Multi-Cluster Protocol for Ad Hoc Mobile Underwater Acoustic Networks

Francisco Salvá-Garau and Milica Stojanovic
Massachusetts Institute of Technology
Bldg. E38-300
Cambridge, MA, 02139
xsalga@coitt.es, millitsa@mit.edu

Abstract- **An autonomous network of underwater vehicles is considered in which there is no central node, but the vehicles communicate in a distributed manner over multiple hops. The focus of network design is on a scalable multiple access technique which is applicable to varying coverage areas as well as varying number of vehicles. The proposed scheme relies on grouping the adjacent vehicles into clusters, and using time-division multiple access within each cluster. Interference among clusters is managed by assigning different spreading codes to adjacent clusters, while scalability is achieved by spatial reuse of the codes. Network operation begins with an initialization phase, and moves on to continuous maintenance during which mobility is managed. Performance is quantified through measures of connectivity, successful transmission rate, average delay and energy consumption. Simulation analysis is used to obtain optimal cluster size and transmission power for a network with given density of vehicles.**

Keywords- **Underwater Acoustic Networks, Ad Hoc Mobile Networks, Cluster Protocol, Scalability, TDMA, CDMA, Autonomous Underwater Vehicles.**

I. INTRODUCTION

Autonomous operations future naval capability (AOFNC) calls for *collaborative* missions of autonomous underwater vehicles (AUVs). Multiple vehicles, equipped with modern sensors, will find application in environmental monitoring, exploration of natural undersea resources, and gathering of scientific data. An enabling technology for these applications is wireless acoustic underwater networking.

Mobile underwater acoustic networks can be envisioned in two forms: centralized, which rely on an infrastructure of base stations to relay data, and distributed, where nodes communicate to each other directly or by establishing paths through other nodes. The focus of this paper is on the latter type of network, and in particular on an ad hoc network in which vehicles establish communication links autonomously upon deployment.

The proposed system consists of a varying number of vehicles that are required to perform collaborative tasks over a given area. To do so, vehicles must be able, at a minimum, to coordinate their operation by exchanging location and movement information.

Previous work in this area focused on protocol design based on scheduled transmission [1]. While simple for implementation, such a flat network has limited coverage. For example, it was shown in [1], that in order to receive a standard data packet from each vehicle at a 10-second interval, five or so vehicles can be supported over an area of 1 km^2. Due to the limitations inherent to acoustic propagation (namely, the delay caused by low speed of sound propagation), protocols based on time scheduling alone are severely restricted in terms of coverage. Lack of scalability prevents their application to a larger network.

In this work, we present a multiple access scheme based on clustering which provides efficient scalability by spatial reuse of channel resources. Clustering protocols have recently received much attention in the context of ad hoc radio networks ([2], [3], [4]). In [2], a clustering scheme is proposed in which in-cluster communication is achieved through time-division multiple access (TDMA), but adjacent clusters use orthogonal spreading codes to mitigate interference. Any given user in this network transmits using an assigned code within an assigned time slot. When it does not transmit, it listens for another user's transmission, using either the same code or a different code. The single receiving code is selected randomly. While efficient for radio communication, single-code reception implies long turn-around time in an acoustic channel. To improve the efficiency of inter-cluster communication, we use code-division multiple access (CDMA) across clusters. Any given user in the proposed scheme transmits using its assigned code, and it receives simultaneously from adjacent clusters using multiple codes. Detection of multiple coded signals over an underwater acoustic channel has been demonstrated experimentally [5], justifying the use of TDMA/CDMA within the proposed network.

In Section II we give an overview of the system design. System initialization and cluster maintenance algorithms are presented in Sections III and IV, respectively. Section V shows the performance evaluation results. Finally, conclusions are summarized in Section VI.

II. SYSTEM DESIGN

A. Network Design

Upon deployment, nodes in the network are organized into non-overlapping clusters. Transmission within each cluster is scheduled using TDMA. Spatial reuse of time slots is achieved by assigning orthogonal spreading codes to clusters.

Due to node mobility, network topology changes, thus necessitating a cluster maintenance algorithm. For a correct network operation, maintenance algorithm must be executed simultaneously in all clusters. Maintenance is performed in every TDMA frame, and, thus, it requires synchronization of all nodes in the network. This requirement imposes a limitation on the maximal number of nodes that can form a cluster, which is equal to the number of slots in the TDMA frame.

When using TDMA, nodes transmit at certain moments, i.e. during their assigned time slots. A node's transmission is followed by a period, called guard time, which allows the information to reach a certain distance without any interference caused by the subsequent transmission. Transmission distance is determined by transmission power, which is assumed to be equal for all nodes. In an underwater acoustic channel, due to low propagation speed, the guard time can be very long, and, consequently, long TDMA frames may be required. The frame duration determines the time between two successive transmissions from the same node, and, hence, the effective transmission rate.

Simultaneous use of slots across clusters is possible using spreading codes. A limited set of codes is used, and the code assignment pattern is repeated across the volume of the network in a way that ensures that a cluster does not have two neighboring clusters using the same code.

B. *Clustering Concepts*

Clustering involves dividing the network into groups of nodes, based on their geographic location, and defining a mechanism by which the clusters are connected. The purpose of a clustering algorithm is to achieve a more efficient use of network resources through spatial reuse, which leads to an increase in the network capacity, in terms of the number of nodes and the region covered.

Two nodes are connected when they are within each other's transmission range. Nodes can be connected to all or some of the nodes in their cluster as well as to nodes in other clusters. Thus, nodes in a cluster will have different tasks depending on the connectivity. Nodes can be cluster-heads, cluster-connectors or ordinary nodes. Because the nodes are mobile and the network topology changes, all nodes must be able to perform any of the tasks.

Cluster-heads are directly connected to all other nodes in the cluster, so that all nodes in the same cluster are, at most, two hops away from each other. Cluster-heads define a cluster and perform cluster maintenance. Cluster-connectors can communicate with nodes in more than one cluster. Ordinary nodes are directly connected just to nodes in their own cluster. An ordinary node is connected, at least, to its cluster-head.

C. *Spatial Reuse of Resources*

The use of CDMA in clustered networks is based on code assignment across different clusters. Assuming that a node can either transmit or receive, but not both (half-duplex), there are three possible options for code utilization. The first option is to assign a code to every cluster for reception. Then, all transmissions sent to a node in that cluster shall be done using that cluster's code. Secondly, a code may be assigned to every cluster for transmission. Then, all the nodes from that cluster transmit using the same code. Finally, the third option is to assign a code to every transmitter-receiver pair within a cluster [2]. The first and the third options do not prevent collision of transmissions from different clusters using the same code. The second option avoids the possibility of collisions, provided that code assignment is such that a cluster does not have two neighboring clusters operating at the same code.

Following the transmitter-based code assignment, a node in the role of cluster-connector needs to receive on more than one code. In [2] a random selection of the code on which a node listens is proposed. This is a good solution for radio networks; however, in underwater communications, the turn-around time between two receptions of information from the same node can be long. A better solution in this case is to listen to multiple transmissions simultaneously. In the simplest case, this can be accomplished by using a bank of single-user receivers. In [5], an experimental four-user system was investigated, showing excellent results for signal-to-interference ratios as low as -12 dB. The amount of interference that such a system can withstand is related to the coding gain used. The alternative to using a bank of single-user receivers is a multi-user receiver. Potentially, this receiver is capable of withstanding higher levels of interference; however, the computational complexity may be high.

The number of codes and the processing gain are limited by the affordable bandwidth reduction. Also, the number of simultaneous receptions is limited by implementation complexity. For a total available bandwidth B, and a processing gain L, the information bandwidth is reduced to B/L. The number of codes is proportional to the code length. For example, if code length of 15 is chosen, there are 17 Gold codes of this length, and 4 Kasami codes. For a code length of 63, there are 65 Gold codes and 8 Kasami codes. This limits the number of clusters that use different codes. However, because the signal attenuates with distance, codes can be reused throughout the network. The concept of code reuse across clusters is analogous to the concept of frequency reuse in cellular systems. Following the reuse condition by which a hexagonal cluster cannot have two neighboring clusters using the same code, it is known that a two-dimensional reuse pattern of N is possible for N that can be expressed in terms of integers i an j as $N=i^2+i\cdot j+j^2$. The minimum number of codes needed to accomplish the condition in a two-dimensional network is 7. Fig. 1 shows a hexagonal reuse pattern of 7. Circles indicate transmission range of a node, which is controlled by the transmission power.

A reuse pattern can also be obtained for a one-dimensional scenario, which would be of interest for network deployment along a strip, such as within a beach or surf-zone environment. Three codes would suffice in this case to keep the clusters apart. Three-dimensional reuse patterns can also be obtained which would serve to allocate resources over a volume of water; however, we focus for simplicity on a two-dimensional case, which is relevant for a fleet of vehicles operating near the bottom, or within the same layer of water.

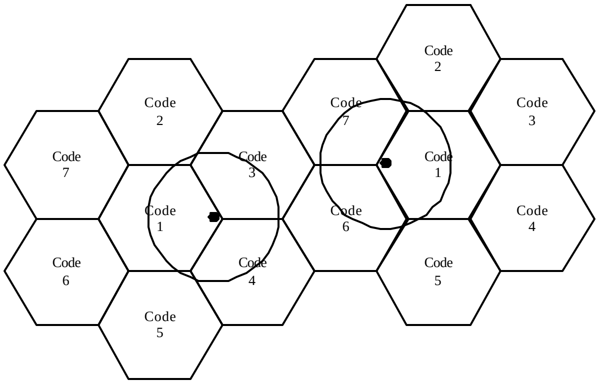


Fig. 1: Code reuse with 7 codes.

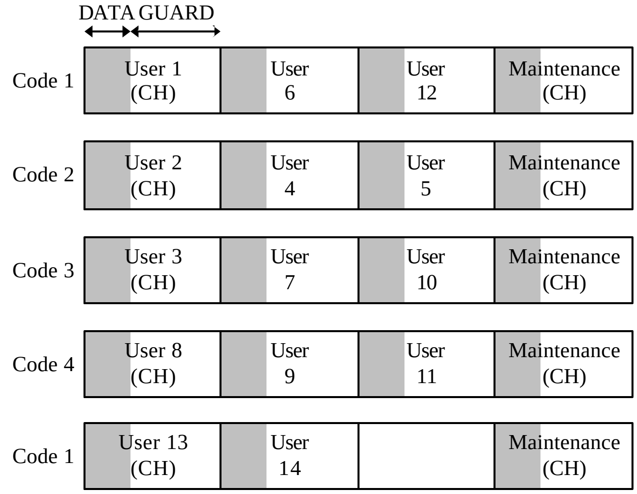
D. Network Operation

The first phase in network operation is initialization. During this phase, network is divided into clusters and time slots and CDMA codes are assigned to each node. The initialization algorithm is presented in Section III. Once the initialization has been completed, nodes begin to exchange information. Nodes must transmit their position and their movement information, and have to relay the information received from other nodes once it has been processed.

TDMA frames are formed by a fixed number of slots, which is equal to the maximal number of nodes allowed to form a cluster. There are two special-purpose slots in each frame: the first and the last slot. While any slot in the frame can be assigned to any node in the cluster, the first one is always assigned to the cluster-head. The last slot is not used for information exchange purposes, but for cluster maintenance. Cluster-heads use this slot to inform the others of changes in cluster formation. Maintenance algorithm and the use of this slot are discussed in Section IV. As the maintenance algorithm must be executed simultaneously at all nodes, if a cluster has fewer nodes than data slots in a frame, the remaining slots are not used. Alternative solutions are possible, which eliminate idle slots; however, we focus for simplicity on the fixed number of slots per frame.

Interconnection between clusters is established by the use of CDMA spreading codes which enable slot reuse. All nodes must be able to detect multiple codes simultaneously. Hence, a node that is within the range of nodes in neighboring clusters, a cluster-connector, can receive simultaneously the information transmitted by the node transmitting in its cluster and by nodes in other clusters. Network operation is half-duplex; hence, during a time slot, a transmitting node cannot receive information from other nodes.

Fig. 2 shows a frame structure for the network partitioning shown in Fig. 3. The maximal number of nodes per cluster in this example is three. The code used by nodes in every cluster is indicated in Fig. 3. Note that the first node to transmit in every frame is the cluster-head and the last slot is reserved for maintenance purposes. In the last cluster, formed by two nodes only, the third slot is not used.



Code 1	User 1 (CH)	User 6	User 12	Maintenance (CH)
Code 2	User 2 (CH)	User 4	User 5	Maintenance (CH)
Code 3	User 3 (CH)	User 7	User 10	Maintenance (CH)
Code 4	User 8 (CH)	User 9	User 11	Maintenance (CH)
Code 1	User 13 (CH)	User 14		Maintenance (CH)

Fig. 2: Frame structure and code assignment. During slot 1, users 1, 2, 3, 8 and 13 transmit simultaneously.

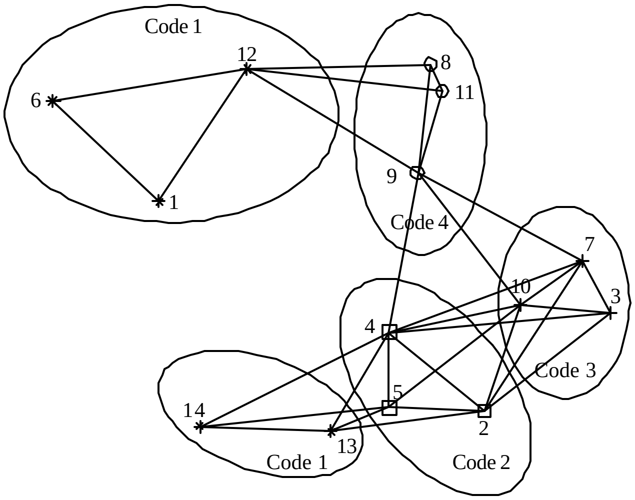


Fig. 3: Network division and resources allocation.

III. NETWORK INITIALIZATION

There are several algorithms used to divide the network into clusters. The most widely used are the lowest-identifier clustering algorithm (LIDCA) and the highest-connectivity clustering algorithm (HCCA) ([3], [4]).

Nodes in the network have a unique identifier. LIDCA organizes the network based on this identifier, giving the role of a cluster-head to the node with the lowest ID in a neighborhood. The operation of HCCA is similar to LIDCA, but it divides the network based on the connectivity of each node, selecting the nodes with highest connectivity – those with most neighbors – as cluster-heads. In this case, the lowest ID rule is only used when more than one node has the same number of neighbors. In both algorithms, every cluster is identified using the ID of its cluster-head.

Both algorithms are based on an operation where nodes first send a packet indicating their position and, depending on it, clusters are built following specific rules.

The clustering algorithm chosen for an underwater mobile network is the lowest ID algorithm. The advantage of this choice is that network partitioning does not depend on the number of node's neighbors, which makes the algorithmmore stable when nodes are mobile.

Upon deployment, nodes transmit their position in a single TDMA frame, with enough power to reach all the other nodes. By doing so, the nodes exchange location information and build a map of all nodes' positions. Alternatively, the nodes can be deployed at pre-specified locations, known to all the network participants. The nodes then compute the cluster division, simultaneously and in a distributed manner, following the lowest ID algorithm, and for a given maximum of cluster members.

Once every node has built the map and identified the cluster division, time slots are allocated. Within a cluster, nodes transmit in order of their ID.

Code assignment is performed next. The assignment algorithm is applied consecutively for all clusters, following the cluster ID order. For every cluster, its neighboring clusters and the neighbors of these clusters are listed and the codes already assigned to them are taken out of the set of spreading codes. Then, the first remaining code in the set is assigned to the cluster. The process continues with the next cluster, until all the clusters have been served.

IV. CLUSTER MAINTENANCE

Because the nodes move, network topology must be continuously updated. Cluster maintenance is based on the position and movement information received from each node. This information contains a stamp of the time when it was generated, so that every node can predict the position of other nodes at the time of network topology upgrade. For purpose of extrapolating positions, the direction is considered not to have changed, that is, nodes are considered to have moved straight. With extrapolated positions, every node builds an updated map of the network. If the information from any other node is too old (i.e., more than a certain number of frames), that node is assumed absent from the network and is not connected to other nodes in the map.

The maintenance algorithm consists of three phases. In the first phase, every node evaluates its position with respect to the rest of the nodes in its cluster, and decides whether it can still belong to the same cluster during the following frame. Mobility may lead to a situation where two clusters using the same code become adjacent to a third cluster, or even to each other. In this situation, one of them must change its transmission code. When this occurs, the cluster with highest cluster ID changes its code. Cluster-heads are in charge of this task during the first phase of maintenance.

In the second phase, cluster-heads transmit the cluster structure they have obtained in previous phase for the next frame. In the third phase, this information is received. Nodes that have not been accepted into the desired cluster and nodes that are alone in their cluster check if they are neighbors of a cluster-head whose cluster has free slots. If this is so, they will transmit in one of the free slots during the following frame, which will be considered by the cluster-head as a query to join the cluster. If there is no such possibility, nodes will have to declare their own cluster and select a code.

Due to a high concentration of nodes in a small region, a situation may arise in which a cluster cannot get a transmission code. Such a situation is illustrated in Fig. 4. Nodes in this situation will try to join a cluster during the next maintenance phase. This situation is similar to that of nodes that join the network after the initialization phase.

V. PERFORMANCE EVALUATION

The performance of the proposed system has been evaluated in simulation. Performance is quantified through measures of connectivity, successful transmission rate, average delay and energy consumption. These measures are studied for varying node density and transmission range. The goal of simulation analysis was to demonstrate system scalability and to provide a design tool for determining the optimal system parameters.

The first question that arises is that of the cluster size. Cluster size is determined by the maximal number of nodes per cluster. For a given node density and maximal number of nodes per cluster, transmission range determines network connectivity. By increasing the transmission range, network connectivity increases, but so does the interference. An increase in interference may result in loss of packets, which effectively lowers the network connectivity. Hence, there is a trade-off in the selection of transmission power. Simulation analysis is used to assess this trade-off, and determine the optimal transmission power for a given system configuration.

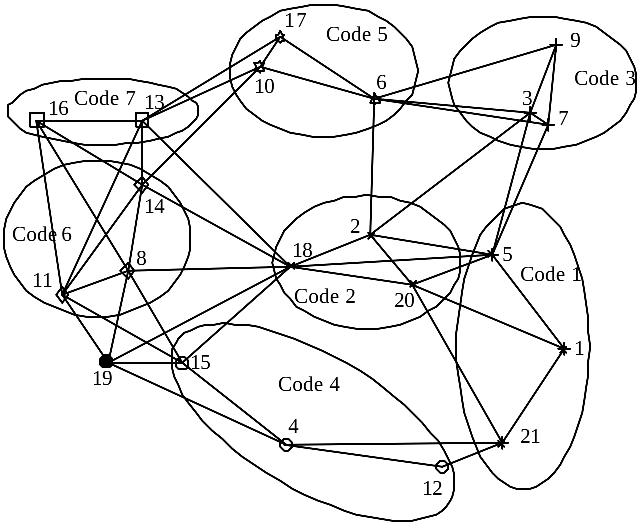


Fig. 4: When all the codes are exhausted among the first and second-level neighbor clusters, a node is left without a code. Node 19 can listen, but it cannot get a code to transmit.

Interference consists of two major components: signals from neighboring clusters that use different spreading codes, and signals from remote clusters that reuse the same code. Interference from neighboring clusters can be reduced by increasing the processing gain; however, for a fixed and limited bandwidth, the information rate then decreases. Trade-offs in system design are many, and often difficult to assess analytically. The effects of selecting the processing gain were evaluated in simulation.

A. *Performance Metrics*

The metrics used to evaluate system performance are the following:

- Network connectivity, defined as the ratio between the number of pairs of nodes that can be connected through a path and the total number of pairs of nodes.
- Delay, calculated as the time that passes between the moment a node transmits its own information and the moment every node successfully receives it. The average delay is the result of averaging over all successful receptions of updated information.
- Energy consumption, evaluated as the total amount of energy used for transmissions during a frame.
- Node connectivity, defined as the percentage of transmitter's neighbors (those nodes to which it is directly connected) that successfully receive a transmission. This value is also studied in absolute terms, that is, total number of neighbors that successfully receive a transmission. A packet is successfully received if the signal-to-interference-plus-noise ratio (SINR) at receiver is above a pre-specified threshold.
- Successful transmission rate, defined as the percentage of successful transmissions for node connectivity over 50%. A transmission is defined successful when its node connectivity is over a certain percentage.

B. *Simulation Scenario*

The area covered by the network is a square of 5 km by 5 km, with a varying number of nodes. Nodes are placed randomly over the area, and are moving at a constant speed of 5 knots (2.5 m/s). On the average, every node changes its direction once every 3 minutes. When a direction change occurs, it can be of 45 or 90 degrees to the left or to the right, according to the probabilities shown in Fig. 5.

A reuse pattern of seven codes is employed, and processing gain of 15 is used for the initial set of simulations. Carrier frequency is 15 kHz, and the system bandwidth is fully utilized at the chip rate of 4,000 chips/second. Packet duration is 4 seconds (1067 bits at 15 chips/bit).

A signal-to-interference-plus-noise ratio (SINR) threshold of 8 dB is used to determine whether a node receives successfully.

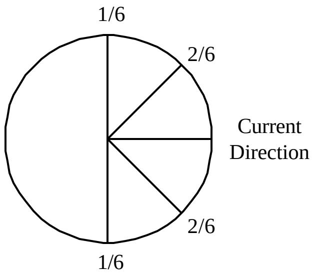


Fig. 5: Direction change probabilities.

For maintenance purposes, a node considers that another node is absent from the network when the last information about that node is older than two TDMA frames.

All values are averaged over ten simulations. The duration of every simulation is twenty network minutes. During this time, each node moves 3 km and transmits 35 times.

C. *Cluster Size*

The optimal cluster size is determined as the number of nodes per cluster for which the network connectivity is maximized. Connectivity is first assessed within a flat network, as a function of transmission range. There is only one cluster in this case, and, hence, no interference. Network connectivity in a flat network of 25 km^2 is shown in Fig. 6 for a total of 30, 40, 50 and 60 nodes, i.e., for four different node densities (1.2, 1.6, 2 and 2.4 nodes/km^2). It can be seen that good connectivity (more than 90%) is obtained for transmission range greater than 1 km with 60 nodes, or 1.2 km with 40 nodes.

To find the optimal number of nodes per cluster, network connectivity after cluster formation is evaluated. For a transmission range of 1.5 km, which provides good flat connectivity for all node densities, network connectivity as a function of cluster size is shown in Fig. 7. Smaller cluster sizes and shorter TDMA frames have an advantage of higher transmission rate. For 30 or 40 nodes, highest connectivity is achieved with cluster size of 6, and we select this value for further analysis.

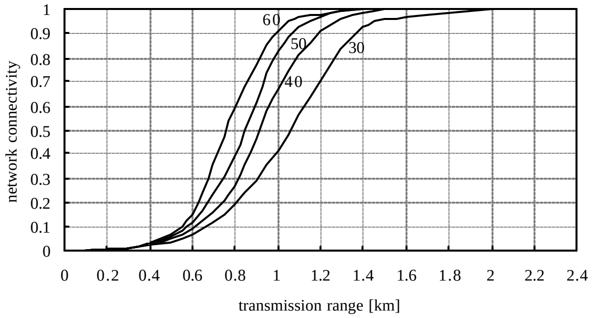


Fig. 6: Network connectivity in a flat network of 25 km^2 with 30, 40, 50 and 60 nodes.

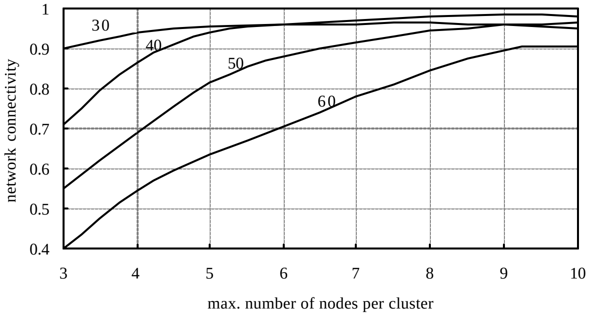


Fig. 7: Network connectivity after cluster formation (25 km^2, transmission range 1.5 km).

D. Average Delay

Delay in the TDMA/CDMA network is caused by two factors: delay on a single hop, which depends on the node's order of transmission, and delay in relaying across multiple hops. Fig. 8 shows the average delay of a successful transmission, as a function of transmission range. Initially, the average delay decreases with increasing transmission range as more direct paths are established. However, with further increase in transmission power, interference increases, causing packets to be lost on direct paths and necessitating alternate routes with more hops.

The number of nodes in the network is also important from the interference point of view. For a greater density of nodes, more simultaneous transmissions occur, resulting in higher interference as compared to that of a lower density network at the same transmission power.

E. Energy Consumption

Fig. 9 shows the total normalized energy consumption for this system. Acoustic transmission loss is calculated as in [1] for the carrier frequency of 15 kHz. Horizontal lines in the plot represent the energy consumed in a flat network of the same dimension (25 km^2) where every node must transmit with sufficient power to reach every other node. In this situation, the transmission range is $5\sqrt{2}$ km, and the packet time is reduced by a factor equal to the processing gain. These lines also represent the energy consumed during the system initialization. Substantial energy savings are obtained with the clustering algorithm.

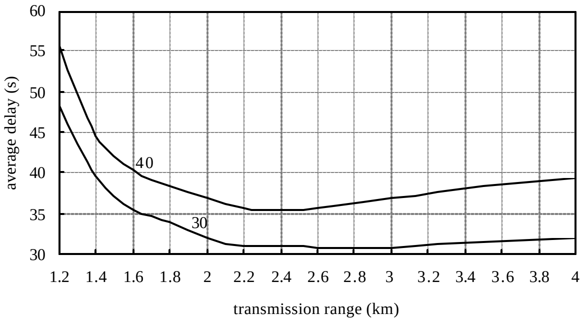


Fig. 8: Average delay (25 km^2, 6 nodes/cluster)

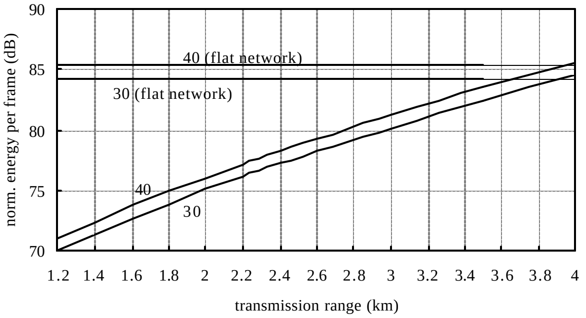


Fig. 9: Total (normalized) energy consumption per frame (25 km^2, 6 nodes/cluster).

F. Node Connectivity and Successful Transmission Rate

Fig. 10 and Fig. 11 show the node connectivity, while Fig. 12 shows the successful transmission range for a node connectivity of 50%.

Results of Fig. 10 show a maximum in performance for transmission range between 1.4 and 1.6 km. For this range, node connectivity is around 35%, which, in absolute terms, means that on average, every transmission is received with a SINR over the threshold by 4 nodes.

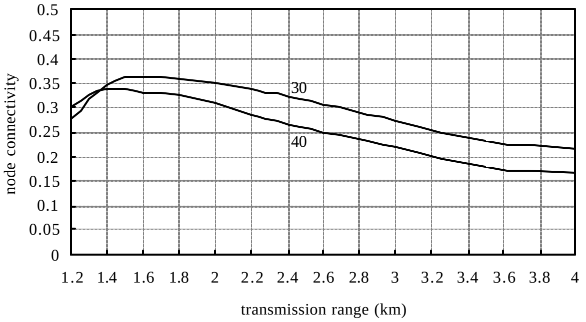


Fig. 10: Node connectivity (25 km^2, 6 nodes/cluster).

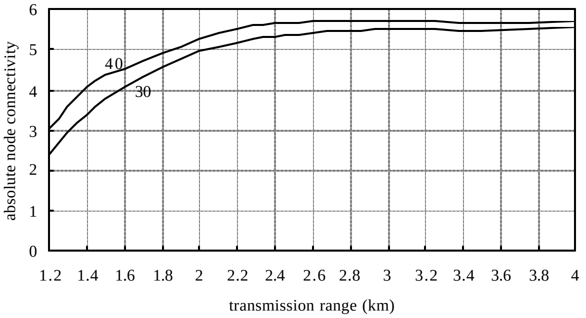


Fig. 11: Absolute node connectivity (25 km^2, 6 nodes/cluster).

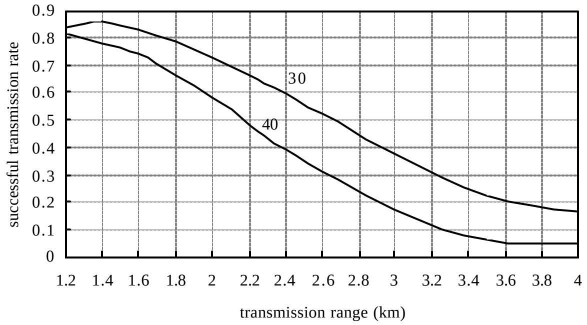


Fig. 12: Successful transmission rate (25 km2, 6 nodes/cluster).

G. *The Effect of Processing Gain*

Selection of the processing gain is governed by the trade-off between interference suppression capability and information throughput within a fixed bandwidth. While the interference reduction obtained by increasing the processing gain can help to increase the connectivity for each node, the data rate is reduced.

Fig. 13 and Fig. 14 show the results of a simulation conducted with the same parameters as before, but with the processing gain changed from 15 to 63. Two cases are considered. In the first one, the amount of data to transmit remains the same (1067 bits per packet) but the packet duration changes (from 4 seconds to 16.8 seconds). In the second case, the packet time remains constant, but due to the bandwidth limitation, the amount of data is reduced to 254 bits per packet. Results show that, for short transmission distances and the same amount of information per packet (curves labeled “packet size”) the gain obtained in node connectivity is small compared to the excessive delay introduced. In the second case, the enhancement obtained in delay is not worth the loss in data rate. For longer transmission ranges, where interference dominates the performance of the system, increasing the processing gain provides substantial enhancement in node connectivity, while keeping the packet size constant does not cause an excessive increase in delay.

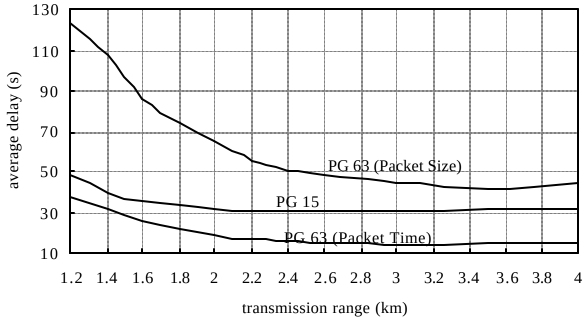


Fig. 13: Average delay for different processing gains.

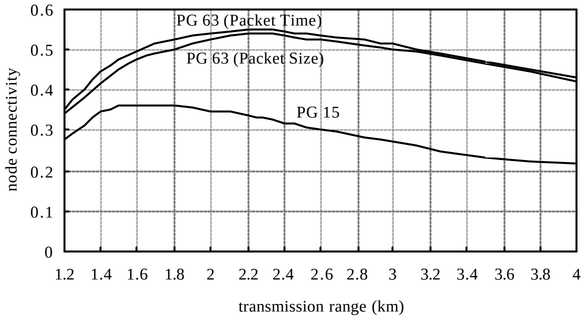


Fig. 14: Node connectivity for different processing gains.

H. *Scalability*

In the preceding sections, network performance was evaluated for a fixed area of 25 km^2, with 30 or 40 nodes. In this section, we study the network performance as both the coverage area and the number of nodes increase. Specifically, we want to show that network performance for a given node density does not deteriorate as the coverage area increases. The same number of nodes per cluster (six) and the same transmission range (1.5 km) as before are used.

Results in Fig. 15 and Fig. 16 demonstrate that as the area of coverage increases, performance indeed does not deteriorate. Node connectivity slightly decreases initially as the coverage area doubles, but reaches a steady state thereafter. The reason for the initial decrease is that code reuse takes effect with additional clusters, i.e. co-channel interference appears.

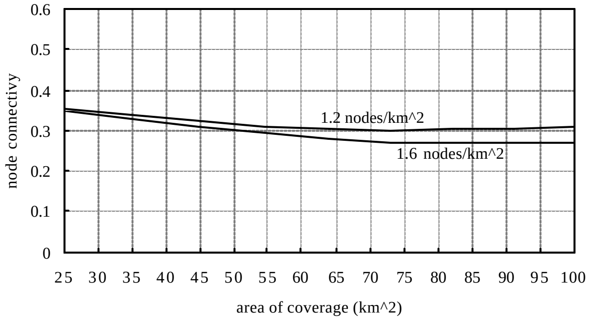


Fig. 15: Node connectivity variation with area of coverage (transmission range 1.5 km, 6 nodes/cluster).

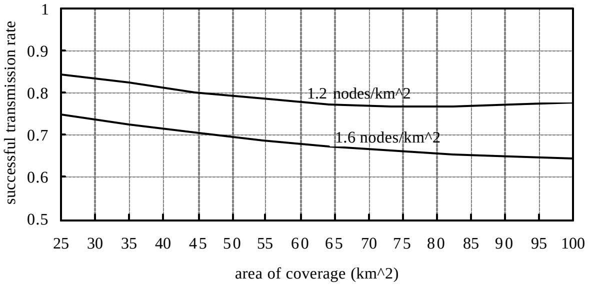


Fig. 16: Successful transmission rate variation with area of coverage (transmission range 1.5 km, 6 nodes/cluster).

VI. CONCLUSIONS

A design for a distributed ad hoc network formed by multiple autonomous underwater vehicles has been proposed. The network is partitioned into clusters, and transmissions in each cluster are scheduled following a TDMA algorithm. CDMA is used to enable spatial reuse of slots throughout the network. Network scalability is attained by reusing CDMA codes in distant clusters. Connectivity between nodes in different clusters is achieved using receivers that are capable of simultaneous detection at multiple spreading codes.

Performance evaluation has been carried out through simulation, and a procedure for network parameter optimization has been outlined. Given the coverage area of the network and the desired node density, optimal cluster size is obtained as the maximal number of nodes per cluster for which the overall network connectivity is maximized. Once this parameter has been determined, optimal transmission range is found by studying various performance metrics (average delay, energy consumption, node connectivity and successful transmission rate). The effect of CDMA processing gain on the network performance was quantified through the node connectivity/delay trade-off. Simulation results demonstrate the network scalability.

The network design and the results presented were focused on multiple access, with particular goal of providing a technique that is scalable, i.e. equally applicable to a network of any size. Future work in this area will focus on analytical assessment of system capacity. Network design will move on to specific routing algorithms that meet the requirements of ad hoc deployable mobile underwater networks.

REFERENCES

[1] M. Stojanovic, L. Freitag, J. Leonard and P. Newman, “A Network Protocol for Multiple AUV Localization,” in Proc. IEEE OCEANS 2002, vol. 1, pp. 604-611, 2002.

[2] C.R. Lin and M. Gerla, “Adaptive Clustering for Mobile Wireless Networks,” IEEE J. Sel. Areas Commun., vol. 15, no. 7, pp. 1265-1275, Sept. 1997.

[3] L. Kai and L. Jiandong, “Mobile Cluster Protocol in Wireless Ad Hoc Networks,” in Proc. Intl. Conf. on Commun. Tech. (WCC - ICCT 2000), Vol. 1, 2000.

[4] M. Gerla and J.T.-C. Tsai, “Multicluster, Mobile, Multimedia Radio Network;” J. on W’less Networks (ACM - Kluwer Academic Publishers), vol. 1, no. 3, pp. 255-265, March 1995.

[5] M. Stojanovic and L. Freitag, “Multiuser Undersea Acoustic Communications in the Presence of Multipath Propagation,” in Proc. IEEE OCEANS 2001, vol. 4, pp. 2165-2169, 2001.