Peer-Reviewed Technical Communication

Software-Defined Underwater Acoustic Modems: Historical Review and the NILUS Approach

Henry S. Dol, *Member, IEEE*, Paolo Casari, *Senior Member, IEEE*, Timo van der Zwan, and Roald Otnes, *Senior Member, IEEE*

Abstract—Flexible/adaptive acoustic modems that are reprogrammable/reconfigurable at all layers of the communication stack, either by a user or by means of autonomous decisions, are considered as an important enabler for interoperability and cognitive networking in the underwater domain. In this paper, we review the existing literature on software-defined acoustic modems (SDAMs) for underwater communications and networking, considering past and ongoing academic efforts, as well as industrial developments and European collaborations centered on software-defined modem structures and functionalities. We then zoom in on relevant R&D efforts currently taking place in a defense cooperation between The Netherlands and Norway, targeting the design of a software-defined modem for the NILUS MK 2 sensor node. This modem is built using general-purpose (GP) computing architectures running open-source operating system and tools, thereby making a further step toward software-defined open-architecture underwater acoustic modems. The first field tests of the NILUS MK 2 node in multihop underwater acoustic networks are presented, showing satisfactory performance in shallow and open waters.

Index Terms—Open architecture, sea trials, software-defined modems, underwater acoustic communications, underwater wireless networks.

I. INTRODUCTION

HISTORICALLY, the development of reconfigurable, reprogrammable, and software-defined underwater acoustic modems has been driven by universities and research institutes, mainly because of the flexibility required by scientific experiments on underwater communications and networks. More recently, lacking standardization of underwater communication protocols, the modem industry has been gradually opening their commercial off-the-shelf (COTS) systems to the (skilled) end user as well. This provides better opportunities for interoperability, among modems made by different manufacturers, and adaptivity, i.e., being able to switch protocols or parameter settings

H. S. Dol and T. van der Zwan are with the The Netherlands Organisation for Applied Scientific Research (TNO), NL-2509 JG The Hague, The Netherlands (e-mail: henry.dol@tno.nl).
P. Casari is with the IMDEA Networks Institute, Madrid, Spain (e-mail: paolo.casari@imdea.org).
R. Otnes is with the Norwegian Defence Research Establishment (FFI), NO-3191 Horten, Norway (e-mail: Roald.Otnes@ffi.no).

depending on an evolving environment or on changing application requirements. However, the reconfiguration/reprogramming options for commercial modems today are typically limited, and any change requires a thorough knowledge of the specific hardware and software architecture. Therefore, an interesting next step would be the introduction of open-architecture/open-source modems, fostering the involvement and support of a global user community. This would boost the establishment of standards and, more importantly, their acceptance. Modem manufacturers could still make a profitable business, but their business model would change by focusing more on efficient hardware and proprietary high-performance algorithms.

In this paper (Section II), we provide a review of past and current efforts toward the construction of reconfigurable/reprogrammable and software-defined modems, listing relevant hardware and software specifications when available. We proceed by detailing a joint effort of The Netherlands Organisation for Applied Scientific Research (TNO) and the Norwegian Defence Research Establishment (FFI) toward the development of a fully flexible and high-level programmable modem, which still complies with operational requirements in terms of processing power, size, and endurance (Section III). Since this software-defined modem runs a Linux operating system, the Netherlands/Norway defense cooperation will contribute to the development of a Software-Defined Open-Architecture Modem (SDOAM). This and other concluding remarks are provided in Section IV.

II. REVIEW OF RECONFIGURABLE, REPROGRAMMABLE, AND SOFTWARE-DEFINED ACOUSTIC MODEMS

Following developments in terrestrial radio systems, an upcoming concept for underwater acoustic modems is software-defined, possibly open-architecture, reconfigurable/user-programmable modems. The current state of these developments, which mainly takes place in a noncommercial setting, is described in this section. The research on reconfigurable acoustic modems is mainly driven by the need for an adaptive platform, whose characteristics (in terms of physical-layer schemes, or network protocols, or both) can be changed to suit the need of different experiments/scenarios.

The first commercial modems to be available on the market basically served as serial link emulators, providing remote data telemetry via acoustic communications. Modem systems were mostly opaque to the end user, as their physical-layer (PHY) algorithms and bit-stream formats were hardcoded in the modem firmware. Moreover, in some cases, their network (NET) protocol implementations included some form of preliminary or periodic messaging which was also not controllable/reconfigurable. For these reasons, commercial off-the-shelf equipment was usually transparent to the application and therefore not very well suited to research efforts.

More recent modems include at least some form of reconfigurability. A reconfigurable modem gives the user the chance to decide on some parameters by issuing commands that are recognized by the modem's firmware. On-the-fly changes of the digital modulation's bit rate and coding scheme at the PHY layer are a typical example of this reconfiguration capability. With the development of underwater modem hardware and firmware for research projects in academia, the options available for the users have been broadened by reprogrammability. Reprogrammable modems introduce the freedom to replace one or more of the main communication blocks including, e.g., complete modulation schemes, signal processing algorithms, or network-layer protocols. While this possibility clearly represents a valuable tool, reprogramming a modem often requires additional skills such as, e.g., the programming of digital signal processors (DSPs) or field-programmable gate arrays (FPGAs) for PHY signal processing schemes, or the use of specific or uncommon programming languages for the network protocol stack. Software-defined modems (sometimes briefly called softmodems) are a specific case of reprogrammable modems where the whole transmitter- and receiver-side processing chains are programmed in software. The recent development of open-source and open-architecture modems is a significant step toward making modem architectures more accessible to the end user. With open-source modems, users can refer to well-defined programming examples and templates to introduce their own functionalities in the PHY or NET layers, or both. Open-architecture modems make this concept even more accessible by running software and algorithms on standard computing devices, whose architecture is well known. A relevant example is the use of general-purpose (GP) processors to replace specialized hardware such as DSPs.

Jones [1] was one of the first advocates in the open literature for software- and cognitive-radio concepts in underwater systems. She advocated methods for changing the modulation scheme without modifying the whole system architecture. Depending on the degree of reconfigurability/reprogrammability sought, research projects on reconfigurable modems may focus on the flexibility of the physical layer, of the network protocols, or of both. In the following, we will describe the relevant research projects on reconfigurable modems that we are aware of. The review is divided into past (mainly USA) academic efforts (2006–2010, Section II-A), ongoing academic efforts (1999–, Section II-B), industrial developments (2010–, Section II-C), and European collaborations (2005–, Section II-D). Section II-E summarizes the findings of this survey. For reference, the characteristics of the surveyed modems are summarized in Table I along with reported results of laboratory and/or field experiments for each modem.

A. Past Academic Efforts

Between 2006 and 2010, seven academic projects on reconfigurable and reprogrammable underwater acoustic modems have been running, mostly in USA, which have set the first milestone for the development of software-defined (open-architecture) underwater acoustic modems.

1) MIT Sea Grant's rModem (USA, 2006–2008): The Massachusset Institute of Technology (MIT) rModem project dates back to 2006–2008. The project aimed at building a piece of hardware which was reprogrammable in terms of PHY techniques for communication and signaling, and that could provide typical PHY-related measurements (e.g., reliability) and services (e.g., high-resolution power control) [2], which were not typically available in off-the-shelf modems, not even Woods Hole Oceanographic Institution's Micro-Modem [3] (Section II-B1), that was the most reconfigurable underwater modem available at the time. The rModem is composed of one motherboard and one daughter card. Its main features include four configurable input/output (I/O) channels suitable for multiple-input–multiple-output (MIMO) algorithms, analog anti-aliasing fil-

ters, an onboard FPGA and a 300-MHz floating-point DSP with 32-MB SDRAM and 32-MB flash memory [2]. The daughter card hosts signal conditioning and amplification stages, and is detached from the motherboard to improve the platform flexibility whenever, e.g., the transducer connected to the modem has to be changed. The FPGA hardware is fast enough to process any signal in the 1–100-kHz band. At the time, Sözer and Stojanovic [2] reported successful communication experiments in the 9–14-kHz band using a Teledyne AT-408 transducer.

The hardware can be operated in real time (if the DSP supports the complexity of receiver-side signal processing) or in pseudo real time (whereby signal samples are stored before being processed by the DSP, therefore introducing some processing delay). A specific implementation of a transmitter and a receiver on the rModem is documented in [4], where successful reception of quadrature phase-shift keying (QPSK) signals is reported at 550 b/s within a tank, using a carrier frequency of 12 kHz. An adaptive decision-feedback equalizer (DFE) is employed for phase-coherent reception, which embeds the recursive least squares (RLS) algorithm and a phase-locked loop. A convolutional code is used for error correction.

2) UCD/UCI/IST Tmote Invent-Based Underwater Node (Ireland/USA/Portugal, 2006–2007): The first known design of a low-cost, software-defined underwater acoustic modem has been carried out in the context of a collaboration between University College Dublin (UCD, Dublin, Ireland), University of California Irvine (UCI, Irvine, CA, USA) and Instituto Superior Técnico (IST, Lisbon, Postugal) [5], [6]. The objective was to create an inexpensive platform for underwater communications that relied on integrated low-cost components to carry out modulation and demodulation tasks. After a test with laptops [5], Moteiv's Tmote Invent/Sky platform was chosen [6] and adapted for underwater usage via waterproof latex membranes. The devices rely on a microphone sensitive to the 0.1–20-kHz band, and on a speaker working in the 0.4–20-kHz band, which are employed as acoustic receiver and transmitter, respectively. A frequency-shift keying (FSK) scheme was designed by employing eight tones that exhibited good performance at several distances between 1 and 13 m in preliminary performance measurements. The modulation and demodulation chains are programmed in software within the node's microcontroller. The system achieves error-free communication at 24 b/s (and possibly up to 48 b/s) up to a distance of 17 m. As the Tmote Invent is a very resource-constrained platform, it is not clear whether its microcontroller could host both PHY and NET code simultaneously. All the same, the result marks an achievement in terms of software-defined low-cost modems.

3) MIT CSAIL's AquaNodes (USA, 2007): In MIT's Aqua Nodes project, the PHY implementation is fixed, but the user is free to reprogram network protocols [7]. AquaNodes can communicate by means of either acoustic or optical signals, although the latter can only be operated within a limited distance of 2 m and when the transceivers are aligned within a tolerance of 90°. The AquaNodes embed an ATmega128 microcontroller, 128 kB of flash memory, 4 kB of RAM, and 512 kB of external storage.

Acoustic communications are realized via an FSK modulation scheme at a carrier frequency of 30 kHz. Data rates of 330 b/s up to a range of 400 m are reported. Optical communications are obtained by interfacing the motherboard to a special-purpose board through a serial link. The optical board generates signals at a wavelength of 532 nm, and a communication range of 2.2 m (up to 8 m with beam-narrowing lenses and precise pointing). The maximum data rate reported is 320 kb/s. A lithium-ion battery of 56 Wh provides sufficient power supply for one year (respectively, 1–2 weeks) at minimum (respectively, maximum) duty cycle. The reconfiguration of the networking components of the AquaNodes is achieved through a communication

TABLE I
SUMMARY OF THE HARDWARE, COMMUNICATION CHARACTERISTICS AND TEST RESULTS FOR THE MODEMS SURVEYED IN THIS PAPER.

	System features						Communications features						Test results	
	μProcessor	DSP, FPGA and/or μController	RAM	Storage	Supported frequency band	Other hardware included	Center frequency f_c	Bandwidth	Default modulation (bit rate)	Specific signal processing algorithms mentioned	MIMO	TX power or source level	Schemes (bit rates)	Max range(s)
rModem (§II-A1)	N/A	TI TMS320 C6713 DSP (225 MHz) Altera Cyclone II FPGA	32 MB	32 MB flash	1–100 kHz	Detachable amplifier	11.5 kHz	4 kHz	QPSK (550 b/s)	1/2-rate conv. coding, DFE, RLS, PLL	Up to 4×4	N/D	*	approx. 16 m
UW Tmote (§II-A2)	N/A	TI MSP430 (8 MHz)	10 kB	48 kB flash	0.4-20 kHz	N/A	900 or 1500 Hz	1 kHz	8-FSK (up to 48 b/s)	N/D	N/A	Max. 1 W	* (24 b/s) * (48 b/s)	10 m / 3 m
AquaNodes (§II-A3)	ATmega128 (16 MHz)	AD Blackfin BF533 DSP (600 MHz)	4 kB	128 kB flash, 512 kB ext.	0-125 kHz	Optical link (2 m range)	30 kHz	N/D	FSK (330 b/s)	N/D	N/A	N/D	*	400 m
UANT (§II-A4)	Intel Q8200 (2.33 GHz)	N/A	4 GB	N/D	175 kHz	USRP TX/RX cards	Depends on transducer	Up to 16 MHz	GMSK, (0.25-500 kb/s)	GNURadio components	N/A	N/D	GMSK, 100-175 kHz band (7.5, 10, 15 kb/s)	Pool, tank
UCSD (§II-A5)	N/A	Xilinx Spartan3 xc3s4000 FPGA	N/A	N/A	6 kHz	N/A	35 kHz	1 kHz	FSK (200 b/s)	N/D	N/A	170 dB[a]	*	Bucket, 350 m detection tests in bay
UNSW (§II-A6)	N/A	Altera Cyclone III FPGA (50 MHz)	N/A	N/A	100-1000 kHz	N/A	800 kHz	80 kHz	BPSK (80 kb/s)	Costas loop, NCO	N/A	N/D	*	Up to 50 m
CAS (§II-A7)	N/A	AD Blackfin 561 DSP (600 MHz)	N/A	N/A	10 kHz	N/A	25 kHz	10 kHz	QPSK (8 kb/s)	Turbo or conv. coding	N/A	Max 190 dB[a]	QPSK (8 kb/s) DSSS (≈1 kb/s) OFDM (20 kb/s)	8-10 km / 5 km / 1 km
Micro-Modem 1 (§II-B1)	N/A	TI TMS320 VC5416 DSP (up to 160 MHz)	≈2 MB	16 MB flash	3-30 kHz	Optional co-proc. board	10 kHz 15 kHz 25 kHz	4 kHz	FH-FSK (80 b/s) PSK (300 b/s − 5 kb/s)	Channel coding (DFE for PSK)	Up to 1×8 [b]	185 dB[a]	*	Several tests in shallow and deep waters
Micro-Modem 2 (§II-B1)	N/A	AD Blackfin BF548 DSP (533 MHz)	16 MB	64 MB+ 32 GB flash	1-100 kHz	"	"	"	"	"	"	"	*	"
UCONN (§II-B2)	N/A	TI TMS320 C6713 (225 MHz) and TI TMS320 C6416 (1 GHz)	4 MB	N/A	9-15 kHz	N/A	12 kHz	6 kHz	OFDM (3.2 kb/s, or 6.4 kb/s w/ MIMO)	1/2-rate conv. coding, LDPC	2×2	N/D	*	N/D

TABLE I
SUMMARY OF THE HARDWARE, COMMUNICATION CHARACTERISTICS AND TEST RESULTS FOR THE MODEMS SURVEYED IN THIS PAPER. *(Continued)*

	System features						Communications features							Test results
	μProcessor	DSP, FPGA and/or μController	RAM	Storage	Supported frequency band	Other hardware included	Center frequency f_c	Bandwidth	Default modulation (bit rate)	Specific signal processing algorithms mentioned	MIMO	TX power or source level	Schemes (bit rates)	Max range(s)
AquaSeNT[c] (§II-B2)	N/D	N/D	N/D	N/D	21-27 kHz	N/D	24 kHz	6 kHz	OFDM (up to 9 kb/s)	N/D	N/D (4RX)	<25 W	*	Up to 5 km
Scripps (§II-B3)	Intel Pentium-M (1.4 GHz)	TI TMS320 C6713 DSP (up to 225 MHz), Xilinx Virtex-II FPGA	1 GB	2 GB	10-32 kHz	GPS	21 kHz	22 kHz	N/D	N/D	8 RX	180 dB[a]	N/D	Up to 8 km
ITACA (§II-B4)	N/A	Silabs C8051f360	N/D	N/D	1 kHz	N/A	85 kHz	1 kHz	FSK (1 kb/s)	PLL	N/A	>12 mW	*	100 m
UNET-2 (§II-B5)	Toradex Collibri PXA270 (520 MHz)	AD TigerSHARC TS201 DSP (up to 600 MHz), Altera Stratix-III FPGA	64 MB	N/D	Up to $2f_c/3$	N/D	Up to 78 kHz	12.5 kHz	OFDM (up to 7.5 kb/s)	Conv. coding, 1-norm Viterbi dec.	N/A	185 dB[a]	Various	>2 km
FAST (§II-B6)	Laptop	N/A	Laptop	Laptop	Laptop's sound card	N/A	16.5 kHz	2 kHz	FSK (100 to 600 b/s)	GnuRadio components	N/A	N/A	*	6 m
Harbin (§II-B7)	ARM 6410 (533 MHz)	N/D	N/D	Flash	N/D	N/D	N/D	N/D	OFDM, SS	N/D	N/A	N/D	* (N/D)	30 km
SeaModem (§II-B8)	Beaglebone's ARM Cortex A8 (1 GHz)	TI TMS320C5000 DSP (up to 200 MHz)	N/D	N/D	25-35 kHz	Beaglebone	30 kHz	10 kHz	FSK (up to 2.25 kb/s) JANUS	N/D	N/A	Max. 40 W	*	100 m
Xi'an (§II-B9)	N/A	TI OMAP-L138 (456 MHz) TI C6000 DSP (456 MHz)	640 kB	N/D	0-24 kHz	N/A	N/D	4 kHz	OFDM (4 kb/s)	Equalization, polyphase resampling	N/A	N/D	*	10 km
SeaTrac (§II-B10)	N/A	ARM Cortex M4 (150 MHz)	N/D	N/D	24-32 kHz	3-element USBL	28 kHz	8 kHz	LFM (100 b/s) DSSS (1.4 kb/s)	RS coding	N/A	176 dB[a]	*	1.5 km (LFM SS) 1.6 km (DSSS)
Buffalo/ NEU (§II-B11)	Laptop (if present)	Xilinx Spartan 3A-DSP 3400 FPGA or ARM M4 (Teensy)	N/D	N/D	0-170 kHz	USRP TX/RX cards	100 kHz	22 or 24 kHz	OFDM, DSSS, B-CSS	1/2-rate conv. coding, RS coding, AMC	N/A	N/D	OFDM (up to approx. 25 kb/s)	100 m
SEANet-G3 (§II-B11)	2x ARM Cortex-A9 (866 MHz)	Avnet PicoZed XC7Z020-1CLG400 DSP/FPGA	256 kB + 1 GB DDR	128 MB flash	200-800 kHz	N/A	500 kHz	600 kHz	ZP-OFDM	3/4-rate conv. coding	N/A	N/D	* (500-800 kHz band)	approx. 10 m

TABLE I
SUMMARY OF THE HARDWARE, COMMUNICATION CHARACTERISTICS AND TEST RESULTS FOR THE MODEMS SURVEYED IN THIS PAPER. *(Continued)*

	System features						Communications features						Test results	
	μProcessor	DSP, FPGA and/or μController	RAM	Storage	Supported frequency band	Other hardware included	Center frequency f_c	Bandwidth	Default modulation (bit rate)	Specific signal processing algorithms mentioned	MIMO	TX power or source level	Schemes (bit rates)	Max range(s)
Develogic HAM (§II-C1)	N/A	TI TMS320 C6713 (300 MHz)	128 MB	64 GB flash	2.5-70 kHz	Detached transducer option	6 kHz	Up to 25 kHz	OFDM	Two-stage error correction	N/A	Max. 200 W	* (scheme N/D)	23 km
EvoLogics WiSE (§II-C2)	N/D	ADSP21363 SHARC (333 MHz), Altera Cyclone II FPGA	N/D	Up to 32 GB	18-34 kHz 48-78 kHz	N/A	26 kHz 63 kHz	16 kHz 30 kHz	Sweep-spread carrier	N/D	N/A	Max. 80 W	* (13.9 kb/s) * (31.2 kb/s)	3.5 km 1 km
SubNero (§II-C3)	Marvell PXA320 (806 MHz)	AD TigerSHARC DSP, Altera Stratix III FPGA	N/D	N/D	Typ. 18 kHz (upper lim. N/D)	N/D	27 kHz	$2f_c/3$	OFDM + FH-BFSK (10 + 0.5 kb/s)	N/D	N/A	185 dB^a	* Up to 10 kb/s	3 km
Buffalo / TB^d (§II-C4)	Gumstix	TI C64x+	512 MB	8 MB + 2x32 GB flash	9-14 kHz 16-21 kHz 22-27 kHz	N/A	Depends on band	5 kHz	DS-CDMA (256 b/s)	N/D	1×2	Max. 20 W	*	20 m
UCAC (§II-D1)	Intel Pentium-M (2 GHz)	N/A	N/D	N/D	1.5-5 kHz	N/A	3.25 kHz	3.5 kHz	OFDM (4-80 b/s)	Turbo coding	N/A	190 dB^a	* b/s	20 km
CLAM (§II-D2)	Gumstix	N/D	N/D	N/D	21-31 kHz	N/A	26 kHz	10 kHz	FSK DSSS	N/D	N/A	190 dB^a	*	4 km
RACUN / Saab (§II-D3)	Intel i7 (2.1 GHz)	N/A	8 GB	160 GB HD	4-8 kHz	N/A	6 kHz	4 kHz	FMT, OFDM, SCTE	Depends on modulation	N/A	185 dB^a	FMT (340 b/s) OFDM (205 b/s) SCTE^e (624 b/s)	Depends on modulation
Proteus II (SUNRISE) (§II-D4)	N/A	2x ARM Cortex M4 (150 MHz)	64 kB	Flash	20-40 kHz	Sensors	21 kHz	2 kHz	FSK (max. 200 b/s)	N/D	N/A	8 W	* (up to 150 b/s)	30 m
NILUS SoftModem (§III)	TI OMAP-4460 (ARM Cortex A9, 1.5 GHz)	N/A	1 GB	8 GB flash	3-11 kHz	N/A	6 kHz	4 kHz	FRSS (90 to 1200 b/s)	1/2-rate conv. coding, multichannel equalizer, LMS, PLL	N/A	N/D	* (N/D)	In excess of 5 nautical miles

Asterisks (*) indicate that the tested modulation scheme(s) and the resulting bit rate(s) are the same as indicated in the "Communication Features" columns.

"N/A" means that the element is not applicable or not relevant.

"N/D" means that the information is not declared in the public literature.

^{a} Relative to 1 $\mu Pa^2\, m^2$.

^{b} To the authors' best knowledge, the MicroModem has up to eight data acquisition channels but only one receive chain, hence MIMO is only possible via offline processing.

^{c} While several characteristics of the AquaSeNT modem are not explicitly declared, they can be expected to be akin to those of the UCONN modem development [17]–[20].

^{d} To the authors' best understanding, these results are obtained by employing the Teledyne-Benthos modem hardware to transmit a custom DS-CDMA scheme.

^{e} Filtered multitone (FMT) and Single-Carrier Turbo Equalization (SCTE) are two modulation schemes that have been used in the context of the RACUN project.

^{f} The reported performance figures refer to a selection of models. The full list of models is available on the web site of the manufacturer [75].

channel to the modems (a wired serial cable, a Bluetooth connection, acoustic or optical wireless communications). In [7], a time-division multiple-access (TDMA) channel access system with self-synchronization capabilities is employed.

4) University of California Los Angeles' UANT (USA, 2009): The Underwater Acoustic Networking PlaTform (UANT) was probably the first software-defined, open-source underwater platform prototype specifically designed to be reprogrammable at both the physical layer and the network layer [8]. PHY reprogramming is achieved via GNU Radio (a widely adopted platform in the software-defined radio community) whereas NET reprogrammability is provided by TinyOS (a standard solution for programming network protocols in wireless sensor networks). Since GNU Radio and TinyOS are open-source/open-architecture developments (for the terrestrial radio environment), the UANT project was well ahead of its time.

GNU Radio is run on a Universal Software Radio Peripheral (USRP) that, albeit more expensive than a low-cost audio board, provides more processing capabilities via dedicated hardware. With the choice of GNU Radio, many of the signal processing blocks already programmed for the radio world can be ported to underwater communications, and can be assembled to (re)configure basic and advanced modulation schemes. Specific changes to GNU Radio include allowing custom medium-access control (MAC) protocols, setting the bit rate, and making the traffic source and sink TinyOS-based so that traffic can be generated and processed within the networking components of the UANT. The changes have been kept limited to maximize the support for future expansions to GNU Radio's digital communication blocks.

GNU Radio has been interfaced with TOSSIM, the simulator of TinyOS. This makes it possible to implement network protocols with a tool commonly employed for programming wireless sensor networks. The interface intercepts the packets coming from the transducer and passes them to a MAC protocol in TOSSIM's stack. From there, further network protocol processing can start. The use of TOSSIM also enables application-level (APP) reprogramming. UANT runs on the following hardware [8]: Intel Core 2 Quad Q8200 processor, 4-GB RAM, USRP low-frequency transmitter (TX) and receiver (RX) daughter cards (0–50 MHz), custom preamplifier board, and RESON TC4013 hydrophone (RX bandwidth 0.1–175 kHz). The extension of the UANT with channel allocation and time synchronization protocols is described in [9].

5) UCSD's Low-Cost Modem (USA, 2010): As experimentation costs represent a heavy burden for the budget of academic underwater networking projects, some underwater acoustic modems were specifically designed to be both software defined and low cost.

The University of California San Diego (UCSD) low-cost modem design is tailored to extract the most from a $10 Steminc SMC26D22H13 ceramic ring transducer [10]. This is achieved via a highly linear 75%-efficient mixed-class AB+D power amplifier, a high-gain receiver, and output power control. The transmitter- and receiver-side signal processing are hosted on an FPGA mounted on a DINI Group DNMEG AD-DA evaluation board. The resulting system is based on a 200-b/s FSK modulation scheme working in the 32–38-kHz band, at a final estimated cost of $\sim$ $600, when produced in large volumes. Controlled tests in Mission Bay, San Diego, CA, USA, confirmed the capability to detect signals from a distance of 350 m.

6) University of New South Wales's High-Rate Modem (Australia, 2010): Nowsheen *et al.* detail a preliminary implementation of a software-defined modem consisting of a BPSK modulation-demodulation chain programmed on an FPGA [11], [12]. Most of the code has been written in Matlab and tested on recorded waveforms before actually implementing the same functions on the FPGA. Successful open water tests are reported to have achieved a bit rate of 80 kb/s using a carrier frequency of 800 kHz at a distance of up to 50 m. The same modem technology

has been tested over a set of different synthetic (i.e., modeled) channels. The rationale and results of these tests are reported in [13].

7) CAS Software-Defined Modem (China, 2010): Li and Huang designed a software-defined acoustic modem (SDAM) around a core Blackfin 561 DSP module [14]. The modem is programmed with different modulation schemes and coding rates, which are tested in a sea trial over a point-to-point link. A bit rate of up to 8 kb/s over a range of 8 km is reported to be achieved using the QPSK modulation and a convolutional channel code within a bandwidth of 10 kHz at a carrier frequency of 25 kHz. A higher rate OFDM modulation scheme using turbo channel coding is reported to have achieved 20 kb/s over a range of 1 km. Network tests with four nodes employing a carrier-sense multiple access/collision avoidance (CSMA/CA) MAC protocol are also mentioned. It remains unclear whether the network protocols can be reconfigured/reprogrammed and on which system component they are run.

B. Ongoing Academic Efforts

Since 1999, four USA academic efforts on software-defined underwater acoustic modems are still active, of which one is being sold not-for-profit (Section II-B1), a second one is now generating a commercial spinoff (Section II-B2), a third one is still functional as part of an operational trial system (Section II-B3), and the fourth one is a more recent development that has already seen a number of tests in the water (Section II-B11). Since 2012, an important contribution to the development of software-defined underwater acoustic modems comes from Singapore (Section II-B5). More recently, an Italian effort targeted the easy reconfiguration of the modem's PHY and was used for double-PHY tests in the water (Section II-B8). Around the same time, a doubly-reprogrammable UK modem seeking robustness in harsh environments has been proposed (Section II-B10). All three academic developments (from Singapore, Italy, and UK) have generated a commercial spinoff. Furthermore, two Chinese academic efforts are being proposed, based on different types of reprogrammable platforms (Sections II-B7 and II-B9). Finally, two academic efforts also worth mentioning focus on low power (Spain, Section II-B4) and low cost (Pakistan, Section II-B6), respectively.

1) WHOI's MicroModem (USA, 1999–): The MicroModem of the Woods Hole Oceanographic Institution (WHOI) [3] has been an important communication platform since 1999 [15, Table I], equipped with multichannel PSK (subject to USA export restrictions) and lower rate frequency-hopped FSK (FH-FSK, available for export) modulations. The MicroModem was licensed nonexclusively to Hydroid/Kongsberg to be mounted on the REMUS autonomous underwater vehicles (AUVs). Since about 2010, the MicroModem has been superseded by a more powerful version [15]. The proprietary PHY schemes are largely opaque to the end user with limited reconfiguration possibilities, and user reprogrammability is not explicitly supported. However, the technical details about the modem architecture as provided on the WHOI MicroModem website [16] may allow independent developers to implement their own modulations and network stack on the DSP system, possibly partly on the optional Gumstix expansion board. Compared to the original MicroModem, the newer version embeds a more powerful fixed-point Analog Devices Blackfin DSP, a data-acquisition FPGA, and increased memory sizes [15]. The PSK scheme runs separately on a floating-point coprocessor, and the multichannel variant uses an additional analog interface board. The coprocessor tasks can, in principle, be taken over by the DSP.

2) University of Connecticut's OFDM Modem (USA, 2007–): The Underwater Sensor Network (UWSN) lab at the University of Connecticut has been actively developing an underwater

modem based on MIMO orthogonal frequency-division multiplexing (OFDM) [17]. A real-time DSP implementation of both a single-input–single-output (SISO) and a MIMO version of the modem is detailed in [18]. In [17], Li *et al.* report results based on channel recordings from the RACE'08 campaign. QPSK up to 64-QAM with two transmitters, QPSK up to 16-QAM with three transmitters, and QPSK up to 8-QAM with four transmitters are used with a bandwidth of 4.8 kHz. The achievement of a 125.7-kb/s data rate with two transmitters, 16-QAM modulation, rate-1/2 coding, and a bandwidth of 62.5 kHz is reported related to a second set of experiments in 2008.

The implementation detailed in [18] demonstrates real-time capabilities with both a floating-point (TI TMS320 C6713) and a fixed-point (TI TMS320 C6416) DSP, achieving data rates of 3.2 and 6.4 kb/s, with a SISO and a 2×2 MIMO modem (where the notation indicates the number of transmitter-side and receiver-side elements, respectively). A QPSK modulation scheme and a rate-1/2 convolutional channel code within a 6-kHz bandwidth were employed. The use of a low-density parity-check (LDPC) code in real time is also demonstrated.

Since 2012, the OFDM modem efforts of the University of Connecticut have generated a commercial spin-off called AquaSeNT. The final design of the modem includes a carrier frequency of 24 kHz, a bandwidth of 6 kHz, and five transmission modes that mix different subcarrier modulation schemes (BPSK, QPSK, and 16-QAM) and different code rates (1/2 and 3/4) at a maximum bit rate of 9 kb/s [19, Ch. 18]. The different modulation and coding schemes are tested via simulations and a sea trial in [20]. A framework called SeaLinx implements networking components as operating system processes, which inherently supports real-time operations [21], [22].

3) UCSD Scripps' SDAM (USA, 2010–): The SDAM development by the Scripps Institution of Oceanography at UCSD is mainly aimed at multichannel and MIMO communications in the 10–32-kHz band [23]. Real-time acquisition and processing for eight channels is obtained via an NI-7833R board (100-kHz sampling rate at 16 b per sample). A central processing unit (CPU) board (CP306, containing a 1.4-GHz Pentium-M on a single PCI board, 1-GB RAM, USB 2.0, 1-Gb/s LAN) can be programmed to handle nonintensive computations and networking-related operations. A storage capacity of 2 GB, a precision GPS, and a high-efficiency custom pulse width modulation (PWM) power amplifier complement the board electronics. Hard disks provide storage capacity for both the operating system of the board and real-time data acquisition. The source–receive array (SRA) is composed of eight ITC-1001 spherical transducers.

UCSD's SDAM is doubly reprogrammable: the FPGA on the digital acquisition board can be leveraged for upshifting and resampling, while a DSP board allows for more complex baseband processing, whereas the GP CPU can run network protocols and higher level functions.

4) University of Valencia's Low-Power ITACA Modem (Spain, 2011–): The idea behind the ITACA modem design is that the microcontroller should be relieved from burdensome tasks such as continuous lowpass filtering [24]. This pushed forward the design of a very-low-power modem, employing a Humminbird XP 9 20 transducer resonating at 85 kHz, and transmitting via a coherent FSK scheme (1 kb/s over a bandwidth of 1 kHz) implemented on a C8051f360 Silabs microcontroller. A custom push–pull class-B-based digitally controlled power amplifier also contributes to the low power consumption. Experiments in Murcia, Spain, proved communication capabilities over 100-m-long horizontal channels. Later improvements [25] include a longer communication range, an RFID-based low-power wakeup module, and in-hardware clear-channel assessment.

5) National University of Singapore's UNET-2 Modem (Singapore, 2012–): The UNET-2 modem is a research platform for underwater communications that offers different types of reprogrammability [26]. The modem builds on a previous development at the Na-

tional University of Singapore's (NUS) Acoustic Research Laboratory (ARL), namely the so-called ARL modem. This modem employs an OFDM modulation scheme to communicate up to a declared distance of 3 km and 7.5 kb/s at a 31.25-kHz carrier frequency with 12.5 kHz of bandwidth [27]. The UNET-2 modem extends this design by supporting carrier frequencies up to 78 kHz, and a modulation bandwidth up to 2/3 of the chosen carrier frequency. The preconfigured modulation schemes consist of an incoherent and a coherent OFDM scheme whose parameters can be dynamically controlled. The applied coding scheme and the source level of the modem can also be dynamically controlled.

The system is composed of a Toradex Collibri PXA270 single-board computer (SBC), an Analog Devices TigerSHARC TS201 DSP, and an Altera Stratix-III FPGA, where the FPGA carries out detection and passband processing, the DSP implements PHY processing and coding, and all NET layer processing resides on the SBC. The latter can be reprogrammed by implementing new Java Agent Development Framework (JADE) components, whereas PHY schemes need be programmed and cross-compiled for the TS201 DSP. Chitre *et al.* [26] report in-water tests to showcase the reconfiguration capabilities of the modem and a link error control protocol.

6) FAST-NUCES' Low-Cost Modem (Pakistan, 2012–): Ahmed *et al.* [28] detail a do-it-yourself (student) modem design that employs common off-the-shelf materials (e.g., car tweeters, vegetable oil, etc.), and reuses common computing equipment (e.g., laptops) to carry out signal processing tasks. The latter is achieved via the GNU Radio software, plus an interface that makes it possible to attach custom network protocols to the communications framework. Tank tests demonstrated the feasibility of communications at a rate of 100–500 b/s over distances up to 6 m. Excluding the laptops, the cost of additional components ranges from approximately $25 to $50 if a high-performance ceramic tweeter is used to achieve a longer communication range.

7) Harbin IMMP Platform (China, 2012–): Zhang *et al.* [29] propose to centralize the control of the different modem components into a management platform, which is built around an ARM 6410 processor. While the modem attached to the platform is not reprogrammable, the platform offers sufficient flexibility to (re)configure the modem, the storage, the wake/sleep subsystems, etc. The platform also offers facilities to program and implement network protocols. The implementation of the ALOHA, CSMA, and TDMA MAC protocols is reported in [29] and is tested over a network of three nodes deployed in the Yellow Sea. The authors report successful communications over a distance of up to 30 km.

8) University of Calabria and AppliCon's SeaModem (Italy, 2014–): The University of Calabria, Italy, in collaboration with their spinoff AppliCon, have been developing Sea Modem, a low-cost underwater acoustic modem that supports both physical- and network-layer reprogrammability [30]. The modem operates in the 25–35-kHz band, and is composed of a DSP board and an ARM-based BeagleBone board for GP reprogrammability. A flash memory hosts the possible system configurations, including a read-only factory default and up to four rewritable configurations. Reconfigurable PHY parameters include the number of tones of the native M-FSK modulation (2, 4, or 8, corresponding to bit rates of 750, 1,500, and 2250 b/s, respectively) and the gain of the class-D power amplifier, which outputs up to 40 W of power. A point-to-point link between two SeaModems has been preliminarily tested in a marina environment, where ranging estimations performed via the modem's two-way communication patterns were reported to achieve a relative error of about 0.5%–3%, depending on the actual range.

In [31], Petroccia *et al.* present a more advanced configuration involving two separate physical layers, namely the SeaModem's native BFSK scheme and the JANUS FH-BFSK scheme [32]. The experi-

ment involved three static underwater nodes and one unmanned surface vehicle (USV). The latter was instructed along a trajectory that sequentially led it within the coverage range of the underwater nodes. The USV would use JANUS to exchange first-contact information, and then switch to the proprietary BFSK physical layer of the SeaModem. JANUS was observed to be quite robust against communication errors, and for this reason it was also used as a means of remote control in case of unexpected behaviors. The SeaModem is reported to achieve a packet error rate below 10% at a distance below 400 m, although it is not clear what packet size this figure applies to, and whether the difference between the native scheme and the JANUS scheme comes from the higher bit rate, the lack of selection diversity, convolutional coding and Viterbi decoding at the receiver, or from specific channel patterns. In any event, Petroccia *et al.* [31] present the first preliminary experiment of a dual-PHY modem involving JANUS. The authors remark that the version of JANUS employed for this test is not fully compliant with the upcoming NATO standard in two respects: operational bandwidth (25-35 kHz instead of 9.44–13.6 kHz) and MAC (CSMA/CA with binary exponential backoff replaced by ALOHA).

9) Xi'an Underwater Acoustic Modem (China, 2014–): Wang *et al.* [33] describe an underwater acoustic modem design based on the TI OMAP-L138 processor, which integrates C6000's DSP facilities and an ARM processor. The modem is the result of a collaboration between the Northwestern Polytechnical University of Xi'an and the China Ship Research and Development Academy. The authors implemented an OFDM modulation scheme with 512 QPSK-modulated subcarriers at a total bit rate of 4 kb/s. The connection between the modem hardware and a PC is required to feed the correct transmission and reception parameters to the DSP. The modem architecture has been tested using data recorded from a lake experiment.

10) Newcastle University and Blueprint Subsea's SeaTrac Modem (UK, 2014–): Blueprint Subsea developed the SeaTrac modem based on front-end specifications by Newcastle University, UK, in the context of the EU-FP7 CADDY project [34]. The aim of the project is to provide reliable communication links among a diver, an AUV, and an autonomous surface vehicle (ASV). The acoustic front end operates in the 24–32-kHz band, with a single transmitting piezoelectric transducer element and four smaller elements above, forming a 3-D ultrashort baseline (USBL) array. Signal processing is implemented on the microcontroller embedded in the ARM Cortex M4 architecture. The modem supports 1) a low-data rate mode (about 100 b/s), which involves up-sweeping and down-sweeping linear chirps that represent the ones and zeros of a binary digital signal, and 2) an intermediate-data rate mode that achieves a link throughput up to 1.4 kb/s via QPSK symbols spread by short chunks of a 8191-point binary m-sequence (via direct-sequence spread spectrum, DSSS), block Doppler compensation and Reed–Solomon (RS) coding. The low-rate modulation is also leveraged on for USBL processing. The class-D amplifier of the modem outputs source levels of up to 176 dB re 1 μPa^2 m^2. A network layer is not yet implemented as it is not needed for the short-range star-topology network of CADDY.

Tests carried out in marinas and pools featuring harsh multipath conditions showed the high robustness of the low-bit-rate scheme and the good robustness of the medium-bit-rate scheme, both in terms of error rate and in terms of range estimates. The modem described in [34] is not presented as a reprogrammable-by-design platform. In addition, it mounts some specialized hardware (e.g., four 16-b analog-to-digital converters), and from the information in [34] and on the manufacturer's website, it does not seem to operate on top of a GP operating system (OS). Nevertheless, with sufficient memory available, the same architecture could double as a controller for the network protocols. The latter configuration is considered and proven by the NILUS modem, which works by leveraging on the components already available on a

standard system-on-module board, and an open-source OS to drive the modem. The NILUS approach is presented in Section III.

11) University at Buffalo and Northeastern University's SEANet and SDAM Prototypes (USA, 2015–): A collaboration between the University at Buffalo, NY, USA and the Northeastern University, Boston, MA, USA, led to the design of a software-defined acoustic networking framework [35] and of a reprogrammable acoustic modem [36] constructed using COTS components. In particular, the high-rate reprogrammable version of the SDAM detailed in [36] and [37] is built around software-defined radio components. The system includes an USRP N210 based on the Xilinx Spartan 3A-DSP 3400 FPGA, with two daughter cards providing an operational frequency range of 0–30 MHz, which is adequate for the RESON TC4013 transducer (range from 1 Hz to 170 kHz). External preamplifier and power amplifier circuitry completes the setup. The baseband signal processing algorithms and the network protocols are executed on a host PC running software such as GNU Radio and Matlab, and are connected to the SDAM via a gigabit Ethernet link.

The system provides reconfigurability options that include switching the data and coding rate of the modulation scheme in use, as well as switching between different schemes. Adaptive mechanisms to switch between different PHY configurations have been implemented and tested both in a tank and in a lake experiment. The platform provides support for full PHY and network reprogrammability by acting on the software in the host PC and on the FPGA onboard the USRP, if needed. A low-end version is detailed and tested in a tank in [35], and incorporates the Teensy 3.1 development platform in place of the USRP N210.

The latest development of the platform, named SEANet-G3 [38], is based on a Xilinx Zynq Z-7020 system-on-chip architecture incorporating a dual ARM Cortex-A9 (866 MHz) and an Avnet PicoZed XC7Z020-1CLG400 FPGA. Recent lake tests demonstrated that the platform can achieve a data rate of about 500 kb/s over short-range horizontal links (e.g., 10 m).

C. Industrial Developments

Since 2010, several commercial modem manufacturers have been taking steps toward more reconfigurability and reprogrammability for the (skilled) end user. This concerns two European companies (Germany), one Asian company (Singapore), and one USA-based company.

1) Develogic HAM Series (Germany, 2010–): The first commercial underwater acoustic modem manufacturer that opened their system to the end user is Develogic (Germany), by releasing their hydroacoustic modem (HAM) series in 2010. Using a JTAG interface and Texas Instrument's Code Composer Studio, the TI DSP of the Develogic modem can be reprogrammed in C/C++ with another PHY protocol in a GNU Radio-based framework (including a signal-processing library) [39]. The off-the-shelf HAM modems do not contain a network layer, but within a European collaboration (Section II-D3) it was shown that network-layer functionality can be successfully added to the Develogic modem by adding an additional external processor board.

2) EvoLogics WiSE Series (Germany, 2012–): A few years later, EvoLogics released a COTS underwater acoustic modem, named White-line Science Edition (WiSE) [40], which embeds a network-development framework in a sandbox (including, e.g., a preinstalled C/C++ toolchain and interpreters for scripting languages). This enables support for network-protocol simulation and development packages such as ns2: as an example, the ns2/MIRACLE-based DESERT Underwater [41], [42] and SUNSET [43] frameworks leverage on this flexibility to drive EvoLogics modems. In addition to the above, EvoLogics modems provide a protocol called D-MAC [44]

that can be exploited to access both the burst-data transmission and the instant-message schemes of the EvoLogics modems [sweep-spread carrier (S2C) modulation], as well as more advanced functions including positioning [45] and high-precision transmission timing.

More recently, EvoLogics has designed a framework for underwater networking and positioning called EviNS [46], which is written using the Erlang language. The objective of the framework is to provide lightweight concurrency, soft real-time operations, easier updates while the software is running, and robustness against software errors. The latter is inherent to the Erlang philosophy, where different processes are isolated by design, such that software misbehaviors due to erroneous programming do not propagate to the whole system and can be easily recovered from.

3) Subnero's Modem and UnetStack (Singapore, 2013–): In October 2013, NUS ARL and the Subnero company released v1.1.1 of UnetStack, an agent-oriented Java/Groovy implementation of an underwater networking stack [47]. Born from one of the first unified simulation and experimentation frameworks proposed and successfully demonstrated in water using the ARL modems [27], UnetStack was introduced and thoroughly discussed in [48], and was tested first on the NUS UNET-2 modems [26] (Section II-B5), on which the Subnero modem [49] is based. UnetStack makes it possible both to reprogram the network protocols and to reconfigure their parameters. Subnero's modems widen the reprogramming options to include PHY schemes and parameters (frequency-hopped binary phase-shift keying). A typical configuration includes a 500-b/s command link running along with a 2–10-kb/s data link, a 27-kHz carrier frequency, 18 kHz of bandwidth, error-correction codes, and a communication range of 2–3 km. All parameters are dynamically customizable.

PHY reprogramming has been showcased in [50] (for OFDM), making the Subnero modem the first commercial modem that is natively reprogrammable at all layers of the OSI stack. However, it requires the programming of a TigerSHARC DSP and of a Stratix III FPGA. Alternatively, the UnetStack baseband service makes it possible to reprogram the modem's PHY using the Java/Groovy language. This is achievable as long as the resources of the programmable logic allow the user to employ the FPGA/DSP as a GP processor. For the most computationally efficient implementation, the direct programming of the FPGA/DSP would still be required. UnetStack makes a simulation mode available to test the developed software and agents. The same software can be adapted to work on UnetStack-compatible hardware modems. In September 2013, EvoLogics announced that their WiSE modem series was upgraded to also support UnetStack, in collaboration with Subnero [51].

4) Teledyne Benthos' Steps Toward Reconfigurability (USA, 2013–): Teledyne Benthos has been alleviating some constraints that made their modem closed to external modifications and reprogrammable only by specialized personnel. With reference to the SM-(9)75 modem, a networking API can now be provided to bypass the modem's native networking functions and command the data-link and physical layers (which remain unchanged) through a modem management protocol (MMP) [52], [53]. Based on this flexibility, the University at Buffalo has been working on a reconfigurable underwater acoustic testbed, where the nodes mount an SM-(9)75 system, complemented by an auxiliary data processor and a Gumstix running control software [52], [54]. The SM-(9)75 modem natively supports the transmission of custom waveforms.

D. European Collaborations

In parallel with the (mainly) USA academic efforts (Sections II-A and II-B), a series of European projects on underwater (network) communications have been running, some of them financed by the national Ministries of Defense of the participating nations (Sections II-D1, II-D3), and others (co)financed by the European Commission (Brussels) (Sections II-D2, II-D4).

1) EUROPA MoU UCAC (2005–2008): In a defense cooperation between Sweden, Norway, Finland, Denmark, The Netherlands, Germany, and Italy, under the EUROPA Memorandum of Understanding (MoU) [55], covert point-to-point underwater acoustic communications was investigated. In this project, called Unmanned Underwater Vehicle (UUV) Covert Acoustic Communications (UCAC) [56], successful reception at very low signal-to-noise ratios (down to -18 dB) was achieved using a very robust multiband OFDM modulation [57]. This modulation was developed in Matlab by TNO and Delft University of Technology [58] and included as Matlab-compiled C libraries in the modem software by Patria. The implementation ran in real time under Linux on the Intel Pentium-M processor (2 GHz) of the UCAC modems built by Teledyne RESON (transducer, casing) and Patria (hydrophone, electronics). Transmissions were made between a Kongsberg HUGIN 1000 AUV and its mothership in a 1.5–5-kHz band over ranges up to 20 km at data rates of 4–80 b/s. The maximum achieved communication range during early sea tests was 52 km [58].

2) EU-FP7 UAN (2008–2011) and EU-FP7 CLAM (2011–2013): Publicly funded European research efforts explicitly targeting reconfigurable and reprogrammable underwater acoustic modems started in 2008 with the Underwater Acoustic Networks (UAN, 2008–2011) project under the 7th Framework Programme (FP7) of the European Union. In UAN, a multilingual modem was created on which one of three PHY protocols could be selected via a specific offline command (no online/in-mission adaptivity). Later on, in the EU-FP7 CLAM project (CoLlAborative eMbedded networks for submarine surveillance, 2011–2013), a reprogrammable network layer was added to these modems. The modems were cNODE systems of Kongsberg Maritime, interfaced with a Gumstix [59] platform running the SUN-SET framework [43].

3) EDA RACUN (2010–2014): In 2010, the UCAC consortium (Section II-D1), minus Finland and Denmark, extended their activities to the other OSI layers in the European Defence Agency (EDA) project Robust Acoustic Communications in Underwater Networks (RACUN, 2010-2014, [60]), in which an *ad hoc* deployable underwater acoustic network was successfully demonstrated for relevant military scenarios in shallow waters (mine hunting, maritime surveillance) [61].

In the context of the EDA-RACUN project, two different modems were employed. Develogic's modem (Section II-C1) featured a reprogrammable TI TMS320C6713 300-MHz 32-b DSP that could host the signal processing required by PHY algorithms. The platform included 128-MB SDRAM and TI subsystems for low-power management and signal conditioning, and was quite compact yet reprogrammable at the physical layer. In RACUN, the network-layer reprogrammability for the Develogic modem was provided via a serial connection to an external Gumstix [59] or NetDCU [62] platform. In parallel, Saab worked on a research platform supporting both physical- and network-layer reprogrammability. The platform is based on the open publish–subscribe (OPS) architecture [63], which offers asynchronous, efficient, and type-secure publish–subscribe operations between processes running on different platforms written in multiple programming languages. During the RACUN sea trials in 2013 and 2014, the Saab modem framework, named SMUM, was running on a high-performance mini-PC (Avalue EPS-QM67: Intel i7, 2.1-GHz, 8-GB RAM, 160-GB drive, Windows 7 Pro) resulting in quite large nodes, comparable in size to the Scripps system (Section II-B3). For the future, the development of a more compact version is foreseen. Two options were implemented for reconfiguring both the Develogic modem and the Saab modem at the network layer,

TABLE II
SUMMARY OF THE RECONFIGURABLE, REPROGRAMMABLE, AND SOFTWARE-DEFINED MODEMS SURVEYED IN THIS PAPER.
THE MEANING OF THE COLUMNS AND BOX COLORS ARE EXPLAINED IN SECTION II-E

	PHY	NET	Available	GP			PHY	NET	Available	GP	
Section II-A rModem	green	red	green	red		µModem	Export restrict.	red	green	red	**Section II-B**
UW Tmote	green	Not tested	red	red		UCONN/AquaSeNT	yellow	green	green	red	
AquaNodes	red	green	red	red		Scripps	green	green	green	green	
UANT	green	green	red	green		ITACA	green	red	green	red	
UCSD	yellow	red	red	red		UNET-2	green	green	green	(c)	
UNSW	green	Not tested	red	yellow		FAST	Laptop	Laptop	green	green	
CAS	green	red	yellow	red		Harbin	red	green	yellow	green	
Section II-C Develogic	green	(a)	green	red		SeaModem	green	green	green	green	
EvoLogics	(b)	green	green	yellow		Xi'an	green	red	green	red	
Subnero	green	green	green	(c)		SeaTrac	green	Not tested	green	green	
Buffalo/TB	red	(b)	green	red		Buffalo/NEU	green	green	green	(d)	
Section II-D UCAC	green	red	red	green		Saab/RACUN	green	green	green	green	
UAN/CLAM	yellow	green	yellow	red		Proteus II	green	green	green	red	
						Section III NILUS	green	green	yellow	green	

Full/empty circles facilitate reading in a black-and-wide print (one full circle for red, two for yellow and three for green).

(a) The NET protocols attached to the Develogic modem were made reconfigurable in the context of the RACUN project.

(b) To the best of the authors' knowledge, the PHY layer of the EvoLogics modem and the NET layer of the Teledyne modem were reprogrammed in the context of the SUNRISE project.

(c) The UNET-2/Subnero also offer the option to employ the FPGA/DSP as if they were general purpose-processors, and to reprogram them using Java/Groovy.

(d) The Buffalo/NEU entry summarizes the three generations of the SEANet reconfigurable modem prototype: versions 1 and 2 [35], [36], respectively, incorporate an FPGA-based USRP and an ARM Cortex M4-based microcontroller, whereas the latest version [38] is built around a Zynq System-on-Chip architecture with 2x ARM Cortex A9 + FPGA.

namely a custom C-based framework and the DESERT Underwater framework [42], [64].

4) EU-FP7 SUNRISE (2013–2016) and University of Twente's Proteus II: The EU FP7 SUNRISE project [65] has the objective to build a consolidated set of underwater networking testbeds [66] and explicitly targets the development of an SDOAM. Among the SUNRISE partners, the NATO STO CMRE has been advocating the design of SDOAMs [67]–[69]. The conceptual architecture of CMRE's SDOAM involves multiple protocols at all layers of the OSI stack, whose selection is administered by a policy engine. In turn, this is a starting point to evolve toward cognitive architectures.

The NEMO architecture [70] makes progress toward a concrete implementation of the CMRE SDOAM concept. NEMO is a collection of modules (providing network protocols and functions), services (PHY switching, synchronization, navigation, etc.) drivers, and NATO-specific information bundling such as the codex system [71], tied together by the interprocess communication functionalities of the MOOS middleware [72]. The latest reported development was tested during the REP'14 cruise in July 2014 and included: relative clock synchronization; an information dispatcher, named "herald," that handles the spread of information in the network with a tunable level of accuracy; the codex for information representation; flooding-based routing with duplicate suppression; and a packet-agnostic queuing mechanism.

The Proteus II modem [73] of the University of Twente (The Netherlands) has been developed in the context of SUNRISE. The Proteus II inherits and improves over the previous design of the SeaSTAR modem [74]. The modem is designed for low power consumption and is based on two ARM Cortex M4 microcontrollers, one to run the transmitter- and receiver-side processing chains of the acoustic modem (implementing 10–200-baud FSK as software radio components) and a second one to run network protocols and perform sensor data processing, including automatic time synchronization functionalities. The modem also exposes an antenna connector that can be used to turn the unit into a radio/acoustic gateway. The power consumption of the modem is sufficiently limited to allow unattended operations for about 1 month using a 5-Ah battery [73].

Buffalo's testbed (Section II-C4) is also part of the SUNRISE project.

E. Summary

For the reader's convenience, we summarize the characteristics of the discussed modems in Table I. In addition, Table II classifies the modems by coding the level of reprogrammability (of PHY and NET), the availability, and the generality of the architecture (GP) of each modem in different colors. In Table II, a green PHY box means that the modem's physical layer is reprogrammable by the end user, a red box means that it is not, while a yellow box means that the user is allowed to reconfigure some PHY parameters, albeit not to completely reprogram the PHY algorithms. The NET boxes indicate the same for the network protocol stack. Availability can be off-the-shelf (green), none (red), or only under the cap of a joint cooperation (yellow). In the context of such a cooperation, it may also be possible to reprogram PHY schemes or NET protocols that would normally be only reconfigurable or closed. Finally, the programmable platform can be a GP processor (green), a dedicated processor such as a DSP or microcontroller (red), or both (yellow). To facilitate reading in a black-and-white print, we provided three full/empty circles in the top-right corner of each table cell, where one full circle represents "red," two represent "yellow," and three represent "green." The GP column gives an idea of the accessibility and flexibility of a modem to the user. In general, a thorough knowledge of the specific hardware and/or software architecture is needed to modify a modem implementation on a dedicated architecture, such as a DSP or microcontroller. On the contrary, specialized knowledge is less needed when the modem code is written using standard languages (e.g., C/C++) and runs under an operating system on a GP processor. The blue frames highlight doubly reprogrammable standalone underwater acoustic modems. Both Tables I and II also include the NILUS SoftModem, which is presented and evaluated in the following section.

III. THE NILUS SOFTMODEM

The NILUS bottom node is a lightweight underwater sensor node developed by FFI [76]. Extensive experience with this system has been gained in the NATO Joint Research Project Next Generation Autonomous Systems (NGAS), the EDA project RACUN [60], and in national Norwegian trials. In the framework of a bilateral Netherlands/Norway defense research cooperation (2014–), a new version of the NILUS node (MK 2) was developed, having an improved sensor suite and more processing capabilities. Also, the Teledyne Benthos modems of the original NILUS (MK 1) nodes are (optionally) replaced by a software-defined modem developed at TNO, which is presented below. The modem has been designed to be able to run on GP processors only, and can thus be fully reprogrammed using well-known programming languages acting on well-defined interfaces. In the same vein, the modem architecture is devoid of DSPs and FPGAs, which require specialized programming efforts and tools. The NILUS MK 2 differentiates itself from most currently available reprogrammable and reconfigurable modems, which often rely on specialized hardware to carry out PHY operations (see Table II). In this respect, the NILUS MK 2 makes significant steps toward the realization of an SDOAM.

A. Modem hardware

The modem's transducer is mounted on a subsurface buoy (see Fig. 1) hovering above the NILUS frame, about 6 m above the seafloor. The buoy includes the amplifier, the impedance-matching network, and the projector (3–11 kHz, −3-dB bandwidth). The receiving hydrophone and preamplifier are mounted on the tripod structure, approximately 1 m above the seafloor. This hydrophone is the top element of a tetrahedral array: the other hydrophone elements are only used for the sensor

Fig. 1. The NILUS MK 2 bottom node, with the new software-defined modem (transducer in special float shape for stability in currents).

capabilities of the NILUS node for the time being, but are potentially also available for modem reception.

The NILUS SoftModem framework, consisting of the PHY, MAC, and NET layers, as well as audio and data interfaces, is running under GNU/Linux on the Variscite System-on-Module board VAR-SOM-OM44, which is based on a Texas Instruments OMAP4460 1.5-GHz Dual-Core ARM Cortex-A9 processor [77]. This compact and energy-efficient yet powerful computational platform, developed for the smartphone/tablet industry, has sufficient resources (7500 MIPS, 1-GB RAM, 8-GB MicroSD) for running a state-of-the-art communications stack programmed in an architecture-independent high-level programming language (ISO C, C++), while still being compact and economic in terms of cost and power usage, and substantially easier to program than DSP boards. The VAR-SOM-OM44 is equipped with a dedicated high-quality audio codec, the TWL6040 [78].

B. Modem Software

Fig. 2 shows the architecture and data-flow diagram of the NILUS SoftModem, where the "NILUS Modem Software" block contains the modulation and demodulation routines (PHY), which interfaces with the "MAC and Network layers" via standard TCP connections. The connection between the PHY block and the "Transducer and Preamp," including buffering, makes use of the Advanced Linux Sound Architecture (ALSA) driver for the integrated sound card. The data connection between the NET layer and the "NILUS Software" (APP layer, on a separate processing board) is realized via a UART interface. The data from the receiving hydrophone preamp comes in by the "Common System Bus." The hydrophone data are published on this digital bus in an ALSA-compatible PCM format. The "ALSA Audio Control" can therefore rely on the ALSA driver for retrieving the hydrophone data.

The modulation scheme employed on the NILUS MK 2 node involves modulating root-raised cosine-windowed QPSK pulses, which are spread through the frequency-repetition spread-spectrum (FRSS) scheme [80], a coherent multiband spread-spectrum method based on [81]. Fig. 3 shows an example of an FRSS signal carrying 256 b at a bit rate of approximately 90 b/s, having a length of approximately 2.9 s. The scheme was originally developed and extensively tested in Matlab at TNO and FFI. The method applies a linear (decision feedforward) multichannel equalizer with an ultrawideband phase-locked-loop solution to several "frequency-band channels" containing the same information but multiplied by a spreading code across frequency bands. The
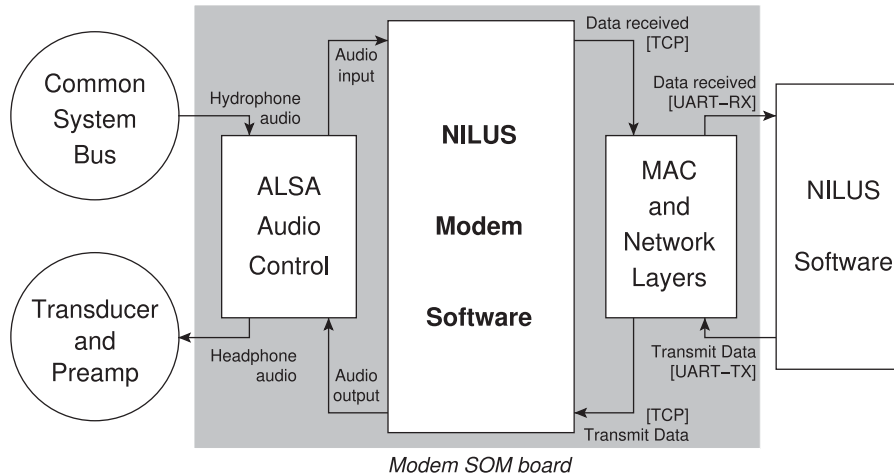
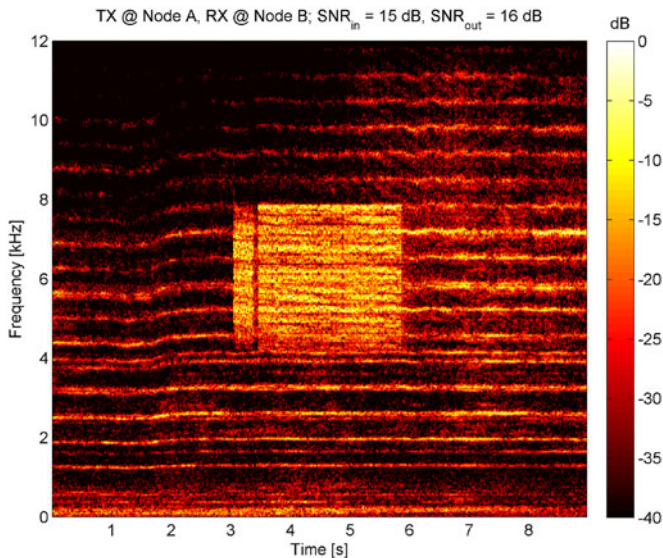Fig. 2. Architecture and data-flow diagram of the NILUS SoftModem [79].

Fig. 3. Example FRSS signal (approx. 2.9 s, 256 info bits, approximately 90 b/s) in the presence of nearby ship noise. The message was correctly decoded at node B of Fig. 4(a).

filter taps are updated by the LMS algorithm, and a Viterbi algorithm is used to decode the rate-1/2 convolutional coding. Using the output SNR [82], the bit probabilities are computed, which are used as "soft information" for the Viterbi decoder. The FRSS modulation is programmed in C++11 in a modular Qt framework. Public GPL-licensed libraries are employed for the FFT (FFTW), linear matrix algebra operations (Armadillo), and resampling routines (Secret Rabbit Code).

For the MAC protocol, ALOHA with carrier sensing (CS) is employed, while a restricted flooding network protocol named Dflood, developed by FFI in the context of the RACUN project (see Section II-D3), is employed at the network layer [83], [84]. The APP layer expresses commands and information reports of different types using standard packets of size equal to 128 b. The entire network protocol stack including MAC and Dflood runs under the ns2-based DESERT Underwater framework [41], [42], with ns2 running in parallel with the modem (PHY) on the SoftModem processor board. By virtue of this, it is possible to use exactly the same software for the network protocol stack in network simulations and at sea.

C. Experimental Results

The first open-water tests for the new NILUS nodes were successfully performed in November 2014 in the inner harbor of Horten (Indre Havn), Norway [82]. The applied frequency band is the RACUN band, from 4 to 8 kHz, which corresponds to the −1-dB bandwidth of the modem's transducer. In a four-node network with an average internode distance of about 1 km, an overall packet delivery ratio (PDR) at the physical layer of approximately 0.86 was achieved. This PHY PDR is defined as the ratio between the number of messages that were received error free and the number of detected messages. Given the initial difficulties of a first trial, the use of a relatively low source level (SL), and the presence of a motorboat sailing through the network, this PDR was found to be satisfactory. This judgement was confirmed by the next trial in March 2015 in the shallow Haringvliet estuary in The Netherlands, where a (PHY) PDR of about 0.95 was achieved in a four-node network with an average internode distance of about 3 km (at a higher SL). Both trials featured direct line of sight between all nodes.

After these initial trials, a project demonstration trial was performed in April 2015, again in the inner harbor of Horten and this time extended to deeper waters in the Oslofjord. One of the scenarios tested was a relay connection between the harbor and the fjord; see Fig. 4(a). This experiment was conducted in the morning of April 16, in nice weather conditions (clear sky, some wind, sea state approximately 1). The nodes were deployed at bottom depths of approximately 8 m (position C) 18 m (positions A, B, and D), and 80 m (positions E, F, and G). Diverse bottom sediments were present in the trial area (mud, sand, rocks). The properties of the water column were sampled using a CTD probe. Not much temporal variability was observed in the collected sound-speed profiles. The mainly upward-refracting sound-speed profile resulted in a strong influence of the sea-surface scattering on the time delay (multipath, reverberation) and Doppler spread. Furthermore, there was some ship traffic, especially in the Oslofjord.

The (PHY) PDR statistics in the multihop network were collected for more than an hour, resulting in the network connectivity figures displayed in Fig. 4(b). The number of packets received by each node during this time varied between 7 and 29, providing a total of 273 investigated packet receptions in the network. The overall PDR was $218/273 \approx 0.80$ for internode distances from about 450 to 2350 m (average about 1210 m). The results show that the network featured long-range point-to-point connections in deep water with direct line of sight (e.g., F ↔ G), whereas hopping was needed for large ranges in shallow water without direct line of sight (e.g., C ↔ F, up to three
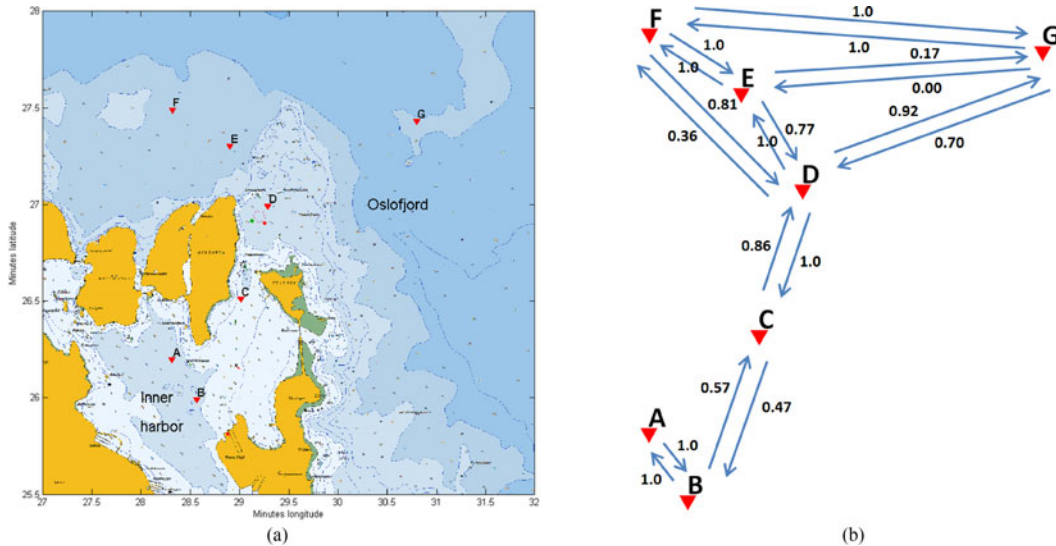
Fig. 4.   (a) Map of Horten Harbor and Oslofjord (mosaic of TELchart/C-Map screenshots) with node positions indicated (in minutes relative to 59°N, 10°E). (b) PHY packet delivery ratios for the field test performed on April 16, 2015.

hops). Without relay nodes C and D, the nodes in the harbor would have been disconnected from the nodes in the fjord. In fact, as node C was originally not present in the network, nodes B and D were initially disconnected. After about 40 min, the node at position A (base station at barge) was moved to position C (base station at support vessel), thus connecting the fjord with the inner harbor. The PDR statistics for both time intervals have been merged in the connectivity graph. On the network-layer level, the PDR was 1.0 between all NILUS nodes and the base station, achieved through the retransmission scheme of the applied NET protocol (i.e., no unique packets were lost after maximum one retransmission).

It is interesting to observe that perfect links $(F \leftrightarrow G, E \leftrightarrow F)$ coexist close to paths with (almost) no connection at all $(E \leftrightarrow G)$. This extreme difference is believed to be mainly due to the presence of an underwater hill between positions E and G. At these positions, the local depth is about 80 m, while the hill rises up to a broad plateau of about 12-m depth around the line E–G. With the transmitter being approximately 6 m above the bottom and the receiver approximately 1 m, there is no direct line of sight between the nodes at E and G and all communication between E and G goes via surface-reflected paths. On the other hand, there is a direct line of sight between E and F. For the line F–G, an offshoot of the hill just seems to prevent a direct line of sight, but still this link was working well, most probably because F is much farther from the hill than E. Messages were received by node G from nodes D, E, and F in the same time frame, so the background noise level was the same for all receptions. Also, signal collisions were equally spread between messages from all three nodes. The fact that the longest direct link $(F \leftrightarrow G)$ was perfect suggests that the modem source level was more than sufficient and that the point-to-point communication range was limited by acoustic shadowing and/or delay-Doppler spreading, especially in the very shallow sections of the network area. This nicely illustrates the usefulness of multihop networks with alternative routes to the destination.

The successful demonstration of the NILUS MK 2 bottom nodes, including NILUS SoftModems, show that generic (open) architectures can be applied to underwater acoustic modems, while still meeting (semi) operational requirements in terms of processing power, size, and endurance. The processing power should be sufficient for real-time execution of a "heavy-duty" network stack (robust, high throughput), the size should be small enough to fit into "man-portable" sensor sys-

tems [76], while the required endurance depends on the application and operational scenario.

IV. CONCLUSION

Our historical review showed that the most flexible reprogrammable underwater acoustic modem is a software-defined modem, where specific tools make it possible for the user to apply changes to (part of) the modem firmware without involving the modem manufacturer. For some modems, reprogrammability applies only to the networking components, for other modems only to the physical layer, and in several cases, only some level of parameter reconfiguration is allowed to the user. Notably, in seven cases presented in the review (UANT, Scripps, UNET-2, SeaModem, Buffalo/NEU, Subnero, and Saab) both the physical and the network layers are fully reprogrammable by the end user. These doubly reprogrammable modems are marked in Table II with blue frames, together with the NILUS SoftModem. Open-architecture modems, featuring a fully transparent and free software/hardware architecture, are still missing for the underwater community, but they are advocated by several research efforts at different institutions, and further advances are expected in the coming years.

Notably, a number of the reprogrammable modems reviewed in this paper (AquaNodes, UCONN/AquaSeNT, Scripps, SeaModem, SeaTrac, Buffalo/NEU, Develogic, EvoLogics, UNET-2/Subnero, UCAC, UAN/CLAM, Saab, Proteus II) have actually seen seaborne tests and/or a development phase in shallow-water environments, while other reprogrammable modems were at an earlier stage of development (e.g., detached electronics, in-tank transducer, offline tests). Some modems also demonstrated MIMO functionalities.

Whereas most reviewed modems are research prototypes that are not commercially available, several COTS modems implement software-defined functions. Good examples are Develogic, who opened their physical layer, EvoLogics, who implemented a flexible network layer, and Subnero, who made it possible to customize both the physical layer and the network protocols.

In addition to the review of these activities, we presented the specific setup of the NILUS SoftModem, jointly designed by TNO and FFI. This modem is based on powerful GP-architecture computation platforms, such as the Dual-Core ARM Cortex-A9 board, which enables the real-time execution of both physical-layer and network-layer

protocols programmed in generic high-level languages. Therefore, the PHY/NET programmer does not have to worry about the hardware details, since one only has to comply with well-defined APIs in a high-level programming framework. To the best of the authors' knowledge, the NILUS SoftModem is the only one among all surveyed modems showing this level of flexibility, along with a compact rugged design and (semi) operational readiness. In some cases (notably, Develogic [39], UNET-II [26] and SubNero [47]), the system is arranged so that high-level programming languages can be employed to directly reprogram the DSP/FPGA that run PHY algorithms on the modem. This is possible as long as the DSP/FPGA memories are large enough, but it will not result in the most efficient implementation. These considerations are overcome by the design of the NILUS SoftModem, which is fully based on GP computing architectures. Such compact and efficient GP platforms, product of smartphone/tablet industry, may provide the bridge to software-defined open-architecture underwater acoustic modems becoming common practice.

We remark that the flexibility of a softmodem running under an operating system on a GP processor comes at a price, such as increased power usage and wakeup time. This is a tradeoff that has to be made by the end user and will depend on the use case, on the scenario, and on flexibility requirements. However, as long as there are no heavy-duty underwater communication stacks being standardized and implemented on all COTS underwater modems, a softmodem may be the only way to achieve interoperability in a heterogeneous network, e.g., as set up by allied navies in a joint operation. Our vision for the future is to further exploit the NILUS SoftModem's flexibility by developing a smart adaptive communication stack, i.e., a modem that can switch autonomously between protocols, frequency bands, data rates, message lengths, etc., based on environmental conditions and operational needs.

REFERENCES

[1] E. Jones, "The application of software radio techniques to underwater acoustic communications," in *Proc. IEEE/OES OCEANS*, Aberdeen, Scotland, Jun. 2007

[2] E. M. Sözer and M. Stojanovic, "Reconfigurable acoustic modem for underwater sensor networks," in *Proc. ACM WUWNet*, Los Angeles, CA, USA, Sep. 2006.

[3] L. Freitag *et al.*, "The WHOI Micro-Modem: An acoustic communications and navigation system for multiple platforms," in *Proc. MTS/IEEE OCEANS*, Washington, DC, USA, 2005.

[4] M. Aydinlik, A. T. Ozdemir, and M. Stojanovic, "A physical layer implementation on reconfigurable underwater acoustic modem," in *Proc. MTS/IEEE OCEANS*, Québec City, QC, Canada, Sep. 2008.

[5] R. Jurdak, C. V. Lopes, and P. Baldi, "Software acoustic modems for short range mote-based underwater sensor networks," in *Proc. IEEE OCEANS*, Singapore, May 2006.

[6] R. Jurdak, P. Aguiar, P. Baldi, and C. V. Lopes, "Software modems for underwater sensor networks," in *Proc. IEEE/OES OCEANS*, Aberdeen, Scotland, Jun. 2007.

[7] I. Vasilescu, C. Detweiler, and D. Rus, "AquaNodes: An underwater sensor network," in *Proc. ACM WUWNet*, Montréal, QC, Canada, Sep. 2007.

[8] D. Torres, J. Friedman, T. Schmid, and M. B. Srivastava, "Software-defined underwater acoustic networking platform," in *Proc. ACM WUWNet*, Berkeley, CA, USA, Nov. 2009.

[9] D. Torres *et al.*, "Software-defined underwater acoustic networking platform and its applications," *Ad Hoc Netw.*, vol. 34, pp. 252–264, 2015.

[10] B. Benson *et al.*, "Design of a low-cost underwater acoustic modem," *IEEE Embedded Syst. Lett.*, vol. 2, no. 3, pp. 58–61, Sep. 2010.

[11] N. Nowsheen, C. Benson, and M. Frater, "Design of a high frequency FPGA acoustic modem for underwater communication," in *Proc. MTS/IEEE OCEANS*, Sydney, N.S.W., Australia, May 2010.

[12] N. Nowsheen, C. Benson, and M. Frater, "A high data-rate, software-defined underwater acoustic modem," in *Proc. MTS/IEEE OCEANS*, Seattle, WA, USA, Sep. 2010.

[13] C. Benson, M. Ryan, and M. Frater, "Towards robust high data-rate hydro-acoustic modems," in *Proc. MTS/IEEE OCEANS*, Hampton Roads, VA, USA, Oct. 2012.

[14] Y. Li and H. Huang, "The design and experiment of a software-defined acoustic modem for underwater sensor network," in *Proc. MTS/IEEE OCEANS*, Sydney, N.S.W., Australia, May 2010.

[15] E. Gallimore *et al.*, "The WHOI micromodem-2: A scalable system for acoustic communications and networking," in *Proc. MTS/IEEE OCEANS*, Seattle, WA, USA, 2010.

[16] Woods Hole Oceanographic Institution, Acoustic Communications Group, "Micromodem pages," accessed Jun 2016. [Online]. Available: http://acomms.whoi.edu/micro-modem/.

[17] B. Li *et al.*, "MIMO-OFDM for high-rate underwater acoustic communications," *IEEE J. Ocean. Eng.*, vol. 34, no. 4, pp. 634–644, Oct. 2009.

[18] H. Yan *et al.*, "DSP based receiver implementation for OFDM acoustic modems," *Phys. Commun.*, vol. 5, no. 1, pp. 22–32, Mar. 2012.

[19] S. Zhou and Z. Wang, *OFDM for Underwater Acoustic Communications*, New York, NY, USA: Wiley, May 2014.

[20] L. Wan *et al.*, "Adaptive modulation and coding for underwater acoustic OFDM," *IEEE J. Ocean. Eng.*, vol. 40, no. 2, pp. 327–336, Apr. 2015.

[21] S. N. Le, Z. Peng, J.-H. Cui, H. Zhou, and J. Liao, "SeaLinx: A multi-instance protocol stack architecture for underwater networking," in *Proc. ACM WUWNet*, Kaohsiung, Taiwan, Nov. 2013.

[22] Y. Su *et al.*, "A versatile lab testbed for underwater sensor networks," in *Proc. MTS/IEEE OCEANS*, San Diego, CA, USA, Sep. 2013.

[23] W. S. Hodgkiss and J. D. Skinner, "A multichannel software-defined acoustic modem," in *Proc. ECUA*, Istanbul, Turkey, Jul. 2010.

[24] A. Sánchez, S. Blanc, P. Yuste, and J. J. Serrano, "A low cost and high efficient acoustic modem for underwater sensor networks," in *Proc. IEEE/OES OCEANS*, Santander, Spain, Jun. 2011.

[25] A. Sánchez, S. Blanc, P. Yuste, A. Perles, and J. J. Serrano, "An ultra-low power and flexible acoustic modem design to develop energy-efficient underwater sensor networks," *MDPI Sensors*, vol. 12, no. 6, pp. 6837–6856, 2012. [Online]. Available: http://www.mdpi.com/1424-8220/12/6/6837.

[26] M. Chitre, I. Topor, and T.-B. Koay, "The UNET-2 modem: An extensible tool for underwater networking research," in *Proc. MTS/IEEE OCEANS*, Yeosu, South Korea, May 2012.

[27] S. Shahabudeen, M. Chitre, M. Motani, and A. Siah, "Unified simulation and implementation software framework for underwater MAC protocol development," in *Proc. MTS/IEEE OCEANS*, Biloxi, MS, USA, Oct. 2009.

[28] N. Ahmed, W. bin Abbas, and A. A. Syed, "A low-cost and flexible underwater platform to promote experiments in UWSN research," in *Proc. ACM WUWNet*, Los Angeles, CA, USA, Nov. 2012.

[29] J. Zhang, G. Qiao, and C. Wang, "An intelligent management platform for underwater acoustic modem," in *Proc. ISCID*, Hangzhou, China, Oct. 2012.

[30] G. Cario, A. Casavola, M. Lupia, and C. Rosace, "SeaModem: A low-cost underwater acoustic modem for shallow water communication," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[31] R. Petroccia, G. Cario, M. Lupia, V. Djapic, and C. Petrioli, "First in-field experiments with a "bilingual" underwater acoustic modem supporting the JANUS standard," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[32] J. Potter *et al.*, "The JANUS underwater communications standard," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[33] C. Wang *et al.*, "Implementation of underwater acoustic modem based on the OMAP-L138 processor," in *Proc. IEEE ICSPCC*, Guilin, China, Aug. 2014.

[34] J. A. Neasham, G. Goodfellow, and R. Sharphouse, "Development of the "Seatrac" miniature acoustic modem and USBL positioning units for subsea robotics and diver applications," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[35] E. Demirors, B. G. Shankar, G. E. Santagati, and T. Melodia, "SEANet: A software-defined acoustic networking framework for reconfigurable underwater networking," in *Proc. ACM WUWNet*, Washington, DC, USA, Oct. 2015.

[36] E. Demirors, G. Sklivanitis, T. Melodia, S. Batalama, and D. Pados, "Software-defined underwater acoustic networks: Toward a high-rate real-time reconfigurable modem," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 64–71, Nov. 2015.

[37] E. Demirors, G. Sklivanitis, G. E. Santagati, T. Melodia, and S. N. Batalama, "Design of a software-defined underwater acoustic modem with real-time physical layer adaptation capabilities," in *Proc. ACM WUWNet*, Rome, Italy, Nov. 2014.

[38] E. Demirors and T. Melodia, "SEANet G3: High-data-rate software-defined underwater acoustic network platform," in *Proc. UComms*, Lerici, Italy, Sep. 2016.

[39] F. Berning, T. Radtke, S. Rautenberg, M. Motz, and I. Nissen, "A realization of the software defined radio concept in an underwater communication modem," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[40] EvoLogics, "White line science edition (WiSE) modems," accessed: Jun. 2016. [Online]. Available: http://www.evologics.de/en/products/developer_tools/index.html.

[41] R. Masiero *et al.*, "DESERT underwater: An NS-miracle based framework to DEsign, simulate, emulate and realize test-beds for underwater network protocols," in *Proc. IEEE/OES OCEANS*, Yeosu, South Korea, May 2012.

[42] P. Casari *et al.*, "Open-source suites for the underwater networking community: WOSS and DESERT underwater," *IEEE Network*, vol. 28, no. 5, pp. 38–46, Sep. 2014.

[43] C. Petrioli, R. Petroccia, and D. Spaccini, "SUNSET version 2.0: Enhanced framework for simulation, emulation and real-life testing of underwater wireless sensor networks," in *Proc. ACM WUWNet*, Kaohsiung, Taiwan, Nov. 2013.

[44] O. Kebkal, M. Komar, K. Kebkal, and R. Bannasch, "D-MAC: Media access control architecture for underwater acoustic sensor networks," in *Proc. IEEE/OES OCEANS*, Santander, Spain, Jun. 2011.

[45] O. Kebkal, M. Komar, K. Kebkal, and R. Bannasch, "Long-baseline hydro-acoustic positioning using D-MAC communication protocol," in *Proc. MTS/IEEE OCEANS*, Yeosu, South Korea, May 2012.

[46] O. Kebkal, V. Kebkal, and K. Kebkal, "EviNS: A framework for development of underwater acoustic sensor networks and positioning systems," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[47] NUS ARL, Subnero, "UNET—The Underwater NETworks project," accessed: Jun. 2016. [Online]. Available: http://www.unetstack.net.

[48] M. Chitre, R. Bhatnagar, and W.-S. Soh, "UnetStack: An agent-based software stack and simulator for underwater networks," in *Proc. MTS/IEEE OCEANS*, St. John's, NL, Canada, Sep. 2014.

[49] Subnero, "Underwater modem overview," accessed: Jun. 2016. [Online]. Available: http://subnero.com/technology/modem/.

[50] M. Chitre, R. Bhatnagar, M. Ignatius, and S. Suman, "Baseband signal processing with UnetStack," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[51] "EvoLogics newsletter," accessed: Jun. 2016. [Online]. Available: http://www.evologics.de/en/news.html?newsman_news_id=51

[52] H. Kulhandjian, L. Kuo, and T. Melodia, "Extended abstract: Development of a reconfigurable underwater networking testbed," in *Proc. ACM WUWNet*, Los Angeles, CA, USA, Nov. 2012.

[53] D. Green and S. McManus, "Smart modems—Underwater capabilities beyond communications," in *Proc. MTS/IEEE OCEANS*, Yeosu, South Korea, May 2012.

[54] H. Kulhandjian, L.-C. Kuo, T. Melodia, D. A. Pados, and D. Green, "Towards experimental evaluation of software-defined underwater networked systems," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2012.

[55] European Defence Agency (EDA), "European Understandings for Research Organisation, Programmes and Activities (EUROPA)," accessed: Jul. 2015. [Online]. Available: http://www.eda.europa.eu/RandTUserGuide/docs/default-document-library/europa_mou_version_signed_150501.pdf.

[56] P. van Walree *et al.*, "UUV covert acoustic communications," in *Proc. UAM*, Nafplion, Greece, Jun. 2009.

[57] H. Dol and P. van Walree, "Underwater acoustic communication research at TNO—Past and present," in *Proc. IEEE/OES OCEANS*, Santander, Spain, Jun. 2011.

[58] G. Leus and P. A. van Walree, "Multiband OFDM for covert acoustic communications," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 9, pp. 1662–1673, Dec. 2008.

[59] "Gumstix home page," accessed: Jun. 2016. [Online]. Available: http://www.gumstix.com/.

[60] J. Kalwa, "The RACUN project: Robust acoustic communications in underwater networks—An overview," in *Proc. IEEE/OES OCEANS*, Santander, Spain, Jun. 2011.

[61] P. Casari *et al.*, "Ad hoc acoustic networks of heterogeneous nodes for tactical underwater missions," 2016, [submitted].

[62] F & S Elektronik Systeme GmbH, "NetDCU platform," accessed: Jun. 2016. [Online]. Available: http://www.fs-net.de/en/products/netdcu/.

[63] A. Gravenstam, "OPS open publish-subscribe middleware," accessed: Jun. 2016. [Online]. Available: http://code.google.com/p/ops/.

[64] C. Tapparello *et al.*, "Performance evaluation of forwarding protocols for the RACUN network," in *Proc. ACM WUWNet*, Kaohsiung, Taiwan, Nov. 2013.

[65] "SUNRISE FP7 project web site," accessed: Jun. 2016. [Online]. Available: http://fp7-sunrise.eu/.

[66] C. Petrioli *et al.*, "The SUNRISE gate: Accessing the SUNRISE federation of underwater test-beds," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[67] P. Potter *et al.*, "Underwater communications protocols and architecture developments at NURC," in *Proc. IEEE/OES OCEANS*, Santander, Spain, Jun. 2011.

[68] J. Potter *et al.*, "Software-defined open architecture modem development at CMRE," NATO STO CMRE, Tech. Rep. CMRE-FR-2013-023, 2013.

[69] J. Potter *et al.*, "Software defined open architecture modem development at CMRE," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[70] A. Vermeij, T. C. Furfaro, and J. Alves, "NEMO: An architecture for software communications research in the maritime domain," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[71] NATO Standardization Agency, "ADatP-3(A) NATO message text formatting system (FORMETS), concept of FORMETS (CONFORMETS)," Brussels, Belgium, 2009, NATO/EAPC UNCLASSIFIED, Releasable to Australia, MD, ICI.

[72] "MOOS-IvP Home Page," accessed: Jun. 2016. [Online]. Available: http://www.moos-ivp.org/.

[73] W. A. P. van Kleunen, N. A. Moseley, P. J. M. Havinga, and N. Meratnia, "Proteus II: Design and evaluation of an integrated power-efficient underwater sensor node," *Int. J. Distrib. Sensor Netw.*, Aug. 2015. [Online]. Available: http://dx.doi.org/10.1155/2015/791046.

[74] W. van Kleunen, N. Moseley, N. Meratnia, and P. J. M. Havinga, "Experiments with aLS-Coop-Loc cooperative combined localization and time-synchronization," in *Proc. WiNMeE*, Hammamet, Tunisia, May 2014.

[75] EvoLogics, "Underwater acoustic modems," accessed: Jun. 2016. [Online]. Available: https://evologics.de/en/products/acoustics/index.html.

[76] R. Otnes, "NILUS—An underwater acoustic sensor network demonstrator system," in *Proc. 10th Int. Mine Warfare Technol. Symp.*, Monterey, CA, USA, May 2012. [Online]. Available: http://www.ffi.no/no/Publikasjoner/Documents/NILUS2012.pdf.

[77] Variscite, "VAR-SOM-OM44 V1.1 datasheet texas instruments OMAP4-based system-on-module," accessed: Jun. 2016. [Online]. Available: http://www.variscite.com/images/stories/DataSheets/VAR-SOM-OM44_v1.1_datasheet_v1.24.pdf.

[78] Texas Instruments, "8-channel high-quality low-power audio codec for portable applications," accessed: Jun. 2016. [Online]. Available: http://www.ti.com/lit/gpn/twl6040.

[79] H. Dol, P. Casari, and T. van der Zwan, "Software-defined open-architecture modems: Historical review and the NILUS approach," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[80] P. van Walree, H. Buen, and R. Otnes, "A performance comparison between DSSS, M-FSK, and frequency-division multiplexing in underwater acoustic channels," in *Proc. UComms*, Sestri Levante, Italy, Sep. 2014.

[81] P. A. van Walree and G. Leus, "Robust underwater telemetry with adaptive turbo multiband equalization," *IEEE J. Ocean. Eng.*, vol. 34, no. 4, pp. 645–655, Oct. 2009.

[82] H. Dol, M. Colin, P. van Walree, and R. Otnes, "How smartphone industry made UComms easier: Moore's law goes underwater," in *Proc. MTS/IEEE OCEANS*, Genova, Italy, May 2015.

[83] R. Otnes and S. Haavik, "Duplicate reduction with adaptive backoff for a flooding-based underwater network protocol," in *Proc. MTS/IEEE OCEANS*, Bergen, Norway, Jun. 2013.

[84] R. Otnes, P. A. van Walree, H. Buen, and H. Song, "Underwater acoustic network simulation with lookup tables from physical-layer replay," *IEEE J. Ocean. Eng.*, vol. 40, no. 4, pp. 822–840, Sep. 2015.

Henry S. Dol (M'15) received the M.Sc. and Ph.D. degrees in applied physics from Delft University of Technology, Delft, The Netherlands, in 1993 and 1998, respectively. His education was in the field of computational fluid dynamics (CFD) of thermally driven turbulent flows.

After a few years in telecommunications research at KPN, Leidschendam, The Netherlands, he continued his career in the CFD field at the National Aerospace Laboratory (NLR), Amsterdam, The Netherlands. In 2006, he switched to underwater acoustics research at The Netherlands Organisation for Applied Scientific Research (TNO), The Hague, The Netherlands, and is working as a Senior Scientist and Project Manager on underwater acoustic communications, and active and passive sonar applications.

Paolo Casari (S'05–M'08–SM'13) received the Ph.D. degree in information engineering from the University of Padova, Padova, Italy, in 2008.

After being on leave at the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2007, his research progressively focused on underwater communications and networks. He collaborated to several funded projects including CLAM (FP7), RACUN (EDA), and several U.S. Army Research Office (ARO), Office of Naval Research (ONR), and National Science Foundation (NSF) initiatives. He was Technical Manager of the NAUTILUS and WISE-WAI projects. In 2015, he joined the IMDEA Networks Institute, Madrid, Spain, where he leads the Ubiquitous Wireless Networking group. His research interests include many aspects of underwater communications, such as channel modeling and simulation, network protocol design and evaluation, localization, detection, and field experiments.

Dr. Casari has been guest editor of a special issue on "Underwater Communications and Networking" published in the *Journal of Electrical and Computer Engineering*.

Timo van der Zwan received the M.Sc. degree in technical informatics from Delft University of Technology, Delft, The Netherlands, in 2005.

His specialization is in information systems engineering. Since 2005, he has been employed at The Netherlands Organisation for Applied Scientific Research (TNO), The Hague, The Netherlands. He is a Software and Systems Engineer there with a focus on the underwater defense domain. This domain includes systems development and deployment for active sonar, passive sonar, and underwater communications.

Roald Otnes (S'00–M'03–SM'13) received the M.Sc. and Ph.D. degrees in telecommunications from the Norwegian University of Science and Technology, Trondheim, Norway, in 1998 and 2003, respectively.

Since 2003, he has been a Scientist at the Norwegian Defence Research Establishment (FFI), Horten, Norway. His current research interests include underwater acoustic communications and sonar signal processing.