SCIENCE AND TECHNOLOGY ORGANIZATION
CENTRE FOR MARITIME RESEARCH AND EXPERIMENTATION

**Reprint Series** **CMRE-PR-2019-129**

# Real-time clock synchronisation in underwater acoustic networks

Arjan Vermeij, Andrea Munafò

June 2019

Originally published in:

OCEANS 2015, 18-21 May 2015, Genoa, Italy,
10.1109/OCEANS-Genova.2015.7271664

# About CMRE

The Centre for Maritime Research and Experimentation (CMRE) is a world-class NATO scientific research and experimentation facility located in La Spezia, Italy.

The CMRE was established by the North Atlantic Council on 1 July 2012 as part of the NATO Science & Technology Organization. The CMRE and its predecessors have served NATO for over 50 years as the SACLANT Anti-Submarine Warfare Centre, SACLANT Undersea Research Centre, NATO Undersea Research Centre (NURC) and now as part of the Science & Technology Organization.

CMRE conducts state-of-the-art scientific research and experimentation ranging from concept development to prototype demonstration in an operational environment and has produced leaders in ocean science, modelling and simulation, acoustics and other disciplines, as well as producing critical results and understanding that have been built into the operational concepts of NATO and the nations.

CMRE conducts hands-on scientific and engineering research for the direct benefit of its NATO Customers. It operates two research vessels that enable science and technology solutions to be explored and exploited at sea. The largest of these vessels, the NRV Alliance, is a global class vessel that is acoustically extremely quiet.

CMRE is a leading example of enabling nations to work more effectively and efficiently together by prioritizing national needs, focusing on research and technology challenges, both in and out of the maritime environment, through the collective Power of its world-class scientists, engineers, and specialized laboratories in collaboration with the many partners in and out of the scientific domain.



**NOTE:** The CMRE Reprint series reprints papers and articles published by CMRE authors in the open literature as an effort to widely disseminate CMRE products. Users are encouraged to cite the original article where possible.

# Real-time clock synchronisation in underwater acoustic networks

Arjan Vermeij and Andrea Munafò
NATO STO Centre for Maritime Research and Experimentation (CMRE)
La Spezia, Italy.
Email: [arjan.vermeij,andrea.munafo]@cmre.nato.int

***Abstract*—This paper proposes an algorithm for real-time clock synchronisation in underwater acoustic networks. The algorithm models modem clocks as linear functions and embeds the data necessary for the clock synchronisation procedure as time stamps that can be communicated in the payload of later messages and only when necessary, hence limiting the communication overhead. The proposed solution takes explicitly into account the limitations of the acoustic communication channel and network node mobility. Specific implementation issues are discussed for its deployment within operational networks. Experimental results are given from the COLLAB-NGAS14 campaign, held in October 2014 off the coast of West Italy.**

***Index Terms*—Clock synchronisation, underwater acoustic networks, acoustic communications, multi-agent systems, localisation and navigation, underwater sensor network, optimisation.**

## I. Introduction

Underwater there is no 'absolute' clock, like the one provided by the GPS system. At the same time, the availability of a shared time is critical for a number of applications ranging from implementation of Media Access Control (MAC) layers in acoustic underwater networks (e.g. it allows for a TDMA algorithm to be utilised over a multi-hop network), to bistatic sonar applications, where the receiver needs a precise estimate of the moment the source starts transmitting [1], to localisation and navigation of Autonomous Underwater Vehicles (AUV), where an advantage of synchronised clocks lies in the ability to navigate through one-way travel times (OWTT), as opposed to two-way travel time (TWTT). Synchronous-clock acoustic navigation is described in [2] using high stable oscillators. Although effective, using dedicated hardware increases the cost and the energy requirement of the nodes, and it is oftentimes not desirable in persistent autonomous networks, especially in those composed of small and cheap devices.

Several clock synchronisation protocols have been designed for terrestrial networks. Although these networks do share many of the problems of the underwater ones (e.g. unreliable packet delivery, limited communication ranges, etc.), clock synchronisation protocols for terrestrial wireless networks cannot be transposed to the underwater domain, as they rely on assumptions (e.g. negligible signal propagation time) that are not valid underwater or they induce a communication burden that is not sustainable in acoustic-based communications. Terrestrial clock synchronisation protocols, - e.g. Network Time Protocol (NTP) - are not readily applicable to underwater acoustic networks. Factors such as long propagation times, low packet delivery success rates, communication ranges that vary over time in an unpredictable manner require the development of acoustic-specific solutions.

The clock synchronisation algorithm presented in this work is explicitly built to tackle the peculiarities of underwater acoustic networks and of specific scenarios. For this reason, it is important here to specify what kinds of networks and applications this paper is concerned with. Firstly, the acoustic network considered in this work can be composed of both fixed and mobile nodes, and these nodes are considered to be at a maximum distance of 7.5 km. Nodes might come into and go out of the communication range due to changes in the communication channel, and/or due to the movement of nodes themselves (e.g. AUVs). Some of the network nodes might be persistently part of the network, while others might be taken out and put back in the water after a prolonged period of time.

Within this scenario, the proposed algorithm is able to continuously estimate the relative clock offset and drift between the nodes. It does so, operating at application level and relying on timing features made available by the physical layer , namely the availability of precision transmission and reception time stamps. The opportunistic exchange of these time stamps makes the algorithm easily implementable within underwater sensor networks where packet delivery success rates are not optimal and/or no communication is possible for extended periods of time. Furtheremore, the approach can be tuned to the requirements of the application. It can optionally reduce the synchronisation communication overhead when more important messages must go through the network. Finally, the algorithm is able to detect and respond to modem resets, and hence is robust against asynchronous events (e.g. interruptions due to maintenance, failures, etc.) affecting the nodes and hence their internal clocks.

The remainder of the paper is organised as follows: Section II describes the mathematical framework for the proposed clock synchronisation algorithm. Section III goes into the details of the on-line implementation. Section IV presents results from the COLLAB-NGAS14 sea trial. Conclusions are given in Section V.

## II. Problem settings

Let $t_m$ be the time reported by modem $m$'s clock. At time $t_{p,0}$, or $p_0$ for short, modem $p$ sends a packet to modem $q$, which receives it some time later, at time $t_{q,1}$, or $q_1$. At $t_{q,2}$

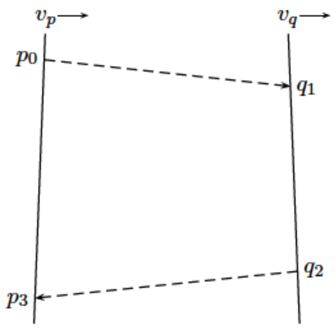


Fig. 1. A two-way packet exchange.

(or $q_2$) modem $q$ responds with a packet of its own, which modem $p$ receives at time $t_{p,3}$ (or $p_3$). This sequence is referred to as one *two-way packet exchange*, see Fig. 1.

Modem $m$ is attached to *node* $m$, which is assumed to have some computing capability. The modem reports the time stamp of each packet that is sent or received, to its node, in the local time $t_m$. After a two-way packet exchange, node $p$ knows time stamps $p_0$ and $p_3$, and node $q$ knows time stamps $q_1$ and $q_2$. Node $q$ may then send time stamps $q_1$ and $q_2$ to node $p$ in the payload of a later packet.

The clock synchronisation problem is hence that to calculate on node $p$ a function $f_{pq}$ able to map time stamps reported by modem $q$ to modem $p$'s clock:

$$t_p = f_{pq}(t_q) \quad . \tag{1}$$

Some modem features are assumed. Each modem clock is considered to be linearly drifting. Furthermore, each modem must be able to accurately register send and receive time stamps and to make this information available to the node. Finally, if relative speed is to be used, each modem is considered capable of accuratly determine the velocity relative to the transmitting modem, typically through the Doppler shift of incoming packets [3]. Note that this is not an excessive demand on the modems since they cannot decode incoming packets without the accurate determination of the Doppler shift.

Finally, it is important to point out here that nodes can have more than one clock. Depending on the implementation, the application, transport, network, MAC, and physical layer can all have their own clock. This work assumes that the time stamps reported by the modem are physical layer time stamps. In wireless sensor networks, at the transmitter node the delay due to packet transport from the application layer, down the communications stack, to the physical layer, causes errors in clock synchronisation. And at the receiver node the packet transport from the physical layer, up the communications stack, to the application layer, introduces additional errors. The use of physical layer time stamps eliminates errors due to delays introduced by the communication stack [4].

## III. Real-time clock synchronisation

If the modem time of node $p$, $t_p$, is modelled as a linear function with an offset $\Delta_{pq}$ and a constant drift $\delta_{pq}$, relative to the time of another modem $t_q$,

$$t_p = (1 + \delta_{pq})\, t_q + \Delta_{pq} \quad . \tag{2}$$

then, it is possible to calculate the relative clock drift $\delta_{pq}$ and the offset $\Delta_{pq}(p_i)$ using, for example, the least square method to fit a linear regression model (constant clock drift assumption). This has been discussed in [5], where the relation between the relative clock drift $\delta_{pq}$ and the offset $\Delta_{pq}$ is shown to be:

$$\begin{aligned}\Delta_{pq}(p_1) = \frac{1}{2}&\left(-(p_0+p_3)+q_2+q_1+\delta_{pq}(q_2-q_1)\right)\\ &+\frac{\delta_v}{2c_p}(1+\delta_{pq})(q_2-q_1)\\ &-\frac{v_p}{2c_p}\left(p_3-p_0-(1+\delta_{pq})(q_2-q_1)\right) \quad .\end{aligned} \tag{3}$$

Equation (3) depends on node velocities and on acoustic packets reception and transmission time stamps. Knowing the maximum speed of both nodes $p$ and $q$, as well as $\delta_v = v_q - v_p$ (e.g. it is provided by the acoustic modem), makes it is possible to calculate intervals of possible values $\Delta_{pq}(p_1)$, that are then converted into single values taking the centre of the interval. As noted in [5], the presence of the complicating factor $\delta_{pq}$ in (3) can be solved using an iterative approach, where the input to each iteration cycle is the $\delta_{pq}$ calculated in the previous iteration cycle, being zero for the first iteration cycle. The iteration stops when the output of the linear regression model has converged to a stable value.

In order to implement the algorithm proposed in [5], some additional issues must be considered:

- The nodes do not have a way of knowing packet transmission times a priori, nor the modems encode, automatically at physical level, transmission or reception times. Rather, each packet generates the necessary time stamps, and all this data is communicated in the payload of later packets.
- The application level, at each node, must be responsible for the encoding of the transmission and reception time stamps into dedicated messages.
- Given that each node receives from others a set of transmission and reception time stamps, it has to correctly associate them to form sequences of packet round-trips, that feed into the described clock synchronisation algorithm. This association procedure is complicated by the prevalence of packet loss.

Each modem registers the corresponding transmission ($p_0$) and reception ($p_3$) time stamps (see Fig. 1). To share this information with the rest of the network, each node, at regular intervals, encodes its modem's address and the previously registered time stamps into a message to be broadcast. Note that, for each reception time stamp the corresponding source address must also be included. To limit the communication footprint and to allow the user to trade off the clock synchronisation performance and the communication requirements some of the efficient encoding schemes of [6] are put into practice (e.g. relative encoding).

Given a sequence of transmissions $T_a = \{t_0, \dots t_{n-1}\}$ and receptions $R_b = \{r_0, \dots r_{m-1}\}$, between two nodes, $a$ and $b$, to calculate the clock offset and drift it needs to be understood which $t_i$ corresponds to which $r_j$ or, in other words, which $t_i$ and $r_j$ were generated by the same modem packet. The association problem is then that to find the largest partial bijection(s) $f : T_a \to R_b$.Note that, the bijection is partial because in general it cannot be guaranteed that $T_a$ contains all transmission time stamps, or that $R_b$ contains all reception time stamps (e.g. packet loss). Only on node $a$, $T_a$ will contain all transmission time stamps. The solution $f$ does not necessarily have to be unique and multiple solutions when $T_a$ and/or $R_b$ are smaller sets.

The association problem can be further reduced, considering that sevaral constraints must hold for each association. First, associations cannot "cross" each other, like in Fig. 2. Let

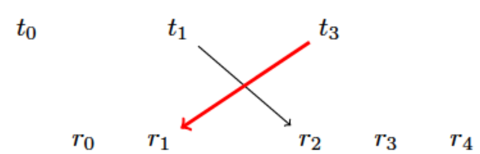


Fig. 2. Constraint 1: Modem packets associations must be ordered (the figure shows an invalid association).

$t_i \mapsto r_j$ stand for a transmission and reception time stamp pair that were generated by the same modem packet. Let both sequences be sorted in ascending order, i.e. $t_{i-1} < t_i$ and $r_{j-1} < r_j$. Then if $t_i \mapsto r_j$, then $\forall k < i, l > j, \neg(t_k \mapsto r_l)$, and $\forall k > i, l < j, \neg(t_k \mapsto r_l)$ (constraint 1).

Additionally, there are physical limits on how much the distance can change between two associations, see Fig. 3. Assume $t_i \mapsto r_j$ and the existence of $t_k, k \neq i$, and $r_l, l \neq j$.

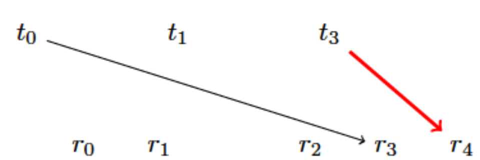


Fig. 3. Constraint 2: there are physical limits on the change in distance between two associations.

The elapsed time between $r_j$ and $r_l$ can be calculated as $|r_l - r_j|$, and the change in distance as $|t_k - t_i + r_j - r_l|\, c$, where $c$ is the sound velocity. The estimated speed $s_{k,l}$ is the change in distance divided by the elapsed time. Let $v_a$ and $v_b$ be the maximum speed of node $a$ and $b$ respectively. Then it follows that $t_i \mapsto r_j$ and $t_k \mapsto r_l$ only if $s_{k,l} \leq v_a + v_b$ (constraint 2).

With these constraints the association problem can be formulated as a Linear Assignment Problem (LAP), where the cost function $C_{k,l}$ to be minimised for the association of $t_k, k \neq i$ to $r_l, l \neq j$ is defined as

$$C_{k,l} = \begin{cases} s_{k,l} & \text{if constraints 1 and 2 are met,} \\ \infty & \text{otherwise.} \end{cases} \quad (4)$$

It is worth noting that the choice of the specific cost values when constraints 1 and 2 are met do not play a key role, as long as they are greater than zero. The specific choice made here, which finds the association at *minimum node velocity*, reflects the fact that most of the nodes in our network are static.

From an implementation stand point, the association of transmission and reception time stamps is done solving the Linear Assignment Problem using the Jonker-Volgenant algorithm [7].

Since the computation time increases with the number of time stamps that have to be associated, the optimisation is done using a running window of time stamps. Considering a time limited history let the algorithm adapt to changing conditions, as for instance to change in the quality of the acoustic channel, and it help improving the performance when the modem clock resets.

## IV. Field Trials

The algorithm was tested in the field during the COLLAB–NGAS14 experimental campaign, which was held from October 29 to October 30, 2014 off the coast of Tuscany, Italy.

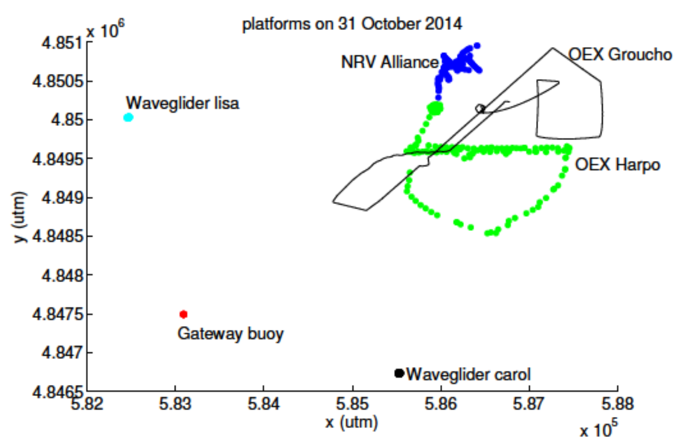


Fig. 4. Platforms deployed on October 31, 2014, during the COLLAB-NGAS14 campaign. Coordinates are shown in UTM, Zone 32N. Similar deployment used throughout the experimental period.

The data presented in this paper were collected at the site shown in Fig. 4 near 43.7754N, 10.03333E. The entire area of operation is a 7.5 km square. Water depth in the area decreases gradually from around 25 m (north-east region) to around 60 m (south-west part).

The sound velocity profiles, as measured at a location close to the north-east corner of the operational area, is shown in Fig. 5. Note that in the first 25 m the sound speed is nearly constant for October 29 and October 31, with more variability registered on October 30.

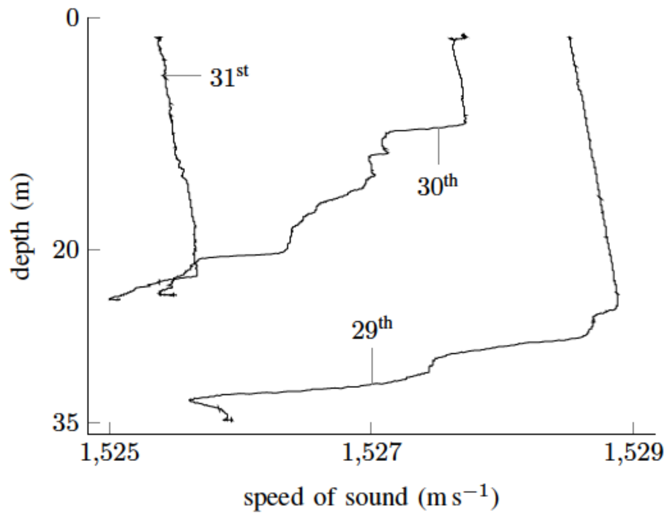


Fig. 5. Sound velocity profiles measured on October 29–31, 2014.

### A. Experimental setup

The deployed acoustic network was composed of one moored gateway buoy, two Wave Gliders (Carol and Lisa), two CMRE Ocean Explorer (OEX) AUVs (OEX Groucho and OEX Harpo) and one modem mounted on NRV *Alliance*. Fig. 4 shows the deployment on October 31, with very similar deployments installed for the other operational days. All nodes were equipped with the EvoLogics 7–17 kHz acoustic modem. The modems of the gateway and NRV *Alliance* were all deployed at a fixed depth of 25 m. Wave Gliders (WGs) had their modem at 6 m depth. All surface nodes were equipped with a GPS receiver that provided accurate position information. OEX Groucho was commanded to navigate at 25 m depth; OEX Harpo was kept at a depth of 15 m. Both vehicles relied on a high-quality inertial navigation unit for positioning and navigation (navigation error $\sim 0.1\,\%$ of the distance travelled). Note that although the Wave Gliders are mobile assets, they were station keeping throughout the trial.

### B. Results

This section describes the experimental results. Unfortunately, an implementation bug on the detection of modem resets affected the real-time results. Wrong and too frequent detection of modem clock resets reduced the number of time stamps available for the clock estimation process not making it possible to correctly converge most of the times. For this reason, this works presents results obtained replaying the software process that was calculating clock offset and drifts but modified to better handle modem clock resets. This allows to have more data, and to better characterise the estimation performance. Note however that the modem time stamps are exactly and only those measured during the in-the-field deployment, and that the only difference with the deployed version lies in removing the modem reset detection issues.

Furthermore, this section presents a post-processing analysis of the clock synchronisation. The post-processing estimate uses all clocks offsets and drifts using the entire history of collected time stamps, as if all transmission and reception time stamps were known on all nodes and with perfect knowledge of transmission-reception time stamp association. In this respect, the post-processing results are also intended to form a ground truth against which the on-line results can be compared.

Post-processing results of the synchronisation between the two OEX AUVs are shown in Fig. 6. The close fit of the model shows that indeed the linear clock model represents a reasonable assumption. The difference between clocks is dominated by a linear clock skew.

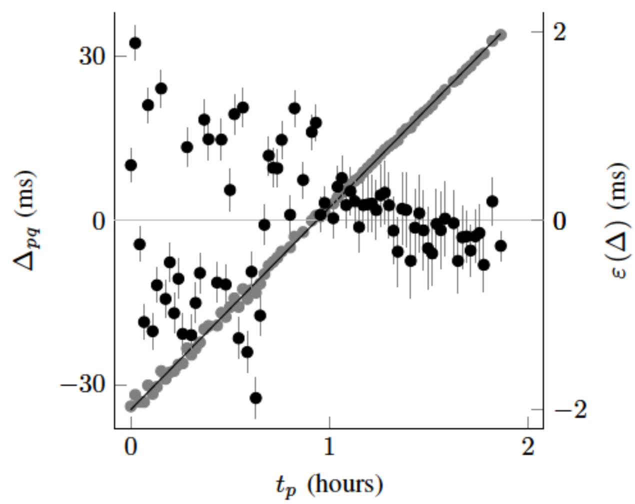


Fig. 6. The grey dots show the calculated $\Delta_{pq}$'s for the clock offset between OEX Groucho (node 2) and OEX Harpo (node 3) on October 29, 2014, the diagonal line shows the result of the linear regression calculated according to (3), normalised such that the line is centred around 0. The black dots show the residual $\varepsilon(\Delta_{pq})$, the vertical bars the interval due to the uncertainty in the nodes' speed. The vertical bars are small because the distance between nodes 2 and 3 was short, from 250 m to 750 m. The large number of outliers is due to the vehicles being on the surface during the first hour.

OEX Groucho and NRV *Alliance* synchronisation performance (still obtained in post-processing) for 8 hours of operation is shown in Fig. 7. The two nodes were at about 1 km of distance. Similar results were obtained for other node pairs and are not reported in this paper.

Clock synchronisation on-line results obtained on October 30 between OEX Groucho and OEX Harpo are shown in Fig. 8. The running window length was set to 60 time stamps. The figure shows the difference between the incrementally calculated offsets and the offset calculated during post-processing (open dots), and between the incrementally calculated drifts and the drift calculated during post-processing (closed dots). Note how the incremental solution is updated as newer measurements become available. Note also the presence of gaps in the estimation due to gaps in the communication (e.g. after around 3 hrs). The algorithm is however able to continue as soon as new data are produced. Finally, note the presence of a modem reset happening after 6 hrs of activity.

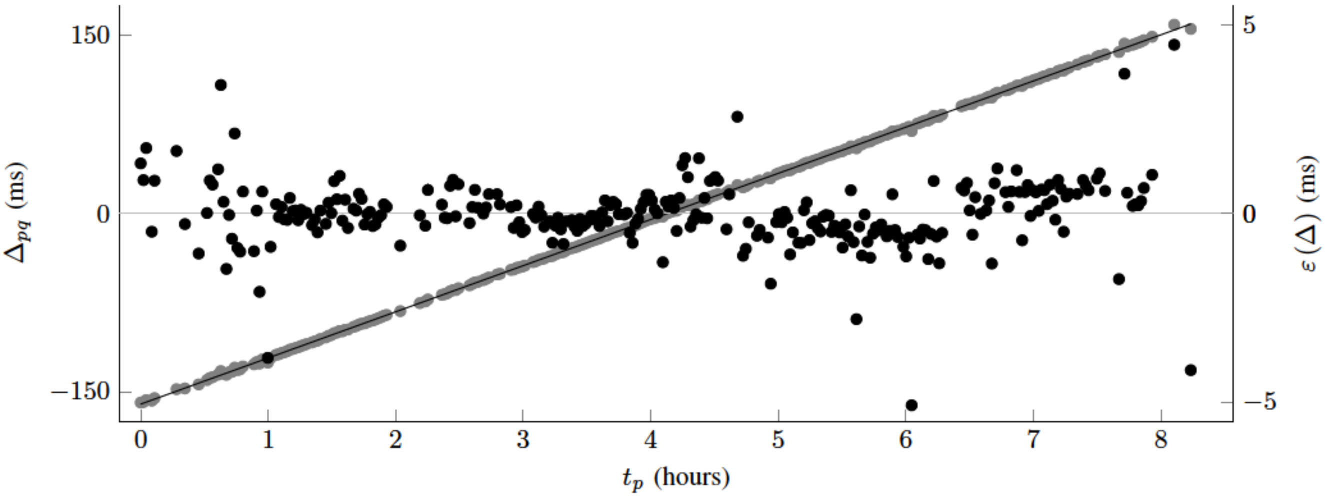


Fig. 7. Clock synchronisation result obtained on October 30, 2014. The gray dots show the calculated $\Delta_{pq}$'s for the clock offset between OEX Groucho (node 2) and NRV *Alliance* (node 1), the diagonal line shows the result of the linear regression calculated according to (3), normalised such that the line is centred around 0. The black dots show the residual $\varepsilon(\Delta_{pq})$. Distance between the nodes was, on average, around 1 km.

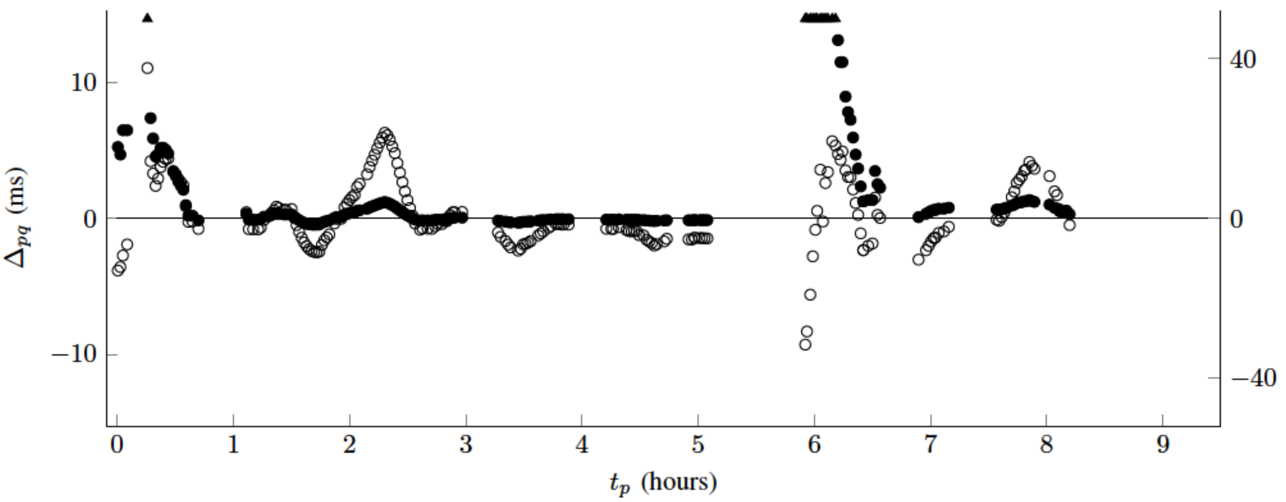


Fig. 8. Incremental output on OEX Groucho (node 2), October 30, 2014, for the clock offset and drift between OEX Groucho (node 2) and OEX Harpo (node 3). The open dots shows the difference between the incrementally calculated offsets and the offset calculated during post-processing, the closed dots the difference between the incrementally calculated drifts and the drift calculated during post-processing.

The modem reset was correctly detected and forced the clock synchronisation to re-start. The difference between the off-line and the on-line results is partially due to the fact that the two nodes were constantly moving and no Doppler data was transmitted during this sea trial and hence used for this replayed calculation.

## V. Conclusions

This paper reports details on the implementation and deployment of a clock synchronisation algorithm for underwater acoustic networks. The algorithm is able to continuously estimate inter-node clock offset and drifts, based on the opportunistic exchange of modem packets including transmission and reception time stamps. It operates at application level, with some reliance on physical layer time stamps provided by the modem. The benefit is that it is able to deal explicitly with the constraints of acoustic communications, such as long propagation delays, intermittent communications, and network node mobility. Some additional acoustic overhead is necessary, but this can be opportunistically distributed among network traffic, for example after transmission of higher priority data. Details on how the approach was implemented in an operational acoustic network are reported, including results from experimental deployments.

## ACKNOWLEDGEMENT

This work has been supported by the NATO Allied Command Transformation (ACT), Collaborative Anti-Submarine Warfare Programme, Project 1 and Project 2.

The authors would like to thank everyone involved in the COLLAB-NGAS14 sea trial, Kevin LePage and Ryan Goldhahn (Scientists In Charge), everyone in the Engineering Department, and the captain and crew of NRV *Alliance*.

## REFERENCES

[1] G. Ferri, A. Munafò, R. Goldhahn, and K. Lepage, "A non-myopic, receding horizon control strategy for an AUV to track underwater targets in bistatic sonar scenarios: proposed approach and experimental results from COLLAB13 exercise," in *Proc. Intl. IEEE Conference On Decition and Control*, June 2014.

[2] R. Eustice, H. Singh, and L. L. Whitcomb, "Synchronous-clock, one-way-travel-time acoustic navigation for underwater vehicles." *J. Field Robotics*, vol. 28, no. 1, pp. 121–136, 2011. [Online]. Available: http://dblp.uni-trier.de/db/journals/jfr/jfr28.html#EusticeSW11

[3] K. G. Kebkal and R. Bannasch, "Sweep-spread carrier for underwater communication over acoustic channels with strong multipath propagation," *The Journal of the Acoustical Society of America*, vol. 112, no. 5, pp. 2043–2052, 2002. [Online]. Available: http://scitation.aip.org/content/asa/journal/jasa/112/5/10.1121/1.1504855

[4] K. Kebkal, O. Kebkal, V. Kebkal, and R. Petroccia, "Synchronization tools of acoustic communication devices in control of underwater sensors, distributed antennas, and autonomous underwater vehicles," *Gyroscopy and Navigation*, vol. 5, no. 4, pp. 257–265, 2014. [Online]. Available: http://dx.doi.org/10.1134/S2075108714040063

[5] A. Vermeij and A. Munafò, "Clock synchronisation in underwater acoustic networks," in *Conf. on Underwater Communications Networking (UComms14)*. IEEE, 2014.

[6] A. Vermeij, T. C. Furfaro, A. Munafó, and J. Alves, "The digital communications stack, from NEMO towards a NATO underwater networked communications architecture and implementation," CMRE, Tech. Rep. CMRE-MR-2014-008, oct 2014, NATO Unclassified.

[7] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, no. 4, pp. 325–340, 1987. [Online]. Available: http://dx.doi.org/10.1007/BF02278710

# Document Data Sheet

| Security Classification | | Project No. |
|---|---|---|
| | | |

| Document Serial No. | Date of Issue | Total Pages |
|---|---|---|
| CMRE-PR-2019-129 | June 2019 | 6 pp. |

*Author(s)*

Arjan Vermeij, Andrea Munafò

*Title*

Real-time clock synchronisation in underwater acoustic networks

*Abstract*

This paper proposes an algorithm for real-time clock synchronisation in underwater acoustic networks. The algorithm models modem clocks as linear functions and embeds the data necessary for the clock synchronisation procedure as time stamps that can be communicated in the payload of later messages and only when necessary, hence limiting the communication overhead. The proposed solution takes explicitly into account the limitations of the acoustic communication channel and network node mobility. Specific implementation issues are discussed for its deployment within operational networks. Experimental results are given from the COLLAB-NGAS14 campaign, held in October 2014 off the coast of West Italy.

*Keywords*

Clock synchronisation, underwater acoustic networks, acoustic communications, multi-agent systems, localisation and navigation, underwater sensor network, optimisation

*Issuing Organization*

NATO Science and Technology Organization
Centre for Maritime Research and Experimentation

Viale San Bartolomeo 400, 19126 La Spezia, Italy

*[From N. America:*
STO CMRE
*Unit 31318, Box 19, APO AE 09613-1318]*

Tel: +39 0187 527 361
Fax:+39 0187 527 700

E-mail: library@cmre.nato.int