Adapting the Appropriate RTT Timeout of TCP NewReno in Submarine Communication Networks

Hajar. Bennouri[1], Amine. Berqia[1], and N'Guessan Koffi Patrick[2]
[1] Smart Systems Laboratory (SSL) ENSIAS, Mohammed V University in Rabat, Rabat, Morocco
[2] Atos-France, Grenoble, France
Email: hajarbenn@gmail.com; berqia@gmail.com; patrickoffi32@gmail.com

Abstract**—**In the future, aquaculture and aquaponics could play a more important role to meet the increasing demand of a growing world population and meet future demand for food. Underwater Wireless sensor networks (UWSNs) are increasingly used to measure various water parameters in order to monitor the evolution of an aquaculture. Adapting this system network in this environment requires several studies for reliable communication. This work is the continuation of many improving studies of the performance of Transmission Control Protocol (TCP) in marine environment. It aims at adapting the NewReno TCP in an UWSN by finding the appropriate parameters. Moreover, in this paper we deal with another parameter in order to evaluate its effect on the performance of this TCP in UWSNs. We propose a performance improvement of the NewReno TCP by fitting the Round Time Trip (RTT) value in accordance with the UWSN characteristics. This new adaptation is simulated on the environment of the Aqua-sim simulator of Ns2 tool. The results show that with the new parameter setting of NewReno TCP, performances in terms of packet transmission gain and packet delivery retransmission rates give better results compared to those obtained while using the original NewReno TCP protocol in a UWSN.

Index Terms**—**TCP, TCP NewReno, RTT, UWSNs, ns2, Aqua-sim

I. Introduction and Motivation

Scientists have complete information about what is happening on the moon, but they do not know what is happening in the ocean. In recent years, the scientific community has placed more emphasis to explore the seabed. To do this, several sophisticated submarine robots equipped with multiple sensors and hydrophones are set up to discover the mystery of this environment [1]. In general, this type of exploration technology is usually composed of a large number of sensors equipped with an acoustic communication system designed to explore the seabed. Fig. 1 depicts an example of architecture and components of an Underwater Wireless Sensor Network (UWSN) equipped with autonomous underwater vehicles.

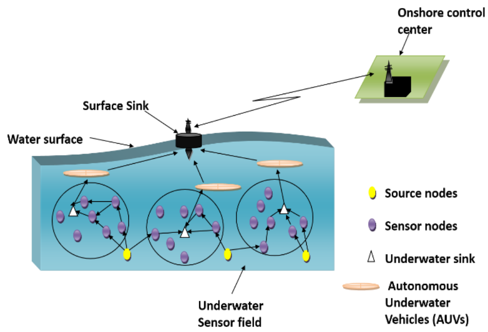

Fig. 1. Example of UWSNs Architectures with different components

The use of radio wave communication in terrestrial environment is largely preferred due to its faster propagation speed compared to the acoustic communication [2]. On the other hand, the use of electromagnetic waves in the submarine domain is not preferred because water and especially salt water is characterized by its dissipative nature related to its high conductivity, thus causing an extreme attenuation to these electromagnetic waves and makes them rapidly inoperative [3], [4].

The only vector for carrying information under the sea with fairly easily exploitable characteristics is acoustic waves, which are mechanical vibrations of the propagation medium. They propagate in seawater very favorably, these waves will compensate to a certain extent, the virtual absence of electromagnetic and light waves in aquatic environment.

In the water, things are somewhat different. Acoustic waves have better transmission characteristics than in the air: they have a higher propagation speed, can reach very high levels, and above all they undergo less attenuation and can therefore spread distances. However this was counterbalanced by other constraints as in particular the useful signals are disturbed by a lot of ambient noise and false echoes [1]. The underwater sound environment is therefore very different from the terrestrial, it must be addressed in a more sophisticated way.

Several fields of application have been initiated for environmental purposes to observe the environmental conditions and disaster management such as tsunamis and underwater monitoring. So, they are used in assisted navigation at sea exploration as well as in ocean sampling [5]. Aquaculture is also one of the applications that uses underwater wireless sensor network, it takes place at sea

Manuscript received February 7, 2019; revised November 2, 2019.
Corresponding author email: hajarbenn@gmail.com
doi:10.12720/jcm.14.12.1191-1197

or in natural water bodies to measure various water parameters. Innovations in this area include air and waterborne sensors and marine drones, which inspect equipment and anchorages, monitor the environment and fish, and help optimize aquaculture activities. In the fishing sector, navigation aids such as GPS make it possible to report fishing zones, record fishing trips and plan low-energy expeditions. The collection of information from underwater sensors used to locate fish, seabed and underwater debris, with exit reports, which provides new data sets and thus improves system efficiency and optimizes the production [6].

According to the 2017 United Nations for Food and Agriculture Organization (FAO) Conference in Rome [7], it was recommended to develop a policy and field programs to encourage countries to invest in the development of fisheries value chains and aquaculture that are nutrition-focused. In this context, Morocco with his two large coastal areas namely the Mediterranean Sea and the Atlantic Ocean can benefit and establish large aquaculture for more profitable production.

The hostile nature of this environment especially the submarine wireless channel makes submarine sensor connection a difficult task [8]. Therefore, establishing such a network imposes very deep studies on the communication of the network and the transmission of data in this underwater environment is impacted by various aspects such as bandwidth usage limitation, surrounding noise and large acoustic propagation delays. However, communication itself is an outstanding challenge. Being able to connect these subsea devices and create a reliable communication channel between them are the keys to make all these applications viable, including unmanned vehicles and robots in harsh environments for humans [9].

The well-known traditional TCP (Transmission Control Protocol), one of the most used transport protocol on the internet, is not suitable to enable this technology. Even though TCP variants for the wireless network are not foolproof in an underwater environment, their use could probably be more difficult in such a multi-hop communication system. We have chosen NewReno for our study. This variant is a modern implementation that includes the four congestion control algorithms. These algorithms have proved to be effective when it comes to terrestrial networks which could be a basis for our study.

In addition, NewReno is known for its algorithm of recovery of several segments lost within the same sending window.

In our case, in the underwater environment, the transport layer is totally unexplored area, unlike the physical, network and data link layers that have been significantly enhanced [10]. In this kind of environment the acoustic communication is more reliable than using RF communication. Improving TCP in UWSNs may seem a big challenge to get out with a reliable and adapted data transmission protocol adapted to acoustic channel in this environment.

Detecting lost packets in a data stream was one of the problems known to TCP Reno [11]. To solve this problem an improved version gave birth to NewReno, a more efficient version to detect lost packets.

In this paper, we focus on studying the behavior of TCP NewReno in a submarine network. We propose an improvement to enhance its performances according to the UWSN characteristics by modifying the parameters values mainly adjusting the RTT timeout. In other words, we propose an appropriate parameterization of the TCP variant NewReno which is used especially for the UWSNs.

The rest of this paper is presented as follows: The upcoming section provides some related work on the literature improvement of TCPs in WSNs and UWSNs. In Section 3 we found an overview of the proposed work. Section 4 explains the parameter setup of the simulations. Section 5 presents the experimental results analysis. Finally, a general conclusion with the future perspective is presented in the last section.

II. Related Work

This section provides some improvements of TCP performance in UWSN and WSN as well. This may constitute a basis for our work.

The adaptation of TCP has been studied in different cases of Ad-hoc networks. In MANETs, congestion is not the only cause of packet loss. To determine the cause of the loss of the data, a behavioral study is done by [12], where the failure of the connection was the cause of data loss. In [12] they proposed a new TCP type named TCP-WELCOME, this variant is adapted to this type of MANET networks. This new TCP is based on two mechanisms; it begins by analyzing the lost data and the delay to specify the cause of the loss. Then, it recovers the appropriate lost packets based on the identified loss cause.

To resolve the problem of congestion, an improved variant of TCP is proposed by [13] called TCP NewBR. They select the more accurate ACK time interval based on the bottleneck links to adaptively estimate the available bandwidth. NewBR TCP uses modified algorithms for faster retrieval depending on the length of the bottleneck link queue.

Another variant was the subject of study in wireless networks. Authors in [14] have studied the SmoothTCP variant comparing it with the standard TCP in simulated media with and without parasitic errors. The experimental results show that in both cases, SmoothTCP performs better with respect to the variation of round-trip time.

Authors propose in [15] a method which limits the maximum window size. As a result, this leads to increase the throughput. In this multi-hop architecture, the author used fixed nodes numbered from 0 (the transmitter) to N (the receiver) with a range of 70 m between nodes and a link bandwidth set at 150 kbps. They come out with two criteria: the effect of the number of hops and the effect of the maximum sending window limitation over Hybla.

And then, he has compared the results between improved Hybla and those related to NewReno. It has been shown that: First, although Hybla presented better throughput than NewReno when the number of hops is between 2 and 5, it still requires an adequate setting of RTT0 to reduce the buffer overflow. But this represents a challenging issue because the RTT is not stable in UWSN environment. Second, Hybla did not experience any significant improvement when applying the maximum sending window limitation. Contrariwise, NewReno throughput is improved in all the packet size cases.

In [16], authors provide a model in order to ensure reliable data deliveries from sensor nodes to surface sink. For this purpose, they propose an algorithm which determines the suitable data packet size for efficient data transfer. The solution proposed by authors is based on the cross-layer approach which extends the HbH-ACK (Hop-by-Hop ACKnowledgement) scheme to 2H-ACK scheme. This new scheme proposes that three nodes be involved in the transmission of one segment with 2 nodes retaining the copy of the relevant segment so that in the case of an unwanted event such as destruction of the node, another copy is always available on the network. This method has shown advances, among other things, the improvement of packets lost from 92 (HbH-ACK case) to 24 (2H-ACK case) or +72.3% gain.

A Collaborative Transport Control Protocol (CTCP) was proposed in [17], which presents a method to recover the packets discarded by a buffer overflow in the multi-hop data transmission. It is based on two levels of reliability. Level 1 where only two nodes are involved in the forwarding of the packet like HbH-ACK scheme; and level 2 resembling the 2H-ACK scheme discussed in [17] but with the difference that the transmitter erases the segment from its buffer after receiving a double ACK guaranteeing increased reliability detailed in section 4B. This solution shows performance improvement with the average delivery rate of 29.82%, 61.11% and 98.15% respectively without transport protocol, with CTCP level-1 and CTCP level-2 situations.

Another study was done in [18], where authors present an initial study on the behavior of the TCP New Reno to prove that the control of the Maximum Window can improve the performance of TCP NewReno in the underwater environment, simulations with different scenarios are done to validate their approach.

In [19] M. Albuquerque, J.H. Kim and S. Roy were interested to study the effect of TCP Packet size to the performance of a single TCP-Reno [11] connection over a lossy, congested link as a function of TCP packet size on the forward link and the ratio of forward-to-reverse link capacity measured in terms of packets/set, they support their study by performing a simulation results using ns-2 Network Simulator.

III. Proposed Work

Our study used the underwater architecture described in Fig. 1. This architecture is based on 100 static nodes emerged and fixed at the bottom of the water where we vary the number of TCP NewReno transmitters, in the middle of this cluster we have an underwater sink for the collection of packets sent.

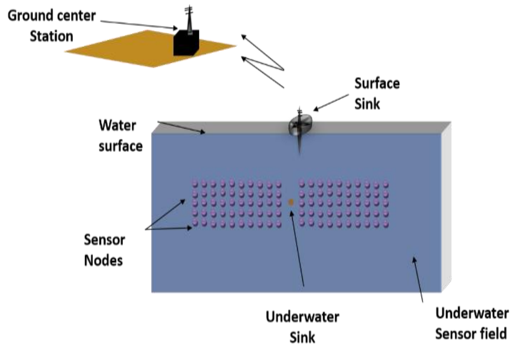

Fig. 2. Network architecture [18]

The characteristics of the channel in the UWSN differ particularly in terms of propagation time that varies in a manner that affects the RTT timeout. As a result, the estimation of the RTT waiting time becomes a necessity to improve the management of the flow control.

A rough calculation of the value of RTT timeout is done by adjusting the "rtxcur_init" parameter which initializes the "t_rtxcur" parameter. This value varies between 1 and 20 with a step size of 1 by taking into account the directions proposed in [20] which includes the introduction of the value of the initial window (IW) and the maximum size of the sender segment (SMSS) as presented in equation (1).

$$IW = \begin{cases} 4*SMSS, & SMSS <= 1095, & (MAX\ segments=4) \\ 3*SMSS, & 1095 < SMSS <= 2190, & (MAX\ segments=3) \\ 2*SMSS, & SMSS > 2190, & (MAX\ segments=2) \end{cases} \quad (1)$$

For the value of the maximum window, it was guided by the result found in our contribution [18] where several studies and simulations have allowed us to find the adapted values taking into account the nature of the underwater environment.

To adjust these changes, we implemented the Pseudo Code 1 to modify the source code in Aqua-sim [21] tool on ns2 simulator network [22] coding on C++ language.

To evaluate the adapting value of these parameters we used two performance metrics which are the Packet Delivery and the Packet Retransmission rate.

- Packet Delivery: It indicates the number of packets received in each communication or simulation. This metric defines the good functioning of the studied TCP. When the number of received packets increases one can say that the performances of the protocol TCP are good [18].
- Retransmission rate: It represents the average time of each packet has been received. For this metric the smaller its value, the better the performance of the TCP used [18].

IV. Simulation Setup

The study was conducted by running simulations using tools called Aqua-sim which is based on the NS2-

30 network simulator [22]. The effectiveness of Aqua-sim to simulate acoustic signal attenuation, packet collisions in underwater sensor networks and support three-dimensional deployment are discussed in [21], Fig. 3 depicts the class diagram of Aqua-Sim.

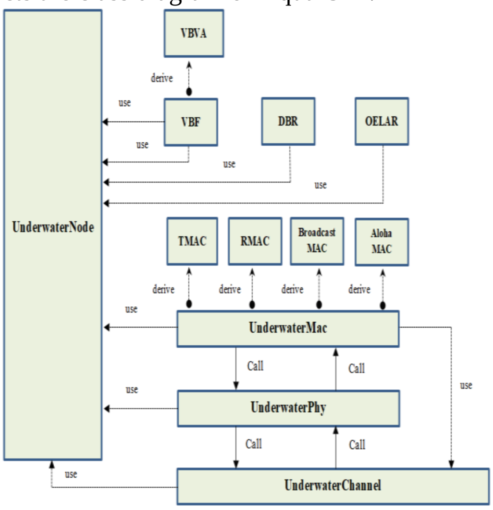

Fig. 3. Class diagram of Aqua-sim [21]

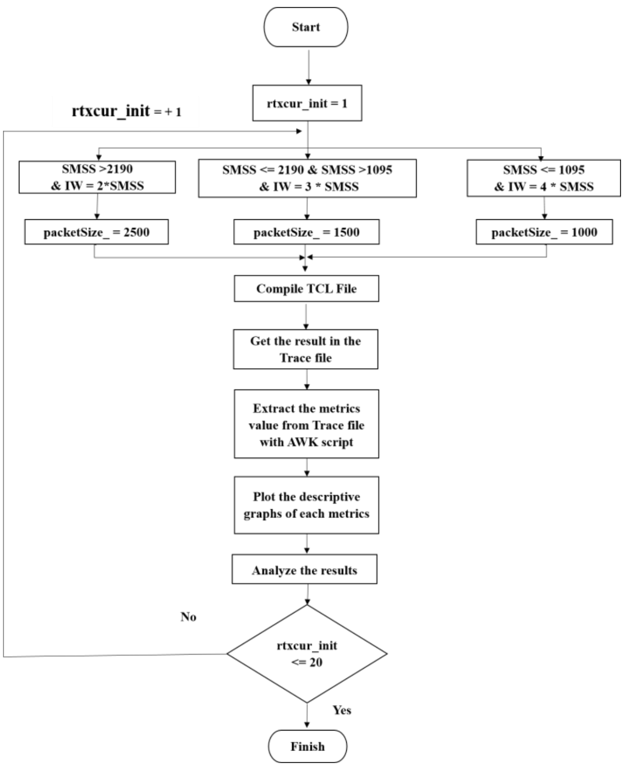

Fig. 4. Flow chart of the implementation

The simulations are designed to compare the performance of the behavior of the standard version of TCP NewReno with the adapted version after the change of the values of the study parameters in this work. We then, adapt changes by performing experiments in multiple scenarios with different parameter values to measure the performance of the new version of TCP NewReno in UWSNs.

The scenarios studied in this research aim to test the behavior of TCP NewReno after adapting the estimate of RTT in different environments. We chose to vary the number of TCP densities in the network in each scenario. The first scenario had 12 NewReno TCP transmitters and the second scenario had 25 NewReno TCP sources. In each scenario we used the implementation as explains the Flow chart below in Fig. 4.

PSEUDO CODE 1: ADJUSTING TCP NEWRENO PARAMETERS

```
for < i=1; 20 > do
rtxcur_init ← i ;
        if SMSS > 2190 then
                IW← 2* SMSS ;
        else if  SMSS > 1095 & SMSS <= 2190
then
                IW← 3* SMSS ;
        else if SMSS <= 1095 then
                IW ← 4* SMSS ;
        end if
end for
```

TABLE I: SIMULATION PARAMETRES

Parameters	Value
Channel	UnderwaterChannel
Propagation	UnderwaterPropagation
PHY	UnderwaterPhy
Antenna	OmniAntenna
Distance	75m, 84m
Frequency	25khz
3MAC protocol	BroadcastMac
Mac_bit_rate	10kbps
Delay	25 us
Routing protocols	DSDV
TCP agent	NewReno
Simulation Time	500s

TABLE II: VALUE OF NEWRENO TCP PARAMETERS

Parameters	Value
Window_	1 2 5 12 16
SMSS	1000 1500 2500
IW	2*SMSS 3*SMSS 4*SMSS
Number of TCP NewReno sources	12 25

The simulation parameters of this work is summarized in Table I. It describes the general simulation parameters on underwater environment, which consist of some parameters and values such as: the type of the used Channel, the type of the Propagation and kind of Antenna and so on. Based on previous work [23], we choose to use DSDV [24] as Routing protocol.

Regarding the values used for New Reno TCP parameters, Table II gives us an idea about the values that were used in this research.

V. Results & Discussion

The results of the first scenario concern a network that contains 12 sources of TCP NewReno. To estimate the delay time RTT we will start by setting the parameter "rtxcur_init" from 0 to 20 which initializes the parameter “t_rtxcur" and we vary the packet size between 1000, 1500 and 2500 bytes.

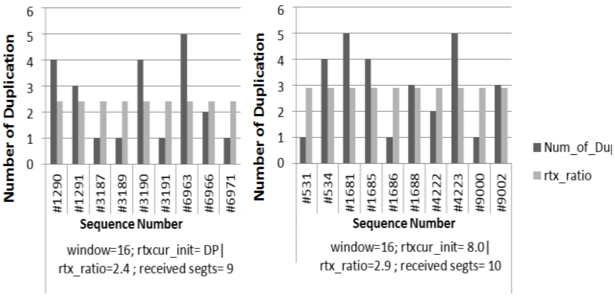

Fig. 5. Performances before & after with SMSS= 1000 bytes

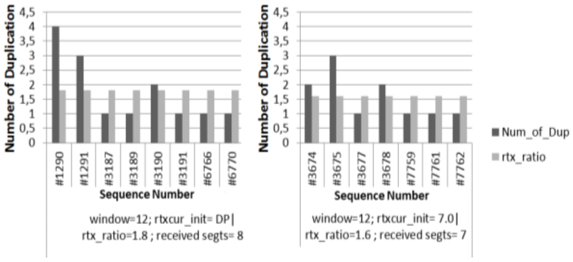

Fig. 6. Performances before & after with SMSS= 1500 bytes

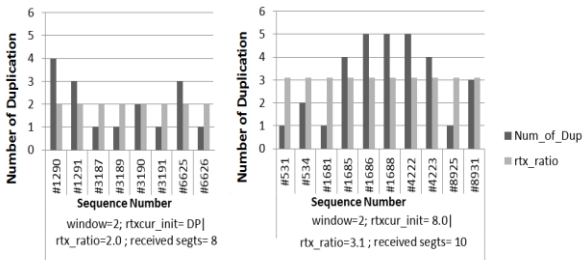

Fig. 7. Performances before & after with SMSS= 2500 bytes

Fig. 5 and Fig. 7 describe the results before and after adapting the appropriate "rtxcur_init" value of TCP NewReno parameters. As we can see for the Packet Delivery, those two cases show a relative increase in gain +11.1% and +25.0% respectively.

On the other hand, the case while using a Packet Size of 1500 bytes the results depicted in Fig. 6 shows a decrease of -12.5%, which may seem to be a disadvantage because the possibility of improving the number of delivered packets (up to 9) is possible for “rtxcur_init” values at 8. However, the disadvantage is the retransmission rate which proves to be high.

Moreover, taking into consideration the ‘rtxcur_init’ parameter it presents divergent results. When the 1500 bytes case shows an improvement with a change from 1.8 to 1.6 or -11.1% gain in Retransmission Fig. 6, the other two cases show improvements of +55% and +20.8% respectively illustrated by Fig. 5 and Fig. 7.

These situations show the difficulty of correctly estimating the value of ‘rtxcur_init’ in this configuration.

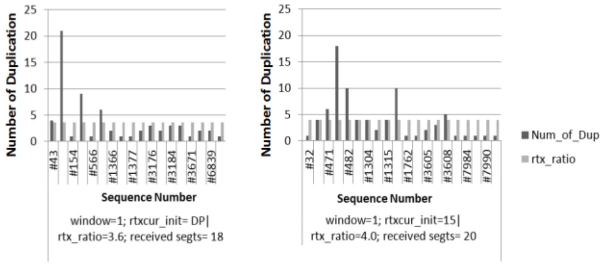

Fig. 8. Performances before & after with SMSS= 1000 bytes

The second Scenario uses 25 nodes sources of TCP NewReno. The improvement of the packet delivery is observed for the appropriate value of “rtxcur_init” as well in this second scenario. We note an evolution from 18 to 20 which is a gain of +11.1% as shown in Fig. 8, from 17 to 23 which is a gain of +35.3% as described in Fig. 9 and from 17 to 27 which represents a gain of +58.8% as depicted in Fig. 10.

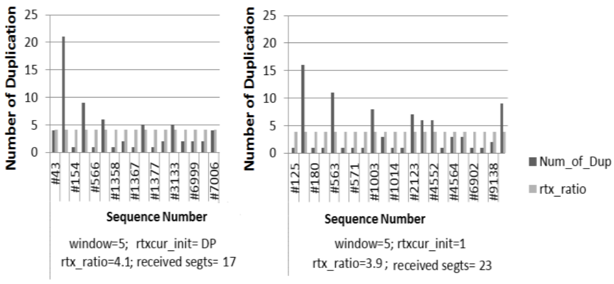

Fig. 9. Performances before & after with SMSS= 1500 bytes

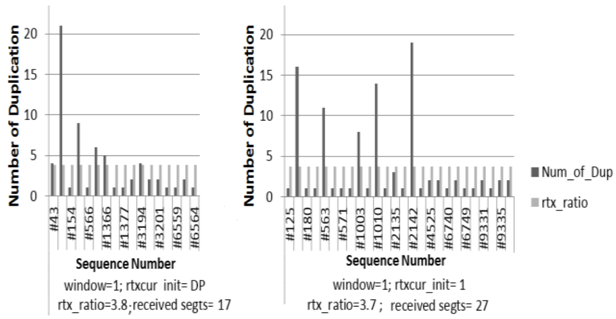

Fig. 10. Performances before & after with SMSS= 2500 bytes

The value of “rtxcur_init” is uniformly 1sec for the last two cases (cf. Fig. 9 and Fig. 10), while for the 1000 bytes case this value is 15 sec. However, in the 1000 bytes case, the best performance with a small value of this parameter is reached at 2 sec with a number of received segments is remaining at 18 but the reverse is the retransmission ratio which increases considerably.

The principle of reducing the retransmission rate is achieved in all cases when it is applied, adjustment the value of "rtxcur_init" as follows: -2.6% in Fig.10 and -4.9% in Fig. 9. However, an exceptional increasing by +10% is observed for the 1000 bytes case in Fig. 8.

VI. Conclusions

The Underwater environment is a network that is characterized by its unique conditions, which affect its performance in general. The estimation of retransmissions in this system is essential since it imposes a load on the energy consumption on the sensors.

In this paper we were able to find an appropriate value of the 'rtxcur_init' used for initializing the RTT timeout to improve the performances of TCP NewReno in an UWSN.

The simulations results, after adjusting the value of 'rtxcur_init' and then finding an appropriate RTT timeout, show that with this adaptation, TCP NewReno offers better performance in terms of packet received and retransmission packets compared to the results found while using the standard parameters of TCP NewReno in UWSNs. As perspective of our work, we will implement our new variant of TCP NewReno in a real UWSN environment.

References

[1] X. Lurton, Acoustique Sous-marine: Présentation et Applications, Editions Quae, 1998.

[2] I. Nor-Syahidatul, A. H. Liban, and S. H. S. Ariffin, "Analyzing the performance of acoustic channel in underwater wireless sensor network (UWSN)," in *Proc. Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*, 2010.

[3] M. Stojanovic, *Acoustic (Underwater) Communications*, Wiley Encyclopedia of Telecommunications, 2003.

[4] I. F. Akyildiz, D. Pompili, and T. Melodia, "Challenges for efficient communication in underwater acoustic sensor networks," *ACM SIGBED Review*, 2004.

[5] N. Saeed, T. Y. Al-Naffouri, and M. S. Alouini, "Outlier detection and optimal anchor placement for 3d underwater optical wireless sensor networks localization," IEEE *Transactions on Communications*, p. 1, 2018.

[6] FAO. 2018, La situation mondiale des pêches et de l'aquaculture 2018, Atteindre les objectifs de développement durable, Rome. Licence: CC BY-NC-SA 3.0 IGO

[7] FAO. 2017. La situation mondiale des pêches et de l'aquaculture 2017, Mettre les systèmes alimentaires au service d'une transformation rurale inclusive, Rome. ISBN 978-92-5-209873-7

[8] A. Celik, N. Saeed, T. Y. Al-Naffouri, and M. S. Alouini, "Modeling and performance analysis of multihop underwater optical wireless sensor networks," in *Proc. IEEE Wireless Commun. and Netw. Conf.*, Apr. 2018, pp. 1–6.

[9] H. Bennouri, A. Yazidi, and A. Berqia, "A pursuit learning solution to underwater communications with limited mobility agents," in *Proc. Conference on Research in Adaptive and Convergent Systems*, Honolulu, Hawaii, October 2018, pp. 112-117.

[10] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Networks*, vol. 3, no. 3, pp. 257-279, May 2005.

[11] V. Jacobson, "Modified TCP Congestion Avoidance Algorithm," Technical Report, 30 Apr. 1990.

[12] G. Seddik, A. Yacine, G. Doudane, and S. M. Senouci. "Coupling loss and delay differentiation to enhance TCP performance within wireless multi-hop ad-hoc networks," *Journal of Communications*, vol. 7, no. 12, pp. 859-872, 2012.

[13] W. U. Rui-Qing, J. Hong, and D. Nan, "An improved TCP congestion control algorithm of based on bandwidth estimation in heterogeneous networks," *Journal of Communications*, vol. 9, no. 10, pp. 792-797, 2014.

[14] V. Elvis and M. Bauer, "Round-trip time variation in smooth TCP in the face of spurious errors," *Journal of Communications*, vol. 35, no. 2, pp. 48-56, 2006.

[15] N. Likubo, M. Nakatsuka, J. Katto, and H. Kondo, "Improving TCP performance over underwater sensor networks," in *Proc. International Workshop on Under Water Networks*, 2008.

[16] M. Ayaz, L. T. Jung, A. Abdullah, and I. Ahmad, "Reliable data deliveries using packet optimization in multi-hop underwater sensor networks," *Journal of King Saud University-Computer and Information Sciences*, vol. 24, no. 1, pp. 41–48, 2012.

[17] E. Giancoli, F. Jabour, and A. Pedroza, "Ctcp: Reliable transport control protocol for sensor networks," in *Proc. International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, 2008, pp. 493–498.

[18] H. Bennouri, A. Berqia, and N. K. Patrick, "Controlling maximum window of TCP new reno in underwater wireless sensor networks," in *Proc. International Symposium on Advanced Electrical and Communication Technologies*, 2018, pp. 1-6.

[19] M. Albuquerque, J. H. Kim, and S. Roy, "Effect of packet size on TCP-Reno performance over lossy, congested links," in *Proc. MILCOM Proceedings Communications for Network-Centric Operations: Creating the Information Force*, vol. 1, pp. 705-710, 2001.

[20] M. Allman, V. Paxson, and E. Blanton, "TCP congestion control," RFC 5681, September 2009.

[21] P. Xie, Z. Zhou, Z. Peng, H. Yan, T. Hu, J. H. Cui, Z. Shi, Y. Fei, and S. Zhou, "Aqua-Sim: An NS-2 based simulator for underwater sensor networks," in *Proc. OCEANS 2009, MTS/IEEE Biloxi - Marine Technology for Our Future: Global and Local Challenges*, March 2010.

[22] S. McCanne and S. Floyd, "ns-LBL Network Simulator", 1997. Obtain [Online]. Available: http://www-nrg.ee.lbnl.gov/ns/

[23] H. Bennouri and A. Berqia, "The impact of TCP packet size and number of TCP connections in underwater wireless sensor networks," in *Proc. International Conference on Advances in Computing and Communication Engineering (ICACCE)*, Paris, France, 2018, pp. 121-126.

[24] G. He, "Destination-sequenced distance vector (DSDV) protocol," Networking Laboratory, Helsinki University of Technology, 2002, pp. 1-9.

Hajar H. Bennouri was born in Rabat, Morocco, in 1989. She received the B.S. degree on Physics -electronics and the M.S. degree on Security Networks & Embedded Systems from the Faculty of Sciences at Mohamed 5 University of Rabat Morocco. She is currently pursuing the Ph.D. degree with Smart Systems Laboratory (SSL), ENSIAS Rabat IT Center at Mohamed 5 University of Rabat, Morocco. Her research interests on improving Routing protocols and performance analysis of wireless communication systems especially on Underwater Wireless Sensor Networks.

Prof. Amine A. Berqia, is full professor in University Mohammed V in Rabat, ENSIAS Engineering School, Morocco. He earned the B.Sc. and the M.Sc. in Applied Mathematics from University Mohammed V, Morocco. He holds a Ph.D. degree in Computer Sciences from Dijon University, France. He was Assistant Professor at University of Geneva and Coordinator of the Swiss Virtual Campus Project VITELS from 2000 to 2003. He is member of IEEE. He has served as chair or co-chair in several international conferences (ISDA'2015, iCEER 2013, NGNS´12, IEEE EDUCON 2012, NGNS'10, EATIS2007, NOTERE2007...). He is taking part of several international conference program committees. He published more than 50 papers in journals and conferences in the areas of Networks performance, New Generation of learning, Online Laboratories. In 2012, Prof. Amine Berqia was awarded by IEEE Education Society in recognition and appreciation for his valued services and contributions in Educon 2012. In 2013, Prof. Amine Berqia was awarded by iNEER Society in recognition and appreciation for his valued services and contributions in iCEER 2013.

Patrick P. N'guessan Koffi received the M.S. degree on Engineering Informatics. He is currently Engineer at Atos Grenoble, France.