Wireless teleoperation with visual feedback for ArduSub based underwater ROVs

Diego Centelles[1], Antonio Soriano[2], Raul Marin [1], Jose V. Martí [1], José M. Claver[2], Pedro Sanz [1]

***Abstract*— An upgrade of the BlueROV architechture is introduced in this work, so that it enables wireless teleoperations using a channel with extremelly narrow bandwith. A cross layer protocol, appart from the TCP/IP stack is proposed. A progressive image compression algorithm, named DEBT (Depth Embedded Block Tree), is integrated within the teleoperation system. Thus, providing visual feedback. UWSim simulator is used as a human-robot interface (HRI) virtual reality extension.**

I. Introduction

Robots are more and more present in underwater activities related with archeology, marine research or maintainance operations in oil and gas industries, since they contribute to reduce the cost and risk associated whith these activities. The increasing complexity of these tasks sometimes require the cooperation of serveral robots. The communication based on umbilical cables condition operational capabilities, specially in cooperative activities with several robots. Thefore, the interest in underwater wireless communications has rised in recent years.

Acoustic communication is the most popular in underwater scenarios. It allows the communication of devices separated several kilometers. But they are affected by multipath effect in the vicinity of solid objects like other robots. Visual light communication (VLC) [2] can be considered as an alternative in situations where acoustic links are not feasible. VLC has a larger bandwith than acoustic links, but its range is limited to several meters due to the strong attenuation of light in water. Another alternative to acoustic links is based on radiofrequency (RF) modems [1], which is not affected by the turbidity of water like VLC.

In this work is presented a communication architecture to allow wireless teleoperation with visual feedback in underwater scenarios. The proposed architecture has been optimized for the small bandwith provided by RF modems. This work has been conducted within the framework of the MERBOTS project, which has been funded by the Spanish Government. MERBOTS is aimed at the development of tools and techniques that permit the cooperation of several robots to accomplish tasks in underwater scenarios. MERBOTS will demonstrate the usage of these tools in the recovery of archeological objects.

This work was supported by the Spanish Ministry of Economy and Competitiveness under grants DPI2014-57746-C3 (MERBOTS Project), DPI2017-86372-C3-1-R (TWINBOTS) and BES-2015-073112, by Universitat Jaume I under grant P1-1B2015-68 (MASUMIA), by Generalitat Valenciana under grant PROMETEO/2016/066.

[1]Universitat Jaume I, Interactive and Robotic Systems Laboratory (IRSLAB), 12071 Castelló (Spain) `centelld | rmarin | vmarti | sanzp@uji.es`

[2]Departament d'Informàtica, Universitat de València. Av de la Universitat, 46100 Burjassot (Spain) `antonio.soriano-asensi | jose.claver@uv.es`

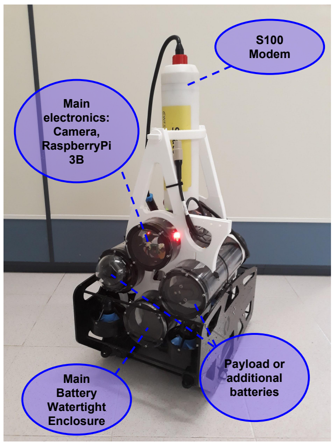


Fig. 1: Experimental platform with the integration of the S100 modem with the BlueROV v1.

II. Experimental Setup

The platform used to evaluate the performance of the proposed architecture is shown in Fig. 1. It is based on a BlueROV (version 1). Wireless communication is based on a pair of RF modems S100 manufactured by Wireless for Subsea (WFS). UWSim [4] has been used as a virtual reality (VR) interface. UWSim has proven to be an excellent tool for the simulation of part of the system in order to evaluate the performance of the proposed architecture in a controlled environment.

The BlueROV is based on a Pixhawk board [3] where the ArduSub software is executed. The comunication between operator and ROV uses an umbilical wire. The BlueROV standard architecture is based on the MAVLink protocol. MAVLink messages are encapsulated into datagrams that are sent through the umbilical wire to the ROV. A software application called MAVProxy extracts the MAVLink messages from the UDP datagrams received by the ROV, and sends the MAVLink messages to the Pixhawk via a serial interface. There is a robot operating system (ROS) named MAVROS that permits the publication of MAVLink messages as topics and services of ROS. The MAVROS node also implements MAVProxy functionalities so that permits the control of the ROV. The wireless communication is implemented with a pair of S100 modems from WFS. MAVLink messages are read from the ArduSub and sent to S100 modems via serial inreface for their transmission.

The MAVROS node facilitates the development of a VR interface based on UWSim which permit the ROV control and the visualization of the image feedback and odometry received in ROV's messages. A specific UWSim module (https://github.com/dcentelles/underwater_simulation) has been also developed in order to test the communication protocol in different transmission scenarios. The model is based on the library network simulator version 3 (NS3) and allows to model the effect of packet loss due to attenuation, packet collisions or discarded packets when the transmission buffer is full. The integration of the simulator in UWSim allows hardware in the loop tests of the ROV and modems separatelly and also the test of the whole system.

III. ARCHITECHTURE FOR UNDERWATER COMMUNICATIONS

The small bandwidth of S100 modems impedes the transmission of MAVLink messages through the RF channel. An ad-hoc communication architecture is therefore required in order to enable wireless teleoperation. It is based on a cross-layer protocol which assumes the functionalities of the link, network and transport layers.

The access to the medium is ensured by means of a token passing strategy. Operator and ROV sides act as master and slave, respectively. The ROV sends a message indicating its odometry only in response to a command message from the operator. The operator sends command messages to the ROV periodically, being the time period between consecutive messages the interpacket gap (IPG). The IPG is large enough so that a complete command from operator to ROV and the answer from the ROV can be successfully transmitted. The point to point communication of S100 modems avoids the requirement of a network layer to transport the messages from operator to ROV and viceversa. A single frame suffices to send the odometry of the ROV, but several frames are required to transmit the visual feedback from ROV to operator. Two specific flags in ROV's response identify the first and the last frames of any image. The protocol itself ensures the delivery of packages in order, so that the information received in several packets can be gathered together to compose the images.

The structure of messages exchanged between ROV and operator is shown in Fig. 2a, where the first byte identifies the start of a new frame. The frame length in bytes is indicated in the second byte. The variable size payload field contains the information transmitted. The 16 bit cyclic redundancy code (CRC) appended at the end of each frame permits to detect any incorrect reception. The payload structure of the messages sent by the ROV are depicted in Fig. 2c. The two inital bits indicate if the message contains the start or the end of an image. The ROV's odometry is codified in bytes 2 to 13. The image trunk field of the message has variable size, and contains information of the images sent from ROV to operator.

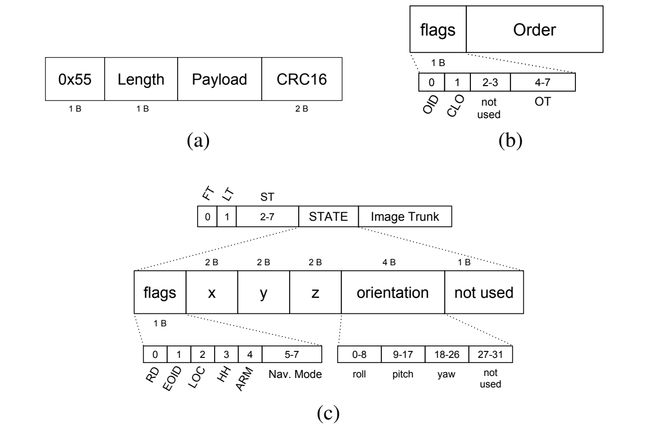


Fig. 2: Format of the protocol frames. (a) Detail of the fields contained in protocol's frames. (b) Structure of the payload of frame sent by the operator. (c) Structure of the payload of the message sent by the ROV.

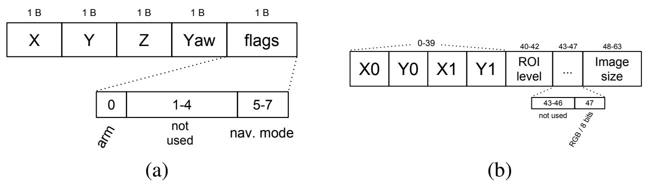


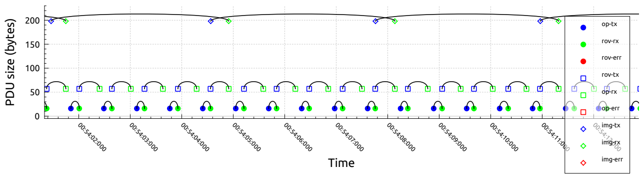
Fig. 3: Format of the orders sent by operator. (a) Positioning command. (b) Image compression command.

The messages sent from operator to ROV are organized as indicated in Fig. 2b. The command sent to the ROV is codified in bits 4th to 7th of the firs byte. The order field contains the information related with the command sent. Its structure is shown in Fig. 3. ROV motion can be controlled indicating its velocity as indicated in Fig. 3a. If the ROV has an autonomous positioning system its movement can also be controlled indicating the desired final position. The payload of this order is the same as in Fig. 3a, but the flags byte is not used.

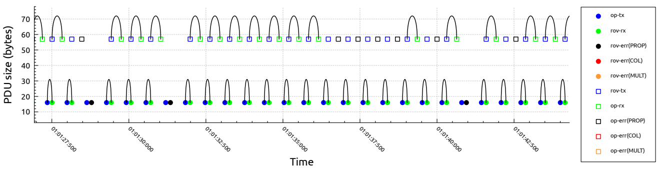
Depth embedded block tree (DEBT) algorithm [5] is integrated into the proposed protocol in order to allow visual feedback. DEBT algorithm permits the selection of a region of interest (ROI) so that image quality is preserved as much as possible in this ROI. The structure of the order sent to configure DEBT algorithm is depicted in Fig. 3b. This message contains information regarding X and Y limits of the ROI, and the level of compression that will be considered by DEBT.

IV. RESULTS

In Fig. 4a and Fig. 4b, a time lapse of the packet sending and receiving events during a teleoperation are shown. Fig. 4a shows the results of a HIL (Hardware In The Loop) experiment with the S100 modems sunk in water and robot motion simulated with the ArduPilot simulator. In this figure, the capture and reception events of an image are also



(a)



(b)

Fig. 4: Time lapse of the S100 communication. Operator to ROV packages have a PDU size of 20 bytes, ROV to operator correspond to 60 bytes packages, and larger PDUs correspond to full image transmission. (a) Time measurements with the S100 modems. (b) Simmulated communication considering a bit error rate (BER) of $0{,}0007$ %.

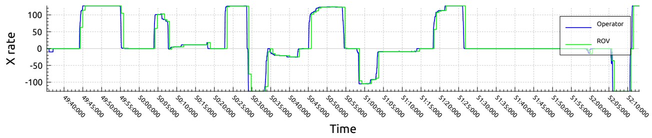
shown. The same teleoperation was replicated in Fig. 4b, but modeling the communications in order to account for transmission errors. An error in any of the packets of an image corrupts its reception. Thus, the correct transmission of images becomes more and more difficult as BER increases. In order to improve image reception, a high level command could be implemented to allow the retransmission of lost packets. Fig. 5 shows the status of the speed controls on the operator and the robot sides during teleoperation. As can be seen, there is only a slight signal lag, caused by the delay, which does not hamper teleoperation in real time.

V. Conclusions

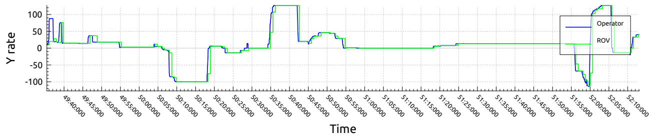
A protocol specific for underwater RF communications implemented within the framework of the MERBOTS project has been presented in this work. Taking into account the limitations of the channels in underwater scenarios, an ad-hoc cross layer network protocol outside the TCP/IP stack is required.

The results achieved with such a protocol demonstrate the feasibility of wireless teleoperation of an underwater ROV. The proposed protocol is capable of provide the operator with visual feedback of the ROV. But the low image rate recommends the usage of either a VR module or a supervised control based on high level commands. The presented system allows both teleoperation in real time and a supervised control of the robot visualizing the result of the commands before being sent. In addition, it allows visual feedback with a region of interest at a constant rate thanks to the integration of the progressive compression algorithm DEBT.

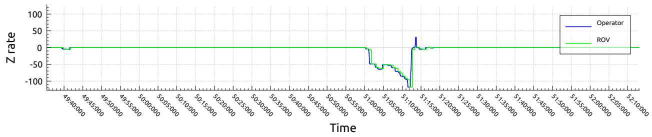
Further work will be devoted to improve the visual feedback. The ability to retransmit los packages will increase the number of images correctly transmitted. The incorporation of a tracking system will facilitate the update of the ROI as the ROV moves. And a semantic analysis of the scene aimed at the recognition of shape, size and orientation of objects will reduce the amount of data required to transmit the visual information. All these improvements will contribute to improve the user experienece during teleoperation.



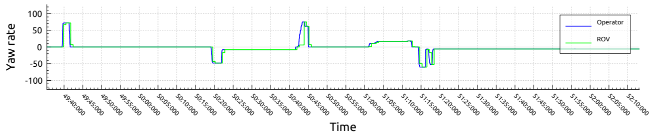
(a)



(b)



(c)



(d)

Fig. 5: Difference between commanded velocities in operator (sender) and ROV (receiver) sides. The velocity control has been modeled considering a fixed BER of $0{,}0007$ %, a bitrate of 1.9 kbps and 85 ms intrinsic delay. (a) x axis. (b) y axis. (c) z axis. (d) yaw.

References

[1] X. Che, I. Wells, G. Dickers, P. Kear, and X. Gong. Re-evaluation of rf electromagnetic communication in underwater sensor networks. *IEEE Communications Magazine*, 48(12):143–151, 2010.

[2] H. Kaushal and G. Kaddoum. Underwater optical wireless communication. *IEEE Access*, 4:1518–1547, 2016.

[3] L. Meier, P. Tanskanen, L. Heng, G. H. Lee, F. Fraundorfer, and M. Pollefeys. Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision. *Autonomous Robots*, 33(1-2):21–39, 2012.

[4] M. Prats, J. Pérez, J. Fernández, and P. Sanz. An open source tool for simulation and supervision of underwater intervention missions. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 2577–2582, 2012.

[5] E. M. Rubino, D. Centelles, J. Sales, J. V. Martí, R. Marín, P. J. Sanz, and A. J. Álvares. Underwater radio frequency image sensor using progressive image compression and region of interest. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Aug 2017.