Short Paper: Reliable Transport and Storage Protocol with Fountain Codes for Underwater Acoustic Sensor Networks

Rui Cao
University of Florida
Gainesville, FL 32611, USA
raycao@ufl.edu

Liuqing Yang
University of Florida
Gainesville, FL 32611, USA
lcyang@ece.ufl.edu

ABSTRACT
Due to high energy efficiency and fast data query, data-centric storage (DCS) is a promising technique for underwater acoustic sensor networks (UASN). However, the harsh sea environment poses new challenges to the design of underwater DCS protocols. First, because of long propagation delay and unreliably data links of underwater acoustic communications, long-distance multi-hop reliable data transport needs to be investigated. Secondly, due to the high node failure probability in the unattended UASN, data reliability demands better protection. To address these two issues together, a fountain codes based reliable transport and storage (RTS) protocol is proposed in this paper. In the RTS protocol, a distributed fountain coding scheme is designed to facilitate reliable data delivery and multiple acknowledgments are adopted to ensure the control message reliability. In addition, to guarantee uniform data reliability, we devise a distributed storage scheme with concatenated fountain codes. Analyses are provided to reveal the performance of the proposed RTS protocol, which include storage reliability, energy consumption and number of retransmissions.

Categories and Subject Descriptors
H.3 [Information Systems]: Information Storage and Retrieval; C.2.2 [Computer Systems Organization]: Computer-Communication Networks—Network protocols

General Terms
Design, performance, reliability

Keywords
Fountain codes, data-centric storage, reliable data transport and storage, underwater acoustic sensor networks

1. INTRODUCTION
Underwater acoustic sensor networks (UASN) are popular network paradigms for many oceanic applications, such as environment monitoring, off-shore exploration and etc. [1]. However, real-time data delivery to the data center on the ground is not suitable for UASN due to the harsh sea environment and scarce node energy. Therefore, in-network data storage is a more favorable solution. For data with frequent query, data-centric storage (DCS) [7] is a preferable protocol because of its fast and energy-efficient data retrieved.

To design an underwater DCS protocol, we need to address two issues: reliable data transport and reliable data storage. In DCS, the data of the same category are stored in one specific location. Thus all sensed data need to be transported correspondingly. However, underwater acoustic communications and networks are featured with long propagation delay and intermittent connectivity, which render dramatic difficulty in reliable multi-hop data transport. On the other hand, the harsh sea environment induces vulnerable sensor nodes due to fouling and corrosion [1], thus data storage reliability is critical to underwater DCS.

In the literature, various protocols have been proposed to enhance the data transport reliability. For routing protocols, the Greedy Perimeter Stateless Routing (GPSR) protocol [4] is adopted to provide flexible data routes by adapting to the network dynamics. In underwater network researches, several reliable data transport protocols are proposed. In [9], traditional erasure codes (EC) are used to facilitate forward error correction (FEC) data transmission schemes. Traditional EC have fixed code rate and high decoding complexity, thus it cannot adapt to the variant link failure rate and do not support large data transmission. In [5], rateless fountain codes are studied for underwater broadcast service. The broadcast performance is analyzed for fountain systems with fixed retransmission ratio and the effects of transmission distance and desired coverage are investigated. To explore the benefits of fountain codes in DCS, we design a fountain codes based Hybrid Automatic Repeat Request (ARQ) data transport scheme for long-distance multi-hop transmissions. Our scheme uses adaptive retransmission ratio. The performance of our proposed scheme for bulk data transport is investigated in terms of number of transmissions and communication overhead. The results will be compared with the traditional ARQ based transport scheme, and the effect of window size is revealed. In addition, the design parameters will be optimized.

On the other hand, to improve the data storage reliability in DCS, various modified DCS protocols are published in the literature. Data replication is adopted in [7] and region-based storage is used in [8]. In another stream of work, fountain codes are introduced into distributed storage to reduce storage cost with better data reliability. In [5], decentralized fountain codes are designed to be applied to distributed data storage in wireless sensor networks. Small space redundancy is achieved with coded storage. However, huge communica-
tion cost is inherent in these schemes, which makes it not suitable for UASN. Thus a zone-based DCS protocol with Raptor codes (R-DCS) is proposed in [2]. In R-DCS, the data encoding is constrained in the storage zone, thus the communication cost is tremendously reduced. The problem that comes with this protocol is that the reliable data transport between the sensor nodes and the storage zone needs to be guaranteed.

In this paper, we design a reliable data transport and storage (RTS) DCS protocol with concatenated fountain codes. In RTS, sensing-zone fountain coding is used to enable reliable data transport. In addition, a second layer of fountain codes is applied in the storage zone to ensure uniform storage reliability. To verify the performance of the proposed RTS protocol, we will analyze the reliability uniformity and communication cost of the concatenated fountain codes.

The rest of the paper is organized as follows. The proposed RTS protocol is described in Sec. 2. Then the storage and transport reliability of the protocol is analyzed in Sec. 3. Summarizing remarks will be given in Sec. 4.

Notation: ⌈a⌉ is the smallest integer that is larger than a.

2. PROPOSED RTS PROTOCOL

In this section, we will describe our proposed RTS protocol. To enable both reliable data transport and storage, three main techniques are utilized. First, concatenated fountain coding is adopted. The first layer of encoding is performed at the sensing zone to facilitate reliable data delivery. The second layer of encoding is conducted in the storage zone to enhance data storage reliability. Secondly, fountain codes based data transport reliability control is designed. Finally, multiple acknowledgements (ACK) are used to increase the reliability of the ACK messages.

2.1 Network Setup

Consider a sensor network with $N$ sensor nodes distributed in an underwater sensing field. Zone-based architecture is considered in our scheme. The sensing field is virtually divided into $N_z$ zones with $N_z$ nodes at each zone. Inside each zone, the nodes are classified into two sets: border nodes and center nodes. If one node has all its neighbors farther than itself to one zone boundary, then it identifies itself as the border node, otherwise it is labeled as a center node. In DCS design, GHT is adopted as data mapping technique. GPSR is chosen as the inter-zone routing protocol, and geocast is used within each zone.

2.2 Protocol Description

The complete RTS protocol consists of three parts. First, the sensed data are encoded in the sensing zone. Secondly, the coded data are sent to the storage zone with reliability control. Finally, the data arrived at the storage zone are further encoding in a distributed manner.

2.2.1 Sensing zone encoding

In the sensing zone, every node senses the environment periodically, and each sensed data is identified by three values: sensing time, data type and zone ID. When DCS cycle comes, each node starts to geocast all its data to the sensing zone. Each sensing zone node temporarily stores a copy of data from all other nodes in the zone. After all data have been spread out in the sensing zone, the encoding process starts at each node.

Assume $k$ data packets are sensed at each node, then $k_x = ⌈(1 + \alpha)(1 + \delta)k⌉$ coded packets are generated from $kN_x$, raw packets at each node. $\alpha$ is the transmission redundancy ratio determined by the route failure rate and $\delta$ is the encoding redundancy ratio affected by the encoding size $kN_x$. Luby Transform (LT) codes [6] are utilized for fountain encoding. To generate one coded packet, a code weight is first randomly drawn from the distribution $\Omega(\alpha)$. Then $d$ raw packets are randomly chosen from the $kN_x$ raw packets and XORed as one coded packet. After encoding, the values of transmission time, total coded packets $k_t = N_xk_x$ and minimum packets for decoding $k_c = (1 + \delta)kN_x$ are added to each coded packet. Finally, the coded packets are forwarded to the storage zone determined by the data type with GHT. In addition, a timer is set up for this DCS period.

2.2.2 Fountain codes based transport

After each coded packet leaves the source node, it is greedily routed to the storage zone with GPSR. At every hop, the packet is forwarded to the neighbor that is nearest to the destination. When the packet arrives at the border node, it is geocast in the storage zone. For each data arrived at one storage node, the node first fetches the zone ID, transmission time and encoding values $k_t$ and $k_c$. Then the zone ID is compared with current zone timers:

- If there is no timer for this zone, the packet is identified as the first packet from this zone. The storage node will set one reception timer and one packet counter for this zone.
- If there is already a timer for this zone, the storage node will read the corresponding zone packet counter and increases it by one.

When one timer decreases to zero, the storage node will first check its packet counter value $n_c$ and compute the route failure rate $p_r = n_c/k_c$. Then the packet counter value $n_c$ is compared with the target number of reception $k_c$. If $n_c \geq k_c$, an ACK message is generated, otherwise a NACK is formed with the number of packets requested $k_t = k_c - n_c$. The route failure rate $p_r$ is also fed back with the message. To enhance the reliability of ACK/NACK messages, a multi-ACK scheme is designed as follows. If the node is a border node, the ACK/NACK message is directly sent to the home zone, otherwise, then the message is sent with probability $1/(1 - p_r)/N_x$.

At the sensing zone, each node keeps tracking its timer. If there is no ACK/NACK received after the timer runs out, the node will regenerate a new set of packets and send them to the storage zone. When a new ACK/NACK arrives in time at the border node, the message is geocast in the sensing zone. At each node, the timer is first discarded on the arrival of an ACK/NACK message. Then the route failure rate $p_r$ is fetched from the message. If an ACK is received, then all raw data are deleted. If a NACK is received, the node will adaptively retransmit $\lceil(1 + \alpha)k_r/N_x\rceil$ new coded packets according to the message, where $\alpha$ is updated by $p_r$. This process continues until either an ACK is received or the maximum number of retransmissions has been tried.

2.2.3 Storage zone encoding

To reduce the storage space requirement, distributed randomized coding is adopted in the storage zone. At each storage node, the total number of needed packets from all zones $n_t$ is determined by adding up all $k_x$. Then, for a predefined storage redundancy ratio $\delta_x$, each storage node computes the number of storage packets as $m = \lceil(1 + \delta_x)n_c/N_x\rceil$. Then, the encoding process starts: (1) Every node opens $m$ storage slots and generates a random weight $d_i$ from the distribution $\Omega(d)$ independently for slot $i$; (2) For each arrived data, the storage node accepts it with probability $d_i/t$ for slot $i$. If the packet is accepted, the corresponding slot will update
its storage by XORing it with the existing data in the slot. Finally, all coded data are generated and stored in the zone. The data retrieval is straightforward. When an underwater vehicle wants to retrieve one type of data, it first finds the storage zone by GHT. Then it can send a query to the storage zone or drive to the storage zone to acquire the data.

3. PERFORMANCE ANALYSIS

To evaluate the performance of the RTS protocol, we will analyze the benefits of the concatenated fountain coding scheme and coded data transport. The data storage reliability, data transport delay and communication cost will be revealed. In addition, the optimal α is investigated.

3.1 Concatenated Fountain Coding

If we remove the second-layer storage-zone coding, the single layer sensing-zone coding can also enable reliable data transport and storage. In single-layer coding scheme, the code packets will be randomly stored in the storage zone without further encoding. The benefits of concatenated fountain coding are illustrated through comparison with single-layer coding scheme.

3.1.1 Storage reliability

One property of fountain codes states that the codes are resilient to any erasure pattern as long as enough packets can be received. Thus, with the second layer storage-zone encoding, the concatenated coding scheme can recover data from all zones under any node failure pattern if enough packets are acquired. Thus uniform protection is assured for data from all zones. On the other hand, the single layer coding scheme is not robust under some node failure patterns. One extreme case is that, if the lost data are all from one zone, then the data recovery for this zone will experience a failure.

3.1.2 Energy efficiency

In LT codes, to recover the original \( kN_n \) raw data from each zone with high probability, the storage zone needs to store at least \( k_c \) packets from each sensing zone. In single layer sensing-zone coding, to overcome a node failure rate of \( p_f \), the sensing zone has to successfully send \( \lceil k_c/(1-p_f) \rceil \) packets on average. On the other hand, in the concatenated fountain coding, the second-layer encoding in the storage zone is performed to resolve the storage node failure problem. Thus only \( k_c \) packets are necessary. The price that the concatenated fountain coding needs to pay is the data flooding inside the storage zone. The communication costs for these two schemes can be computed as follows:

\[
C_{\text{single}} = \frac{N \cdot k_c \sqrt{N} + \sqrt{N_n}}{(1-p_f)}
\]

\[
C_{\text{concat}} = \frac{N \cdot k_c}{1-p_f} (\sqrt{N} + \sqrt{N_n})
\]

Comparison reveals that the concatenated scheme requires less communication if the fault rate is higher than \((\sqrt{N_n} - 1)/(\sqrt{N} + \sqrt{N_n})\). Thus the concatenated scheme is more beneficial in the networks with small zone size and large number of zones.

3.2 Data Transport

To reveal the performance of the fountain code assisted data transport, we evaluate the number of transmissions and communication cost for successful data delivery. The results are compared with traditional transport protocol using ARQ with Selective Repeat (SR).

3.2.1 Average number of transmissions

For the ARQ-SR transport protocol with window size of \( L \), the average number transmissions can be computed in a recursive manner:

\[
T_{\text{ARQ},L} = \frac{1}{1-p_F} \left[ 1 + \sum_{i=1}^{L} \left( \binom{L}{i} p_i (1-p_c)^{L-i} T_{\text{ARQ},i} \right) \right]
\]

On the other hand, for the RTS protocol, the average number of retransmissions \( T_{\text{RTS},L} \) for transmitting \( L \) packets is calculated as:

\[
T_{\text{RTS},L} = \frac{1}{1-p_F} \left[ 1 + \sum_{i=\lceil \alpha L \rceil}^{K-1} \left( \binom{K-1}{i} p_i (1-p_c)^{K-i} T_{\text{RTS},i} \right) \right]
\]

where \( K = L + \lceil \alpha L \rceil \). In Fig. 1, we simulate the average number of transmissions numerically for both the ARQ-SR and RTS protocols. The route failure rate is \( p_f = 0.1 \). It is observed from the figure that, as the transmission window size \( L \) increases, the number of transmissions increases for both protocols. However, the RTS has much smaller \( T \).
than the ARQ-SR protocol. At large window size, the average transmissions of the RTS protocol saturate to about 1.5, while the average transmissions of ARQ-SR increase slowly to around 4. This observation indicates that the RTS protocol is extremely beneficial in terms of end-to-end delay, especially for large number of data delivery. Therefore, it is favorable for the DCS protocol in dynamic underwater acoustic networks.

3.2.2 Communication overhead

Similarly, we can compute the average total number of transmissions required to deliver all $L$ packets with the ARQ-SR protocol as follows:

$$N_{\text{ARQ},L} = \frac{1}{1-p} \left( (L+1)\sum_{i=1}^{\alpha} (K+i) p^i \right) - N_{\text{ARQ},L-1}.$$  

(5)

For the RTS protocol, the number of transmissions is:

$$N_{\text{RTS},L} = \frac{1}{1-p} \left( (K+N_{\text{ACK}})\sum_{i=1}^{\alpha} (K+i) p^i \right) - N_{\text{RTS},L-1}.$$  

(6)

where $N_{\text{ACK}}$ is the number of ACKs. For a route failure rate of $p_r = 0.1$, we simulate the total packet transmissions for both the ARQ-SR and RTS protocols with different window size. To demonstrate the communication cost difference, we define the transmission overhead percentage as the difference ratio between the total number of transmissions and the window size: 

$$\frac{N - L}{L} \times 100\%.$$  

The results are displayed in Fig. 2. For both protocols, the transmission overhead decreases as the window size $L$ increases. For large window size, the ARQ-SR protocol approaches to about 11% overhead, while the RTS protocol uses about 16% overhead. This 5% difference comes from the coding redundancy ratio $\delta = 0.05$. The comparison shows that the RTS protocol achieves similar communication cost as the ARQ-SR protocol if the coding redundancy ratio is very small, which is true for large coding size.

3.2.3 Optimal transmission redundancy $\alpha$

Recall that the transmission redundancy ratio $\alpha$ is tuned according to the route failure rate $p_r$. To achieve the optimal system performance, we can find the best $\alpha$ value according to two criteria: (1) minimizing the number of transmissions in Eqs. (4) and (2) minimizing the transmission overhead in (6). Since the closed-form formula solution for the optimal $\alpha$ is mathematically intractable, we resort to numerical results to find it. In Fig. 3, we plot the optimal $\alpha$ for different route failure rate $p_r$ with respect to the above two criteria. The result shows that the optimal $\alpha$ for both criteria increases almost linearly with $p_r$. To achieve the minimum transmission overhead, the optimal $\alpha$ needs to be almost the same as $p_r$, while for the minimum number of retransmissions, the optimum $\alpha$ is slightly more than $p_r$. Therefore, for different system performance requirements, we can choose the optimal $\alpha$ accordingly.

4. CONCLUSIONS

In this paper, we designed a fountain codes based DCS protocol termed as RTS. To guarantee reliable data transport, the fountain codes based transmission scheme with multi-ACK is designed. To enable reliable data storage, the concatenated fountain coding is adopted. The performance analyses are provided to reveal the benefits of the RTS protocol. The results show that the proposed RTS protocol possesses small number of end-to-end transmissions, low communication cost and uniform data reliability. In addition, the data transmission redundancy ratio needs to be chosen around the route failure rate to obtain the best performance.

5. ACKNOWLEDGEMENT

This work is in part supported by National Science Foundation under grant #0845722.

6. REFERENCES


