WaterCom: Connecting Research Configurations with Practical Deployments

A Multilevel, Multipurpose Underwater Communications Test Platform

Ciarán Mc Goldrick[†], Enrique Segura[‡], Tianyan Wu[‡], Mario Gerla[‡]

[†]School of Computer Science & Statistics, Trinity College Dublin, Dublin, Ireland
[‡]Department of Computer Science, University of California, Los Angeles, California, USA
Ciaran.McGoldrick@scss.tcd.ie, {segcarri123,twu1995629}@ucla.edu, gerla@cs.ucla.edu

ABSTRACT

This paper describes the ongoing evolution of the Water-Com framework developed as part of the NSF funded Ocean-TUNE CRI project. A particular challenge for connecting global researchers and the public with remote, autonomous underwater research infrastructures is that translating and bringing the requesters topology and system properties into physical existence through dynamic reconfigurability of the deployed experimental infrastructure and assets. Previous work described the overall WaterCom framework, system architecture and design. The extension of this system to enable graphical node topology definition and validation, and the procedure for defining, scheduling and deploying experimental configurations on remote hardware is articulated.

1. INTRODUCTION

In recent years significant advances have been made in the underpinning technologies, and systems for wireless underwater communications. In general the focus of many research efforts has been on the transmission hardware[1], and on robust protocols and error handling strategies for encoding, transmitting and decoding of digital data. More recently, researchers have addressed the application of modern software defined radio and networking concepts and principles to the underwater environment[2]. As the complexity of simulated and emulated underwater systems increases, realising these in practical, scalable ways in testbeds and across test infrastructures becomes increasingly challenging.

This paper describes the methodology and approach adopted in permitting remote endusers design and layout their desired experimental topology, define the modem properties and characteristics, validate the proposed configuration(s), and then presents the system and scheme for translating these expressed requirements into a collection of remote modem configurations, schedules and experimental jobs for remote execution.

The scheme is designed and integrated into the Water-Com[4] framework, thereby leveraging the knowledge and expertise already encapsulated in this system. More generally the approach is compatible with the underwater software defined networking overlay described in [2], and can provide a more accessible, intuitive platform for realising and deploying such configurations.

2. WATERCOM FRAMEWORK

The WaterCom[4] experimental framework has been designed to be cloud hosted, extensible and reliable. Experimental nodes can come online at any time, poll the cloud based job scheduler queue, locally execute their experiment set, and periodically reconnect to the cloud service to upload results and execution status information. These operational modalities support permanently connected, occasionally connected, and predominantly disconnected modalities. The actual carrier medium and protocol can range from Delay and Disruption Tolerant network techniques via satellite connections, through to wireless 5G communication links. Practical evaluation identified a significant usability challenge in capturing, translating and mapping an end-users locally simulated/emulated network topology onto actually deployed experimental configurations. This paper documents our efforts in translating and deploying these topologies across remotely deployed experimental installations.

3. REMOTE NODE ACCESS AND CONTROL

To broaden the range of deployed hardware, to support greater geographic diversity of experimental locations, and to facilitate straightforward interconnectivity into the WaterCom framework, a new remote node configuration has been created based on a Raspberry Pi controller. Modems are connected to the Pi, which runs both Sealinx[3] and the WaterCom Local Execution Engine[4] module. The Execution module handles the secure connection to the WaterCom server, the download of the device configuration file, the experiment configuration file(s), and the file(s) for transmission. Connectivity can be by Wi-Fi, 4/5G data or even Satellite links. Experiment execution is synchronised across participating nodes using NTP. New underwater communication devices can be integrated into WaterCom through the simple process of connecting them to a suitable configured Pi device. Existing test facilities, that may not be online, can be integrated and made available through Watercom.

In practice this approach provides great flexibility as it enables the deployment and integration of devices globally - from offshore wave energy installations, through to marine environmental sensing integrations.

4. END USER INTERFACE

The end user interface is used to configure and parameterize the experimental devices and topologies. The GUI supports single hop, multi-hop and mesh topologies. Available devices avatars are positioned by drag and drop. Link connectivities are specified either graphically or parametrically. Planned future functionalities include visual representations of propagation and communication regions for each device type and device geolocation.

4.1 Entities

The entities that are used on the interface comprise devices and links. Devices can be acoustic or optical modems connected to WaterCom via a Remote Pi node. In the acoustic class of devices, we include two variants of the AquaSent OFDM modems. These devices map directly to physical hardware deployed in the UCLA test infrastructure. Additional devices are presented through the interface as they come online and register with the WaterCom server. Offline devices can also register for periodic availability, and can be considered for queued experimental configurations. Links indicate the devices between which connectivity is required. Links can be configured in two different ways - required link characteristics can be specified e.g. approximate throughput or bit rate, or the actual link properties can be specified e.g. modem transmit power level, encoding scheme, etc. The framework handles the validation and creation of the relevant remote device configurations based on the user specified requirements. These are processed by the remote Pi controller and used to configure both the SeaLinx experiments and the attached modems.

4.2 Device Properties

On the web GUI interface, after the users determine the general topology for the network, they can then specify the parameterization for a particular modem. Representative properties for the acoustic modems include maximum and minimum power level, transmission mode, and blocks level, the number of retransmissions, and sender/receiver/router operation mode. The device specific configurations are applied to the appropriate device(s) by the Local Execution Engine on the relevant remote Pi node.

4.3 Supported Configurations

The present system design envisages single hop, multi-hop, and mesh topologies. As different devices have different properties, and support different physical carriers and protocol sets, it is necessary that the end-user requested experimental configurations are validated prior to submission to the local execution framework. Validation must be performed for both individual hops, and for multi-path requirements and constraints.

5. CREATION AND DEPLOYMENT OF THE EXPERIMENTAL CONFIGURATION

The server validates the requested configurations against both requested and available hardware, and against both practical and physical constraints. For instance, device deployment in different water bodies is a limiting factor (geolocation constraint). Moreover, the requirements of a requested network topology e.g hopping configurations and specific device communication capabilities, must be achievable through the devices connected to the remote Pi nodes(s).

5.1 Candidate node sets

The system validates the requested topology against both its current available device set, and against the periodically available device set. For each node to be a viable part of a network, it must be in communication range of another participating node. Nodes must also satisfy the geolocation and maritime geography constraints mentioned above. The Internet Scheduler capabilities described in [4] form a key component in the candidate node selection process. The system seeks to identify the first instance whereby the remote topology can be available and the experiment(s) executed.

Where an existing configuration is suitable, the scheduler will coordinate the addition and actioning of experimental jobs across different Pi attached modems. In doing so the scheduler must ensure the likely availability of all necessary devices at the time the experiment set is submitted for execution. Where no suitable configuration is identified a new configuration set will be created and the job associated with that configuration and availability instance. The scheduler is also responsible for ensuring accurate NTP time sync across the remote online nodes and for a given experiment group.

6. CONCLUSIONS

This paper described recent advances in the the WaterCom system. The system is designed to be straightforward to use, and to mask the complexity of underwater modems and communications from end users. The remote modem connectivity device is introduced and its functional requirements established. Specific design features that allow end-user visual configuration of an experiment set are presented and their implementation discussed. The system functionality required to validate and deploy the requested experiments on the remote modems is elucidated.

7. ACKNOWLEDGEMENTS

8. REFERENCES

[1] K. Domrese et al. Comparison of the ranging function of three types of underwater acoustic modems. In *11th IEEE MASS*, pages 743–748, 2014.

[2] R. Fan, L. Wei, P. Du, C. Mc Goldrick, and M. Gerla. A SDN-Controlled Underwater MAC and Routing Testbed. In *MILCOM'16*. AFCEA, 2016.

[3] S. Le et al. SeaLinx: a multi-instance protocol stack architecture for underwater networking. In *8th ACM WuWNet*. ACM, 2013.

[4] C. Mc Goldrick, M. Matney, E. Segura, Y. Noh, and M. Gerla. WaterCom: A Multilevel, Multipurpose Underwater Communications Test Platform. In *ACM WUWNet'15*. ACM, 2015.