Exploring Random Access and Handshaking Techniques in Large-Scale Underwater Wireless Acoustic Sensor Networks

Peng Xie, Jun-Hong Cui
xp@engr.uconn.edu, jcui@cse.uconn.edu
Computer Science & Engineering Department
University of Connecticut, Storrs, CT 06029-2155

***Abstract*— In this paper, we study the medium access control (MAC) problem in large-scale underwater wireless acoustic sensor networks. We specially explore the random access and handshaking (i.e., RTS/CTS) techniques. We first formally model the two approaches, and then conduct extensive numerical experiments to study their performance in various network conditions. Based on our results, we observe that the performance of random access and RTS/CTS are affected by many factors such as data rate, transmission range, network topology, packet size, and traffic pattern. And our results show that RTS/CTS is more suitable for dense networks with high date rate, low/medium transmission range and bursty traffic, whereas random access is preferred in sparse networks with low data rate and non-bursty traffic. We believe this work will supply useful guidelines for energy-efficient adaptive MAC design in large-scale underwater wireless acoustic sensor networks.**

I. INTRODUCTION

Recently, there are growing interests in underwater networks [2], [4]–[6], [8], [11], which are envisioned as powerful technology for monitoring and exploring aquatic environments. In this paper, we study the medium access control (MAC) problem in large-scale underwater wireless acoustic sensor networks (we use "large-scale underwater sensor networks" for short) [2], [4], [6]. Like nodes in terrestrial sensor networks, underwater sensor nodes are also usually powered by batteries. Thus, besides throughput (or channel utilization), energy efficiency is one of the important goals of the MAC protocol design in underwater sensor networks. Moreover, the characteristics of the underwater acoustic channel, especially low communication bandwidth and long propagation delay, pose unique challenges for MAC protocol design.

In general, MAC protocols can be roughly divided into two main categories: (1) contention-free protocols that avoid collision among transmission nodes, and (2) contention-based protocols where nodes compete for a shared channel, resulting in probabilistic coordination. Contention-free protocols include time-division multiple access (TDMA), frequency division multiple access (FDMA) and code division multiple access (CDMA), where users are separated in time, frequency, or code domains. These protocols have been widely used in modern cellular communication systems. Contention-based protocols include random access (ALOHA, slotted ALOHA) [1], [9], and collision avoidance with handshaking (MACA, MACAW) [3], [7], which is the basis of several widely-used standards including IEEE 802.11. In general, contention-free protocols are not suitable for large-scale underwater sensor networks: TDMA requires centralized control which is not scalable in networks with a large number of nodes; FDMA is not suitable due to the narrow bandwidth of underwater acoustic channel; CDMA is not very practical because it is difficult to assign pseudo-random codes among large numbers of sensor nodes. Moreover, the near-far problem inherited in CDMA is not well addressed in underwater sensor networks [11].

In this paper, we investigate the two key techniques: random access and handshaking (employed by contention-based protocols) in underwater sensor networks. It has been argued that contention-based protocols that rely on handshaking, i.e., RTS (Request-To-Send)/CTS (Clear-To-Send), are not appropriate in underwater communications [8] [2]. The common cited reason is that RTS/CTS involves large end-to-end delay, thus increasing energy consumption. Based on a similar argument, in [10], Rodoplu et al. proposed a random access based MAC protocol for underwater sensor networks, focusing on low duty cycle applications with relatively sparse sensor deployment. To our best knowledge, however, there is no formal comparison of the random access and handshaking techniques in underwater sensor networks.

The questions we strive to answer in this paper are: (1) Is random access an absolute winner? (2) Can RTS/CTS yield better performance than random access in some network conditions? (3) Is it possible to design an adaptive MAC protocol to accommodate various network conditions? In the rest of this paper, we will quantitatively study the two schemes: random access and RTS/CTS. We will first describe our models for these two approaches, and then present numerical results to evaluate their performance.

II. MODELLING RANDOM ACCESS VS. RTS/CTS

A. Network Model and Performance Metrics

We use a simple network model, which is illustrated in Fig. 1. In this model, there is only one receiver, node B, located at the center, and all other nodes such as A, C, D, and E in its transmission range (denoted by R) compete for the channel to send data. We assume the data generation in each node is independent and follows a Poisson process with a rate λ. We use S_p to denote the packet size and T_p to denote the packet duration. Then we have $T_p = \frac{S_p}{bw}$, where bw is the bandwidth of each node. Further, we use n to denote the number of nodes in the network, L to denote the average distance from a sender to the receiver, and v to denote the sound propagation speed in water.

We investigate two performance metrics: throughput and communication overhead. *Throughput* is defined as the number of effective bits, i.e., the size of all the data packets successfully received by the receiver (node B) in one second. In this

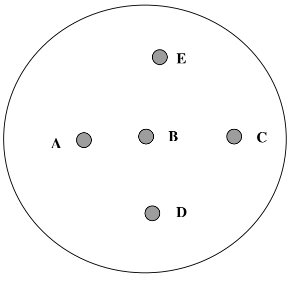

Fig. 1. Network model

paper, we call all the data packets that a node can send at one time as *a message*, and we use m to denote the number of data packets in one message. When a message consists of multiple data packets, we ignore the interval between two consecutive packets since it is usually very small compared with the data transmission time and propagation delay. *Communication overhead* is defined as the ratio of the total number of bits sent by the sender to the number of effective bits received by the receiver.

B. Modelling Random Access

In the random access approach, a sender simply starts sending whenever it has data ready for delivery. When a data packet arrives at the receiver, if the receiver is not receiving any other packet and, during the time of receiving this data packet, there is no incoming data packet (i.e., in a time period of $2\times T_p$, there are no other arriving packets), then the receiver can receive this data packet successfully.

To compute the throughput and communication overhead of random access, we first evaluate the success probability (denoted by P) of one packet sent from one sender (e.g., node A) to the receiver (node B). In fact, P is equivalent to the probability that all the nodes (including node A) in the transmission range of node B do not send any packet in a time period of $2 \times T_p$. If we denote the arriving time of the packet from A as t_0, then for each sender node i, the time period of $2 \times T_p$ is $[t_0 - T_p - D_i, t_0 + T_p - D_i]$, where D_i is the propagation delay from node i to node B. Please note that D_is are significant, as is different in radio-based networks. Considering the Poisson process of data generation at senders, the probability P can be given by

$$P = \Pi P_i = \Pi e^{-2\times\lambda\times T_p} = e^{-2(n-1)\lambda T_p}. \tag{1}$$

Then, the average throughput of the whole network can be calculated as

$$\rho_{random} = (n-1)\times\lambda\times S_p\times e^{-2(n-1)\lambda T_p}. \tag{2}$$

Correspondingly, the communication overhead will be evaluated as

$$\eta_{random} = \frac{1}{P} = e^{2(n-1)\lambda T_p}. \tag{3}$$

When there is bursty traffic in the network, i.e., $m > 1$ for each node, the throughput of random access is different from the case without bursty traffic. We assume that all the sender nodes have the same pattern of bursty traffic, i.e., m is the same for all the sender nodes. Then for any sender, the probability that all m data packets are successfully received by the receiver is given by

$$P_m = e^{-2(n-1)\lambda T_p\times m}, \tag{4}$$

and the probability that exactly i $(0 \le i < m)$ packets are successfully received by the receiver is calculated as

$$\begin{aligned} P_i = ((&e^{-(n-1)\times\lambda\times T_p\times(m+i)} - e^{-(n-1)\times\lambda\times T_p\times(m+i+1)}) \\ &+(e^{-(n-1)\times\lambda\times T_p\times i} - e^{-(n-1)\times\lambda\times T_p\times(i+1)})). \end{aligned} \tag{5}$$

In Equation 5, the first term represents the probability that the first $m-i$ packets (of m packets) collide with preceding burst traffic, and the second term denotes the probability that the last $m-i$ packets (of m packets) collide with subsequent burst traffic. After obtaining P_m and P_i, the average number of packets that are successfully received by the receiver can be easily calculated as

$$\Sigma_0^m i \times P_i. \tag{6}$$

Then, the throughput of random access with bursty traffic can be calculated as

$$\rho_{random}(m) = (n-1)\times\lambda\times S_p\times\Sigma_0^m i\times P_i. \tag{7}$$

Correspondingly, the communication overhead of random access with bursty traffic (i.e., when $m > 1$) will be given by

$$\eta_{random}(m) = \frac{m}{\Sigma_0^m i\times P_i}. \tag{8}$$

C. Modelling RTS/CTS

The basic idea of the RTS/CTS scheme is that a sender has to capture the channel (by handshaking) before sending any data. In underwater sensor networks, due to the long propagation delay, the traditional RTS/CTS model for radio-based networks should be modified. In the following, we use two examples to show why it is infeasible to directly use the traditional RTS/CTS model in underwater acoustic networks.

We first examine the example illustrated in Fig. 2, where node A and node B are 60 meters apart and node B and node C are 90 meters apart. Assuming node A sends an RTS to node B, and then node B replies with a CTS. Considering the propagation speed of acoustic signal in water ($v = 1500 m/s$), it takes 40 ms for the CTS to arrive at node A and 60 ms to arrive at node C. After node A receives the CTS, node A can not send data immediately since node C has not received the CTS from node B yet and possibly sends an RTS to node B, which will collide with the data packets sent from node A. Thus, in underwater sensor networks, to make RTS/CTS work, we devise the following changes: when a node receives a CTS, it can not send data immediately. Instead, it has to wait

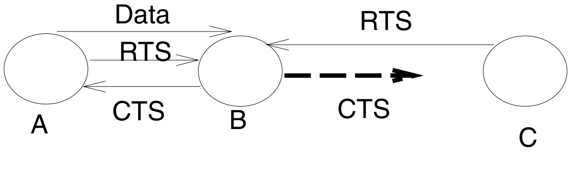

Fig. 2. RTS/CTS example 1

for the CTS to propagate the whole transmission range of the receiver (i.e., the sender of the CTS).

In underwater sensor networks, though the long propagation delay significantly damages the effectiveness of the traditional RTS/CTS mechanism, this network feature allows RTS/CTS exchange and data communication to proceed in parallel. As shown in Fig. 3, while node B is receiving a data packet from node A, node C can schedule to send an RTS to node B. When this RTS propagates to node B, node B just finishes receiving data from node A and is now ready for the RTS from node C.

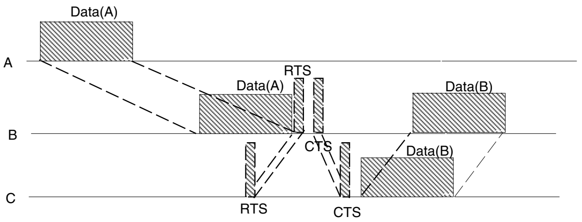

Fig. 3. RTS/CTS example 2

We now evaluate the throughput of the revised RTS/CTS approach. In this approach, the time for one transmission, T_θ, is calculated as $T_{cts} + T_{prop} + T_{trans}$, where T_{cts} is the time that the CTS propagates through the transmission range of the receiver (i.e., $T_{cts} = R/v$), T_{prop} is the time that data propagates from the sender to the receiver (i.e., $T_{prop} = L/v$) and T_{trans} is the data transmission time (i.e., $T_{trans} = S_p/bw$). Due to the effective collision avoidance of RTS/CTS, when the data rate is higher than the channel capacity, the effective data rate for the receiver reaches the limit, and can not increase any more. Thus, the effective data rate, λ_t, for the receiver can be calculated as $min(\frac{1}{T_\theta}, \lambda \times (n-1))$. Then the throughput of RTS/CTS is evaluated as

$$\rho_{rts/cts} = m \times S_p \times \lambda_t. \tag{9}$$

As for the communication overhead, it can be easily obtained as the following:

$$\eta_{rts/cts} = \frac{2 \times S_c + m \times S_p}{m \times S_p}, \tag{10}$$

where S_c denotes the size of the control packets (i.e., RTS/CTS packets). Please note that in this computation, we ignore the collision of RTS/CTS packets, since when S_c is small, the collision probability of RTS/CTS packets is very low.

III. NUMERICAL RESULTS

In this section, we conduct numerical experiments to compare the throughput and communication overhead of random access and RTS/CTS under various network conditions. We first explore their performance in networks without bursty traffic (referred to as "non-burst-traffic case"), considering the effect of transmission range, network topology, and packet size. Then we investigate the performance of the two approaches in networks with bursty traffic (referred to as "burst-traffic case").

A. General Experimental Settings

Unless specified otherwise, in all the numerical experiments studied in this section, we set the bandwidth of each node, bw, to 10 $kbps$, and the distance between a sender and the receiver, L, to 50 $meters$. In addition, the transmission range of each node, R, is set to 100 $meters$, the data packet size, S_p, to 40 $Bytes$ and the control packet size, S_c, to 4 $Bytes$. Further, the number of nodes in the target network, n, is set to 5.

B. Performance Comparison in Non-Burst-Traffic Case

We first check the performance of random access and RTS/CTS with varying data rates. In this set of experiments, we change the date rate from 0 to 10 $pkts/sec$, and fix all other parameters as in the general experimental settings. We measure the throughput and communication overhead of random access (We use "Random" for short in all the result figures) and RTS/CTS, and the results are plotted in Fig. 4 and Fig. 5 respectively.

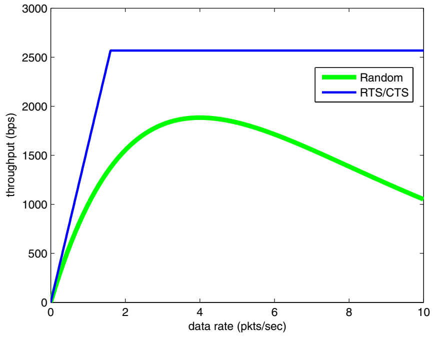

Fig. 4. Effect of data rate on throughput

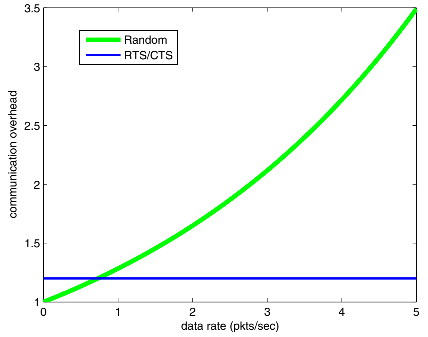

Fig. 5. Effect of data rate on communication overhead

From Fig. 4, we can observe that when the data rate is low, the throughput of both random access and RTS/CTS increases as the data rate is lifted. When the data rate exceeds some threshold, the throughput of both approaches reaches the maximum value. For the RTS/CTS approach, due to the effective collision avoidance, throughput keeps stable at the maximum value when the data rate continues to increase. However, for the random access approach, throughput starts to drop after the data rate exceeds some limit (and will eventually approach 0 when the data rate is extremely large). This is because that in random access, with no collision avoidance, very high data rate means very high chance of collision, translating into very low throughput.

The effect of data rate on the communication overhead of both approaches is illustrated in Fig. 5. From this figure, we can see that the communication overhead of random access increases dramatically as the data rate becomes larger, as can be explained as follows: when the data rate is larger, there is more chance for collision, thus more packets will be wasted, i.e., more communication overhead is introduced. On the other hand, the communication overhead of RTS/CTS does not change with data rate, as is due to its effective scheme of collision avoidance.

Comparing the performance of random access and RTS/CTS in Fig. 4 and Fig. 5, we can observe that when the data rate is low (less than 0.7 $pkts/sec$), the communication overhead of random access is lower than that of RTS/CTS, while the throughput of random access is almost the same as that of RTS/CTS. This indicates that random access is preferred in networks with very low data rate.

1) Effect of Transmission Range: In this set of experiments, we study the effect of transmission range, R, on the performance of random access and RTS/CTS. We vary the data rate and fix other parameters, and plot the throughput of the two approaches for various transmission ranges in Fig. 6.

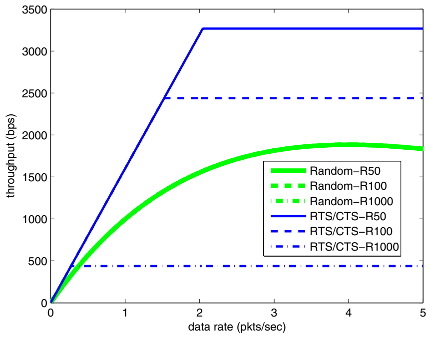

Fig. 6. Effect of transmission range on throughput

From Fig. 6, we observe that the transmission range affects the throughput of RTS/CTS significantly. When the transmission range is very large, for example, 1000 $meters$, the throughput of RTS/CTS is lower than that of random access for a wide range of data rate. This is because that when the transmission range is large, the time wasted for a CTS to propagate through the whole transmission area is long, thus degrading throughput. While for random access, the transmission range has no effect on the performance of throughput, as is consistent with Equation 2.

Based on Equation 3 and Equation 10, we can easily see that the transmission range has no effect on the communication overhead of both approaches. Thus, we omit the figure of the effect of transmission range on communication overhead. To summarize, this study suggests that RTS/CTS does not perform well in networks with large transmission range.

2) Effect of Network Topology: We now check the effect of network topology on the throughput and communication overhead of random access and RTS/CTS. We study two topological metrics: the number of nodes, n, in the transmission range of the receiver (representing the network density), and the distance, L, between a sender and the receiver.

First, we explore the effect of network density (i.e., n) on throughput. We change n from 5 to 15, and plot the throughput of random access and RTS/CTS in Fig. 7. From this figure, we can observe that random access performs worse in a denser network. This can be explained as follows: in a denser network, the random access approach reaches its limit more quickly, thus degrading faster. As for RTS/CTS, the network density has some effect on throughput when the data rate is low, since in such network conditions, throughput is determined by the date rate, and more neighbors means more incoming data, thus higher throughput. However, when the throughput of RTS/CTS climbs to its limit, due to the effective collision avoidance, the network density has no effect on throughput anymore.

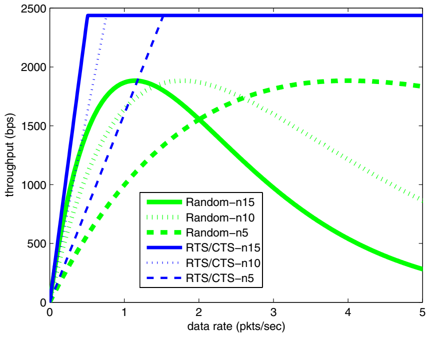

Fig. 7. Effect of network density on throughput

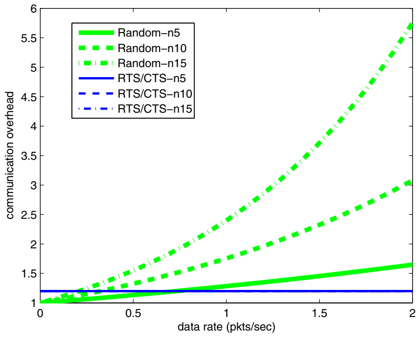

Fig. 8. Effect of network density on communication overhead

The effect of network density on the communication overhead is illustrated in Fig. 8. This figure shows that the network density has no effect on the communication overhead of RTS/CTS. However, it affects the communication overhead of random access significantly: the denser the network, the higher communication overhead, as is consistent with our intuition.

Now, we investigate the effect of distance between a sender and the receiver (i.e., L). From Equations 3 and 10, we know that L has no effect on the communication overhead of both random access and RTS/CTS approaches. However, it affects the throughput of RTS/CTS. We set L to 20 $meters$, 60 $meteres$ and 80 $meters$, and plot the results of throughput in Fig. 9. From this figure, we can see that when the distance between a sender and the receiver is shorter, the throughput of RTS/CTS has a higher maximum value. This, from another aspect, indicates that RTS/CTS performs better in denser networks. On the other hand, the distance L has no effect on the throughput of random access, as is consistent with Equation 2.

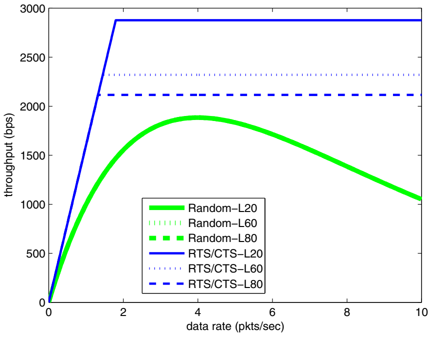

Fig. 9. Effect of distance L on throughput

Comparing the performance of random access and RTS/CTS with different topological parameters in Fig. 7, Fig. 8, and Fig. 9, we can conclude that when the data rate is high, RTS/CTS works better in a dense network than random access, yielding higher throughput while introducing lower communication overhead.

3) Effect of Packet Size: We now investigate the effect of packet size on the throughput of random access and RTS/CTS. We vary the data rate cases, setting the packet size to 40 $Bytes$, 80 $Bytes$ and 160 $Bytes$ respectively. The results are plotted in Fig. 10. From this figure, we can see that the packet size has significant impact on both random access and RTS/CTS approaches. For RTS/CTS, as the packet size increases, the maximum throughout increases as well. On the other hand, the increasing packet size damages the throughput of the random access approach: when the packet size becomes larger, the throughput of random access reaches its limit earlier, and then decreases more dramatically as the data rate continues to increase. Another point worth notice is that when the packet size increases, the maximum throughput of random access does not change, while the maximum throughput of RTS/CTS increases greatly.

Fig. 11 shows the effect of packet size on the communication overhead of random access and RTS/CTS. We observe that larger packet size reduces the communication overhead of RTS/CTS, whereas increases that of random access.

In short, from this set of experiments, we can conclude that random access works better in a small-packet-size network, while RTS/CTS favors the networks with a large packet size.

C. Performance Comparison in Burst-Traffic Case

We now evaluate the performance of random access and RTS/CTS approaches under bursty traffic. Since sensor networks are widely used in the applications of measurement and surveillance, data traffic in such networks is averagely low in long time periods, but may be heavy in some very short time periods when sensor nodes are triggered by some events. We believe bursty traffic would be also typical in underwater sensor networks. In this set of experiments, we use m, the number of packets in a message, to measure the burstiness of data traffic. We set m to 1, 10, and 20, and plot the results of throughput and communication overhead in Fig. 12 and Fig. 13 respectively.

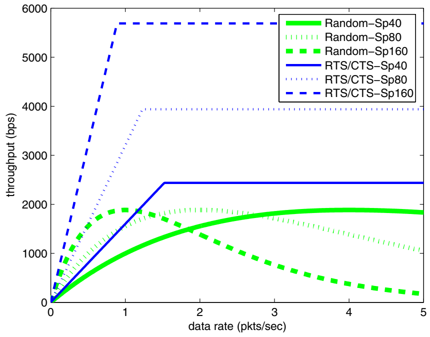

Fig. 10. Effect of packet size on throughput

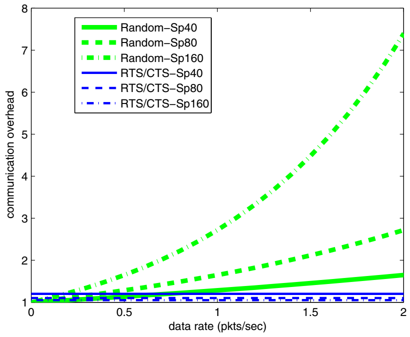

Fig. 11. Effect of packet size on communication overhead

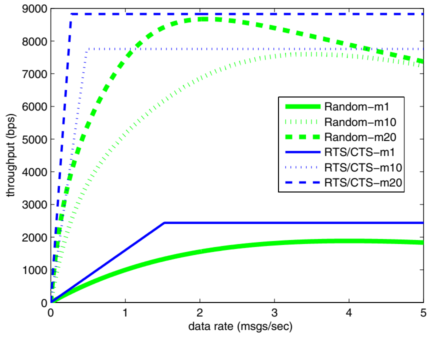

Fig. 12. Throughput under bursty traffic

Fig. 12 shows that bursty traffic significantly improves the maximum throughput of random access and RTS/CTS. Moreover, heavier bursty traffic makes random access achieve

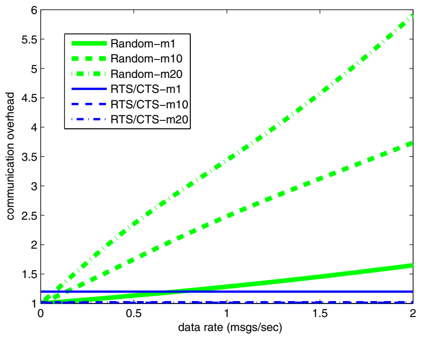

Fig. 13. Communication overhead under bursty traffic

its maximum throughput more quickly and then degrade more sharply as the data rate increases. As for RTS/CTS, with bursty traffic, the system can reach its maximum throughput when the data rate is very low, and the maximum throughput approaches the channel maximum throughput as traffic gets more bursty.

The communication overhead of random access and RTS/CTS under different levels of bursty traffic is plotted in Fig. 13. From this figure, we observe that as the data traffic gets more bursty, the communication overhead of random access increases dramatically; on the other hand, the communication overhead of RTS/CTS decreases as there are more packets in one data burst (i.e., one message). This is due to the fact that RTS/CTS can avoid packet collision effectively, and more packets sent mean low overhead per packet for collision avoidance, while random access has no collision avoidance, and more packets sent mean high probability of collision, thus high communication overhead.

This study indicates that though bursty traffic improves the throughput of both random access and RTS/CTS approaches, random access improves its throughput at the cost of more packet collision, and RTS/CTS improves its throughput more efficiently.

D. Summary

We summarize our observations as follows: (1) Random access has almost the same performance as or better performance than RTS/CTS under very low traffic and sparse deployment (which is consistent with the argument in [10]), while when the data rate increases and the network gets denser, the channel is saturated quickly, resulting in low throughput degradation and high communication overhead; (2) In contrast, RTS/CTS has no significant advantages at low data rate and sparse deployment, but provides more room for performance improvement in dense networks with high data rate; (3) The throughput of RTS/CTS is significantly affected by the transmission range: when the transmission range is large, RTS/CTS has a very low throughput. On the other hand, the transmission range has no effect on random access; (4) The packet size has significant impact on the performance of both random access and RTS/CTS. In general, random access works better in networks with a small packet size, while RTS/CTS outperforms in large-packet-size networks; (5) Bursty traffic improves the throughput of random access, but at the cost of more energy waste on packet collision. As for RTS/CTS, bursty traffic not only improves throughput but also reduces communication overhead.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we have formally studied the random access and handshaking (i.e., RTS/CTS) techniques in underwater sensor networks. We first modelled the two approaches, and then conducted extensive numerical experiments to investigate their performance (in terms of throughput and communication overhead) in various network conditions. Based on our results, we conclude that the simple random access approach is preferred in sparse networks with very low data traffic; while RTS/CTS has better performance when the network gets denser and the data rate becomes higher. Moreover, RTS/CTS does not work well in networks with large transmission range, while random access has no such constraint. Furthermore, RTS/CTS has better performance than random access in large-packet-size networks. Lastly, in the case of bursty traffic, RTS/CTS can achieve higher throughput while at relatively lower communication overhead compared with random access.

Recalling the three questions raised in the introduction, we are now in a good shape to answer them: (1) Random access is not an absolute winner; (2) RTS/CTS can achieve better performance in many network conditions (as summarized earlier); (3) Our study indicates that it is possible to design an adaptive MAC protocol, adopting random access and RTS/CTS dynamically for different networks conditions. Our future work will exactly follow the third line, aiming to devise an adaptive MAC protocol to accommodate the complex conditions in underwater sensor networks.

REFERENCES

[1] N. Abramson. The ALOHA System -Another Alternative for Computer Communications. *AFIP Press*, Vol 37, 1970.

[2] I. F. Akyildiz, D. Pompili, and T. Melodia. Challenges for Efficient Communication in Underwater Acoustic Sensor Networks. *ACM SIGBED Review*, Vol. 1(1), July 2004.

[3] V. Bhargavan, A. Demers, S. Shenker, and L. Zhang. MACAW-A Medium Access Protocol for Wireless LANs. *ACM SIGCOMM*, pages 212–225, 1994.

[4] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou. Challenges: Building scalable mobile underwater wireless sensor networks for aquatic applications. *IEEE Network, Special Issue on Wireless Sensor Networking*, 20(3):12–18, May/June 2006.

[5] L. Freitag, M. Grund, C. von Alt, R. Stokey, and T. Austin. A Shallow Water Acoustic Network for Mine Countermeasures with Autonomous Underwater Vehicles. In *IEEE Oceans Conference*, Washington DC, 2005.

[6] J. Heidemann, Y. Li, A. Syed, J. Wills, and W. Ye. Underwater sensor networking: Research challenges and potential applications. *USC/ISI Technical Report ISI-TR-2005-603*, 2005.

[7] P. Karn. MACA-A New Channel Access Method for Packets Radio. *Proceedings of the ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, September 1990.

[8] J. G. Proakis, E. M. Sozer, J. A. Rice, and M. Stojanovic. Shallow Water Acoustic Networks. *IEEE Communications Magazines*, pages 114–119, November 2001.

[9] L. G. Robert. ALOHA packet system with and without slots and capture. In *ACMSOGCOMM Comput. Communication review*, pages 28–42, 1975.

[10] V. Rodoplu and M. K. Park. An Energy-Efficient MAC Protocol for Underwater Wireless Acoustic Networks. In *IEEE Oceans Conference*, Washington DC, 2005.

[11] G. G. Xie and J. Gibson. A Networking Protocol for Underwater Acoustic Networks. In *Technical Report TR-CS-00-02*, Department of Computer Science, Naval Postgraduate School, December 2000.