]. As a result, this allows the research community to conduct advanced networking experiments as required.

As shown in Figure 7, by leveraging the networking API, the Modem Management Protocol (MMP) was developed in the Teledyne Benthos modem. The native protocol layers of this modem are bypassed completely by shifting its original responsibilities to Gumstix processor. With MMP, the modem can be controlled or reconfigured by exchanging messages continuously between the Gumstix processor and the modem while keeping the original implementation of physical and data link layers intact. To facilitate the cross-layer design, a new module called cross-layer controller (XLC) has been designed for controlling and regulating the information exchange among different protocol layers. XLC requires the assistance of MMP to obtain the information of physical and Logical Link Control layers by monitoring the status of the modem. Typically, a set of parameters related to the physical status of the modem are monitored, stored and updated in XLC. Similarly, the link and network layer information, such as link reliability, channel contention, path latency, and so on, are stored and updated in XLC for cross-layer design. The testbed includes both SM-975 and SM-75 modems of Telesonar Benthos brand, and it integrates with an underwater emulator to simulate different underwater channels for comprehensive experiment scenarios. It also has an IP-compatible protocol stack for underwater Internet access [107].

Features and limitations: UW-Buffalo modifies the commercial Telesonar Benthos (SM-75) modem to provide a reconfigurable cross-layer design-enabled platform. With APIs and the XLC module, researchers can use the low-level information from the link or physical layer to design cross-layer optimized protocols at the higher layer (e.g., MAC, network, or even the application layer). Furthermore, it has IP-compatible protocol stacks (e.g., IPv4 and IPv6) to facilitate Internet access. The limitation is that its channel emulator is not comparable with real underwater channels and may need a ray-tracing-based underwater acoustic model to improve the channel accuracy.

Table III. Main Features of Lab-Level Experimental Testbeds

Testbed	Software or Hardware	Modem Support	Channel Adaptability	Cost	Cross-layer Support
Aqua-Lab	Written in C	WHOI Micro-Modem	Low	Low	—
UPPER	GNU Radio	Low-cost Hydrophone	—	Ultra Low	—
SeaNet	C++ with Teensyduino	Teledyne RESON TC4013	High	Low	High among Four Different Layers
UANT	TOSSIM on TinyOS	Low-cost Transducer	Medium	Medium	High among Three Different Layers
Aqua-Sim	ns-2	Benthos Acoustic Modems	Medium with Rogers Models	Medium	—
UW-Buffalo	Gumstix on Linux	Telesonar Benthos SM-75	Medium with Channel Emulator	Medium	Medium with a Cross-layer Control Module

Furthermore, its ability in terms of large-scale simulation is weaker than that of the famous ns-2 simulator or TOSSIM.

4.2. Comparison among Lab-Level Testbeds

In Table III, we compare different lab-level experimental testbeds in terms of a set of features, such as hardware or software component, supported modem, channel adaptability, cost, and cross-layer design flexibility. In the following, we provide a more elaborate comparative study among these testbeds based on the aforementioned prominent and relevant design criteria.

Reconfigurability and cross-layer design support: Reconfigurability option provides researchers the flexibility to adopt cross-layer interaction support while designing experiments or to change the configuration of the testbed in accordance with application-oriented or environmental dynamics. Both Aqua-Lab and UW-Buffalo have flexible architecture and are equipped with reconfigurable modems. Aqua-Lab provides APIs to configure WHOI modems for various options, such as setting up diverse functional parameters or selecting different functional modes, and so on, whereas UW-Buffalo provides reconfiguration flexibility of its Teledyne Benthos modem. The reconfiguration flexibility option of SeaNet is due to its SDR-based architecture. On the other hand, the flexible architecture of UPPER and UANT is also based on SDR and GNU radio. Basically, UPPER adopts SDR and GNU radio to provide the acoustic modulation and demodulation features in communication, and it also allows researchers to configure modulation modes, frequency, and data rate. UANT utilizes GNU radio to provide a configurable physical layer, in which the transmission bit rate and center frequency are easily configured. Since embedded devices are not suitable for GNU radio because of their computational complexity, a PC is typically used to run GNU radio.

UW-Buffalo, UANT, and SeaNet support cross-layer design as well. UW-Buffalo is equipped with an XLC module, which makes connections between the physical layer and higher layers. Consequently, researchers can implement cross-layer optimized protocols via the software-defined protocol stack running on a Gumstix board. Lavish cross-layer design support across several layers further helps researchers to figure out the factors that affect the functionalities of their applications easily and quickly. As a supplement platform of rModem (which only provides reconfiguration option at lower protocol layers), UANT is probably the first underwater platform to provide reconfiguration flexibility at three different layers, such as network, MAC, and physical layers. SeaNet provides even more flexible cross-layer design support across the four layers. These four layers are physical, data-link, network, and application layers.

Efficiency and accuracy: To improve channel adaptation ability of different applications in simulation, both UW-Buffalo and Aqua-Sim provide channel emulators. The emulator in UW-Buffalo is developed based on the Teledyne (SM-75) modem, and it allows researchers to conduct small-scale laboratory-centered experiments. Aqua-Sim integrates the Rogers and PE models and provides more accurate underwater acoustic channels to improve the simulation accuracy at the granular level. Aqua-Sim is also the first open source complete packet-level underwater network simulation platform. Both Aqu-Sim and UANT are developed based on well-known powerful open source simulators, while others use their own proprietary implementation using C, C++, or other programming languages. Aqua-Sim is developed on the top of ns-2, but it does not patch on the existing ns-2 wireless package. Instead, a new simulation package paralleled with the Carnegie Mellon University (CMU) wireless package is designed, and hence it has the ability to evolve independently. UANT uses TOSSIM of TinyOS in simulation to facilitate its real underwater deployment and improves efficiency. One shortcoming of UANT is that TOSSIM and TinyOS are originally developed with the IEEE 802.15.4 standard for terrestrial WSNs. Consequently, more supports in TOSSIM are required to bring reconfiguration and adaptation facilities if it is used for UWA-SNs.

Seamless transition support: Although six testbeds in this group are suitable for field-level experiments, their main focus is only on lab-level experimental evaluation and development. Aqua-Lab mainly focuses on emulation using its emulator and configurable WHOI modems to evaluate protocols in an environment closer to the reality. By the emulator of Aqua-Lab, researchers can set up network topology, propagation delay, and so on, according to their preference. Although UPPER can be used to conduct experiments in a lake, it is suitable for early-level experiments in the lab environment due to its cost constraints. SeaNet is also suitable to construct early-level testing prototypes quickly using customized hardware components and especially to verify protocols in different layers, such as physical, data link, networks, and application layers. UANT is a framework that is based on TOSSIM although does not use TOSSIM solely as a simulation tool. However, TinyOS is an embedded OS for sensor nodes, and hence researchers can transfer the verified code blocks for simulation to real network environments with little effort. Aqua-Sim, which benefits from the powerful simulator ns-2, is equipped with numerous protocols, and it supports three-dimensional virtual environments in the deployment. However, currently, Aqua-Sim mainly focuses on only simulation and emulation.

Modem support: Modem plays an important role in protocol emulation, verification, and testing. Aqua-Lab, SeaNet, UW-Buffalo, and Aqua-Sim all are equipped with real commercial underwater modems. As shown in Table III, Aqua-Lab supports WHOI Micro-Modem, SeaNet supports Teledyne RESON TC4013, UW-Buffalo supports Telesonar Benthos SM-75, and Aqua-Sim supports Benthos Acoustic modems. On the other hand, UPPER and UANT are equipped with low-cost hydrophone and transducers as the substitution of a modem.

Free accessibility of the resources: Among six testbeds, currently, only Aqua-Sim provides an open source software package for the research community, and a new version (2.0) of Aqua-Sim was released recently [125]. Moreover, the implementation of all these six testbeds are not that much more sophisticated, and hence researchers can easily emulate similar or different underwater test environments with the help of published literature.

Remote accessibility: Since lab-level experimental testbeds are mainly used in the lab, remote access and control is not an important metric for this group of testbeds.

However, some testbeds have remote access functionality. The remote access functionality of UPPER is achieved by Remote Procedure Call, which requires API extension in the shim layer. UW-Buffalo has an IP-compatible protocol stack that will provide easy underwater Internet access in the future.

Deployment cost: Since all these testbeds are deployed in the lab environment, their deployment costs are relatively low. Among them, the deployment cost of both Aqua-Lab and UPPER are the lowest. Although Aqua-Lab belongs to the early version of simulation testbeds, it provides a low-cost configurable platform with WHOI modems. Therefore, it is suitable for new researchers to build confidence and to evaluate their designed protocols at early stages of the design process. Similarly to Aqua-Lab, UPPER focuses on lowering costs to let more researchers access them. Although it may not be suitable to deploy UPPER in sea or ocean over a large portion of area because of high cost, it is considered as economically feasible to evaluate designed protocols at the early stage before real field-level testing or experiments. SeaNet is also built on cheap commercial hardware components to reduce deployment cost.

5. OVERVIEW OF SHORT-TERM EXPERIMENTATION PLATFORMS AND THEIR COMPARISON

Until now, the testbeds that fall into this category are Aqua-Net/Mate, SeaLinx, UnetStack, DESERT, and SUNSET. In this section, we describe each of them briefly, and then we present a comparative study among them based on the aforementioned design criteria.

5.1. The List of Short-Term Experimental Testbeds

5.1.1. Aqua-Net/Mate. Aqua-Net (2009) and Aqua-Net Mate (2013) were developed by University of Connecticut in 2009 and 2013, respectively [83, 124]. Differing from the prominent simulator ns-2 or TOSSIM, Aqua-Net was developed from the scratch, and it supports cross-layer design. Since the original Aqua-Net was designed for a real-system without the simulation functionalities, it was extended to provide such functionalities and was named Aqua-Net Mate [124]. The event scheduling efficiency of Aqua-Net Mate is the same as that of Aqua-Net because of its real-time scheduling mechanism, and researchers can effortlessly switch between simulation and experimental testing as required. Aqua-Net can run on the top of an embedded Linux OS, in which it uses Micro-Modem as the communication device and Gumstix as the controller. Aqua-Net was tested (an experiment called Aqua-TUNE [81]) in a lake on 2011 as well as in the Atlantic Ocean in 2013 [80].

The architecture of Aqua-Net is depicted in Figure 8. Several types of modems at the physical layer are supported to implement underwater networks with real-world scenarios via a wrapper. The wrapper is developed under compliance with the National Marine Electronics Association for serial communication to provide instructions for connectting underwater devices with non-marine devices. The socket interface is for researchers to create protocols or reuse existing implementation for new applications. To implement cross-layer design, the information of different layers is stored in a system database. Then, via the cross-layer interface, the information is utilized by all layers to improve the network performance. As illustrated in Figure 9, three layers are added in Aqua-Net Mate, which are Adapter, Virtual Modem, and Virtual Channel. The Virtual Modem is used to emulate real modem, which is linked to Aqua-Net via an Adapter. The Virtual Channel is the communication channel between different Virtual Modems.

A version of Aqua-Net, developed on top of Linux, was tested in a lake (Mansfield Hollow Lake) in 2011, which is called Aqua-TUNE [81]. The deployed experiment Aqua-TUNE used four nodes. Each node, accompanied with a floating buoy, is equipped with both radio and underwater acoustic modems for surface and underwater

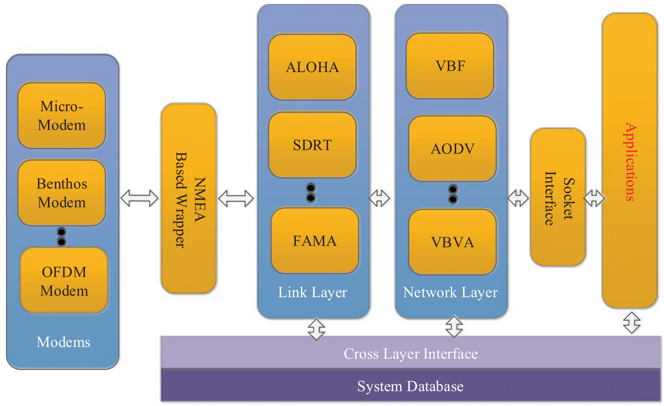

Fig. 8. The architecture Aqua-Net.

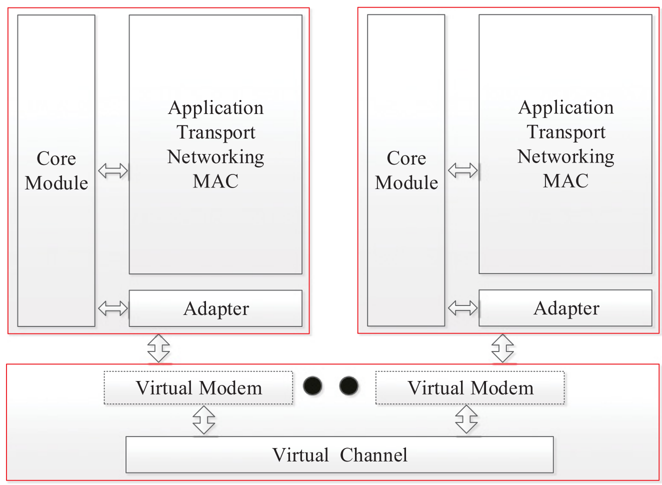

Fig. 9. The architecture of Aqua-Net Mate.

communications. In addition, the node also had GPS (Global Positioning System) facilities to obtain its instantaneous location information. With Aqua-TUNE, researchers can tune some features or functionalities, such as node localization, time synchronization, transmit power of node, network protocols, and so on, to verify their designed protocols. In 2013, Aqua-Net was tested in the Atlantic Ocean [80]. That field-level experiment had 11 deployed nodes over 10km open sea area. The objective of the experiment was to identify, study, and analyze various factors that affect the performance of network protocols.

Features and limitations: Aqua-Net was developed from the scratch, and, consequently, it is easier for researchers to tailor underwater network protocols more easily. Compared to other simulation tools, such as ns-2, Aqua-Net is associated with a smaller footprint, which eventually benefit embedded sensor networks. One shortcoming of Aqua-Net is that it allows only one protocol in a modem that results in inefficient usage of resources.

5.1.2. SeaLinx. SeaLinx (2013) stems from Aqua-Net and the extended framework after overcoming some drawbacks of Aqua-Net [56]. With the ability to optimize cross-layer design, SeaLinx provides a better solution. As shown in Figure 10, SeaLinx

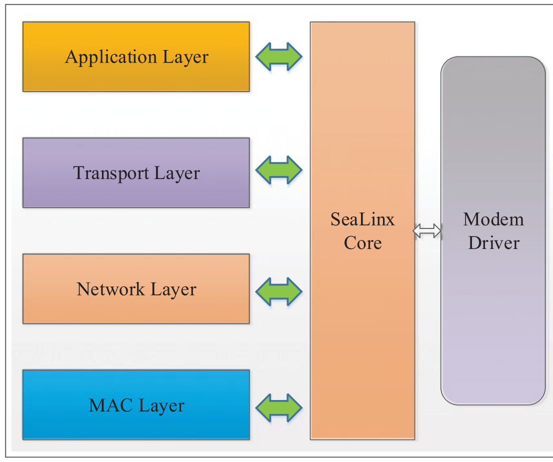

Fig. 10. The architecture of SeaLinx.

includes three parts, which are protocol module, SeaLinx Core, and Modem Driver. Such architecture makes SeaLinx a cooperative tool rather than a monolithic tool. In a typical protocol stack, each protocol layer can only communicate with its adjacent layer directly, such as the layers above or below of it. On the contrary, in SeaLinx, each layer can directly communicate with any other layer of the protocol stack via SeaLinx Core.

In Aqua-Net, each modem can only handle one protocol, which leads to inefficient usage of resources. Therefore, in SeaLinx, each individual protocol is operated by an individual process. In this way, each protocol can independently be altered or restarted. If one protocol fails to function properly, then it has minimal effect on the rest of processes in the protocol stack. Furthermore, the simulation code blocks in SeaLinx can also be used for field-level experiments without any change. Based on SeaLinx, in [51], a real sea-level test with the orthogonal frequency division multiplexing– (OFDM) based modem is conducted.

Features and limitations: Since Sealinx runs on real-time systems, the event scheduling in this tool is nearly precise and accurate. Furthermore, the operational architecture is distributed by nature, and therefore the code blocks for simulation can be used for emulation without any change. It also supports multi-process and multi-threaded applications. Therefore, it incurs higher accuracy in timing, which is suitable to test time-sensitive protocols. The limitation is that the tests conducted on this may suffer from efficiency-loss because of real-time operations.

5.1.3. UnetStack. UnetStack (2014) is a new type of underwater network testbed, which is based on Fjage agent framework [21]. The most noticeable characteristic of this testbed is that it is not based on traditional layer-based protocol stack. Instead, UnetStack consists of many software agents and constructs a service-oriented architecture. These agents are allowed to share information and communicate with each other, and hence cross-layer optimization is achieved more easily compared to the traditional layered architecture. Furthermore, UnetStack takes Java Virtual Machine (JVM) as part of its component, and it supports both discrete event simulations and real-time operations. Therefore, with the help of JVM, once the protocols and applications are developed and tested via simulation on UnetStack, the simulation code blocks can be used in any modem that is compliant with UnetStack for further field-level experiments. Consequently, researchers can avoid the hassles of writing code separately for simulation and field-level experiments. Until now, UnetStack supports several types of

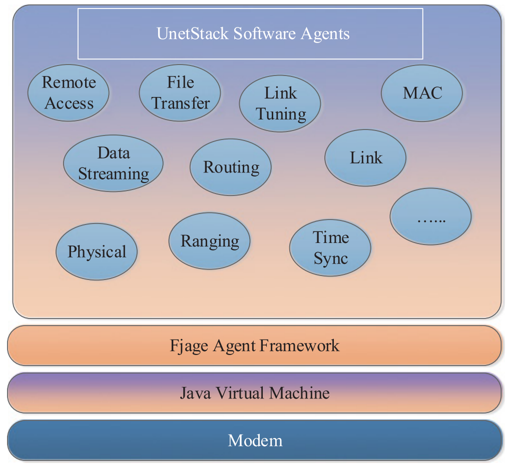

Fig. 11. The architecture of UnetStack.

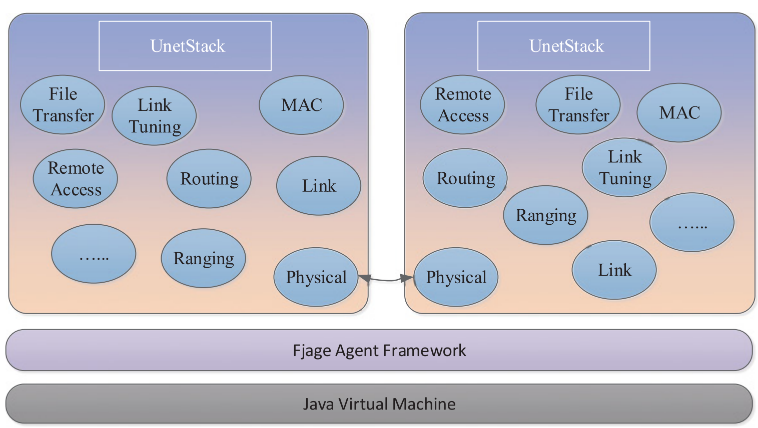

Fig. 12. The architecture of UnetStack-based UnetSim.

modems, such as Subnero, Evologics, and ARL UNET-II [23]. UnetStack also supports SDMs, and [20] provided an implementation of an OFDM modem for UnetStack.

The architecture of UnetStack is shown in Figure 11, which includes UnetStack agents, a fjage agent framework, Java VM, and a modem. The agents in the stack provide well-defined functionalities similar to the layers in the traditional network protocol stack. Generally, the agents communicate with each other via different types of messages, such as request, response, and notifications. In addition to these low-level agent-to-agent communications, UnetStack also supports high-level communications to monitor or control other agents. Besides message-based agent-to-agent communication, UnetStack also supports message broadcasting service, by which a set of subscribed agents receive messages of a certain topic to which are subscribed. Note that the service-oriented agent architecture is highly extensible, and hence it enables researchers to develop, test, and add new functionalities to a process that was not supported in the traditional layer-based network architecture. Figure 12 depicts the architecture of UnetSim, in which nodes interact with each other through physical agents deployed on them. The researchers can access the protocol stack directly or remotely via an

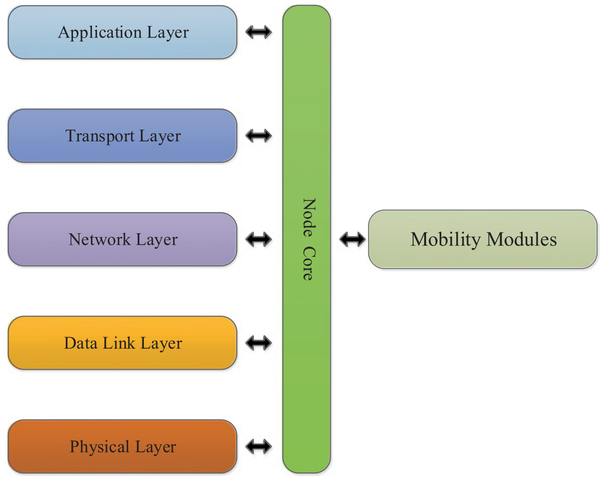

Fig. 13. The architecture of DESERT.

open source fjage agent framework with text-based commands and can use the real hardware modem via the physical agent (driver) when it is in simulation mode.

Features and limitations: The agent-based, service-oriented architecture of UnetStack facilitates rapid and flexible cross-layer design support for applications, simulation, field-level testing, and deployment. Furthermore, since UnetStack has a JVM component, the compiled binary code blocks written for simulation can be directly transferred to UnetStack-compatible modems for further field-level experiments.

5.1.4. DESERT. DESERT (2012) is an open source simulation, emulation, and experimental tool, based on ns-2 and ns2-Miracle, and was developed at the University of Padova [15, 68]. Ns2-Miracle is the extended version of ns-2 that adds many new libraries to enhance information exchange mechanism for cross-layer design and optimization [75]. In ns2-Miracle, several modules can co-exist simultaneously in the same layer with little interaction. Moreover, its highly modular-based architecture enables researchers to reuse many code blocks written for ns-2 with only minor modification. It further exploits the modularity property to optimize system design while taking cross-layer information into full consideration.

The architecture of DESERT is depicted in Figure 13. The physical layer provides the interface between the simulator and real hardware modems (e.g., Micro-Modems, Evologics). The mobility modules implement four different two-dimensional (2D) or 3D mobility models to simulate underwater robot movement. Similarly to real hardware devices, the architecture of Ns2-Miracle is logical, and hence it provides valid outcomes while evaluating the performance of user-designed applications and protocols. Furthermore, ns2-Miracle improves its simulation accuracy by connecting itself to specific underwater channel model tools (e.g., WOSS) [41]. Ns2-Miracle can also integrate real hardware modems to the simulation process for creating a more realistic environment while developing high-performance protocols and applications.

DESERT supports both emulation and testbed settings. In the emulation setting, as shown in Figure 14, multiple acoustic modems are connected to a single PC or a laptop, which runs a single ns2-Miracle process to control all modems and their connected acoustic transducer. The other one is the testbed setting, as depicted in Figure 15, in which unique instances of ns2-Miracle or related devices take charge of each of the modems. RECORDS (Remote Control Framework for Underwater Networks) allows DESERT to be remotely controlled in real field-level trials [111], which has been

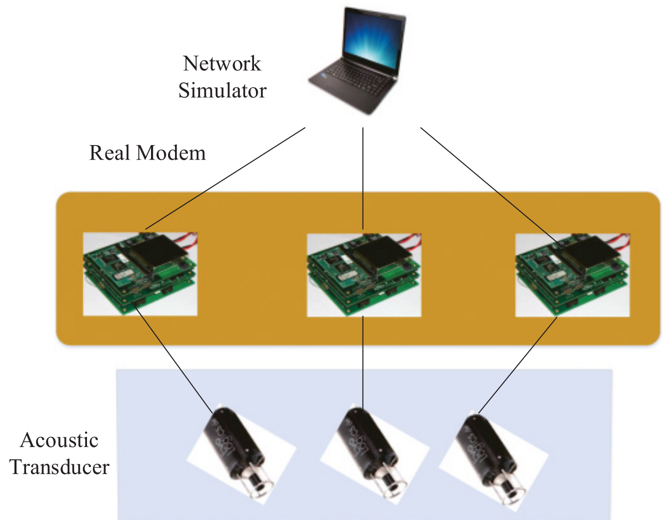

Fig. 14. A sample emulation setting of DESERT.

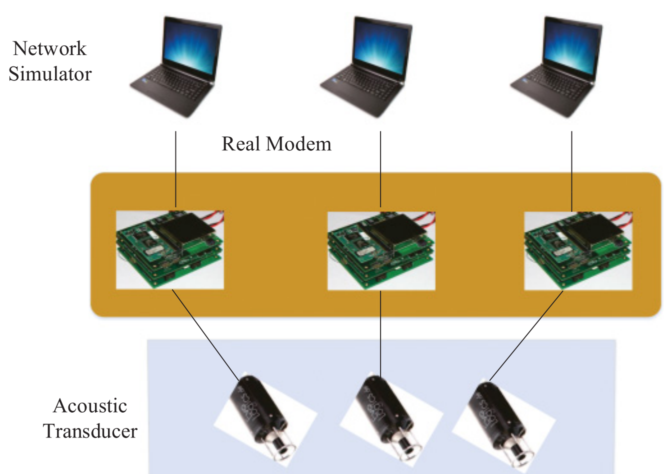

Fig. 15. A sample testbed setting of DESERT.

successfully experimented in Center for Maritime Research and Experimentation (CMRE) [67].

Features and limitations: DESERT is developed on top of ns-2 and ns2-Miracle and can integrate simulation, emulation, and field-level testing in a seamless manner. It is also an open source downloadable framework from the Internet. With a simple architecture, although DESERT provides seamless transition flexibility among simulation, emulation, and field-level testing, it has limitations, such as high resource and energy consumption.

5.1.5. SUNSET. SUNSET (2012) is also an open source platform developed from ns-2 and ns2-Miracle. Similarly to DESERT, it provides the similar facilities, such as seamless transition support among simulation, emulation, and field-level testing while

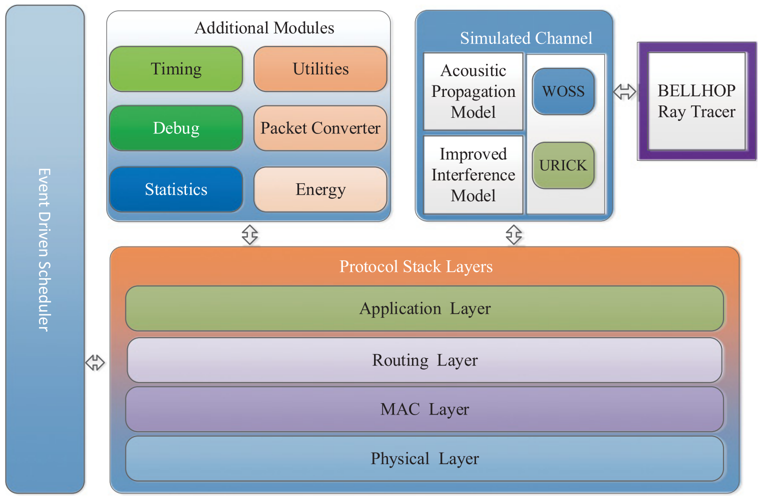

Fig. 16. A sample setup of SUNSET in the simulation mode.

evaluating the performance of novel applications and protocols [84]. For example, researchers can first use the simulation mode of SUNSET to implement, evaluate, and improve their designed applications and protocols. Then, as required, they can use the simulation code blocks for emulation or sea-site testing. When researchers switch from simulation to emulation or sea-site testing, they do not need to modify the code blocks written to implement their designed applications and protocols significantly. Currently, several acoustic modems are used for SUNSET, including Teledyne Benthos modem, Kongsberg, Evologics, and Micro-Modem. In addition to that, SUNSET also integrates various surface-level or underwater mobile vehicles (AUVs or autonomous surface vehicles (ASVs)) and different nodes to measure temperature, CO_2, or other physical or environmental properties of the ocean. It also supports cross-layer optimized protocols or applications design.

SUNSET released version 1.0 and version 2.0 for the research community in 2012 and 2013, respectively. The latest version of SUNSET provides mechanisms to control or reconfigure the testbed remotely with the help of underwater acoustic communication media and back-seat driver module [88]. With SUNSET, many experiments have been conducted by placing its components in three different countries [25]. Such deployment was possible with SUNSET because of its cooperative and remote management facilities. SUNSET has extensively been evaluated via more than 15 tests and experiments. It is chosen as a standard testbed for EU-FP7 project SUNRISE, which is an Underwater Internet of Things–based project deployed in Europe and North America [5, 67].

The functional block diagram of SUNSET is shown in Figures 16 and 17 for the simulation mode and the emulation mode, respectively. There are three main modules in SUNSET, which are core components, network protocols, and emulation components [86]. In Figure 17, new modules, including application driver, packet converter, generic driver, modem driver, channel emulator, and multi-threaded real-time scheduler, are added. The application driver allows easy integration with any external device for data exchange or interaction by providing methods to set/get parameters to/from external devices or to execute desired actions. The external devices typically include real sensing platforms and underwater automatic vehicles, such as AUV/ASVs. Packet converter is used to reduce the packet header size as much as possible before transmitting to the

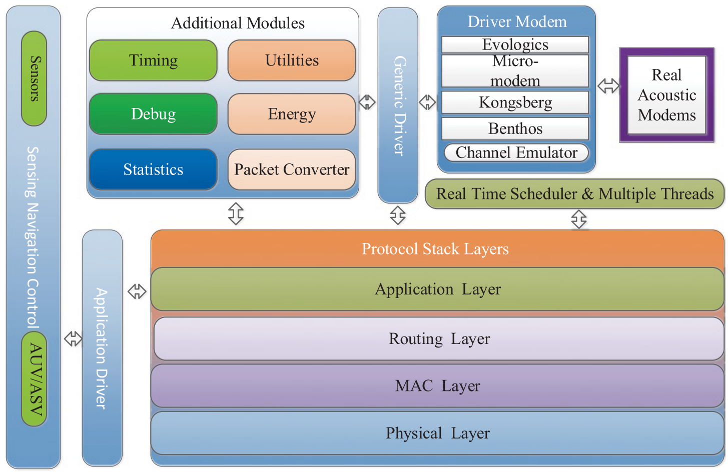

Fig. 17. A sample setup of SUNSET in the emulation mode.

actual devices. This is because in the simulation, a packet includes some redundant header to contain additional information, which is useful for debugging or statistical purposes. The Generic driver is responsible for assisting in the interaction between the real hardware modem and the ns-2 simulator. SUNSET supports a multi-threaded real-time scheduling technique. This allows some arbitrary program (handling one network protocol) to have multiple threads, in which the main thread is associated with the main program and the secondary thread could be the connections that bridge communication with external devices. Because of the real-time scheduling mechanism, SUNSET is transparent between the protocol stack and the external hardware.

Features and limitations: SUNSET supports simulation, emulation, and field-level testing in a seamless manner via the re-usage of code to a great extent. It was selected as a standard testbed in SUNRISE, which was extensively experimented over a period of four years. It is also an open source platform, the latest version of which can be downloaded from the Internet. Even though SUNSET has the ability to simulate real scenarios with long duration (in real time) within a short period of time, it may be difficult to identify time synchronization and accuracy-related problems as all events are executed in a single thread sequentially even with its real-time scheduler. Still, because of the real-time scheduling mechanism, in the simulation, SUNSET may achieve higher accuracy in event scheduling while avoiding the overlapping of events with each other.

5.2. Comparison among Short-Term Experimentation Platforms

As shown in Table IV, we choose software components, hardware components, supported modems, channel adaptability, cost, and seamless transition ability as a feature set to make a brief comparative study among different short-term experimental testbeds. In this section, we further present a detailed comparative study among them based on the aforementioned feature set, which is as follows.

Reconfigurability and cross-layer design support: All five testbeds provide a reconfigurability option by exploiting their different architectures. Each testbed with its distinct architecture has its own pros and cons. Aqua-Net and SeaLinx are the platforms in which the protocol stack is layer based. However, their protocol stacks are built from the scratch, and hence it is easier to design a new protocol (that will

Table IV. Main Features of Short-Term Experimental Testbeds

Testbed	Software or Hardware	Modem Support	Channel Adaptability	Cost	Seamless Transition Support
Aqua-Net/Mate	Gumstix on Linux	WHOI Micro-Modem, Teledyne Benthos modem	Medium with virtual channel	Medium	Low
SeaLinx	Extension of Aqua-Net	WHOI Micro-Modem, Teledyne Benthos modem	Medium	Medium	Medium
UnetStack	Agent-based architecture	ARL UNET-II, Subnero, Evologics modem	Medium	Medium	High from simulation to field-level testing
DESERT	Ns-2 and ns2-Miracle	WHOI Micro-Modem, EvoLogics S2C	High with channel modeling tool WOSS	High	High with simulation, emulation, and field-level testing
SUNSET	Ns-2 and ns2-Miracle	WHOI, Evologic, Kongsberg and Teledyne Benthos modem	High with WOSS-based BELLHOP Ray tracer and URICK	High	High with simulation, emulation, and field-level testing

be added to the existing stack) that is especially suitable for UWA-SNs. DESERT and SUNSET were developed on top of ns-2 and ns2-Miracle. UnetStack provides a testbed reconfigurability option via its flexible agent-based architecture and configurable UNET-2 modem. The agent-based architecture is flexible, which allows us to share information and promote interaction among different agents. Consequently, it is easier to implement new protocols and applications that require us to take cross-layer information into full consideration. The flexible framework is also extensible by adding new agents as required. In addition to that, UNET-2 is an SDM that is flexible and adaptive in accordance with the operations of UnetStack. Via firmware in the modem, researchers are able to configure the modem on the fly as per the demands of their designed applications.

Cross-layer optimization is also supported in all five testbeds of this group. Although the protocol stack of Aqua-Net is based on layer-based architecture, it has a core module for exchanging messages among different layers, which provides the flexibility in having cross-layer optimization. SeaLinx is the enhanced version of Aqua-Net in terms of cross-layer design. Unlike using the Unix socket in Aqua-Net, SeaLinx has a Core module (acts as a server) to handle all relevant communications for cross-layer optimization. SeaLinx can also run multiple protocol modules in parallell even in the same layer. On the other hand, UnetStack has a service-oriented architecture, which allows the agents to share information as required. Consequently, because of the advantages in cross-layer interaction, the cross-layer optimization is achieved more easily compared to other platforms equipped with a traditional layer-based protocol stack. Both SUNSET and DESERT are developed on the top of original ns2-Miracle to provide cross-layer design support. Ns2-Miracle is a modular-based framework, where different modules are developed to handle different protocols in different layers (e.g., PHY (Physical Layer), MAC, routing, transport, application layer protocols, and so on).

The cross-layer optimization algorithms are employed in plug-in modules, and a Node Core is developed to act as a bridge among different modules, and plug-ins are used to facilitate cross-layer design.

Efficiency and accuracy: Five testbeds in this group adopt different approaches to achieve efficiency and accuracy while experiments/simulations are conducted on them. In Aqua-Net and SeaLinx, each protocol module can run as an independent process. At the same time, they can communicate with each other. Since all events are scheduled by the embedded system in a real-time scale, their footprints are smaller, and the resultant event execution control is better. UnetStack supports both discrete-event and real-time operation modes, and so the compiled binary simulation code can be copied to any UnetStack-compliant modem (e.g., UNET-II) for further field- or lab-level testing without requiring any additional cross-compilation. Both DESERT and SUNSET are extended from ns-2, and hence they are also constrained with discrete-event characteristics. For example, both platforms have a main single-threaded process, and the events are executed sequentially scheduled by a strict event scheduler that is sensitive to time-restricted events. Different event scheduling mechanisms have different simulation performance. For example, SUNSET may outperform SeaLinx for shorter simulation/experimentation duration; however, SeaLinx ensures better accuracy in terms of event-timing as SeaLinx schedules events from multiple threads. Although DESERT and SUNSET have high compatibility, the system architecture of DESERT is simpler, which can be learned and used more easily by the users. DESERT has another significant advantage, which is that multiple protocol instances can be run simultaneously in each layer. On the contrary, in SUNSET, the events must run one after another sequentially, which is unfavorable, especially when an event involves I/O (In/Out) operations. DESERT has the ability to adapt with the variation of channel while following the guidelines of WOSS. SUNSET has also the same ability while taking the advantage of the WOSS-based BELLHOP ray tracer and URICK (Urick). Among all testbeds in this group, SeaLinx is chosen as a standard platform in Ocean-TUNE, while SUNSET is chosen for the field-level testbed project SUNRISE.

Seamless transition support: The original version of Aqua-Net does not support simulation mode, and Aqua-Net Mate developed a new simulation module to provide simulation functionalities along with the experimentation facilities. SeaLinx has a special modem driver that facilitates communication between two different types of models with distinct physical layer via making marginal or negligible changes in the modules of its protocol stack. UnetStack allows researchers to evaluate protocols both in the simulation and in sea trials in a seamless manner. Since UnetStack supports both discrete-event simulation and real-time experimentation modes, with the help of JVM, it allows us to transfer the entire compiled binary simulation code to any UnetStack-compliant modem for field-level experiments without requiring any additional cross-compilation. Both DESERT and SUNSET support simulation, emulation, and field-level experimentation modes and allow users to seamlessly transit from simulation to sea-site tests. When researchers use DESERT to transit from the simulation mode to the emulation mode, they do not need any extra module for this transition. Therefore, it not only reduces the simulation/experimentation time but also helps researchers in avoiding additional errors associated with code rewriting. However, the packet conversion method in DESERT is not that convenient, which may incur a higher packet conversion overhead [85]. Moreover, it is pointed out in [89] that experiments conducted via DESERT are less flexible and inefficient compared to that via SUNSET. The underlying reason behind this may be that DESERT uses an ns-2 real-time event scheduler that handles the interaction among multiple threads inefficiently. Therefore, it leads to higher CPU usage, memory, and energy consumption.

Modem support: Currently, Aqua-net and SeaLinx support WHOI Micro-Modem, Teledyne Benthos modems. Moreover, UnetStack supports ARL UNET-II, Subnero, and Evologics, and DESERT supports WHOI Micro-Modem and Evologics S2C. On the other hand, SUNSET supports five different commercial acoustic modems, such as WHOI FSK (Frequency Shift Keying) and PSK (Phase-shift Keying) Micro-Modem, Evologic, Kongsberg, and Teledyne Benthos modems.

Free accessibility of the resources: As shown in Table VI, UnetStack, DESERT, and SUNSET provide free downloadable software components to the research community. Note that, all three testbeds have released new versions of software components recently after their initial versions.

Remote accessibility: SeaLinx, UnetStack, DESERT, and SUNSET all have mechanisms that allow users to access and control the testbeds remotely. SeaLinx integrates a module, namely Acoustic Remote Control (ARC), to remotely access and control network nodes. ARC runs as an application-layer module permanently on network nodes, and it allows researchers to configure network parameters and monitor network status via acoustic communication remotely. UnetStack has a remote access agent to offer remote access and control functionalities (e.g., setting or getting parameters, file transfer, or other remote operations). SUNSET exploits a back-seat driver to remotely control different hardware components of the testbed (e.g., static and mobile nodes) via acoustic communication media. Typical remote control functionalities include protocol stack selection, network parameters tuning, initiation and termination of experiments, and so on. DESERT is equipped with open source software RECORDS to provide remote reconfiguration and control facilities for multi-hop networks. Unlike the back-seat driver in SUNSET, RECORDS can be easily and conveniently transferred to embedded systems without further cross-compilation.

Deployment cost: Compared with lab-level testbeds, the testbeds in this group have relatively higher deployment cost. Aqua-Net, SeaLinx, and UnetStack have medium deployment cost. Although DESERT and SUNSET support simulation, emulation and field-level experimentation operations in a seamless manner, setting up such integrated test environments usually incurs overall deployment cost.

6. OVERVIEW OF LONG-TERM FIELD-LEVEL EXPERIMENTATION PLATFORMS AND THEIR COMPARISON

The testbeds in this group include Seaweb, Ocean-TUNE, and SUNRISE. Since both Ocean-TUNE and SUNRISE are under development, we select University of Connecticut (UCONN), UW, and WaterCom as the representatives for Ocean-TUNE and the Littoral Ocean Observatory Network (LOON) and Porto testbeds as the representatives for SUNRISE to discuss their gradual development process and progress. Then, later in this section, we provide a comparative study among three long-term field-level experimental testbeds based on some pre-specified criteria.

6.1. The List of Long-Term Field-Level Experimentation Platforms

6.1.1. Seaweb. Seaweb (1995–2004) is the most well-known field-level testbed, which is designed mainly for the applications of US Navy military [99]. Over 50 experiments have been conducted on Seaweb in different areas for different objectives [98, 99]. Seaweb brings undersea wireless networks to reality. The network includes static underwater acoustic sensor nodes, AUVs or UUVs, underwater submarines, and intelligent master nodes in buoys with various interfaces to manned command centers. The network facilitates researchers to coordinate appropriate resources for accomplishing a pre-specified mission in an arbitrary ocean environment via commands, control, communication, and navigation.

In 2001, 14 nodes were deployed on a grid topology under the water to test Seaweb on the Loma Shelf near San Diego (CA). The network included 10 underwater telesonar repeater nodes, two Racom buoy gateway nodes, and two experimental sensor nodes of deployable autonomous distributed systems. A submarine maneuvered itself around the network and sent the first successful underwater wireless email to a command center.

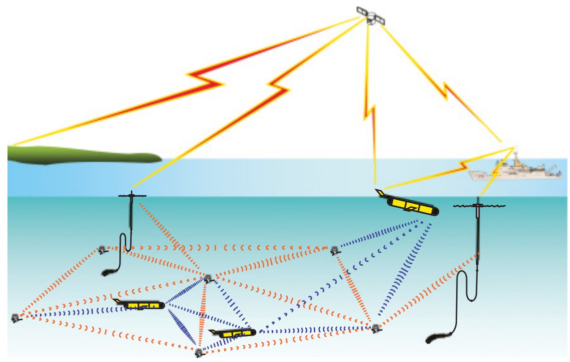

Fig. 18. A sample experimental setup using Seaweb on 2003.

In 2003, as shown in Figure 18, an experiment was conducted in the eastern Gulf of Mexico. The deployed system consisted of six fixed repeater nodes, three AUV glider nodes, two Racom buoy gateway nodes, and a shipboard command center. The experiment mainly tested the underwater mobile sensor nodes of Seaweb. With the deployed Seaweb, researchers could track the positions of the moving nodes continuously and could assist the moving nodes in navigation under the water. Moreover, AUVs are also proved to be excellent mobile gateways without the attendant vulnerable moored gateways.

In 2004, another experiment was conducted on Seaweb in the deep water of the continental shelf. The deployed system with 40 relay nodes followed a grid topology, which covered a large area to provide wireless connections to underwater moving vehicles. The grid networks were then connected to several Racom buoys, which were deployed on the sea surface. With these surface buoys, the underwater networks were eventually connected to a command communication center that is located on a shipboard. The experiment also revealed that Seaweb could deal with bad weather in the ocean well with high reliability as it had successfully overcome two hurricanes.

Features and limitations: Seaweb is the first large-scale multi-hop off-shore field-level experimental testbed in the world, and over fifty experiments have been conducted on this over the decades. The successful experiments conducted via this testbed have greatly inspired and encouraged the underwater research community worldwide and enhanced the development of UWA-SNs. One limitation is that it is constrained with maritime applications, and it is only available to a selected community and not open to the public.

6.1.2. Ocean-TUNE. Ocean-TUNE is a collection of four testbeds placed at the east and the west sides of the US and Gulf coasts (e.g., Long Island Sound (CT), Santa Monica Bay (CA), Hood Canal (WA), and Galveston Bay (TX)), which covers large areas in the coasts of the US to provide diverse testbeds for public access [27]. The deployment areas for this experimentation platform are chosen considering geographical and environmental diversity. It was funded by the National Science Foundation (NSF) Computing Research Infrastructure (CRI) program on 2012 and was developed by four institutions in the US, UCONN, UW, University of California Los Angeles (UCLA), and Texas A&M University (TAMU). Among the individual testbeds of Ocean-TUNE, in this article, we mainly introduce Ocean-TUNE UCONN, developed at UCONN; Ocean-TUNE UW, developed at the UW; and WaterCom, developed at UCLA.

UCONN is an open access testbed that provides remote access, reconfiguration, and control facilities [82]. UCONN uses SeaLinx protocol modules to be run on embedded

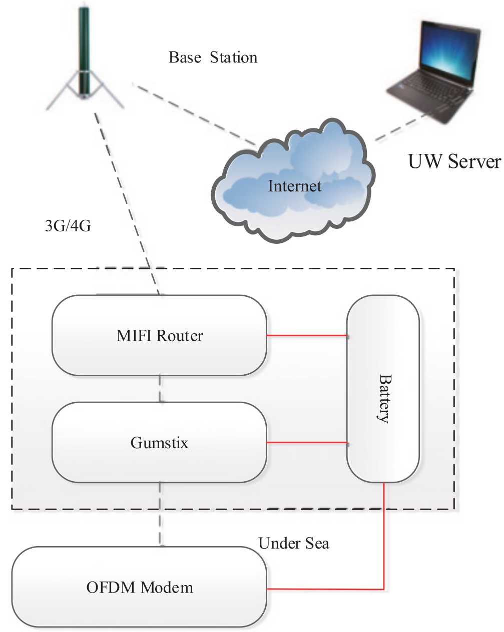

Fig. 19. The architecture of UW testbed.

Linux. The sensor nodes of UCONN are deployed on the sea surface and the bottom of the ocean; several underwater gliders act as mobile nodes in the system. The surface node is hosted on a buoy and acts as a transparent gateway with the Internet, since it has a unique IP address that is accessible through an onshore computer. The surface nodes are also equipped with three types of modems (e.g., RF (Radio Frequency) CDMA (Code Division Multiple Access), Teledyne Benthos, and OFDM modems), so they can communicate and control the nodes deployed on the sea floor via acoustic communication media. Furthermore, UCONN provides a highly reconfigurable option to researchers. With Graphical User Interface and ARC modules, researchers can remotely control or reprogram the sensor network. For example, they can remotely stop or restart ongoing experiments or can even modify application parameters as required. In addition, the testbed has an Adaptive Modulation and Coding (AMC) module to deal with highly varying communication channels. This module also allows a node to transmit data with different amounts of transmission power while studying and validating protocols and applications.

The UW testbed was developed by UW and is set up in Hood Canal [51]. The system architecture of this testbed is depicted in Figure 19. A node equipped with routing facilities is installed in the Hoodsport buoy for RF-based 3G/4G communication with a base station on the shore. The data received by the base station from the testbed are transferred to a server at the university through Internet. The testbed uses a Linux OS along with a Gumstix processor. An OFDM acoustic modem with a maximum data rate of about 9Kbps, is also the part of the testbed [123]. To avoid interference from environmental noise and waves, the modem is deployed 1 to 2m below the sea surface.

WaterCom was developed by UCLA and aims to be a multi-level testbed that has the ability to conduct small-scale, medium-scale, and large-scale experiments separately [69]. The small-scale experimentation suite is a laboratory-based testing platform, which emphasizes short-range communication associated with strong reflection

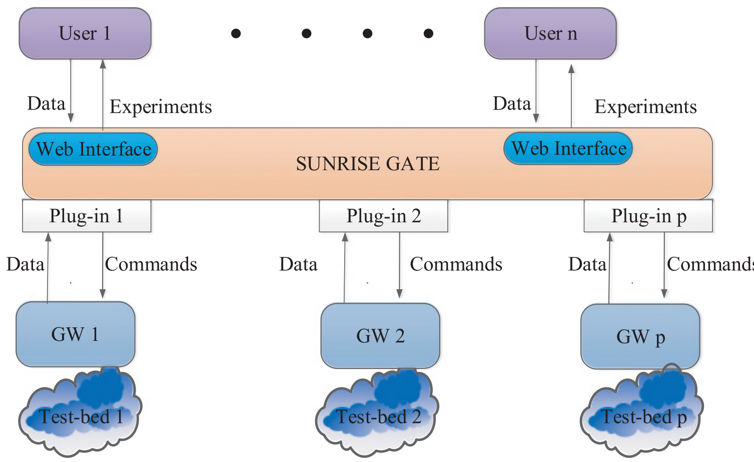

Fig. 20. The architecture of SUNRISE GATE.

and multipath signal effects. The medium-scale experimental suite is to set up a controllable experimentation environment for extended networks (that include mobile sensors) to evaluate traffic detection or other purposes. The large-scale experimental suite supports offshore open-water experimentation. Furthermore, WaterCom supports cloud-based online testing environments that can be reached at any time remotely. With its webserver and job scheduler, researchers can configure hardware or network parameters, initiate and coordinate different tasks relevant to experiments, and then obtain the test results from the testbed via the Internet. WaterCom also allows external simulators to be integrated with itself.

Features and limitations: The aforementioned four testbeds of Ocean-TUNE were deployed in four different sites around the coastline of the US. The experimentation sites were diverse in terms of weather and geography. Furthermore, Ocean-TUNE was the first highly reconfigurable large-scale open access testbed in the US that could be controlled via the Internet to facilitate underwater experiments remotely. It was also the first testbed to deploy and test high speed underwater communication media using reconfigurable multicarrier OFDM modems. Regarding limitations, there were not enough mobile sensor nodes integrated in Ocean-TUNE. Furthermore, more experiments are required to verify its three-level experimentation suites.

6.1.3. SUNRISE. The SUNRISE platform is funded by the European Union through an EU-FP7 project and the CMRE North Atlantic Treaty Organization (NATO) Science and Technology Organization. This platform aims to create an Internet of underwater things by deploying a number of testbeds in different places for long-term underwater scientific research [67]. The personnel of the research team came from universities, companies, and research labs, such as the University of Rome La Sapienza, University of Twente, University at Buffalo, EvoLogics, and SUASIS. The testbeds are located in the sea, canals, and lakes in the Mediterranean area and central parts of Europe. These testbeds support various applications, ranging from harbor monitoring and marine assets protection to marine rescue or search facilities. SUNRISE is also accessible to the researchers via Internet. As shown in Figure 20, the components of the testbed from different sites are connected to SUNRISE GATE (Gate) via gateways and plug-ins. Consequently, researchers around the world have access to the resources of different testbeds and can conduct experiments via the Internet [87]. Currently, the University of Rome La Sapienza focuses on the research involved with underwater MAC and routing

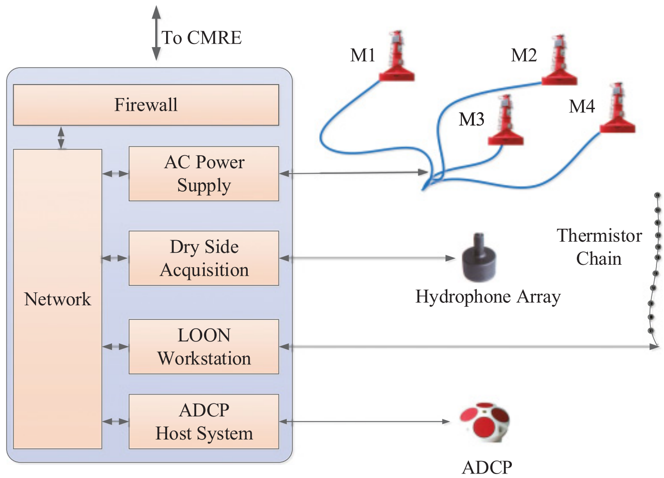

Fig. 21. The architecture of LOON testbed.

protocols. The University of Porto is trying to implement a delay-tolerant underwater network. A JANUS (Janus) physical layer protocol standard is under development by Teledyne Benthos. Since SUNRISE is still under development, we select the LOON [5] and the Porto testbed [67] to describe the features and facilities of the SUNRISE platform.

LOON is the first testbed of the SUNRISE project, developed by CMRE, and was developed based on a unified modular concept, and hence any equipment powered by 24V can be connected to it via a TCP/IP protocol. More importantly, the testbed has a remote accessibility feature that allows scientists and engineers around the world to collaboratively use the testbed for the purpose of scientific research after having authorization and authentication from CMRE. The structure of the testbed is depicted in Figure 21. M1, M2, M3, and M4 are tripod nodes that forms a star network with the control center M3, which is connected to an on-shore base station via Ethernet. These tripod nodes are equipped with commercial acoustic modems, such as WHOI, Evologics, and Teledyne Benthos. Moreover, a tetrahedral hydrophone array is connected to the shore via optical fiber. A thermistor chain is deployed to monitor the temperature of the ocean, and an Acoustic Doppler Current Profiler is deployed upward from the bottom of the ocean to measure ocean wave spectra.

The Porto testbed was developed at Porto University and is targeted for collaborative testing and evaluation of underwater network protocols using surface and underwater AUVs, moored buoys, and underwater sensors. With the centralized OpenVPN sever of the SUNRISE project, users can change operational parameters and control the nodes of the testbed to evaluate MAC, routing, and cross-layer protocols; manage localization and time synchronization issues; collect data from the environment; and so on. The key components of the testbed are AUVs, acoustic localization schemes, ASVs, a remotely operated vehicle (ROV), a Manta gateway, a shore side control station, buoys and moored sensors, and a software toolchain called DUNE (Dune Uniform Navigation Environment).

Features and limitations: SUNRISE provides open access facilities to researchers and engineers around the world after having authorization from CMRE. The testbed also supports disruption-tolerant networking (DTN) protocols besides other standard protocols [90]. Moreover, hybrid networks can be deployed on SUNRISE, especially for the applications of mobile sensor networks.

Table V. Main Features of Long-Term Field-Level Experimental Testbeds

Testbed	Primary Testbed Suites	Deployment Sites	Node Types	System Types	Number of Sea Trials	Sponsor
Seaweb	Testbeds mainly deploymed in ships	Pacific and Atlantic Oceans, Mediterranean and Baltic Seas, Norwegian fjords, and even under the Arctic ice shelf	Buoys, AUVs, Gliders, underwater nodes	Mobile networks and grid topology	50+	Seaweb
Ocean-TUNE	UCONN, UW, WaterCom, and TAMU	Long Island Sound (CT), Hood Canal (WA), Santa Monica Bay (CA), and Galveston Bay (TX)	Surface nodes, bottom nodes, and mobile nodes	Mobile and hybrid networks	10+	NSF-CRI
SUNRISE	LOON and Porto	Atlantic Ocean, Mediterranean Sea, Black Sea, and lakes and canals in Europe	ASVs, AUVs, ROVS, and underwater static nodes	Mobile, DTN and hybrid networks	10+	EU-FP7

6.2. Comparison among Long-Term Field-Level Experimentation Platforms

As shown in Table V, we choose testbed suites, deployment sites, node types, supported testbed types, the duration of sea trial, and sponsors as a list of criteria to make a comparative study among different long-term field-level experimental testbeds. Based on some more pre-specified criteria, in the following, we further review different properties of these long-term field-level experimental testbeds.

Reconfigurability and cross-layer design support: Seaweb can be rapidly deployed by taking components from a variety of platforms. It has the ability to autonomously self-configure itself into an optimal network. It reconstructs itself according to the diverse configurations of underwater sensors and Seaweb modems, which is a notion of its enhanced ability to adapt with varying environments. Both Ocean-TUNE and SUNRISE allow reconfiguration and cross-layer design support, although their primary emphasis is on field-level testing. UCONN provides a reconfigurability option to researchers and engineers via a reconfigurable OFDM underwater acoustic modem. It also adopts an AMC module to deal with temporally and spatially varying channels. Both UCONN and UW adopt protocol modules provided by SeaLinx, and hence they provide cross-layer design facilities. SUNRISE has a plug-in, named SUNRISE2SUNSET, to connect SUNSET with itself for the purpose of reconfiguration. With SUNRISE2SUNSET, researchers can reconfigure the heterogeneous nodes, measure or setup different network parameters, or configure experimental settings. Moreover, via SUNRISE2SUNSET, researchers are able to access, monitor, control, and configure the resources of testbeds to implement and test cross-layer protocols.

Efficiency and accuracy: Seaweb is the first long-term field-level testbed to demonstrate the efficiency and accuracy of underwater acoustic networks after successfully conducting more than 50 sea trials in shallow water as well as in the deep water of the Pacific and Atlantic Oceans, Mediterranean and Baltic Seas, Norwegian fjords, and even under the Arctic ice shelf. Both Ocean-TUNE and SUNRISE have mechanisms to improve the overall efficiency and accuracy of the deployed experiments. Ocean-TUNE has four testbeds deployed in different locations spanning over the coastline of the US to provide 24/7 access via the Internet in an efficient manner. In UCONN, the AMC module provides five different modes associated with different levels of transmit power and data rates, and so researchers have the flexibility to select different modes in accordance with varying channel conditions, which essentially enhances the efficiency of the experiments. For a similar reason, SUNRISE has a GATE to access, monitor, and reconfigure testbeds. To further improve the efficiency and accuracy of the testbed, as a standard part, SUNSET is integrated with SUNRISE. With a software-defined communication stack, researchers have the ability to select different working modes and configure or tune protocol parameters at runtime to achieve optimized performance of applications with the varying acoustic channel conditions and environments.

Seamless transition support: The long-term field-level experimental testbeds mainly focus on real field-level validation and testing. However, Ocean-TUNE takes SeaLinx as its protocol framework and SUNRISE has SUNSET as its standard plug-in. Consequently, both of these testbeds are allowed to adopt the seamless transition facilities from their dependant/connected platforms.

Modem support: Seaweb mainly supports Navy-restricted firmware that operates on a Benthos commercial modem, such as ATM885 telesonar. On the other hand, Ocean-TUNE currently supports two types of acoustic modems: the AquaSeNT OFDM modem and the Benthos ATM-885 modem. Moreover, the WHOI Micro-modem, Teledyne Benthos ATM 900 series modems, and Evologics are typically used by SUNRISE.

Free accessibility of the resources: Seaweb is mainly for marine applications, and it is not open for public access. Both Ocean-TUNE and SUNRISE have open access facilities for researchers. After registration, researchers can use WaterCom to conduct experiments by setting different experimental parameters via a web submission form. Once the experiment is finished, the results of the experiment are sent to researchers via an email. LOON also provides remote open access facilities to researchers for conducting experiments once they are authorized to access the testbed from CMRE.

Remote accessibility: Field-level testbeds are usually deployed in the ocean, and hence remote accessibility is an important feature, especially for the testbeds open to public access via the Internet. Seaweb provides remote control facilities to testbed nodes via underwater acoustic links from ships or RF-based links from the control center located at the shore. Both Ocean-TUNE and SUNRISE have facilities for researchers to access and control them remotely. UCONN has an ARC module and a file transfer module to deal with remote control and monitoring tasks. These tasks include reconfiguring network, transfer collected data, reconfiguring acoustic modem, reprogramming nodes, initiating or terminating experiments, selecting protocol parameters, and so on. UW also has a web-based scheduler to handle testbed experiments remotely. With a cloud-based webserver, WaterCom can be reached at any time by researchers to control the testbed (e.g., to set network parameters, to schedule experiment-related tasks, and to acquire results and status information). SUNRISE has a unified web interface GATE and a control shell for researchers to access, control, reconfigure testbeds, and run experiments remotely after having authentication from SUNRISE GATE.

Deployment cost: Long-term field-level experimental testbeds are the most expensive platforms compared to other platforms due to high boat rental fees, high offshore deployment costs, as well as expensive underwater hardware components, static and mobile sensors, and so on. Seaweb belongs to the first generation of long-term large-scale field-level experimental testbeds (since 1995), and more than 50 experiments have been conducted on it. With Seaweb, many researchers have gained practical experiences with underwater digital communication. Although it is constrained to military applications, Seaweb has given the underwater research community enormous confidence and has greatly enhanced research and development in UWA-SNs. We consider both Ocean-TUNE and SUNRISE as the second generation of long-term field-level testbeds that are still under development. These testbeds actually aim to explore the next-generation underwater acoustic sensor networks with open access facilities for underwater research community. Ocean-TUNE is the first large-scale open underwater networking testbed in the US, while SUNRISE is the largest open access underwater testbed for the European Union.

Table VI. The Download Sites of Open Source Simulation Tools

Simulation Tool	Download Site
Aqua-Sim	http://obinet.engr.uconn.edu/wiki/index.php/Aqua-sim
UnetStack	http://www.unetstack.net/doc/html/downloads.html
DESERT	http://nautilus.dei.unipd.it/desert-underwater
SUNSET	http://svnreti.di.uniroma1.it/SUNSET/

7. RECOMMENDATIONS ON CHOOSING SIMULATION AND EXPERIMENTATION PLATFORMS

As mentioned earlier, to evaluate, validate, or test underwater protocols and applications, researchers may need lab-level or field-level, short-term, or even long-term experimentation platforms at different stages of the development process. From the perspective of users, they may consider many different criteria while choosing a suitable platform. Moreover, each platform has its own pros and cons, and hence it is a difficult task to make this decision. In this section, we provide guidelines in terms of the following metrics to help researchers in selecting the desired simulation tools and testbeds.

—*Free accessibility of the resources:* The early development of UWA-SNs is usually related to maritime applications, and so most of the underwater experimentation platforms are not open to the public. As the development for commercial applications increases, researchers prefer to choose open source simulation tools and freely accessible testbeds. Among the 17 simulation and experimentation platforms, currently only 4 of them are available via the Internet, and 1 field-level testbed provides open remote access facility via the Internet. The four open source simulation tools are Aqua-Sim, UnetStack, DESERT, and SUNSET, and the free downloadable websites for them are shown in Table VI. After initial release, new versions came out that added new features and components. For example, the latest versions of SUNSET, DESERT, and UnetStack are 10, 2.1.2, and 1.3, respectively. As for long-term field-level experimental testbeds, currently only LOON in SUNRISE provides an open remote access facility to researchers. One crucial objective of SUNRISE is to share its testbed components with scientists and researchers around the world and encourage collaborative research on UWA-SNs. To have access to LOON, researchers just need to get the authorization certificate from CMRE.

—*Overall cost and scalability:* When researchers set up their own underwater test environment or select experimentation platform, they prefer to choose the cheaper one. Generally, the lab-level testbeds, such as Aqua-Lab, UPPER, SeaNet, UANT, Aqua-Sim, and UW-Buffalo, are low-cost experimentation platforms and can accommodate small-scale experiments. Since Aqu-lab, UPPER, and SeaNet use cheaper commercial components at a relatively low price, researchers can learn the architecture of those platforms easily to set up their own simple underwater test environment. For beginners, a simple lab-level testbed may be more than enough, and it is the cheapest one. However, using short-term experimentation platforms, one can conduct simulation, emulation, and field-level testing. Aqua-Net, SeaLinx, UnetStack, DESERT, and SUNSET are in this group. With the simulation tools in these platforms, researchers can implement medium- to large-scale experiments at a relatively low cost. Long-term field-level experimentation is the most expensive approach for protocol verification and testing. Lab-level testing, emulation, and simulation can help researchers in the applications/protocols design process. However, since underwater applications will eventually be deployed in the ocean, we still need to test and verify the underwater protocols in these expensive field-level experimentation platforms before real-world deployment. Ocean-TUNE and SUNRISE are long-term field-level testbeds, and WaterCom in Ocean-TUNE tries to provide a multi-level testing platform, which includes small-, medium-, and large-scale testing suites.

—*Reconfigurability and cross-layer design support:* A reconfigurable testbed provides flexibility to researchers in selecting functional modes or adjusting parameters of hardware and software components. Consequently, they can program the testbed as necessary to find out the factors that have great influence on specific applications. Cross-layer design support is another important facility for researchers to design optimized protocols by sharing information across different layers. This facility especially helps researchers to deal with spatio-temporally variant underwater channels by making the information of lower layers available to the higher layers. UANT, UW-Buffalo, SeaNet, Aqua-Net, SeaLinx, UnetStack, DESERT, and SUNSET provide this facility by adopting different approaches or by using different hardware or software components. UANT uses GNU radio to reconfigure three protocol layers in the stack, such as physical, MAC, and network layers. UW-Buffalo carries out cross-layer design by linking a cross-layer controller module and a reconfigurable Teledyne modem. Therefore, the optimized design can be verified with real underwater modems. Using SDR, SeaNet provides fully reconfiguration facility across four layers, such as physical, data-link, network and application layers. Aqua-Net has a cross-layer interface module, and SeaLinx has SeaLinx Core to facilitate this feature. UnetStack uses an agent-based information sharing mechanism to provide cross-layer design support. Finally, ns2-Miracle provides a cross-layer design facility to both SUNSET and DESERT.

—*Adaptability with dynamic channels and modem support:* Many simulation and experimentation platforms adopt complex underwater channel models to handle channel adaptability issues and improve the accuracy of simulation or experiments. Aqua-Sim integrates Rogers and PE models that are sensitive to shallow waters. Both SUNSET and DESERT enhance channel adaptability by implementing a Bellop ray-tracing software model with WOSS. UCONN adopts an AMC scheme to tune communication parameters in accordance with the change of channels. By using software defined communication stack, SUNRISE can adapt with dynamic channels in a better way. More specifically, this communication stack allows researchers to select different functional modes, configure or tune protocol parameters, and so on, to optimize the performance of deployed applications. A real underwater acoustic modem can improve the performance of testbed emulation, and it is crucial for testbeds to support different types of modems that are compatible with different communication media. For example, SUNSET supports five different commercial modems. Except for UPPER and UANT, all other testbeds support commercial or experimental modems.

—*Seamless transition support and turnaround period:* The testbeds that support SEFT facilitate in designing any new protocol or application and can shorten the overall design turnaround period. Furthermore, seamless transition among simulation, emulation, and field-level testing enables researchers in reusing code and speeds up the overall design process and improve design efficiency. Seamless transition implies that there is no any inconsistency among simulation, emulation, and field-level testing at the first step. Second, significant amounts of code can be reused, and it incurs minimal effort while exporting code from simulation to emulation and field-level testing, however, in different approaches. UnetStack, DESERT, and SUNSET support SEFT and seamless transition. Both SUNSET and DESERT are developed on ttop of ns-2 and ns2-Miracle, which are primarily discrete-event simulators. Therefore, while transferring from discrete-event-based simulation to distributed real-time emulation on these platforms, the transition may require significant changes in code and design that may bring additional problems. For example, if centralized global network information is used in the simulation of certain application, while conducting emulation of the same application, special caution needs to be taken when the code is transferred from the purpose of simulation to emulation. For example, it is difficult to identify event-timing related problems in the simulation, and so, while exporting the code written for the simulation to the emulation, many consistency-related problems may happen in this context. UnetStack has an agent-based architecture, and supports real-time simulation. Therefore, the same compiled binary code blocks used for the simulation can be directly deployed on UnetStack-compatible underwater modems without cross-compilation for the purpose of emulation or field-level testing.

—*Remote accessibility:* A remote access and control feature is important, especially for UIOTs (Underwater Internet of Things). UW-Buffalo, UnetStack, DESERT, and SUNSET are remotely controllable and reconfigurable testbeds. UW-Buffalo has an IP-compatible protocol stack to facilitate underwater Internet access. UnetStack has a flexible agent-based architecture, and its remote access agent with a TCP/IP interface provides remote access and control services, which include querying or setting parameters, transferring files and running scripts remotely, and so on. DESERT utilizes RECORDS to remotely control and reprogram the entire multi-hop network. RECORDS is an open source software that can be downloaded from the Internet, and it can easily and conveniently be exported to embedded systems without cross-compilation. SUNSET uses a back-seat driver to remotely control hardware in the testbeds that include static and mobile nodes (e.g., AUVs and ASVs). Via the back-seat driver, researchers can remotely change network topology, activate or configure sensor nodes, initiate or terminate experiments, change experiment parameters, or even switch among different tests easily and quickly.

Ocean-TUNE and SUNRISE support convenient remote access and control facilities for field-level experiments. UCONN has an ARC module to provide remote control and monitoring capability that includes a reprogram node and a start or stop program and that can adjust protocol parameters, file transfer, and so on. WaterCom supports cloud-based experimentation nodes, which are online at any time. Researchers can connect to a cloud-based webserver to submit experiment-related tasks, set up network parameters, wait for the execution, and obtain results and status information. SUNRISE provides a unified web interface, namely GATE, for researchers to access, control, reconfigure testbeds, and run experiments remotely. It also provides a control shell for advanced users to effectively access the low-level, complex, and enhanced functionalities after having authentication from SUNRISE GATE.

—*User-friendliness and learning cycle:* The simulation and field-level testbeds should have a short learning cycle. A steep learning curve implies that researchers need a significant amount of time and effort to get familiar with testbeds. While choosing testbeds, researchers prefer shorter learning curves, so they need to spend less time on mastering them and can speed up application development. The characteristics of the learning curve also depend on the background and knowledge of the researchers. In other words, the same tool may produce different types of learning curves for different researchers. Both DESERT and SUNSET are extended from the well-known open source simulator ns-2, which has a very large community and users. Therefore, researchers who are familiar with ns-2 need not learn a new software framework in case they choose to use DESERT or SUNSET. The user-friendliness is another important criterion, which implies the degree of overall comfort level related to the usage of the testbed. To enhance user-friendliness, testbeds should have user-friendly interfaces so researchers can use those in a convenient manner.

8. FURTHER DISCUSSION AND FUTURE RESEARCH

In recent years, although we have seen significant advancements in developing simulation and experimentation platforms to solve the emerging research problems in UWA-SNs, there are still many important issues and challenges that need to be addressed in this context. In this section, we provide a few suggestions and some possible problems on future research.

8.1. Balance between Simulation and Field-Level Testing

To evaluate protocols or algorithms while developing real applications for UWA-SNs, it is cost-effective to identify potential problems at the simulation phase before expensive field-level deployment. Therefore, determining a way to reduce the gap between simulation and real field-level experiments, and to seamlessly transport simulation code to real devices at the field without rewriting much code is becoming an interesting research topic. Consequently, the feature that integrates simulation and field-level experiments together, and allows transition seamlessly from simulation and emulation to field-level testing, also becomes an important metric of testbed design. Furthermore, it is desireable to not have to change the code blocks when transferring from the simulation to field-levels experiments. Another important issue related to the seamless transformation is to develop an accurate underwater acoustic propagation model [105] that can be used in the simulation platforms to simulate different scenarios and environments thoroughly before real field-level testing. Moreover, to refine the simulation results for future field-level testing may require tracing the actual environments of deployment sites into the simulation.

8.2. Comparison among Simulation and Experimentation Platforms

With the development of more and more simulation and experimentation platforms, more comparison-based studies on some uniform criteria and the corresponding guidelines for researchers are required. This is because each platform has its own pros and cons, and the protocol or application evaluation process is directly affected by that. Research has been conducted recently in this context. For example, the comparison-based studies between SUNSET and SeaLinx, and SUNSET with DESERT, addressed several performance metrics, such as accuracy of the simulation, energy efficiency, memory overhead, and so on. However, these studies are far from sufficient. Besides the aforementioned metrics, other metrics for comparison-based studies could include free accessibility of the resources, scalability, deployment costs, the seamless transition from simulation to field-level testing, channel adaptability, reconfiguration ability, user-friendliness, and so on. Note that, due to the diverse and complex nature of different experimentation platforms, finding a unified comparison strategy among them in an efficient and fair manner is still an open problem [126].

8.3. Optimized Simulation and Experimentation

One of the challenges in designing and testing underwater applications or protocols is dealing with fast spatio-temporally variant underwater acoustic channels [10, 96, 105]. A promising and appealing method to optimize the evaluation of underwater applications and protocols efficiently is AMC [116, 119]. Another method is cross-layer design that utilizes the information of different protocol layers to optimize the evaluation process of different applications [91, 93, 115]. To deal with temporally and spatially varying underwater channels, SDR has recently emerged as a potential technology to design intelligent and self-adaptive testbed environments. Consequently, finding a way to take the advantages of software defined underwater acoustic technology by combining SDM and cross-layer design facility is also an interesting research topic.

8.4. Remote Access and Control Facilities

It is very expensive to set up and maintain simulation and field-level experimentation platforms deployed in the ocean. To lower the evaluation cost of any new application or protocol, it is best to deploy small-, medium-, and large-scale experimentation platforms that are available to a broader group of researchers around the world. The testbed should be accessible remotely via the Internet so all resources in the testbeds, such as static surface sensors, underwater sensors, mobile sensors, AUVs, ASVs, and ROVs, are available to users via a unified and standard interface. More importantly, these software and hardware components should be reconfigurable remotely so researchers can book and use them to conduct research seamlessly via simulation, emulation, and sea-site testing.

8.5. Specific Testbed Design

Currently, most of the aforementioned testbeds are for general testing and evaluation of underwater communication protocols. We propose to design more specific testbeds targeted for specific research (e.g., research on network lifetime and diverse applications, underwater spatial channel reuse, mobile sensor networks, or even the next-generation UWA-SNs). These testbeds can adapt to specific research requirements to achieve maximum results. To further enhance UWA-SN development, many researchers have proposed some novel paradigms especially for the next-generation UWA-SNs [3, 29]. For example, since the existing underwater communication networks take a hardware-based inflexible architecture, in [3], the authors proposed a software-defined networking paradigm, SoftWater, which is a virtualizable network architecture with high network resource utilization and flexibility features. Based on these novel proposed paradigms, more testbeds should be developed to facilitate the development of the next-generation underwater sensor networks.

8.6. Smart Testbed Design

Until now, underwater acoustic sensor network testbeds have not been smart enough to conduct sophisticated simulation and experiments intelligently. In order to acquire the optimized performance, cross-layer design is the most useful mechanism. As sensor nodes are becoming more powerful with larger computational abilities, the advancements in artificial intelligence (AI) technology recently have created new opportunities for progress in UWA-SNs to make them more intelligent in the near future [38, 50]. For example, more machine-learning-oriented protocols or algorithms (e.g., reinforcement learning) have been designed to increase the network real-time, self-adaptability with the harsh ocean environments, as well as to maximize the utilization of network resources. To cope with this research trend, we need to create and design AI-supported simulation and experimentation platforms. For example, we may design deep learning-based platforms that support pre-training for underwater sensor nodes with automatic extracted features from the targeted ocean area to increase node intelligence in terms of environmental adaptability.

9. CONCLUSION

In this article, we present a survey on underwater simulation tools and experimental testbeds. We first categorize 17 typical simulation and experimental platforms into three groups, and then introduce the main components of these platforms. To meet the main objective of this survey, we discuss the architecture, features, and limitations of each platform separately for each group. Within the same group, we further provide a comparative study among the selected platforms. Based on the comparative study, we present some advice for researchers on choosing suitable platforms for their applications. At the end of this survey, based on the research trend and our findings, we provide some directions on future research in this context. We believe that SDR is becoming an emerging technology that has many potential benefits to deal with spatio-temporally variant underwater communication channels and in designing fully reconfigurable, flexible, and self-adaptive testbeds for the next generation UWA-SNs.

More effort should be put into the research on designing shared open source standard platforms, which can be reached from anywhere in the world conveniently via the Internet. Such platforms will allow researchers to reconfigure their resources and carry out experiments remotely with reliable results at relatively low cost. Moreover, we believe that more research efforts on simulation and experimentation platforms will speed up the development of UWA-SNs.

ACKNOWLEDGMENTS

We thank the reviewers and the editors for their valuable advice and the detailed comments.

REFERENCES

[1] Niaz Ahmed, Waqas bin Abbas, and Affan A. Syed. 2012. A low-cost and flexible underwater platform to promote experiments in UWSN research. In *Proceedings of the 7th ACM International Conference on Underwater Networks and Systems*. ACM, 1–4.

[2] Ian F. Akyildiz, Dario Pompili, and Tommaso Melodia. 2005. Underwater acoustic sensor networks: Research challenges. *Ad Hoc Netw.* 3, 3 (2005), 257–279.

[3] Ian F. Akyildiz, Pu Wang, and Shih-Chun Lin. 2016. SoftWater: Software-defined networking for next-generation underwater communication systems. *Ad Hoc Netw.* 46 (2016), 1–11.

[4] Ben Allen, Tom Austin, Ned Forrester, Rob Goldsborough, Amy Kukulya, Greg Packard, Mike Purcell, and Roger Stokey. 2006. Autonomous docking demonstrations with enhanced REMUS technology. *OCEANS 2006* (2006), 1–6.

[5] Joao Alves, John Potter, Piero Guerrini, Giovanni Zappa, and Kevin LePage. 2014. The LOON in 2014: Test bed description. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–4.

[6] AppliedOceanSystems. 2016. SAM1 Wireless Subsea Acoustic Modem. (2016). Retrieved from http://www.applied-ocean.com.

[7] AquaSeNT. 2016. AquaSeNT Subsea Acoustic Wi-Fi. Retrieved from http://www.aquasent.com.

[8] Muhammad Ayaz, Imran Baig, Azween Abdullah, and Ibrahima Faye. 2011. A survey on routing techniques in underwater wireless sensor networks. *J. Netw. Comput. Appl.* 34, 6 (2011), 1908–1927.

[9] Mehmet Aydinlik, A. Turan Ozdemir, and Milica Stajanovic. 2008. A physical layer implementation on reconfigurable underwater acoustic modem. In *OCEANS 2008*. IEEE, 1–4.

[10] Mohsen Badiey, Yongke Mu, Jeffrey Simmen, Steve E. Forsythe, and others. 2000. Signal variability in shallow-water sound channels. *IEEE J. Ocean. Eng.* 25, 4 (2000), 492–500.

[11] Nicola Baldo, Federico Maguolo, Marco Miozzo, Michele Rossi, and Michele Zorzi. 2007. ns2-MIRACLE: A modular framework for multi-technology and cross-layer support in network simulator 2. In *Proceedings of the 2nd International Conference on Performance Evaluation Methodologies and Tools*. 16.

[12] Waqas Bin Abbas, Niaz Ahmed, Chaudhry Usama, and Affan A. Syed. 2015. Design and evaluation of a low-cost, DIY-inspired, underwater platform to promote experimental research in UWSN. *Ad Hoc Netw.* 34 (2015), 239–251.

[13] Brian Borowski and Dan Duchamp. 2010. Measurement-based underwater acoustic physical layer simulation. In *Proceedings of MTS/IEEE OCEANS10.*

[14] Henrique Cabral. 2014. Acoustic modem for underwater communication. Master's thesis. Faculdade de Engenharia da Universidade do Porto, Porto, Portugal.

[15] Filippo Campagnaro, Roberto Francescon, Federico Guerra, Federico Favaro, Paolo Casari, Roee Diamant, and Michele Zorzi. 2016. The DESERT underwater framework v2: Improved capabilities and extension tools. In *Proceedings of the 2016 IEEE 3rd Underwater Communications and Networking Conference (UComms)*. IEEE, 1–5.

[16] Edward A. Carlson, Pierre-Philippe Beaujean, and Edgar An. 2004. Simulating communication during multiple AUV operations. In *2004 IEEE/OES Autonomous Underwater Vehicles*. IEEE, 76–82.

[17] Paolo Casari, Cristiano Tapparello, Federico Guerra, Federico Favaro, Ivano Calabrese, Giovanni Toso, Saiful Azad, Riccardo Masiero, and Michele Zorzi. 2014. Open source suites for underwater networking: WOSS and DESERT underwater. *IEEE Netw.* 28, 5 (2014), 38–46.

[18] Vijay Chandrasekhar, Winston K. G. Seah, Yoo Sang Choo, and How Voon Ee. 2006. Localization in underwater sensor networks: Survey and challenges. In *Proceedings of the 1st ACM International Workshop on Underwater Networks (UUWNet'06)*. 33–40.

[19] Keyu Chen, Maode Ma, En Cheng, Fei Yuan, and Wei Su. 2014. A survey on MAC protocols for underwater wireless sensor networks. *IEEE Commun. Surv. Tutor.* 16, 3 (2014), 1433–1447.

[20] Mandar Chitre, Rohit Bhatnagar, Manu Ignatius, and Shailabh Suman. 2014b. Baseband signal processing with UnetStack. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–4.

[21] Mandar Chitre, Rohit Bhatnagar, and Wee-Seng Soh. 2014a. UnetStack: An agent-based software stack and simulator for underwater networks. In *Oceans-St. John's, 2014*. IEEE, 1–10.

[22] Mandar Chitre, Shiraz Shahabudeen, and Milica Stojanovic. 2008. Underwater acoustic communications and networking: Recent advances and future challenges. *Mar. Technol. Soc. J.* 42, 1 (2008), 103–116.

[23] Mandar Chitre, Iulian Topor, and Teong-Beng Koay. 2012. The UNET-2 modemAn extensible tool for underwater networking research. In *OCEANS, 2012-Yeosu*. IEEE, 1–7.

[24] Salvador Climent, Antonio Sanchez, Juan Vicente Capella, Nirvana Meratnia, and Juan Jose Serrano. 2014. Underwater acoustic wireless sensor networks: Advances and future trends in physical, MAC and routing layers. *Sensors* 14, 1 (2014), 795–833.

[25] Nuno A. Cruz, Bruno M. Ferreira, Oleksiy Kebkal, Aníbal C. Matos, Chiara Petrioli, Roberto Petroccia, and Daniele Spaccini. 2013. Investigation of underwater acoustic networking enabling the cooperative operation of multiple heterogeneous vehicles. *Mar. Technol. Soc. J.* 47, 2 (2013), 43–58.

[26] Jun-Hong Cui, J. Kong, M. Gerla, and S. Zhou. 2006. The challenges of building scalable mobile underwater wireless sensor networks for aquatic applications. *IEEE Netw.* 20, 3 (2006), 12.

[27] Jun-Hong Cui, Shengli Zhou, Zhijie Shi, James O'Donnell, Zheng Peng, Mario Gerla, Burkard Baschek, Sumit Roy, Payman Arabshahi, and Xi Zhang. 2012. Ocean-TUNE: A community ocean testbed for underwater wireless networks. In *Proceedings of the 7th ACM International Conference on Underwater Networks and Systems*. ACM, 17.

[28] Emrecan Demirors, Bharatwaj G. Shankar, G. Enrico Santagati, and Tommaso Melodia. 2015. SEANet: A software-defined acoustic networking framework for reconfigurable underwater networking. In *WUWNET'15.*

[29] Emrecan Demirors, George Sklivanitis, G. Enrico Santagati, Tommaso Melodia, and Stella N. Batalama. 2014. Design of a software-defined underwater acoustic modem with real-time physical layer adaptation capabilities. In *Proceedings of the International Conference on Underwater Networks & Systems*. ACM, 25.

[30] Develogic. 2016. DevelogicIntegrated subsea solution. Retrieved from http://www.develogic.de.

[31] Henry Dol, Paolo Casari, and Timo van der Zwan. 2014. Software-defined open-architecture modems: Historical review and the NILUS approach. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–5.

[32] Henry Dol, Paolo Casari, Timo van der Zwan, and Roald Otnes. 2017. Software-defined underwater acoustic modems: Historical review and the NILUS approach. *IEEE Journal of Oceanic Engineering* (2017), 1–16. DOI:http://dx.doi.org/10.1109/JOE.2016.2598412, pending publication.

[33] Katherine Domrese, Andrew Szajna, Shengli Zhou, and Jun-Hong Cui. 2014. Comparison of the ranging function of three types of underwater acoustic modems. In *Proceedings of the 2014 IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 743–748.

[34] DSPComm 2016. AquaCommUnderwater wireless modem. Retrieved from http://www.dspcomm.com.

[35] Melike Erol-Kantarci, Hussein T. Mouftah, and Sema Oktug. 2011. A survey of architectures and localization techniques for underwater acoustic sensor networks. *IEEE Commun. Surv. Tutor.* 13, 3 (2011), 487–502.

[36] EvoLogics 2016. EvoLogicsUnderwater Acoustic Modems. (2016). Retrieved from http://www.evologics.de.

[37] Ruolin Fan, Li Wei, Pengyuan Du, Ciarán Mc Goldrick, and Mario Gerla. 2016. A SDN-controlled underwater MAC and routing testbed. In *IEEE Military Communications Conference (MILCOM 2016-2016)*. IEEE, 1071–1076.

[38] Eric L. Ferguson, Rishi Ramakrishnan, Stefan B. Williams, and Craig T. Jin. 2016. Deep learning approach to passive monitoring of the underwater acoustic environment. *J. Acoust. Soc. Am.* 140, 4 (2016), 3351–3351.

[39] Eric Gallimore, Jim Partan, Ian Vaughn, Sandipa Singh, Jon Shusta, and Lee Freitag. 2010. The WHOI micromodem-2: A scalable system for acoustic communications and networking. In *OCEANS 2010*. IEEE, 1–7.

[40] GNUradio 2016. E. Blossom Exploring GNU Radio. (2016). Retrieved from http://www.gnu.org/software/gnuradio.

[41] Federico Guerra, Paolo Casari, and Michele Zorzi. 2009. World ocean simulation system (WOSS): A simulation tool for underwater networks with realistic propagation modeling. In *Proceedings of the 4th ACM International Workshop on UnderWater Networks*. ACM, 4.

[42] GumstixDeveloperCenter 2016. Gumstix Developer Center. Retrieved from http://www.gumstix.org.

[43] Guangjie Han, Chenyu Zhang, Lei Shu, Ning Sun, and Qingwu Li. 2013. A survey on deployment algorithms in underwater acoustic sensor networks. *Int. J. Distrib. Sens. Netw.* 9, 12 (2013), 314049.

[44] John Heidemann, Milica Stojanovic, and Michele Zorzi. 2012. Underwater sensor networks: Applications, advances and challenges. *Philos. Trans. Roy. Soc. A* 370, 1958 (2012), 158–175.

[45] John Heidemann, Wei Ye, Jack Wills, Affan Syed, and Yuan Li. 2006. Research challenges and applications for underwater sensor networking. In *Proceedings of the IEEE Wireless Communications and Networking Conference 2006 (WCNC 2'06)*, Vol. 1. IEEE, 228–235.

[46] Feng Hong, Bozhen Yang, Yuliang Zhang, Ming Xu, Yuan Feng, and Zhongwen Guo. 2014. Time synchronization for underwater sensor networks based on multi-source beacon fusion. In *Proceedings of the 2014 20th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 218–224.

[47] Feng Hong, Yu-liang Zhang, Bo-zhen Yang, Ying Guo, and uo. 2013. Review on time synchronization techniques in underwater acoustic sensor networks [J]. *Acta Electron. Sinica* 5 (2013), 020.

[48] Lu Hong, Feng Hong, Bozhen Yang, and Zhongwen Guo. 2013. ROSS: Receiver oriented sleep scheduling for underwater sensor networks. In *Proceedings of the Eighth ACM International Conference on Underwater Networks and Systems*. ACM, 4.

[49] Jens Horneber and Anton Hergenröder. 2014. A survey on testbeds and experimentation environments for wireless sensor networks. *IEEE Commun. Surv. Tutor.* 16, 4 (2014), 1820–1838.

[50] Tiansi Hu and Yunsi Fei. 2010. QELAR: A machine-learning-based adaptive routing protocol for energy-efficient and lifetime-extended underwater sensor networks. *IEEE Trans. Mobile Comput.* 9, 6 (2010), 796–809.

[51] Shae Hurst, Xinyu Xie, Shwan Ashrafi, Sumit Roy, and Payman Arabshahi. 2014. Puget sound underwater networking TestBed. In *Oceans-St. John's, 2014*. IEEE, 1–5.

[52] Muhammad Imran, Abas Md Said, and Halabi Hasbullah. 2010. A survey of simulators, emulators and testbeds for wireless sensor networks. In *Proceedings of the 2010 International Symposium in Information Technology (ITSim)*, Vol. 2. IEEE, 897–902.

[53] Jules Jaffe and Curt Schurgers. 2006. Sensor networks of freely drifting autonomous underwater explorers. In *Proceedings of the 1st ACM International Workshop on Underwater Networks*. ACM, 93–96.

[54] Emma Jones. 2007. The application of software radio techniques to underwater acoustic communications. In *OCEANS 2007-Europe*. IEEE, 1–6.

[55] Hovannes Kulhandjian, L. Kuo, Tommaso Melodia, Dimitris A. Pados, Dale Green, and Teledyne Benthos. 2012. Towards experimental evaluation of software-defined underwater networked systems. *Proc. IEEE UComms* (2012), 1–9.

[56] Son N. Le, Zheng Peng, Jun-Hong Cui, Hao Zhou, and Janny Liao. 2013. Sealinx: A multi-instance protocol stack architecture for underwater networking. In *Proceedings of the 8th ACM International Conference on Underwater Networks and Systems*. ACM, 46.

[57] LinkQuest 2016. LinkQuestUnderwater Acoustic Modems. Retrieved from www.link-quest.com.

[58] Lanbo Liu, Shengli Zhou, and Jun-Hong Cui. 2008. Prospects and problems of wireless communication for underwater sensor networks. *Wireless Commun. Mobile Comput.* 8, 8 (2008), 977–994.

[59] Hanjiang Luo, Yuejiao Gong, Lionel Ni, and others. 2016. Localization for drifting restricted floating ocean sensor networks. *IEEE Trans. Vehic. Technol.* 65, 12 (2016), 9968–9981.

[60] Hanjiang Luo, Zhongwen Guo, Wei Dong, Feng Hong, and Yiyang Zhao. 2010. LDB: Localization with directional beacons for sparse 3D underwater acoustic sensor networks. *J. Netw.* 5, 1 (2010), 28–38.

[61] Hanjiang Luo, Zhongwen Guo, Kaishun Wu, Feng Hong, and Yuan Feng. 2009. Energy balanced strategies for maximizing the lifetime of sparsely deployed underwater acoustic sensor networks. *Sensors* 9, 9 (2009), 6626–6651.

[62] Hanjiang Luo, Kaishun Wu, Zhongwen Guo, Lin Gu, and Lionel M Ni. 2012. Ship detection with wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* 23, 7 (2012), 1336–1343.

[63] Hanjiang Luo, Kaishun Wu, Zhongwen Guo, Lin Gu, Zhong Yang, and Lionel M Ni. 2011. Sid: Ship intrusion detection with wireless sensor networks. In *Proceedings of the 2011 31st International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 879–888.

[64] Hanjiang Luo, Kaishun Wu, Jiang Xiao, and Zhongwen Guo. 2014. LDSN: Localization scheme for double-head maritime sensor networks. In *Proceedings of the 20th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 233–240.

[65] Hanjiang Luo, Yiyang Zhao, Zhongwen Guo, Siyuan Liu, Pengpeng Chen, and L.M. Ni. 2008. UDB: Using directional beacons for localization in underwater sensor networks. In *Proceedings of the 14th IEEE International Conference on Parallel and Distributed Systems, 2008 (ICPADS'08)*. 551–558.

[66] Robert Martin, Yibo Zhu, Lina Pu, Fei Dou, Zheng Peng, Jun-Hong Cui, and Sanguthevar Rajasekaran. 2015. Aqua-sim next generation: A NS-3 based simulator for underwater sensor networks. In *Proceedings of the 10th International Conference on Underwater Networks & Systems*. ACM, 18.

[67] Ricardo Martins, Joao Borges de Sousa, Renato Caldas, Chiara Petrioli, and John Potter. 2014. SUNRISE project: Porto university testbed. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–5.

[68] Riccardo Masiero, Saiful Azad, Federico Favaro, Matteo Petrani, Giovanni Toso, Federico Guerra, Paolo Casari, and Michele Zorzi. 2012. DESERT underwater: an NS-miracle-based framework to DEsign, simulate, emulate and realize test-beds for underwater network protocols. In *OCEANS 2012*. IEEE, 1–10.

[69] Ciarán Mc Goldrick, Mark Matney, Enrique Segura, Youngtae Noh, and Mario Gerla. 2015. WaterCom: A multilevel, multipurpose underwater communications test platform. In *WUWNET'15*.

[70] Bartosz Musznicki and Piotr Zwierzykowski. 2012. Survey of simulators for wireless sensor networks. *Int. J. Grid Distrib. Comput.* 5, 3 (2012), 23–50.

[71] Youngtae Noh, Dustin Torres, and Mario Gerla. 2015. Software-defined underwater acoustic networking platform and its applications. *Ad Hoc Netw.* (2015).

[72] Nusrat Nowsheen, Craig Benson, and Michael Frater. 2010. A high data-rate, software-defined underwater acoustic modem. In *OCEANS 2010*. IEEE, 1–5.

[73] Ns-2. 2016. Ns-2. Retrieved from http://www.isi.edu/nsnam/ns.

[74] Ns-3. 2016. Ns-3. Retrieved from http://www.nsnam.org.

[75] Ns2-Miracle. 2016. NS-MiracleThe Network Simulator: NS-Miracle. Retrieved from http://dgt.dei.unipd.it/download.

[76] OMNet 2016. OMNetOMNeT++ Community Site. Retrieved from http://www.omnetpp.org.

[77] OPNET 2014. OPNET OPNET Technologies. Retrieved from http://www.opnet.com.

[78] Jim Partan, Jim Kurose, and Brian Neil Levine. 2006. A survey of practical issues in underwater networks. In *Proceedings of ACM WUWNet'06*. 17–24.

[79] Zheng Peng, Jun-Hong Cui, Bing Wang, Keenan Ball, and Lee Freitag. 2007. An underwater network testbed: Design, implementation and measurement. In *Proceedings of the 2nd Workshop on Underwater Networks*. ACM, 65–72.

[80] Zheng Peng, Son Le, Michael Zuba, Haining Mo, Hao Zhou, Jun-Hong Cui, Shengli Zhou, Zaihan Jiang, and Jeffrey A Schindall. 2013. Field test experience of an underwater wireless network in the atlantic ocean. In *Proceedings of the 2013 MTS/IEEE OCEANS*. IEEE, 1–10.

[81] Zheng Peng, Son Le, Michael Zuba, Haining Mo, Yibo Zhu, Lina Pu, Jun Liu, and Jun-Hong Cui. 2011. Aqua-TUNE: A testbed for underwater networks. In *Proceedings of the IEEE OCEANS, 2011*. IEEE, 1–9.

[82] Zheng Peng, Li Wei, Zigeng Wang, Lei Want, Michael Zuba, Jun-Hong Cui, Shengli Zhou, Zhijie Shi, and James O'Donnell. 2014. Ocean-TUNE UCONN testbed: A technology incubator for underwater communication and networking. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–4.

[83] Zheng Peng, Zhong Zhou, Jun-Hong Cui, and Zhijie Jerry Shi. 2009. Aqua-Net: An underwater sensor network architecture: Design, implementation, and initial testing. In *MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges (OCEANS 2009)*. IEEE, 1–8.

[84] Chiara Petrioli and Roberto Petroccia. 2012. SUNSET: Simulation, emulation and real-life testing of underwater wireless sensor networks. *Proceedings of IEEE UComms 2012* (2012), 12–14.

[85] Chiara Petrioli, Roberto Petroccia, John R. Potter, and Daniele Spaccini. 2015. The SUNSET framework for simulation, emulation and at-sea testing of underwater wireless sensor networks. *Ad Hoc Netw.* 34 (2015), 224–238.

[86] Chiara Petrioli, Roberto Petroccia, and Daniele Spaccini. 2013. SUNSET version 2.0: Enhanced framework for simulation, emulation and real-life testing of underwater wireless sensor networks. In *Proceedings of the 8th ACM International Conference on Underwater Networks and Systems*. ACM, 43.

[87] Chiara Petrioli, Roberto Petroccia, Daniele Spaccini, A. Vitaletti, Tommaso Arzilli, Davide Lamanna, Alessandro Galizia, and Enrico Renzi. 2014. The SUNRISE GATE: Accessing the SUNRISE federation of facilities to test solutions for the internet of underwater things. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–4.

[88] Roberto Petroccia and Daniele Spaccini. 2013a. A back-seat driver for remote control of experiments in underwater acoustic sensor networks. In *Proceedings of the 2013 MTS/IEEE OCEANS*. IEEE, 1–9.

[89] Roberto Petroccia and Daniele Spaccini. 2013b. Comparing the SUNSET and DESERT frameworks for in field experiments in underwater acoustic networks. In *Proceedings of the 2013 MTS/IEEEOCEANS*. IEEE, 1–10.

[90] José Pinto, Paulo S. Dias, Ricardo Martins, Joao Fortuna, Eduardo Marques, and Joao Sousa. 2013. The LSTS toolchain for networked vehicle systems. In *Proceedings of the 2013 MTS/IEEE OCEANS*. IEEE, 1–9.

[91] Dario Pompili and Ian F. Akyildiz. 2008. A cross-layer communication solution for multimedia applications in underwater acoustic sensor networks. In *Proceedings of the 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS'08)*. IEEE, 275–284.

[92] Dario Pompili and Ian F. Akyildiz. 2009. Overview of networking protocols for underwater wireless communications. *IEEE Commun. Mag.* 47, 1 (2009), 97–102.

[93] Dario Pompili and Ian F. Akyildiz. 2010. A multimedia cross-layer protocol for underwater acoustic sensor networks. *IEEE Trans. Wireless Commun.* 9, 9 (2010), 2924–2933.

[94] Michael B. Porter. 2011. The bellhop manual and users guide: Preliminary draft. Heat, Light, and Sound Research, Inc., La Jolla, CA, USA, Tech. Rep (2011).

[95] John Potter, Joao Alves, Thomas Furfaro, Arjan Vermeij, Nicolas Jourden, Giovanni Zappa, Alessandro Berni, and Diego Merani. 2014. Software defined open architecture modem development at CMRE. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–4.

[96] Parastoo Qarabaqi and Milica Stojanovic. 2013. Statistical characterization and computationally efficient modeling of a class of underwater acoustic communication channels. *IEEE J. Ocean. Eng.* 38, 4 (2013), 701–717.

[97] REMUS 2016. REMUS AUV Woods Hole Oceanographic Institution. Retrieved from http://www.whoi.edu/osl/remus-auv.

[98] Joseph Rice, Bob Creber, Chris Fletcher, Paul Baxley, Ken Rogers, Keyko McDonald, Dave Rees, Michael Wolf, Steve Merriam, Rami Mehio, et al. 2000. Evolution of seaweb underwater acoustic networking. In *Oceans 2000 MTS/IEEE Conference and Exhibition*, Vol. 3. IEEE, 2007–2017.

[99] Joe Rice and Dale Green. 2008. Underwater acoustic communications and networks for the US navy's seaweb program. In *Proceedings of the 2nd International Conference on Sensor Technologies and Applications (SENSORCOMM'08)*. IEEE, 715–722.

[100] P. H. Rogers. 1981. *Onboard Prediction of Propagation Loss in Shallow Water*. Technical Report. DTIC Document.

[101] Daniel L. Rudnick, Russ E. Davis, Charles C. Eriksen, David M. Fratantoni, and Mary Jane Perry. 2004. Underwater gliders for ocean research. *Mar. Technol. Soc. J.* 38, 2 (2004), 73–84.

[102] Sumi A. Samad, S. K. Shenoy, G. Santhosh Kumar, and PRS Pillai. 2011. A survey of modeling and simulation tools for underwater acoustic sensor networks. *Int. J. Res. Rev. Comput. Sci.* 2 (2011), 40–47.

[103] J. Senne, A. Song, M. Badiey, and K. B. Smith. 2012. Parabolic equation modeling of high frequency acoustic transmission with an evolving sea surface. *J. Acoust. Soc. Am.* 132, 3 (2012), 1311–1318.

[104] Ethem Mutlu Sözer and Milica Stojanovic. 2006. Reconfigurable acoustic modem for underwater sensor networks. In *Proceedings of the 1st ACM International Workshop on Underwater Networks*. ACM, 101–104.

[105] Milica Stojanovic and James Preisig. 2009. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Commun. Mag.* 47, 1 (2009), 84–89.

[106] Roger P. Stokey, Alexander Roup, Chris von Alt, Ben Allen, Ned Forrester, Tom Austin, Rob Goldsborough, Mike Purcell, Fred Jaffre, Greg Packard, and others. 2005. Development of the REMUS 600 autonomous underwater vehicle. In *Proceedings of MTS/IEEE (OCEANS'05)*. 1301–1304.

[107] Yifan Sun and Tommaso Melodia. 2013. The internet underwater: An IP-compatible protocol stack for commercial undersea modems. In *Proceedings of the 8th ACM International Conference on Underwater Networks and Systems*. ACM, 37.

[108] Hwee-Pink Tan, Roee Diamant, Winston K. G. Seah, and Marc Waldmeyer. 2011. A survey of techniques and challenges in underwater localization. *Ocean Eng.* 38, 14 (2011), 1663–1676.

[109] Teledyne 2016. Teledyne-BenthosAcoustic Modems. Retrieved from https://teledynebenthos.com.

[110] Dustin Torres, Jonathan Friedman, Thomas Schmid, and Mani B. Srivastava. 2009. Software-defined underwater acoustic networking platform. In *Proceedings of the 4th ACM International Workshop on UnderWater Networks*. ACM, 7.

[111] Giovanni Toso, Ivano Calabrese, Paolo Casari, and Michele Zorzi. 2014. RECORDS: A remote control framework for underwater networks. In *Proceedings of the 2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*. IEEE, 111–118.

[112] Tritech 2016. TritechMicron Data Modem. Retrieved from http://www.tritech.co.uk.

[113] USRP 2016. USRP Brochure. Retrieved from http://www.ettus.com.

[114] Iuliu Vasilescu, Keith Kotay, Daniela Rus, Matthew Dunbabin, and Peter Corke. 2005. Data collection, storage, and retrieval with an underwater sensor network. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*. ACM, 154–165.

[115] Mehmet Vuran and Ian F. Akyildiz. 2008. Cross-layer packet size optimization for wireless terrestrial, underwater, and underground sensor networks. In *Proceedings of the he 27th IEEE Conference on Computer Communications (INFOCOM 2008)*. IEEE.

[116] Lei Wan, Hao Zhou, Xiaoka Xu, Yi Huang, Shengli Zhou, Zhijie Shi, and Jun-Hong Cui. 2013. Field tests of adaptive modulation and coding for underwater acoustic OFDM. In *Proceedings of the 8th ACM International Conference on Underwater Networks and Systems*. ACM, 35.

[117] Yu Wang, Yingjian Liu, and Zhongwen Guo. 2012. Three-dimensional ocean sensor networks: A survey. *J. Ocean Univ. Chin.* 11, 4 (2012), 436–450.

[118] WHOI 2016. WHOIWoods Hole Oceangraphic Institution. Retrieved from http://acomms.whoi.edu/micro-modem.

[119] Lingjuan Wu, Jennifer Trezzo, Diba Mirza, Paul Roberts, Jules Jaffe, Yangyuan Wang, and Ryan Kastner. 2012. Designing an adaptive acoustic modem for underwater sensor networks. *IEEE Embed. Syst. Lett.* 4, 1 (2012), 1–4.

[120] Geoffrey Xie, John Gibson, and Leopoldo Diaz-Gonzalez. 2006. Incorporating realistic acoustic propagation models in simulation of underwater acoustic networks: A statistical approach. In *OCEANS 2006*. IEEE, 1–9.

[121] Peng Xie, Zhong Zhou, Zheng Peng, Hai Yan, Tiansi Hu, Jun-Hong Cui, Zhijie Shi, Yunsi Fei, and Shengli Zhou. 2009. Aqua-Sim: An NS-2 based simulator for underwater sensor networks. In *MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges (OCEANS 2009)*. IEEE, 1–7.

[122] Guo Zhong-Wen, Luo Han-Jiang, Hong Feng, Yang Meng, and Ni Ming-Xuan. 2010. Current progress and research issues in underwater sensor networks. *J. Comput. Res. Dev.* 47, 3 (2010), 377–389.

[123] Shengli Zhou, Baosheng Li, Peter Willett, Milica Stojanovic, and Lee Freitag. 2010. Apparatus, systems and methods for enhanced multi-carrier based underwater acoustic communications. Patent No. 7,859,944.

[124] Yibo Zhu, Son Le, Lina Pu, Xiaoyan Lu, Zheng Peng, Jun-Hong Cui, and Michael Zuba. 2013a. Aqua-net mate: A real-time virtual channel/modem simulator for aqua-net. In *OCEANS-Bergen, 2013 MTS/IEEE*. IEEE, 1–6.

[125] Yibo Zhu, Xiaoyan Lu, Lina Pu, Yishan Su, Robert Martin, Micheal Zuba, Zheng Peng, and Jun-Hong Cui. 2013b. Aqua-Sim: An NS-2 based simulator for underwater sensor networks. In *Proceedings of the 8th ACM International Conference on Underwater Networks and Systems*. ACM.

[126] Yibo Zhu, Lina Pu, Zigeng Wang, Xiaoyan Lu, Rashad Martin, Yu Luo, Zheng Peng, and Jun-Hong Cui. 2014. Underwater acoustic network protocol stacks: Simulator-based vs. OS-based. In *Oceans-St. John's, 2014*. IEEE, 1–7.

[127] Michael Zuba, Zaihan Jiang, TC Yang, Yishan Su, and J. Cui. 2013. An advanced channel framework for improved underwater acoustic network simulations. In *Proceedings of IEEE/MTS OCEANS San Diego*.

[128] Michael Zuba, Aijun Song, and Jun-Hong Cui. 2014. Exploring parabolic equation models for improved underwater network simulations. In *Underwater Communications and Networking (UComms), 2014*. IEEE, 1–5.

Received June 2016; revised January 2017; accepted January 2017