Fountain Codes and their Application to Broadcasting in Underwater Networks: Performance Modeling and Relevant Tradeoffs

Paolo Casari, Michele Rossi and Michele Zorzi

Department of Information Engineering
University of Padova – Italy
{casarip, rossi, zorzi}@dei.unipd.it

This work has been partially supported by
NOAA’s Sea Grant College Program, Project no. NA060AR4170019
Motivation and Objectives

- Broadcast is a fundamental primitive, and will be required for many operations
  - Interest and data dissemination
  - Explicit route discovery
  - Wireless node reprogramming

- Objectives
  - To apply rateless codes to underwater broadcasting and study relevant tradeoffs that arise
  - To find convenient working points as a function of the network parameters and the amount of redundancy
The What and Why of rateless codes - 1

- Rateless codes are special codes that allow to generate redundancy on demand
  - Drawback: they require a certain overhead even in the absence of transmission errors
- Encoding procedure
  - Pick $K$ packets to transmit
  - Extract $n$ binary vectors whose weight (number of ones) is chosen according to a certain distribution $\rho(x)$
  - Generate $n$ encoded packets by XORing the source packets corresponding to ones in the vectors
- Decode by inverting the encoding matrix formed by all received vectors (if possible)

From D.J.C. MacKay, "Fountain Codes", *IEE Proceedings*, vol. 152, n. 6, pp. 1062-1068.
The What and Why of rateless codes - 2

• **Pros**
  – Rateless codes are not bound to a fixed amount of redundancy
    ✓ Generate more redundancy in the presence of more errors
    ✓ Do not require to estimate the error probability beforehand in order to design the code
  – Asymptotically optimal in terms of throughput
  – Suited to underwater networks: can be optimized to yield minimum number of transmissions, that represent the greatest cost in these networks
  – Suited for broadcast (efficient error control)

• **Cons**
  – A non-zero (albeit small) overhead is always required (namely we need $K (1+\varepsilon)$ packets to decode a group of $K$ packets)
    ✓ This overhead is required to make the encoding matrix invertible (recall that all vectors in the matrix are random)
Fountain codes and ARQ

- Simple ARQ scheme
  - The transmitter sends $K+\xi$ packets (e.g., $K=32$ and $\xi=4$)
  - The receivers check whether the message has been correctly received
  - A very short feedback is sent in case of errors
  - If any feedback is received, the transmitter sends $\xi$ additional packets

Round 1 ($K+\xi$ packets)  Round 2 ($\xi$ more packets)

The maximum number of rounds is constrained to $L$ (e.g., $L=5$)
Some preliminary considerations

• Focus of this work: broadcast and *advancement per hop*
• Parameter we can play with:
  – redundancy per round, $\xi$
  – Maximum number of rounds, $L$
  – Target coverage range, transmit power, $P_{tx}$, transmit bandwidth (rate)
• Performance metrics:
  – the number of users that correctly receive the message per hop
  – advancement
  – overall “energy cost” of the broadcast
  – delay (longer reach \(\rightarrow\) fewer transmissions but smaller available bandwidth and larger propagation delay)
• Here, we try to strike a balance between advancement, power consumption, “code rate” (through $L$ and $\xi$), delay
  – Explicitly accounting for macroscopic underwater propagation effects
The bandwidth-distance relationship

\[ SNR(l, f) = \frac{P_\Delta / A(l, f)}{N(f)\Delta f} \]

- Both the frequency center AND the bandwidth vary with distance between nodes

Analysis – 1

- Channel: SNR & bandwidth depend on distance (approx)

\[
SNR(d) = \frac{P_{tx}/B(R_n)}{A(d, f_0(R_n)) N(f_0(R_n))}
\]

\(R_n = \) TX range

- Probability that \(K\) packets are reconstructed from \(x\)

\[
\Psi_K(x) = \prod_{i=0}^{K-1} (1 - 2^{i-x})
\]

Adapted from D.J.C. MacKay, "Fountain Codes", *IEE Proceedings*, vol. 152, n. 6, pp. 1062-1068.
Analysis – 2

- Probability that a user requires more than $j$ rounds to decode, probability that a user is successful, and average number of rounds

$$P_{1}^{>j}(p) = \sum_{e=0}^{K+j\xi} B(K + j\xi, e, p) \left(1 - \Psi_{K}(K + j\xi - e)\right)$$

$\rho$ = packet error probability, $K$ = message packets, $\xi$ = redundancy per round, $e$ = erroneous packets

$$\tilde{P}_{1}^{\leq j} = \frac{1}{R_{n}} \int_{0}^{R_{n}} P_{1}^{\leq j}(p(\ell)) \frac{2\ell}{R_{n}^{2}} d\ell$$

$$E[j] = \sum_{N_{u}=0}^{+\infty} E[j|N_{u}] \frac{\left(\mu A\right)^{N_{u}} e^{-\mu A}}{N_{u}!}$$

$$= \sum_{j=0}^{L-1} \left[1 - e^{-\mu A(1 - \tilde{P}_{1}^{\leq j})}\right]$$

$\mu$ = average user density per unit area

- Delay

$$D = (K + E[j]\xi) T_{D}(R_{n}) + (E[j] - 1) \left(2\tau_{R_{n}} + T_{A}(R_{n})\right) + \tau_{R_{n}}$$
Analysis – 3

- Probability that advancement is greater than $z$ given $N_u$ receivers

$$F(z) = P[\text{adv} > z] = \sum_{N_u=0}^{+\infty} \frac{(\mu A_z)^{N_u}}{N_u!} e^{-\mu A_z} (P_{A_z}(z))^{N_u}$$

$$= e^{-\mu A_z} (1 - P_{A_z}(z))$$

where the probability that a given user inside the grey area is successful is

$$P_{A_z}(z) = \int_0^z P_{\ell < L} (p(\ell)) \frac{2\ell}{z^2} d\ell$$

- Average advancement

$$E[z] = \int_0^{R_n} F(z) dz$$
Results – Illustration of code performance

- **Probability** that more than \( j \) rounds are required so that all receivers in a 5 km range decode correctly

- \( L = 5 \)
- \( \xi = 1, 2, 4, 8 \)
- \( \lambda = 5, 10 \)
- \( \lambda = \) average number of users in a 5 km radius
- Notably, a large \( \xi \) may still require more than 1 round
**Results – One-hop delay performance**

- **Delay** to complete a one-hop transmission versus *distance*
- Depends on $L$, $\xi$
- Depends on $R_n$ because of propagation delay
- Initial increase until large PER
- Sharp transition around $R_n$
- Linear increase (due to larger propagation delay) after $R_n$

$R_n$ sets the TX range (thus power, hence PER)
### Results – Delay vs. Advancement – 1

- **Delay versus Advancement**: tradeoff played by **distance**
- A greater redundancy per round helps improve the advancement (at the price of greater delay)

![Graph showing delay vs. advancement](image)

\( R_n \) sets the TX range (thus power, hence PER)
**Results – Delay vs. Advancement – 2**

- *Delay versus Advancement*: tradeoff played by $\xi$ (redund. per round)
- A larger $\xi$ allows to actually reach (or go slightly beyond) the set TX range $R_n$
- The best performance improvement is observed from $\xi = 1$ to $\xi = 2$
- A larger $\xi$ yields little improvement
Results – Multihop delay

- **Multi-hop delay versus distance**
- We measure the time to cover $Z=100$ km
  $$ZD/E[z]$$
- There exists an optimal distance (slightly less than $R_n$) which is optimal in terms of delay
Summary, Conclusions, Future Directions

• In this work, we employed rateless codes for hybrid ARQ in underwater sensor networks
• We employed a simple retransmission technique and studied the tradeoffs that arise thereby
• Results show that there are some interesting tradeoffs related to the amount of additional redundancy over multiple rounds, which suggest that some optimized strategy would be desirable
• Future directions of this work include the study of policies in the presence of variable error probabilities and the extension to broadcasting over multihop topologies
• Also, we will consider a realistic scenario in which to test the schemes and optimize their parameters. We developed an ns2 extension:
  – [http://telecom.dei.unipd.it/download](http://telecom.dei.unipd.it/download) -> NS2/NS3 -> Download UnderWaterMiracle
Fountain Codes and their Application to Broadcasting in Underwater Networks: Performance Modeling and Relevant Tradeoffs

Paolo Casari, Michele Rossi and Michele Zorzi
Department of Information Engineering
University of Padova – Italy
{casarip, rossi, zorzi}@dei.unipd.it

This work has been partially supported by NOAA’s Sea Grant College Program, Project no. NA060AR4170019