Scheduling Granularity in Underwater Acoustic Networks

Kurtis Kredo II
Electrical and Computer Engineering Dept.
California State University, Chico
kkredo@csuchico.edu

Prasant Mohapatra
Computer Science Department
University of California, Davis
pmohapatra@ucdavis.edu

ABSTRACT

Underwater acoustic networks have many distinct channel characteristics when compared to terrestrial networks. Long propagation delay, one such characteristic, allows scheduling methods with varying granularity that tradeoff schedule quality with protocol overhead. This work examines several channel scheduling methods to determine at what point protocols find the best balance between performance and overhead. To accomplish this, five scheduling options are detailed and then compared through numerical and simulation results between themselves and to other protocols. The results indicate that scheduling links provides the best performance for the resource investment and that other scheduling options either require significant overhead or provide insufficient performance. While the results show that direct sequence spread spectrum techniques, common at the physical layer in underwater networks, do not yield an improved schedule, they do reduce protocol overhead and scheduling complexity by reducing conflicts in the network.

Categories and Subject Descriptors

C.2.5 [**Computer-Communication Networks**]: Local and Wide-Area Networks—*Access schemes*; C.4 [**Computer Systems Organization**]: Performance of Systems

General Terms

Design, Performance

Keywords

underwater acoustic networks, channel scheduling, medium access control

1. INTRODUCTION

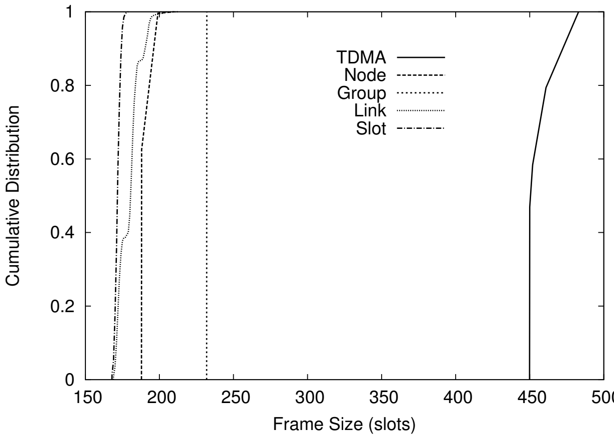
Underwater acoustic networks present new opportunities in environmental sensing, resource management, and security, but complex channel conditions and limited energy resources present challenges to researchers and application developers. One fruitful area of research in underwater acoustic networks lies in medium access control (MAC) protocols, which directly affect energy use and the available data rate by controlling when and how devices use their transceiver. Transceivers often consume significant energy, so MAC protocols are ideally located to balance the tradeoff between service quality and resource requirements.

Two main categories of MAC protocols are scheduled and unscheduled protocols. Unscheduled protocols have a simple implementation that requires minimal protocol overhead, but suffer from increased collisions or only support low data rates. Proposed unscheduled protocols include adaptations to CSMA/CA [17, 1, 6, 12, 15] and ALOHA [19, 2], loosely scheduled protocols [13, 20], and CDMA techniques [16, 4]. Scheduled protocols provide the potential for higher data rates and reduced energy losses from collisions and idle listening, but require synchronization protocols and schedule formation overhead. Researchers have proposed several protocols that attempt to limit scheduled protocol overhead while also providing improved performance [9, 8, 5, 11, 7].

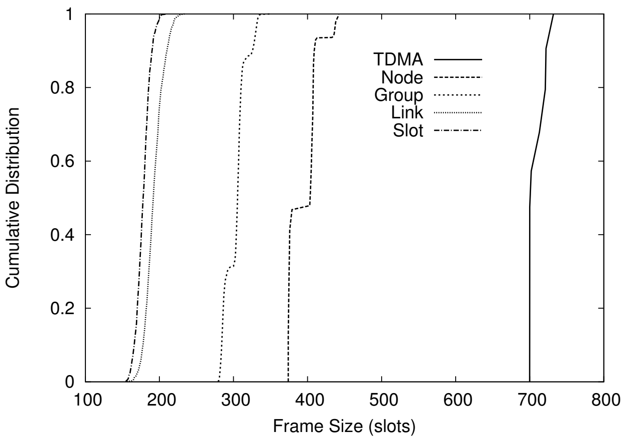
Scheduled protocols are particularly well suited for underwater networks since they can leverage the long propagation delays inherent in acoustic communication to support multiple simultaneous transmissions without collisions at the receiver, but they must select a scheduling granularity. Schedule granularity determines how devices use transmission opportunities and to whom they transmit at each opportunity. This paper studies the spectrum of scheduling methods in stationary underwater wireless networks that require high data rates by examining five scheduling methods of differing granularity: TDMA, with the coarsest scheduling granularity and smallest overhead, Node, Group, Link, and Slot, with the finest scheduling granularity and largest overhead. At each point along the spectrum there exists a tradeoff between schedule quality (efficiency, throughput, latency, or other metric) and the overhead required for scheduling and state distribution. Finding the correct scheduling granularity ensures MAC protocols achieve the best performance for the given investment in energy consumption and overhead.

Section 2 illustrates the scheduling spectrum for underwater networks and presents the five scheduling methods under investigation. The scheduling problem and related algorithms are described in Section 3. Numerical results are provided in Section 4 comparing the various scheduling methods and Section 5 continues the evaluation, through simulation, by considering metrics unavailable through numerical modeling. Lastly, Section 6 concludes the paper.

WUWNet'11 December 1–2, 2011, Seattle, Washington, USA.



(a) RF Network



(b) Underwater Network

Figure 1: Frame Size CDF.

2. CHANNEL SCHEDULING SPECTRUM

To illustrate the effectiveness of the scheduling methods and their capability to leverage long propagation delay, consider the cumulative distribution functions (CDFs) of frame size for each method, as shown in Figure 1.

Figure 1(a) shows the frame size CDF for the scheduling methods when used in a terrestrial wireless network. Since terrestrial networks have negligible propagation delay, there is little difference in frame size between Node, Link, and Slot scheduling, but a significant difference between those methods and TDMA scheduling. Group scheduling performs poorly due to assumptions that hold in underwater acoustic networks, but are invalid for terrestrial networks.

In contrast, Figure 1(b) contains the frame size CDF for the scheduling methods in an underwater acoustic network. Now the scheduling methods become more distinct in the opportunities they provide for improved scheduling due to the long propagation delays present in underwater networks.

Improved scheduling enhances network functionality in several ways. First, it may improve some desirable metric, such as throughput or latency, since the additional information about when objects are scheduled and how they are related through propagation delay allows more successful transmissions within a given time period. Additionally, with detailed information about when specific objects are scheduled and who they communicate with, devices can increase the amount of time they remain in sleep mode, which increases network lifetime. In these ways, proper scheduling yields a multifaceted improvement to device performance.

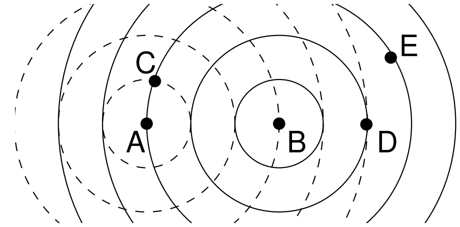


Figure 2: Example Network.

However, the improved schedules come at a cost. As the schedule becomes more specialized the devices have less flexibility in how they may use their transmission opportunities. For example, a schedule that provides node a with 10 slots of time within a frame allows node a to transmit to any neighbor, in any order. However, a schedule that provides link j with 10 slots of time provides much less flexibility on how the device may use link j. If no data is available for transmission over link j, then those time slots must remain idle as other transmissions might cause a collision. A specialized schedule also requires additional state information. Scheduling nodes to transmit for a fixed, known time period requires only the time slot assignment for each device to be distributed through the network, but scheduling links with a variable duration requires each device to share multiple slot assignments and durations. The additional state distribution consumes energy and channel resources that could be used for application data.

2.1 Five Scheduling Methods

Each scheduling method provides a different balance between state distribution overhead and schedule quality. This section introduces five separate schedule points along the spectrum, which later sections develop further. The following explanations use Figure 2, an example underwater network showing signal propagation at time slot intervals, and Figure 3, the associated data frame for each scheduling method. In the following examples using Figure 2, node A has data to send to node C and node B has data to send to node D.

2.1.1 TDMA Scheduling

At one extreme lies traditional TDMA; the simplest, but least efficient scheduling method. With TDMA, nodes utilize a fixed-sized set of slots within a frame that is long enough for a packet transmission and the propagation delay to the maximum interference range. In terrestrial networks the propagation delay is negligible or constitutes a minimal overhead, but underwater acoustic communication results in significant propagation delays for non-trivial node distances. As a result, TDMA time slot assignments require significant overhead to accommodate the propagation delay. Since the reservation duration for each node is fixed, nodes only need to distribute basic topology information and their time slot assignments. From Figure 3, nodes A and B are each assigned 10 slots for transmission and signal propagation, resulting in a frame of 20 slots.

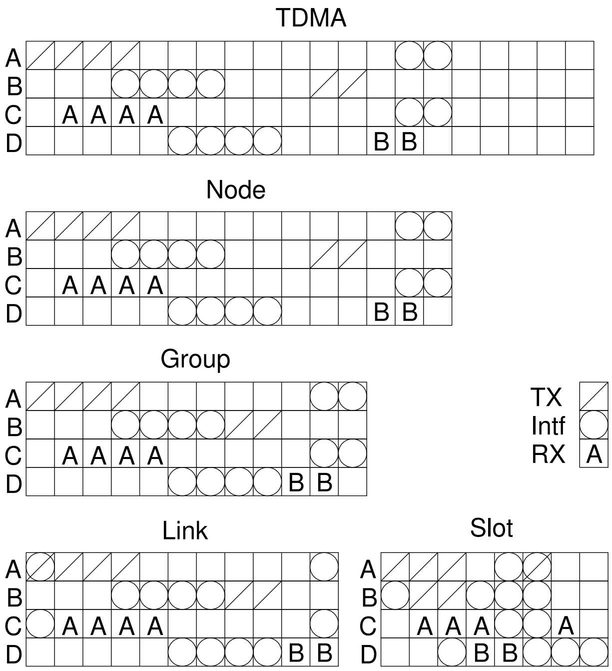


Figure 3: Example Frames.

2.1.2 Node Scheduling

To limit the effect of propagation delay overhead and enable traffic-adaptive scheduling, nodes may utilize varying-sized reservations with Node scheduling. Nodes now determine their transmission duration and the delay to the farthest node to which they cause interference and use this information to reserve time in the schedule. This results in a tighter schedule, but comes at the cost of the overhead of sharing duration and propagation delay (or position information) among neighboring nodes. For example, in Figure 2 node B only requires a transmission time of two slots and a guard period of three time slots, while node A requires 10 slots in total, the same value required with TDMA scheduling. Figure 3 shows how Node scheduling is able to reduce the frame size to 15 slots using the additional traffic and topology information.

2.1.3 Group Scheduling

Further improvements are made with Group scheduling. In this scheduling mechanism, nodes divide their neighbors into groups based on propagation delay estimates and nodes with similar propagation delays are grouped and scheduled together. Devices can now build a schedule that overlaps transmissions so collisions only occur at nodes that are not meant to receive any of the colliding packets. From Figure 2, node B can transmit after the signal from node A has passed it as there is no possibility of collision with packets from node A after that time. Group scheduling is able to use the more specific destination information to reduce the frame size to 12 slots in Figure 3. Nodes do not have to distribute propagation delay estimates to each group if a fixed ring size is used at all nodes. However, nodes must now distribute schedule information about rings, which increases the schedule variables in proportion to the number of occupied rings per node.

2.1.4 Link Scheduling

Schedule refinement continues with Link scheduling, where individual unicast links are scheduled once in each frame. A scheduling algorithm may now take individual propagation delays into account to overlap communications to a larger extent. In Figure 2, node B with Group scheduling reserves transmission time to ring 2 containing nodes A, C, and E at time slot $s_{B,2}$, so nodes A, C, and E cannot transmit in slot $s_{B,2}+3$ as they might receive a packet from node B starting in that slot. However, with link scheduling node B schedules transmissions to the nodes separately, so node A may schedule a transmission to node C while node E receives a packet from node B. With link scheduling, the nodes can use the improved schedule because their packets do not collide at the intended destinations. Figure 3 shows how Link scheduling reduces the frame size to 11 slots. The overhead for Link scheduling grows with the number of utilized and interfering links.

2.1.5 Slot Scheduling

Finally, Slot scheduling allows for the tightest schedule generation by allowing links to transmit multiple times per frame. Figure 3 shows how Slot scheduling yields a smaller schedule than Link scheduling for the links (A, C) and (B, D) from Figure 2. By breaking up the transmission over link (A, C), Slot scheduling yields a frame size of 8 slots.

2.2 Schedule State Requirements

One disadvantage of tighter scheduling is the additional state that must be distributed for conflict-free operation. Table 1 details the state required for each of the scheduling methods. TDMA scheduling only requires that devices distribute their slot assignment as all nodes use a fixed, large duration that covers the maximum duration and propagation delay of any node. Node scheduling requires the additional state variables of transmission duration and device propagation in order to optimize the time each device reserves the channel. Group scheduling further refines the schedule, but requires the distribution of ring numbers and requires that each node maintain multiple assignments. State overhead in Group scheduling grows linearly with the number of groups present in the network. Link scheduling requires approximately the same state as Group scheduling for sparse networks, but requires additional state distribution for dense networks, where there are likely to be significantly more than one neighbor per group. Finally, Slot scheduling requires the most state since nodes maintain and distribute schedule information for every transmitting slot in the frame. As expected, the state overhead for Slot scheduling grows rapidly.

3. SCHEDULING PROBLEM

Each scheduling method yields different transmission opportunities for devices, but they all must ensure destinations receive their packets without collision. A scheduling algorithm prevents collisions by creating a schedule that judiciously selects when devices transmit their packets.

A schedule consists of an assignment of transmission time slots within a frame for each schedule object, where a schedule object may be a node or a link depending on the scheduling method. Every schedule element e receives a starting slot s_e and a duration Δ_e, where the durations are provided by

		State		
Class	**Characteristics**	**S**	**D**	**P**
TDMA	Fixed-length blocks	×		
Node	Variable-length blocks	×	×	×
Group	Group neighbors into rings	×	×	×
Link	Single link transmission	×	×	×
Slot	Multiple link transmissions	×	×	×

State	Description
S	Slot for each element
D	Duration for each element
P	Propagation delay or ring number

Table 1: Scheduling Method State.

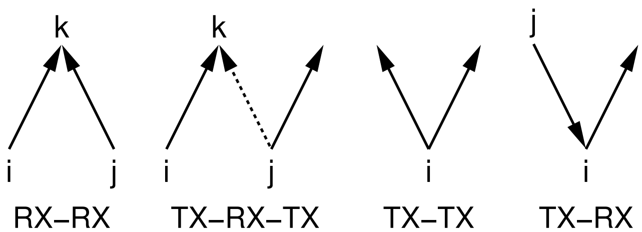


Figure 4: Schedule Conflicts.

the routing layer. The collection of transmit times $\mathcal{S}$, durations Δ, and m, the frame size measured in time slots, define the schedule. Frames are repeated with the same schedule until application or routing changes require the nodes to find a new schedule.

Schedules avoid collisions by resolving network conflicts present in the network topology. Two or more packets collide, and are lost, if they overlap in time at a node and the node is the intended receiver for any of the packets. As shown in Figure 4, there are four types of network conflicts considered: RX-RX, TX-RX, TX-TX, and TX-RX-TX. These conflicts arise from devices having single half-duplex radios capable of single packet reception. Direct sequence spread spectrum (DSSS) techniques are commonly used in underwater networks to eliminate or reduce complications at the physical layer. If devices use DSSS, then TX-RX-TX conflicts do not exist in the network since the destination can, in general, receive the intended packet amid concurrent transmissions. However, a schedule is still required to prevent the other conflicts as devices can only receive one packet at a time.

Network conflicts are resolved by placing constraints on the transmit times of schedule elements, where each network conflict results in a schedule constraint. Each constraint has an associated ordering variable that determines in which order events, such as transmissions and receptions, occur. For example, the ordering variable for the TX-RX conflict from Figure 4 determines if the packet from node j arrives before node i transmits or if node i transmits before receiving the packet from node j. All schedule constraints take the form

$$\underline{B}_{ij} - m \leq s_j - s_i - m o_{ij} \leq \overline{B}_{ij}$$

where $\underline{B}_{ij}$ and $\overline{B}_{ij}$ are the schedule bounds for the conflict between i and j and m is the frame size.

Given the set of schedule constraints and the traffic demand, nodes must solve for a valid schedule. Previous work [9, 10] discussed several ways to find the schedule, both in a centralized and distributed manner. Briefly, devices determine conflicts in the network through local topology discovery, receive duration requirements from routing layer inputs, and estimate propagation delay to neighboring nodes through a synchronization protocol [3, 18]. Devices then set the ordering variables for all conflicts by giving higher priority to schedule elements with longer transmit durations, with ties broken by element number. The resulting set of linear inequalities can be easily solved in a distributed fashion using a Bellman-Ford algorithm.

Optimal schedules can be found by defining an appropriate objective function and using numerical software to solve the integer linear programming problem. Section 4 includes optimal solutions for the scheduling methods using objective functions that minimize the frame size and minimize traffic latency from nodes to a sink.

3.1 Scheduling Constraints

With an understanding of the various scheduling methods and how schedules are found in the network, schedule constraints that prevent collisions in the network are required. The TDMA scheduling constraints are briefly derived and previous work [9, 10] derives the Link constraints. Derivation for the other scheduling constraints are similar, so the details are not included for brevity.

3.1.1 TDMA Scheduling

TDMA scheduling simply uses a large, fixed set of time slots to prevent collisions. Each conflicting node is assigned a set of non-overlapping time slots and the guard periods included in the time slot assignment prevents collisions.

Define Λ, the size of the fixed time slot set, as

$$\Lambda = \max_{a} \left\{ \Delta_a + \left\lceil \frac{p_a}{T} \right\rceil \right\} \qquad \forall a \in \mathcal{N}$$

where p_a is the maximum propagation delay of node a, T is the time slot length, and $\mathcal{N}$ is the set of all nodes.

Within each frame node a starts to transmit in slot s_a and node b in slot s_b. If node a transmits first in the current frame, then

$$s_b \geq s_a + \Lambda.$$

Similarly, in the next frame node a must transmit after node b has finished, so

$$s_a + m \geq s_b + \Lambda.$$

If $o_{ab} = 1$ in this case, and combining a similar set of inequalities for when node b transmits first within the frame ($o_{ab} = 0$), the TDMA schedule constraint becomes

$$\Lambda - m \leq s_b - s_a - m o_{ab} \leq -\Lambda$$

3.1.2 Node Scheduling

Node scheduling is similar to TDMA scheduling, but reserves a unique time slot assignment for each device. The time period for scheduling a node a, Λ_a, equals

$$\Lambda_a = \Delta_a + \frac{p_a}{T} \qquad \forall a \in \mathcal{N}$$

The Node schedule constraint equals

$$\Lambda_a - m \leq s_b - s_a - m o_{ab} \leq -\Lambda_b$$

3.1.3 Group Scheduling

Group scheduling handles several neighbors at the same time when all the neighbors have similar propagation delays. By treating a group of neighbors as a single element, a device can improve scheduling performance. Devices group neighbors into "rings" based on

$$r_{a,b} = \left\lfloor \frac{p_{(a,b)}}{T} \right\rfloor$$

where $p_{(a,b)}$ is the propagation delay from node a to node b and $r_{a,b}$ is the ring of node b relative to node a. Note that $r_{a,b}$ may not equal $r_{b,a}$ due to non-symmetric propagation paths. Ring scheduling assigns devices multiple transmit times per frame, so TX-TX conflicts exist between transmission times of separate rings. For simplicity, rings have fixed size and a node transmits to it's rings contiguously, starting with the furthest (highest numbered) ring and continuing toward closer rings. In this work a ring is wide enough for an acoustic signal to propagate in one time slot.

TX-TX Constraints.

First consider the order in which a device transmits to its rings. In order to limit the time nodes spend idle, each device transmits to nodes sequentially, starting from the furthest occupied ring and moving toward the closest ring.

The TX-TX constraint for Ring scheduling is

$$\Delta_{a,r^-} - m \leq s_{a,r} - s_{a,r^-} - mo_{ar^-,ar} \leq \Delta_{a,r^-} - m$$

where r^- is the next occupied ring closer to node a than ring r and $\Delta_{a,r}$ is the duration for which node a transmits to ring r. Note that since we assign rings contiguous slots for each device, the upper and lower bounds for the TX-TX constraint are identical (the constraint yields equality).

Other Constraints.

For constraints between nodes, Group scheduling uses

$$\Delta_a + \min\left\{r_a^I - r_b, r_{a,b}\right\} + 1 - m \leq s_b - s_a - mo_{a,b} \leq -\Delta_b + \min\left\{r_a^I, r_{a,b}\right\} - 2r_b - 1$$

where r_a^I is the furthest ring in which node a causes interference and r_b is the ring under consideration to which node b transmits.

3.1.4 Link Scheduling

Link scheduling further refines the schedule by dealing with individual unicast links. These are the same constraints from previous work [10] and are repeated here for reference.

TX-TX Constraints.

$$\Delta_i - m \leq s_j - s_i - mo_{ij} \leq -\Delta_j$$

TX-RX Constraints.

$$\Delta_i - \frac{p_j}{T} - m \leq s_j - s_i - mo_{ij} \leq -\Delta_j - \frac{p_j}{T}$$

RX-RX Constraints.

$$\Delta_i + \frac{p_i - p_j}{T} - m \leq s_j - s_i - mo_{ij} \leq \frac{p_i - p_j}{T} - \Delta_j$$

TX-RX-TX Constraints.

$$\Delta_i + \frac{p_i - p_{(j_s,i_d)}}{T} - m \leq s_j - s_i - mo_{ij} \leq \frac{p_i - p_{(j_s,i_d)}}{T} - \Delta_j$$

3.1.5 Slot Scheduling

Slot scheduling uses the same constraints as Link scheduling, but now each individual slot of a link's duration is scheduled separately.

4. NUMERICAL EVALUATION

The paper first examines the effectiveness of the scheduling methods by solving the scheduling problem for each method using an integer linear programming solver. These results enable us to look at the basic features of the scheduling methods and make initial comparisons between them and to protocols presented in the literature.

The scheduling methods are evaluated on networks where nodes are placed on a grid with random perturbations around their indented point to simulate placement error and a sink is positioned at the network center. The grid width, or average node separation, is 3500m to simulate sparse networks.

4.1 Throughput and Latency

First, consider the throughput provided by the scheduling methods. Figure 5 shows how each scheduling method's throughput changes as synchronization error increases. In this section, throughput is measured as the number of time slots spent by the sink in transmission or reception divided by the frame size. Slot and Link scheduling yield similar performance, which is expected considering Figure 1(b). Figure 5 also indicates that the scheduling methods tolerate synchronization error well. Synchronization error affects Slot and Link scheduling the most as those methods have a minimum number of idle time slots between scheduling events and any increase to accommodate errors quickly consumes a significant portion of the frame. However, even at $\sigma = 0.5s$, meaning nodes' clocks may differ by up to one second, the throughput of Link and Slot scheduling only decreases by 22% and still outperforms other methods by a large margin. Figure 5 also indicates the optimal throughput possible using ALOHA. TDMA and Node scheduling do not achieve the optimal throughput of ALOHA, but they prevent collisions, which may make them more effective for underwater networks with stringent energy resources.

Next, consider the latency of packets generated within the network destined for the sink, where latency is measured as the difference between the transmission time of the last schedule element and the first schedule element on a path. Figure 6 shows how the uplink latency varies with synchronization error for the scheduling methods. Similar to throughput, uplink latency improves when using additional information to determine a schedule.

4.2 Performance with DSSS

Finally, consider the operation of the scheduling methods when using direct sequence spread spectrum (DSSS) tech-

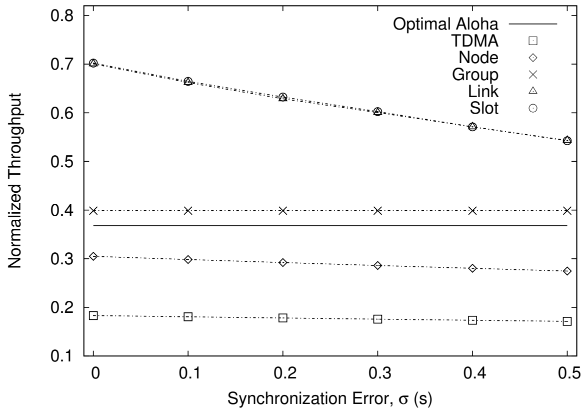


Figure 5: Normalized Throughput.

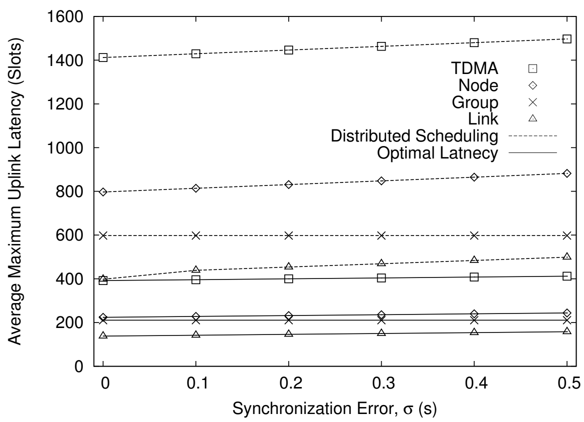


Figure 6: Average Maximum Uplink Latency.

niques. Figure 7 illustrates how the throughput varies when using DSSS with various spreading factors. First notice that DSSS provides no benefit when $SF > 1$ as the increase in transmission time cancels any benefit achieved by removing TX-RX-TX conflicts. Figure 7 also includes results for ST-MAC [8], another scheduled protocol proposed for underwater networks. Figure 7 shows that the improvement of ST-MAC over Slot and Link scheduling, present without DSSS, disappears when using DSSS. ST-MAC achieves a slightly higher throughput without DSSS, but requires a centralized scheduler, which would scale poorly to large networks and consume significant overhead. The scheduling methods proposed here are easily implemented by a distributed algorithm [10].

4.3 Summary

The results in this section indicate that underwater networks yield a wide spectrum of choices for scheduling granularity. Each point along the spectrum offers a different tradeoff between performance and overhead. Based on the numerical results, Slot scheduling offers a marginal improvement over Link scheduling, so the significant increase in overhead when using Slot scheduling is not a prudent investment. Similarly, the small reduction in overhead provided by TDMA when compared to Node scheduling, is unlikely to compensate for the significant decrease in performance.

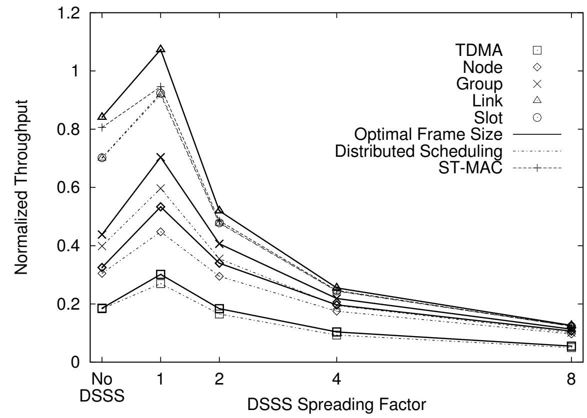


Figure 7: Normalized Throughput using DSSS.

Therefore, TDMA and Slot scheduling appear to be poor choices for scheduling methods and are not investigated in later sections.

5. SIMULATION RESULTS

The numerical results in the previous section provide valuable insight into the scheduling options, but have some limitations that are best resolved through simulation. Two aspects of significant importance are the energy consumption of the scheduling methods and the operation of the distributed scheduling algorithms. This section presents results from running the scheduling methods on a simulator to measure and compare these system characteristics.

The simulation results were collected using the same protocols on the same topologies as the numerical results. Devices generate traffic for the sink randomly based on a Poisson process where data packet sizes are fixed during the simulation. Simulation results also compare the scheduling methods to two random access protocols: an ALOHA protocol modified for underwater networks [14] and Tone Lohi [19].

5.1 Efficiency

First, consider the energy efficiency of the protocols. Efficiency is measured as the number of application data bits delivered to the sink divided by the energy consumed by all non-sink nodes. Figure 8 shows the efficiency of the protocols as traffic load varies. Since the proposed scheduling methods overlap communications while preventing collisions, they achieve high efficiencies and, similar to the numerical results, using detailed information to enhance scheduling leads to improved performance. Notice that the random access protocols achieve low efficiency, which is caused by the large energy losses from collisions. These results include the energy overhead of schedule state distribution and synchronization requirements for the scheduled protocols, but the benefits of improved scheduling greatly outweigh these costs.

5.2 Throughput and Latency

Next, consider the traffic metrics of throughput and latency, where throughput is the number of data bits received at the sink divided by the simulation time and latency is measured from packet creation to its reception at the sink

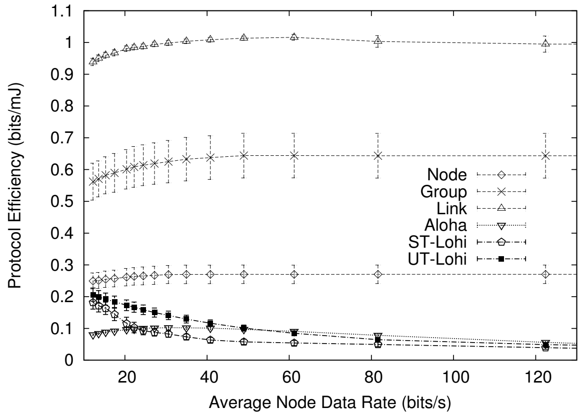


Figure 8: Network Efficiency.

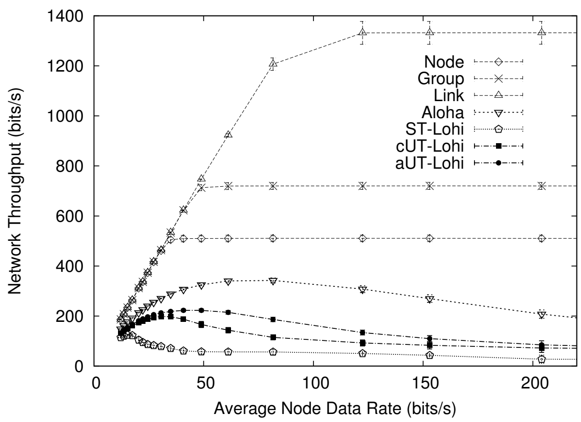


Figure 9: Network Throughput.

(including queueing delays). Figure 9 shows the throughput as the traffic rate varies. The random access protocols reach a much lower maximum throughput at lower data rates than the scheduled methods. Similar to previous results, Link scheduling yields much better performance than Group or Ring scheduling.

Figure 10 illustrates the traffic latency on the network, but it may seem unclear which protocols perform the best from these results. The latency of ALOHA and aUT-Lohi are influenced by selection bias as few packets arrive successfully at the sink and those that do arrive are those that have traversed one hop. Therefore, these protocols would find limited usefulness to users at significant data rates.

5.3 DSSS

DSSS provides many benefits at the physical layer and removes all TX-RX-TX conflicts, but may not provide an overall benefit at the MAC layer, as seen in Table 2 and the previous numerical results. Table 2 lists the frame sizes for Link scheduling as the spreading factor varies. When $SF > 1$, the increase in transmission time outweighs the benefit of removing the TX-RX-TX conflicts.

However, one benefit of DSSS is that devices can limit the amount of information they must share to compute a valid schedule. Since DSSS prevents TX-RX-TX collisions, schedule elements conflict less often, which simplifies scheduling.

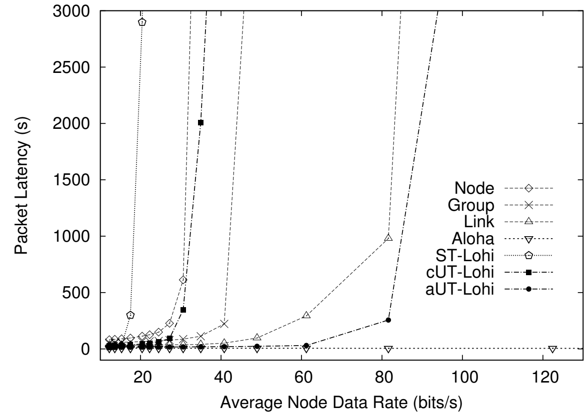


Figure 10: Network Latency.

Spreading Factor, SF	Average Frame Size (slots)
No DSSS	128.2
1	86.1
2	163.5
4	314.1
8	610.8

Table 2: Link Scheduling Average Frame Size.

Table 3 details the size of distributed state packets for the scheduled protocols. For each scheduling method, DSSS reduces by approximately half the amount of state each node must share. Table 3 also shows that the specialized scheduling methods share additional information when compared to the more general methods, but, as seen previously, this information yields in better schedules and higher performance.

5.4 Scheduling Convergence

As detailed in previous work [10], the scheduled protocols must converge on a single schedule before any device uses the schedule to prevent collisions. Determining the length of the scheduling convergence periods must balance the time required for the protocols to stabilize with the desire to adapt quickly to changing network conditions. Table 4 lists the number of frames required for each of the scheduled methods to stabilize, defined as the Epoch. During each Epoch, devices use fixed schedules and routes while distributing state information and calculating the schedule and routes for the next Epoch. More specific scheduling methods require fewer state packet transmissions to stabilize both on average and in the maximum. The reduced state distribution when using DSSS also results in faster scheduling convergence. The results indicate that the distributed scheduling algorithms converge quickly, allowing the schedule to adapt quickly to network changes.

	No DSSS		DSSS	
Scheduling Method	**Mean**	**Max**	**Mean**	**Max**
Node	364.9	848	165.0	464
Group	378.0	1040	178.7	656
Link	347.8	1176	146.5	400

Table 3: Schedule State Size in bits.

	No DSSS		DSSS	
Scheduling Method	**Mean**	**Max**	**Mean**	**Max**
Node	17.4	22	8.2	12
Group	15.2	19	8.4	13
Link	12.2	17	7.6	10

Table 4: Epoch Size in Frames.

6. CONCLUSION

The long propagation delays present in underwater networks offers many challenges to network designers as it invalidates many traditional approaches. However, these delays also provide additional opportunities not available in RF networks. Numerical and simulation results explored these opportunities by considering the spectrum of scheduling options through five specific points along the spectrum: TDMA, Node, Group, Link, and Slot. Each option provides a different tradeoff between the amount of state information that must be distributed and the improvement in scheduling.

This paper showed that the scheduling methods presented here perform well in underwater networks. The numerical studies illustrated that TDMA, on the low overhead end of the scheduling spectrum, provided poor performance due to the large guard periods required to prevent collisions. On the other end, Slot scheduling provided little benefit over Link scheduling, so it would likely not be a good choice for actual use due to the scheduling complexity.

Using the three likely candidates for scheduling in underwater networks, Node, Group, and Link scheduling, simulations provided further evaluation. Simulation results showed that Link scheduling performed the best by achieving the highest throughput, highest efficiency, and lowest latency of the scheduled methods evaluated. The additional overhead of Link scheduling compared to Node and Group scheduling was an acceptable tradeoff for the performance improvement. Further results showed that DSSS provides no benefit to network performance metrics (throughput, latency, efficiency), but it does allow nodes to reduce the state they must distribute and decreases scheduling convergence time.

The combined results show that scheduling links in underwater acoustic networks provides the best performance for the given resource investment and indicates scheduled MAC protocols should focus on this point of the scheduling spectrum.

7. REFERENCES

[1] N. Chirdchoo, W. seng Soh, and K. Chua. RIPT: a Receiver-Initiated Reservation-Based protocol for underwater acoustic networks. *IEEE J. Sel. Areas Commun.*, 26(9):1744–1753, Dec. 2008.

[2] N. Chirdchoo, W. Soh, and K. C. Chua. Aloha-Based MAC protocols with collision avoidance for underwater acoustic networks. In *IEEE INFOCOM*, pages 2271–2275, May 2007.

[3] N. Chirdchoo, W. Soh, and K. C. Chua. MU-Sync: a time synchronization protocol for underwater mobile networks. In *ACM WUWNet*, pages 35–42, Sept. 2008.

[4] R. Diamant and L. Lampe. A hybrid spatial reuse MAC protocol for Ad-Hoc underwater acoustic communication networks. In *IEEE ICC*, May 2010.

[5] Y. Guan, C. Shen, and J. Yackoski. MAC scheduling for high throughput underwater acoustic networks. In *IEEE WCNC*, pages 197–202, Mar. 2011.

[6] X. Guo, M. R. Frater, and M. J. Ryan. Design of a Propagation-Delay-Tolerant MAC protocol for underwater acoustic sensor networks. *IEEE J. Oceanic Eng.*, 34(2):170–180, Apr. 2009.

[7] L. Hong, F. Hong, Z. Guo, and Z. Li. ECS: efficient communication scheduling for underwater sensor networks. *Sensors*, 11(3):2920–2938, Mar. 2011.

[8] C. Hsu, K. Lai, C. Chou, and K. C. Lin. ST-MAC: Spatial-Temporal MAC scheduling for underwater sensor networks. In *IEEE INFOCOM*, pages 1827–1835, Apr. 2009.

[9] K. Kredo II, P. Djukic, and P. Mohapatra. STUMP: exploiting position diversity in the staggered TDMA underwater MAC protocol. In *IEEE INFOCOM Miniconference*, pages 2961–2965, Apr. 2009.

[10] K. Kredo II and P. Mohapatra. Distributed scheduling and routing in underwater wireless networks. In *IEEE GLOBECOM*, Dec. 2010.

[11] Z. Li, Z. Guo, H. Qu, F. Hong, P. Chen, and M. Yang. UD-TDMA: a distributed TDMA protocol for underwater acoustic sensor network. In *IEEE MASS*, pages 918–923, Oct. 2009.

[12] M. Molins and M. Stojanovic. Slotted FAMA: a MAC protocol for underwater acoustic networks. In *IEEE OCEANS*, May 2007.

[13] M. K. Park and V. Rodoplu. UWAN-MAC: an Energy-Efficient MAC protocol for underwater acoustic wireless sensor networks. *IEEE J. Oceanic Eng.*, 32(3):710–720, July 2007.

[14] N. Parrish, L. Tracy, S. Roy, P. Arabshahi, and W. L. J. Fox. System design considerations for undersea networks: Link and multiple access protocols. *IEEE J. Sel. Areas Commun.*, 26(9):1720–1730, Dec. 2008.

[15] B. Peleato and M. Stojanovic. Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks. *IEEE Commun. Lett.*, 11(12):1025–1027, Dec. 2007.

[16] D. Pompili, T. Melodia, and I. F. Akyildiz. A CDMA-based medium access control for UnderWater acoustic sensor networks. *IEEE Trans. Wireless Commun.*, 8(4):1899–1909, Apr. 2009.

[17] E. M. Sozer, M. Stojanovic, and J. G. Proakis. Underwater acoustic networks. *IEEE J. Oceanic Eng.*, 25(1):72–83, Jan. 2000.

[18] A. A. Syed and J. Heidemann. Time synchronization for high latency acoustic networks. In *IEEE INFOCOM*, Apr. 2006.

[19] A. A. Syed, W. Ye, and J. Heidemann. Comparison and evaluation of the T-Lohi MAC for underwater acoustic sensor networks. *IEEE J. Sel. Areas Commun.*, 26(9):1731–1743, Dec. 2008.

[20] P. Xie and J. Cui. R-MAC: an Energy-Efficient MAC protocol for underwater sensor networks. In *Int. Conf. on Wireless Algorithms, Syst. and Applicat. (WASA)*, pages 187–198, Aug. 2007.