Oceans - Europe 2005

An Acoustic Network Protocol for Sub-sea Sensor Systems

A. E. Adams, University of Newcastle upon Tyne,
School of Electrical, Electronic and Computer Engineering,
Newcastle upon Tyne, NE1 7RU, United Kingdom. alan.adams@ncl.ac.uk
G. Acar, Centre for Communication Systems Research, University of Surrey
Guildford, GU2 7XH, United Kingdom

Abstract - **This paper describes a new protocol for the a network of acoustically-linked sub-sea sensors. Much of the work described in this paper took place under the ACME project. This project was funded as part of the European Union's Framework 5 programme and involved the collaboration of researchers in France, the Netherlands and the United Kingdom. The main aim of the project was to develop technologies capable of supporting a data gathering system, able to operate in the difficult conditions of shallow coastal waters and estuaries. Such a system could provide continuous monitoring of important environmental conditions, such as pollution and sediment levels, as well as providing data on more general water parameters such as current flow.**

I. INTRODUCTION

The ability to monitor the underwater environment and/or control underwater equipment over large areas in quasi-real-time is required by various environmental organisations, off-shore oil companies, research communities, navies, and river and harbour authorities. It is anticipated that the need for such systems will grow significantly in the future in response to the demand for more detailed and widespread supervision of ocean conditions, particularly in terms of the early recognition of pollution events and their potential consequences. In the past the solution to the problem of widespread environment monitoring has been to deploy sensors on the sea floor and to transfer the measured data to the shore via cables or through radio links [1]. This approach demands that surface units, equipped with radio transmitters are deployed above the underwater sensors in order to provide the RF links. In many situations, however, the use of surface buoys and sea-floor cables is not possible due to the associated cost, or the environmental conditions, such as waves and strong currents. Further, human activities such as trawling, general navigation and fishing often render such systems inoperable through the destruction of the surface buoys or the sub-sea cables. Under such conditions, underwater acoustic communication links are the only feasible means for the transfer of the data measured at the sensors to the shore, or to enable the control underwater units from the shore [2, 3].

This paper describes a new protocol for such an acoustic network of sub-sea sensors. Much of the work described in this paper took place under the Acoustic Communication Network for the Monitoring of the Underwater Environment in Coastal Areas (ACME) project. This project was funded as part of the European Union's Framework 5 programme and involved the collaboration of researchers in several European countries. Research institutes, companies, both small and large, and a university were engaged in this work. The main aim of the project was to investigate the development of a system capable of operating in a harsh communication environment such as an estuary. Such a system could provide continuous monitoring of important environmental conditions, such as pollution and sediment levels, as well as providing data on more general water parameters. The envisaged operating environment mitigates against the use of wired connections to the data gatherers, because of the very likely, frequent damage to such a system, owing to the passage of large ships and the use of fishing nets. A network based upon acoustic transmission was chosen therefore. However, the environment is also difficult in acoustic terms, the shallow water creating significant multi-path problems and the anticipated shipping traffic producing high background noise levels. The project as a whole considered modem development and signal design, however, this paper will concentrate on the protocol aspects of the system used in the final version of the network – ACMENet. The protocol was exercised in both tank experiments and in the real conditions of the Westerschelde. Bandwidth limitations and long propagation delays provide more serious problems in underwater acoustic communications than they do in radio communications. These limitations make it especially important to minimise the amount of control messaging and to use the long propagation delays to advantage by multiplexing messages through the water. This requires designing scheduling mechanisms that are tolerant of highly variable propagation delays.

The ACME network protocol, ACMENet, is a master-slave protocol intended for small to medium sized underwater sensor networks with arbitrary topologies. It is anticipated that the slave nodes will be responsible for measuring environmental quantities and transmitting data to the master node. Although the majority of data retrievals will be direct transfer of data from a slave node to the master node, ACMENet does support relayed data retrieval from remote slave nodes.

Medium Access Control (MAC) in ACMENet is based on scheduled transmissions, where the master node broadcasts/multicasts transmission schedules that are designed such that data packets from slave nodes arrive at the master node consecutively, without collisions. In the case where a asynchronous multi-user transmission scheme, such as CDMA, is used, the nodes may be scheduled so that groups of synchronised data packets are received simultaneously. Each packet in the group will use a different orthogonal channel described by the multi-user scheme. Further, the transmission schedules are piggy-backed by special instructions to control the transmission power levels and modulation rates at slave nodes. Once all data packets have been received, the master node will then cause the sequence to be repeated by broadcasting other transmit instructions to the remote sensor nodes.

The network controller associated with the master node is a sophisticated system responsible for; the provision of a MMI (Man Machine Interface) for the human network operator; the management of the master acoustic modem; measurement, storage and display of the link error statistics via the MMI; automatically generating appropriate instructions to the remote nodes; interfacing to the sensor controller; and the automatic management of slave transmission power levels and data rates. Special attention will be given to the power/rate control aspects which employ artificial intelligence techniques to achieve maximum reliable data transfer rates with minimum energy consumption at the remote (battery-powered) sensor nodes[4].

II. THE ACME NETWORK PROTOCOL[5]

The general organisation of an ACME network is shown in figure 1. Traffic in the network is managed by control (CTRL) commands sent from the base or master node. The transmission schedule is designed so that data packets from each remote node do not collide but arrive at the base node successively. If a multi-user transmission scheme, such as CDMA, is used, the nodes may be scheduled so that groups of synchronised data packets are received in each reception slot[6]. Under these circumstances, each packet in the group will use a different orthogonal channel described by the multi-user scheme. Once all data packets have been received, the master node will then cause the sequence to be repeated by broadcasting another transmit instruction to the remote nodes.

In order to determine the data transfer schedule the master node must know the time delays to each of the sensor nodes. This information is determined using a setup procedure. This is achieved by a node-by-node initiation procedure controlled by the master node. It is assumed that the master node knows the ID numbers of all remote nodes with which it is to communicate. For each remote node the master will transmit a *Sync* packet.

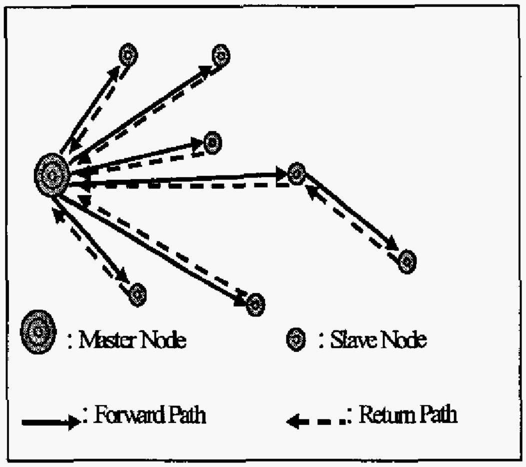

Figure 1: General network scenario

After receiving a *Sync* message a remode node waits a pre-determined length of time from the end of the *Sync* message. Once this time has elapsed the remote nodes respond to the master node with a *SyncAnswer* message. The master node records the elapsed time between it sending the *Sync* message and receiving the *SyncAnswer* message. This process is repeated to determine the transmission delay between the master node and each remote node. The master node is then able to calculate an efficient transmission schedule. Once this has been calculated then the data gathering process can begin.

In some situations conditions may exist in the network which mean that direct communication between the master node and some slaves is not possible, this may be a temporary event, e.g. during the passage of large ships, or on a more permanent, basis owing to range considerations or sea-bed topography[7]. In such cases communication can be achieved using intermediate slave nodes as relays, as illustrated in Figure 2.

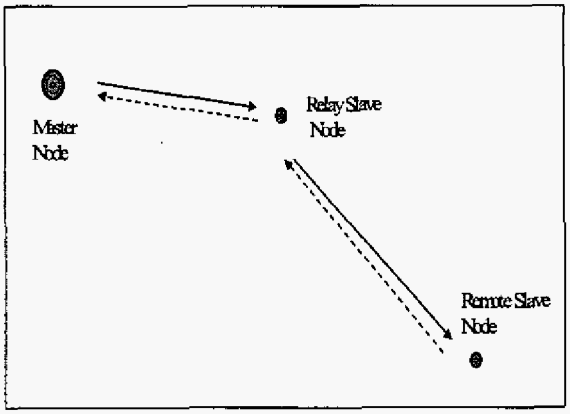

Figure 2: Relayed data operation

A separate power level may be used for relay operations as compared to direct slave-master communication. Because of this, each slave node in the network maintains two sets of power level and modulation type parameters. One set is used in direct data requests from the master node. The second set of parameters is used when the slave node is part of a relay circuit. The joint management of the power level and modulation type at both relay and remote nodes is contolled by the master node. Relaying nodes are made aware of modulation types by permitting them to peek at the instruction in the CTRL packet that it is to relay to the remote node. Note that the relay node only reads instruction to update its relay mode power level and modulation type parameters. In contrast, the remote node not only reads but also executes the instruction in the CTRL packet. This technique simplifies the design of the ACME controller by guaranteeing the same modulation type at both the relay and remote nodes. Moreover, power and rate management in direct data requests and relayed data requests are completely isolated.

III. THE POWER-RATE CONTROLLER

The ACME controller is resposible for six main tasks:

i. provision of a MMI (Man Machine Interface) for the human network operator.
ii. management of the master MATS200 acoustic modem.
iii. measurement, storage and display of the link error statistics via the MMI.
iv. automatic management of slave transmission power levels and modulation type.
v. generating appropriate instructions.
vi. responding to the user interface.

It is intended that the conrtroller should operate autonomously for long periods of time. The system architecture of the ACME Controller is illustrated in figure 3. The Task Manager receives, from the user, the list of slave nodes from which data is to be requested. At the same time, the Power and Rate Controller updates the modulation type and slave power levels according to recent link error statistics. Based on the recommended power levels and modulation type, determined by the Power and Rate Controller, the most recent propagation delay measurements, and the list of target slave nodes, the Task Manager decides whether or not a new transmission schedule must be computed.

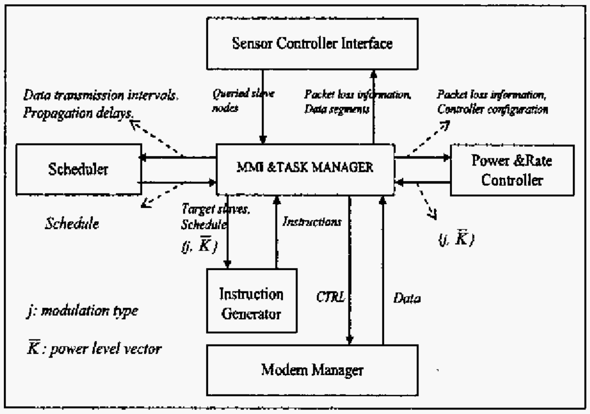

Fig. 3. ACME Controller

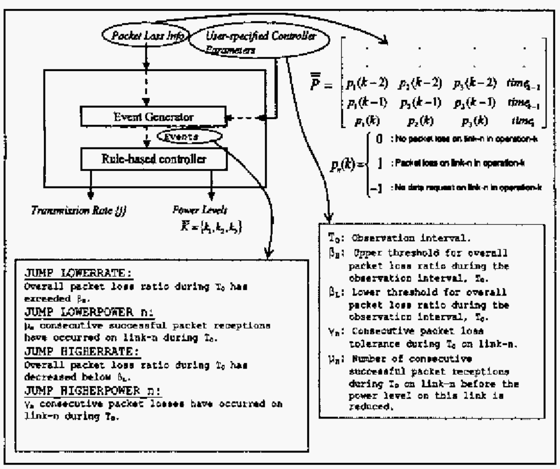

Fig. 4. Power and Rate Controller

As shown in figure 4, the Power and Rate Controller consists of two units; an Event Generator and a Rule-based Controller. According to a recent set of packet loss statistics and the user-specified controller parameters the Event Generator triggers a combination of the events listed below.

- *JUMP_LOWERRATE*: to change to a lower rate modulation type.
- *JUMP_HIGHERRATE*: to change to a higher rate modulation type.
- *JUMP_LOWERPOWER_n*: to decrease transmission power level at slave-n.
- *JUMP_HIGHERPOWER_n*: to increase the transmission power level at slave-n.

The Event generator manages the network modulation type (common to all nodes) in accord with the packet loss statistics for the whole network, whereas the the power level, which is specific to each node, is controlled as a result of monitoring the packet loss statistics for each particular link. As a result of this approach, persistent packet losses on a distant or faulty link are prevented from causing undue changes in the modulation type for other links, which will thus maintain a higher data transfer rate.

Packet loss information is monitored over an observation period, T_o. If the loss over this time is greater than some user determined tolerance level, β_H, the Event Generator initiates a JUMP_LOWERRATE event. If the overall packet loss ratio within T_o decreases below a target level for global packet loss, β_L, a JUMP_HIGHERRATE event is initiated. JUMP_LOWERRATE implies that a lower, more reliable, transmission rate is required to reduce the global packet loss ratio in the network, JUMP_HIGHERRATE indicates that the channel conditions are good enough for a faster modulation type. As well as observing the overall network packet loss ratio within T_o, the Event Generator also considers the packet losses on individual links. If the number of consecutive packet losses on link-n exceeds the user determined value, γ_n, the Event Generator triggers a *JUMP_HIGHERPOWER_n*. However, if the number of consecutive successful packet transmissions on link-n exceeds a further user determined value, μ_n, , a *JUMP_LOWERPOWER_n* event is triggered. *JUMP_HIGHERPOWER_n* involves the implementation of a higher transmission power for slave-n, *JUMP_LOWERPOWER_n* indicates the possibility of energy conservation by using a lower power level at slave-n.

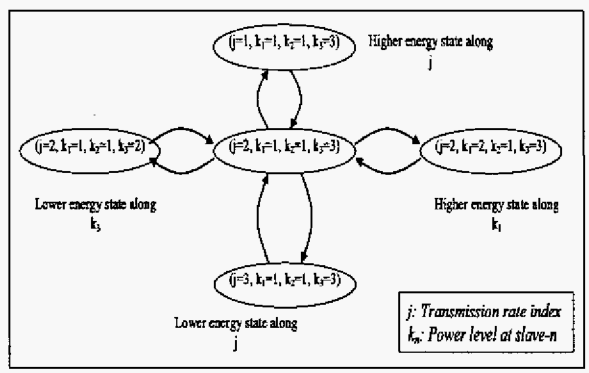

Fig. 5. Energy State Diagram

The energy state diagram that corresponding to a network with three slave nodes, each capable of transmitting at three different power levels and in five different modulation types is shown in figure 5. k_n is the index for the power level for slave-n and j is the index for the transmission rate (modulation scheme) of the network. Note that, as k_n increases so does the power level and the total energy consumption. In contrast, increasing values of j correspond to higher transmission rates. This means shorter transmission times, and thus lower energy consumption/bit.

The purpose of the Rule-based Controller is to translate these events into updates in transmission rates and slave power levels The Rule-based Controller is a forward-chaining system with a conflict resolution strategy that is based on the order of the fired rules in the agenda. Rules are checked in order until either a rule fires or the end of the rule list is reached. When a rule fires the controller goes back to the beginning of the list. If the end of the list is reached without a fired rule, the controller terminates. The order of the rules formulates the conflict resolution strategy of the system.

```
1- IF JUMP_HIGHERRATE AND JUMP_HIGHERPOWER_n => DELETE JUMP_HIGHERRATE
2- IF JUMP_HIGHERRATE AND JUMP_LOWERPOWER_n => KEEP THE JUMP THAT SAVES MORE ENERGY
3- IF JUMP_LOWERRATE AND JUMP_HIGHERPOWER_n => DELETE JUMP_HIGHERPOWER_n
4- IF JUMP_LOWERRATE AND JUMP_LOWERPOWER_n => DELETE JUMP_LOWERPOWER_n
5- IF JUMP_LOWERRATE AND RATE==MIN => DELETE JUMP_LOWERRATE
6- IF JUMP_LOWERRATE AND RATE>MIN => DELETE JUMP_LOWERRATE AND REDUCE RATE
7- IF JUMP_HIGHERRATE AND RATE<MAX => DELETE JUMP_HIGHERRATE AND INCREASE RATE
8- IF JUMP_HIGHERRATE AND RATE==MAX
      => CONVERT JUMP_HIGHERRATE TO JUMP_LOWERPOWER_n SUCH THAT MAX ENERGY IS SAVED
9- IF JUMP_LOWERPOWER_n AND POWER_n==MIN => DELETE JUMP_LOWERPOWER_n
10- IF JUMP_LOWERPOWER_n AND POWER_n>MIN
      => DELETE JUMP_LOWERPOWER_n AND REDUCE POWER_n
11- IF JUMP_HIGHERPOWER_n AND POWER_n<MAX
      => DELETE JUMP_HIGHERPOWER_n AND INCREASE POWER_n
12- IF JUMP_HIGHERPOWER_n AND POWER_n==MAX
      => CONVERT JUMP_HIGHERPOWER_n TO JUMP_LOWERRATE
```

Fig. 6. Power and Rate Controller rule base

IV. SEA TRIALS

The main ACME sea-trials were held in the Westerschelde estuary, Netherlands, in September 2003. The network consisted of one master node and three slave nodes.

The intent of these trials was to demonstrate the autonomous operation of a real-time underwater acoustic sensor network in a busy estuary using the ACMENet protocol. Sensors were connected to the slave nodes to measure, current flow, turbity and temperature. The master modem was mounted on a measurement pole near the town of Hansweert, the three slave nodes were set out in the estuary, close to but not in the main shipping lanes. The ACME Controller itself was 30km away in Middleburg and connected to the master node via a telephone line.

Prior to the actual sea trials, the network was tested in an anechoic basin at TNO-FEL laboratories in the The Hague. During these tests, the ACME Controller was observed to provide autonomous control of the network for four days. More specifically, the Power and Rate Controller successfully managed the modulation scheme and slave power levels according to the configuration parameters input by a human operator, and the Instruction Generator consistently provided the shortest set of instructions for various scenarios.

In addition, numerous relayed data retrievals and relayed synchronisations were performed during the tests in the anechoic basin. MFSK modulation schemes operated with success ratios as high as 90%. Higher data rate schemes using CDMA were less successful, however, and only achieved a success rate of 12%. This may have been due to the very strong surface multipath experienced in the test tank.

The actual sea trials commenced on 1st September. However, due to a series of problems regarding the experimental conditions in the Westerschelde, communication between the master modem and slaves 1 and 3 could not be maintained for any significant amount of time. Consequently, important capabilities of the ACMENet protocol such as relayed data retrieval and scheduled data retrieval could not be demonstrated in the Westerschelde. The Power and Rate Controller successfully performed autonomous control of the modulation scheme and power level at slave 2. Similarly, the Instruction Generator at the ACME Controller functioned satisfactorily. During the actual sea trials, the data retrieval success ratios could be as high as 80%. However, the communication between the master node and slave-2 experienced extended link outages during nearby large ship activity.

IV. CONCLUSIONS

In tank experiments prior to the main sea trials the ACMEnet protocol, including the Power and Rate Controller functions were seen to operate satisfactorily in an autonomous manner. It is unfortunate that conditions during the main trials did not give the opportunity to test the full capabilities of the system. Nevertheless, it is the case that when operating with a single node the system worked independently for more than a week. During this time the Power and Rate controller was continually adjusting power levels and modulation types in order to maintain maximum data transfer rates with minimum power drain. Over this period approximately 80% of data packets were error free.

Acknowledgments

The work described in this paper was funded by the EU under contract No. EVK3-CT-2000-00039, their support is gratefully acknowledged. The other members of the ACME consortium were Thales Underwater Systems and ORCA in France, and TNO and RWS in the Netherlands.

REFERENCES

[1] K. Kawaguchi, K. Hirata, T. Nishida, S. Obana,, H. Mikada, "A new approach for mobile and expandable real-time deep seafloor observation - adaptable observation system", *IEEE Journal of Oceanic Eng*, Volume: 27 , Issue: 2 , pp182 – 192, April 2002.

[2] S. Phoha, E.M. Peluso, R. Culver, "A high-fidelity ocean sampling mobile network (SAMON) simulator testbed for evaluating intelligent control of unmanned underwater vehicles", *IEEE Journal of Oceanic Eng*, Volume: 26, Issue: 4, pp646 – 653, Oct. 2001

[3] B. Hou, O.R. Hinton, A.E. Adams, B.S. Sharif, "A Time Domain Orientated Multiple Access Protocol for Underwater Acoustic Network Communications", *Proc IEEE Oceans '99*, Session 3D, Seattle, Sept. 1999.

[4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey", Computer Networks, vol. 38, issue.4, March 2002.

[5] G. Acar and A.E. Adams, "ACMENet Protocol in the 3rd Sea Trial", *WP3.3 Final Report for the ACME Project*, October 2003.

[6] C. C. Tseminidis, O. R. Hinton, A. E. Adams, and B. S. Sharif, "Underwater Acoustic Receiver Employing Direct-Sequence Spread Spectrum and Spatial Diversity

Combining for Shallow-water Multi-access Networking", *IEEE Journal of Oceanic Engineering,* vol. 26, no. 4, October 2001.

[7] D. L. Codiga, J. A. Rice, and P. A. Baxley, "Networked Acoustic Modems for Real-Time Data Delivery from Distributed Subsurface Instruments in the Coastal Ocean: Initial System Development and Performance", *Journal of Atmospheric and Oceanic Technology*, Vol 21, No 2, pp331-46, Feb 2004.