A Rate-Compatible Sphere-Packing Analysis of Feedback Coding with Limited Retransmissions

Adam Williamson, Tsung-Yi Chen, and Rick Wesel

UCLA Communication Systems Laboratory
arXiv: 1202.1458

July 6, 2012
Variable-length feedback with termination (VLFT) codes [Polyanskiy et al. 2011]:

⇒ Transmitter sees all channel outputs and tells the receiver when to terminate through a separate control channel.

⇒ Transmission may terminate after each symbol.

⇒ (Rate-compatible) random coding.

⇒ General results with numerical examples for BSC and BEC.
This Talk

- **This talk:** Still the basic VLFT framework.
  - Transmitter sees all channel outputs and tells the receiver when to terminate through a separate control channel.
  - Transmission may only terminate at the end of a “packet”.
  - Incremental packet lengths will be optimized.
  - Rate-compatible sphere-packing (RCSP) [Chen et al. 2011].
  - Rate-compatible tail-biting convolutional code.
  - Focused on the AWGN channel.
Forward channel is AWGN with known SNR, $\eta$.

The receiver attempts to decode after each incremental transmission, based on all received symbols.
Transmission Scheme Details (1st transmission)

- $k = \log_2 M = \text{information bits.}$

1st transmission:
- Send $I_1$, decode with $N_1 = I_1$.
- $R_1 = k / N_1 = \text{initial code rate.}$
Transmission Scheme Details (2nd transmission)

- $k = \log_2 M = \text{information bits.}$

2nd transmission:
- Send $I_2$, decode with $N_2 = I_1 + I_2$.
- $R_2 = k/N_2 = \text{code rate.}$
Transmission Scheme Details ($i$th transmission)

- $k = \log_2 M = \text{information bits.}$

- **$i$th transmission**: ($i = 2, \ldots, m$)
  - Send $I_i$, decode with $N_i = N_{i-1} + I_i$.
  - $I_i = \text{incremental step size}$, $N_i = \text{block length at $i$th transmission}$.
  - $R_i = k/N_i = \text{code rate at $i$th transmission}$.

- $m = \text{maximum number of transmissions (before repetition)}$. 
If decoding is unsuccessful after $m$ transmissions, start over by sending $I_1$ bits, then $I_2$ bits, etc. (similar to ARQ).

This is a practical limitation.

Simplifies analysis.
Decoding Error Probability for Sphere-Packing

- \( P[\text{error with block length } N_i] = P(\zeta_i) = P \left( \sum_{\ell=1}^{N_i} z_{\ell}^2 > r_i^2 \right) = 1 - F_{\chi_{N_i}^2} (r_i^2) \),

- \( r_i^2 = \frac{N_i(1+\eta)}{2^{2k/N_i}} \) is the sphere-packing radius (squared),
- \( z_{\ell} \sim \mathcal{N}(0, 1) \) are the noise samples.
An ideal sphere-packing codebook is mythical.

⇒ Upper bound on packing density $\phi$ in $n$ dimensions:

$$\phi \leq \left(\frac{n}{e}\right) 2^{-n/2}.$$ 

... but we will see that a convolutional code can achieve sphere-packing performance.
Marginal vs. Joint Decoding Error Probability

- $\Pr[\text{error with block length } N_i] = \Pr(\zeta_i)$ (marginal)
  
  \[ = \Pr\left( \sum_{\ell=1}^{N_i} z_\ell^2 > r_i^2 \right) = 1 - F_{\chi_{N_i}^2}(r_i^2), \]

- $\Pr[\text{error after } j \text{ transmissions}] = \Pr(\zeta_1, \zeta_2, \ldots, \zeta_j)$ (joint)
  
  \[ = \Pr\left( \sum_{\ell=1}^{N_1} z_\ell^2 > r_1^2, \sum_{\ell=1}^{N_2} z_\ell^2 > r_2^2, \ldots, \sum_{\ell=1}^{N_j} z_\ell^2 > r_j^2 \right) \]
  
  \[ = \int_{r_1^2}^{\infty} \int_{r_2^2-t_1}^{\infty} \cdots \int_{r_{j-1}^2-\sum_{i=1}^{j-2} t_i}^{\infty} f_{\chi_{I_1}^2}(t_1) \cdots f_{\chi_{I_{j-1}}^2}(t_{j-1}) \times \]
  
  \[ \left( 1 - F_{\chi_{I_j}^2}\left( r_j^2 - \sum_{i=1}^{j-1} t_i \right) \right) dt_{j-1} \cdots dt_1. \]
Latency and Throughput (for \( m = 1 \), the ARQ Case)

- \( \lambda = \text{latency} = \) expected number of forward channel uses.

\[
\lambda = I_1 \left( 1 + P(\zeta_1) + P(\zeta_1)^2 + P(\zeta_1)^3 + \ldots \right) \\
= \frac{I_1}{1 - P(\zeta_1)} \\
= \frac{I_1}{F_{\chi^2_{N_1}}(r_{1}^2)}
\]

- \( R_t = \text{throughput} = k/\lambda \).

- Select \( I_1 \) to maximize \( R_t \).
What About $m > 1$?

- **Latency**

  \[
  \lambda = \frac{I_1 + \sum_{i=2}^{m} I_i P \left( \bigcap_{j=1}^{i-1} \zeta_j \right)}{1 - P \left( \bigcap_{j=1}^{m} \zeta_j \right)}
  \]

- **Throughput**

  \[
  R_t = \frac{k}{\lambda}
  \]

- Select \(\{I_1, I_2, \ldots, I_m\}\) to maximize \(R_t\).
RCSP: Latency vs. Throughput for $m = 1$ (ARQ) Using Optimal Step Size $I_1$

SNR = 2.0 dB, Capacity = 0.6851

Latency $\lambda$

Throughput $R_t$

$k = 16$

$k = 32$

$k = 64$

$k = 128$

$k = 256$

Capacity

$m = 1$ analysis
RCSP: Latency vs. Throughput for $m = 1$ to $m = 6$, Using Optimal Step Sizes $I_i$

SNR = 2.0 dB, Capacity = 0.6851

Throughput $R_t$ vs. Latency $\lambda$ for $m=1$ to $m=6$, with optimal step sizes $I_i$. The plot shows the throughput capacity for different values of $k$ and $m$.
RCSP: Latency vs. Throughput for $m = 5$, Using Optimal Step Sizes $I_i$

SNR = 2.0 dB, Capacity = 0.6851
Comparison with [Polyanskiy et al. 2011]
Convolutional Code Simulations for $m = 5$

- Mother codes are rate $1/3$, 64-state and 1024-state convolutional codes from [Lin and Costello 2004].

- Use transmission lengths $\{I^m_1\}$ identified in RCSP optimization for $m = 5$.

- High-rate codes obtained by pseudo-random puncturing of mother codes.

- **Maximum likelihood (ML) decoding.**
  - ML decoding regions completely fill the power constraint sphere.

- Tail-biting implementations used for throughput efficiency.
Convolutional Code Achievability, $m = 5$

SNR = 2.0 dB, Capacity = 0.6851

- 90% of AWGN capacity in ~100 symbols.
Decoding Error Trajectory

SNR = 2.0 dB, Capacity = 0.6851, $k = 64$

$m = 5$ RCSP analysis
Decoding Error Trajectory

SNR = 2.0 dB, Capacity = 0.6851, $k = 64$

$m = 5$ RCSP analysis
$m = 5$ 64-state conv. code
Decoding Error Trajectory

SNR = 2.0 dB, Capacity = 0.6851, \( k = 64 \)

- \( m = 5 \) RCSP analysis
- \( m = 5 \) 64-state conv. code
- \( m = 5 \) 1024-state conv. code
Decoding Error Trajectory

SNR = 2.0 dB, Capacity = 0.6851, $k = 64$

- $m = 5$ RCSP analysis
- $m = 5$ 64-state conv. code
- $m = 5$ 1024-state conv. code
- VLFT code ($m=5$ block lengths)
Decoding Error Trajectory

SNR = 2.0 dB, Capacity = 0.6851, $k = 64$

- $m = 1$ RCSP analysis
- $m = 2$ RCSP analysis
- $m = 3$ RCSP analysis
- $m = 4$ RCSP analysis
- $m = 5$ RCSP analysis
- $m = 6$ RCSP analysis
- $m = 5$ 64-state conv. code
- $m = 5$ 1024-state conv. code
- VLFT code ($m=5$ block lengths)
SNR = 2.0 dB, Capacity = 0.6851, \( k = 64 \)

Marginal \( P(\zeta) = 1 - F_{\chi^2_N} (r^2) \)

- \( m = 1 \) RCSP analysis
- \( m = 2 \) RCSP analysis
- \( m = 3 \) RCSP analysis
- \( m = 4 \) RCSP analysis
- \( m = 5 \) RCSP analysis
- \( m = 6 \) RCSP analysis
- \( m = 5 \) 64-state conv. code
- \( m = 5 \) 1024-state conv. code
SNR = 2.0 dB, Capacity = 0.6851, $m = 5$

$R_1 > C$
Concluding Thoughts

- Feedback improves achievable rate for finite block lengths.
  - Feedback after every bit is best.
  - When transmissions must be grouped, pick the sizes wisely.
- Find good codes by matching RCSP error trajectories.
- Questions?