An introduction to coding theory

Adrish Banerjee

Department of Electrical Engineering
Indian Institute of Technology Kanpur
Kanpur, Uttar Pradesh
India

Feb. 20, 2017

Lecture #11C: Problem solving session-V

Convolutional codes

- **Problem # 1:** Feedforward encoder inverse: How do we find an estimate of the information sequence from the received sequence without decoding?

Convolutional codes

- **Problem # 1:** Feedforward encoder inverse: How do we find an estimate of the information sequence from the received sequence without decoding?
- **Solution:** Since $\mathbf{v}(\mathbf{D}) = \mathbf{u}(\mathbf{D})\mathbf{G}(\mathbf{D})$, we need an inverse matrix $\mathbf{G}^{-1}(\mathbf{D})$ such that $\mathbf{G}(\mathbf{D})\mathbf{G}^{-1}(\mathbf{D}) = D^l$ for some positive l.

Convolutional codes

- **Problem # 1:** Feedforward encoder inverse: How do we find an estimate of the information sequence from the received sequence without decoding?
- **Solution:** Since $\mathbf{v(D)} = \mathbf{u(D)G(D)}$, we need an inverse matrix $\mathbf{G^{-1}(D)}$ such that $\mathbf{G(D)G^{-1}(D)} = D^l$ for some positive l.
- Then $\mathbf{v(D)G^{-1}(D)} = \mathbf{u(D)G(D)G^{-1}(D)} = \mathbf{u(D)I}D^l = \mathbf{u(D)}D^l$

Convolutional codes

- **Problem # 1:** Feedforward encoder inverse: How do we find an estimate of the information sequence from the received sequence without decoding?
- **Solution:** Since $\mathbf{v(D)} = \mathbf{u(D)G(D)}$, we need an inverse matrix $\mathbf{G^{-1}(D)}$ such that $\mathbf{G(D)G^{-1}(D)} = D^l$ for some positive l.
- Then $\mathbf{v(D)G^{-1}(D)} = \mathbf{u(D)G(D)G^{-1}(D)} = \mathbf{u(D)I}D^l = \mathbf{u(D)}D^l$
- For a rate R=1/n code, a feedforward inverse exists if and only if

$$\text{GCD}\{\mathbf{g^0(D)}, \mathbf{g^1(D)}, \cdots, \mathbf{g^{n-1}(D)}\} = D^l$$

Convolutional codes

- **Problem # 1:** Feedforward encoder inverse: How do we find an estimate of the information sequence from the received sequence without decoding?
- **Solution:** Since $\mathbf{v}(\mathbf{D}) = \mathbf{u}(\mathbf{D})\mathbf{G}(\mathbf{D})$, we need an inverse matrix $\mathbf{G}^{-1}(\mathbf{D})$ such that $\mathbf{G}(\mathbf{D})\mathbf{G}^{-1}(\mathbf{D}) = D^l$ for some positive l.
- Then $\mathbf{v}(\mathbf{D})\mathbf{G}^{-1}(\mathbf{D}) = \mathbf{u}(\mathbf{D})\mathbf{G}(\mathbf{D})\mathbf{G}^{-1}(\mathbf{D}) = \mathbf{u}(\mathbf{D})\mathbf{I}D^l = \mathbf{u}(\mathbf{D})D^l$
- For a rate R=1/n code, a feedforward inverse exists if and only if

$$\text{GCD}\{\mathbf{g}^{\mathbf{0}}(\mathbf{D}), \mathbf{g}^{\mathbf{1}}(\mathbf{D}), \cdots, \mathbf{g}^{\mathbf{n-1}}(\mathbf{D})\} = D^l$$

- For a rate k/n code, a feedforward inverse exists if and only if

$$\text{GCD}\{\mathbf{\Delta}_{\mathbf{i}}(\mathbf{D}) = D^l\}$$

where $\{\mathbf{\Delta}_{\mathbf{i}}(\mathbf{D})\}$ is the set of all determinants of $k \times k$ submatrices.

Convolutional codes

- Let $\mathbf{G}(\mathbf{D}) = [1 + D^2 + D^3 \quad 1 + D + D^2 + D^3]$.

Convolutional codes

- Let $\mathbf{G(D)} = [1 + D^2 + D^3 \quad 1 + D + D^2 + D^3]$.
- GCD $\{1 + D^2 + D^3, 1 + D + D^2 + D^3\} = 1$. Hence feedforward inverse exists.

Convolutional codes

- Let $\mathbf{G(D)} = [1 + D^2 + D^3 \quad 1 + D + D^2 + D^3]$.
- GCD $\{1 + D^2 + D^3, 1 + D + D^2 + D^3\} = 1$. Hence feedforward inverse exists.
- In this case,

$$\mathbf{G^{-1}(D)} = \begin{pmatrix} 1 + D + D^2 \\ D + D^2 \end{pmatrix}$$

Adrish Banerjee Department of Electrical Engineering Indian Institute of Technology Kanpur Kanpur, Uttar Pradesh India
An introduction to coding theory

Convolutional codes

- Let

$$\mathbf{G(D)} = \begin{pmatrix} 1+D & D & 1+D \\ D & 1 & 1 \end{pmatrix}$$

Convolutional codes

- Let

$$\mathbf{G(D)} = \begin{pmatrix} 1+D & D & 1+D \\ D & 1 & 1 \end{pmatrix}$$

- 2×2 determinants are given by $\{1+D+D^2, 1+D^2, 1\}$

Convolutional codes

- Let

$$\mathbf{G(D)} = \begin{pmatrix} 1+D & D & 1+D \\ D & 1 & 1 \end{pmatrix}$$

- 2×2 determinants are given by $\{1+D+D^2, 1+D^2, 1\}$
- GCD $\{1+D+D^2, 1+D^2, 1\} = 1$

Convolutional codes

- Let

$$\mathbf{G(D)} = \begin{pmatrix} 1+D & D & 1+D \\ D & 1 & 1 \end{pmatrix}$$

- 2×2 determinants are given by $\{1+D+D^2, 1+D^2, 1\}$
- GCD $\{1+D+D^2, 1+D^2, 1\} = 1$
- Hence, feedforward inverse exists and is given by

$$\mathbf{G^{-1}(D)} = \begin{pmatrix} 0 & 0 \\ 1 & 1+D \\ 1 & D \end{pmatrix}$$

Convolutional codes

- Catastrophic encoders do not have a feedforward inverse.

Convolutional codes

- Catastrophic encoders do not have a feedforward inverse.
- Example. Consider a convolutional encoder with $G(D) = [1 + D \quad 1 + D^2]$.

Convolutional codes

- Catastrophic encoders do not have a feedforward inverse.
- Example. Consider a convolutional encoder with $G(D) = [1 + D \quad 1 + D^2]$.
- GCD $\{1 + D, 1 + D^2\} = 1 + D$.

Convolutional codes

- Catastrophic encoders do not have a feedforward inverse.
- Example. Consider a convolutional encoder with $G(D) = [1 + D \quad 1 + D^2]$.
- GCD $\{1 + D, 1 + D^2\} = 1 + D$.
- There doesn't exist any feedforward inverse.

Convolutional codes

- Catastrophic encoders do not have a feedforward inverse.
- Example. Consider a convolutional encoder with $G(D) = [1 + D \quad 1 + D^2]$.
- GCD $\{1 + D, 1 + D^2\} = 1 + D$.
- There doesn't exist any feedforward inverse.
- Let $\mathbf{u}(\mathbf{D}) = 1(1 + D) = 1 + D + D^2 + \cdots$, then

$$\mathbf{v}(\mathbf{D}) = \mathbf{u}(\mathbf{D})\mathbf{G}(\mathbf{D}) = 1/(1 + D)[1 + D \quad 1 + D^2] = [1 \quad 1 + D]$$

Convolutional codes

- Catastrophic encoders do not have a feedforward inverse.
- Example. Consider a convolutional encoder with $G(D) = [1 + D \quad 1 + D^2]$.
- GCD $\{1 + D, 1 + D^2\} = 1 + D$.
- There doesn't exist any feedforward inverse.
- Let $\mathbf{u}(\mathbf{D}) = 1(1 + D) = 1 + D + D^2 + \cdots$, then

$$\mathbf{v}(\mathbf{D}) = \mathbf{u}(\mathbf{D})\mathbf{G}(\mathbf{D}) = 1/(1 + D)[1 + D \quad 1 + D^2] = [1 \quad 1 + D]$$

- This is a catastrophic encoder since infinite input weight sequence will result in finite weight output sequence.

Convolutional codes

- **Problem # 2:** If a class of rate 1/2 nonsystematic feedforward convolutional encoders with $\mathbf{G(D)} = [\mathbf{g^0(D)} \quad \mathbf{g^1(D)}]$ satisfy

$$\mathbf{g^0(D)} + D^{\beta}\mathbf{g^1(D)} = D^{\alpha}$$

They are known as quick-look in (QLI) encoders. Show that QLI encoders are noncatastrophic encoders.

Convolutional codes

- **Problem # 2:** If a class of rate 1/2 nonsystematic feedforward convolutional encoders with $\mathbf{G(D)} = [\mathbf{g^0(D)} \quad \mathbf{g^1(D)}]$ satisfy

$$\mathbf{g^0(D)} + D^{\beta}\mathbf{g^1(D)} = D^{\alpha}$$

They are known as quick-look in (QLI) encoders. Show that QLI encoders are noncatastrophic encoders.

- **Solution:** QLI encoders have a simple feedforward inverse

$$\mathbf{G^{-1}(D)} = \begin{pmatrix} 1 \\ D^{\beta} \end{pmatrix}$$

hence are noncatastrophic

Convolutional codes

- **Problem # 2:** If a class of rate 1/2 nonsystematic feedforward convolutional encoders with $\mathbf{G(D)} = [\mathbf{g^0(D)} \quad \mathbf{g^1(D)}]$ satisfy

$$\mathbf{g^0(D)} + D^\beta \mathbf{g^1(D)} = D^\alpha$$

They are known as quick-look in (QLI) encoders. Show that QLI encoders are noncatastrophic encoders.

- **Solution:** QLI encoders have a simple feedforward inverse

$$\mathbf{G^{-1}(D)} = \begin{pmatrix} 1 \\ D^\beta \end{pmatrix}$$

hence are noncatastrophic

- Further, the information sequence $\mathbf{u(D)}$ can be recovered directly from $\mathbf{v(D)} = [\mathbf{v^0(D)} \quad \mathbf{v^1(D)}]$ using an encoder inverse with only two taps.

$$\mathbf{v(D)G^{-1}(D)} = \mathbf{v^0(D)} + D^\beta \mathbf{v^1(D)} = D^\alpha \mathbf{u(D)}$$

Convolutional codes

- **Problem # 3:** An important distance measure for convolutional codes is the column distance function. Let

$$[\mathbf{v}]_l = (v_0^{(0)} v_0^{(1)} \cdots v_1^{(n-1)}, v_1^{(0)} v_1^{(1)} \cdots v_1^{(n-1)}, \cdots, v_l^{(0)} v_l^{(1)} \cdots v_l^{(n-1)})$$

denote the lth truncation of the codeword $\mathbf{v}$ and let

$$[\mathbf{u}]_l = (u_0^{(0)} u_0^{(1)} \cdots u_1^{(n-1)}, u_1^{(0)} u_1^{(1)} \cdots u_1^{(n-1)}, \cdots, u_l^{(0)} u_l^{(1)} \cdots u_l^{(n-1)})$$

denote the lth truncation of the information sequence $\mathbf{u}$. The column distance function of order l, d_l is defined as

$$\begin{aligned} d_l &= \min_{[\mathbf{u}']_l, [\mathbf{u}'']_l} \{d([\mathbf{v}']_l, [\mathbf{v}'']_l) : [\mathbf{u}']_0 \neq [\mathbf{u}'']_0\} \\ &= \min_{[\mathbf{u}]_\mathbf{j}} \{w[\mathbf{v}]_l : [\mathbf{u}]_0 \neq 0\} \end{aligned}$$

where $\mathbf{v}$ is the codeword corresponding to the information sequence $\mathbf{u}$. Prove that for noncatastrophic encoders

$$\lim_{l \to \infty} d_l = d_{\text{free}}$$

Convolutional codes

- **Solution:** By definition, d_{free} is the minimum weight path that has diverged from and remerged with the all-zero state.

Convolutional codes

- **Solution:** By definition, d_{free} is the minimum weight path that has diverged from and remerged with the all-zero state.
- Assume that $[\mathbf{v}]_j$ represents the shortest remerged path through the state diagram with weight d_{free} .

Convolutional codes

- **Solution:** By definition, d_{free} is the minimum weight path that has diverged from and remerged with the all-zero state.
- Assume that $[\mathbf{v}]_{\text{j}}$ represents the shortest remerged path through the state diagram with weight d_{free} .
- Let $[d_l]_{\text{re}}$ be the minimum weight of all remerged paths of length l, it follows that $[d_l]_{\text{re}} = d_{\text{free}}$ for all $l \geq j$.

Convolutional codes

- **Solution:** By definition, d_{free} is the minimum weight path that has diverged from and remerged with the all-zero state.
- Assume that $[\mathbf{v}]_{\text{j}}$ represents the shortest remerged path through the state diagram with weight d_{free} .
- Let $[d_l]_{\text{re}}$ be the minimum weight of all remerged paths of length l, it follows that $[d_l]_{\text{re}} = d_{\text{free}}$ for all $l \geq j$.
- Also, for a noncatastrophic encoder, any path that remains unmerged must accumulate weight.

Convolutional codes

- **Solution:** By definition, d_{free} is the minimum weight path that has diverged from and remerged with the all-zero state.
- Assume that $[\mathbf{v}]_{\text{j}}$ represents the shortest remerged path through the state diagram with weight d_{free} .
- Let $[d_l]_{\text{re}}$ be the minimum weight of all remerged paths of length l, it follows that $[d_l]_{\text{re}} = d_{\text{free}}$ for all $l \geq j$.
- Also, for a noncatastrophic encoder, any path that remains unmerged must accumulate weight.
- Let $[d_l]_{\text{un}}$ be the minimum weight of all unmerged paths of length l, it follows that

$$\lim_{l \to \infty} [d_l]_{\text{un}} \to \infty$$

Convolutional codes

- **Solution:** By definition, d_{free} is the minimum weight path that has diverged from and remerged with the all-zero state.
- Assume that $[\mathbf{v}]_{\text{j}}$ represents the shortest remerged path through the state diagram with weight d_{free} .
- Let $[d_l]_{\text{re}}$ be the minimum weight of all remerged paths of length l, it follows that $[d_l]_{\text{re}} = d_{\text{free}}$ for all $l \geq j$.
- Also, for a noncatastrophic encoder, any path that remains unmerged must accumulate weight.
- Let $[d_l]_{\text{un}}$ be the minimum weight of all unmerged paths of length l, it follows that

$$\lim_{l \to \infty} [d_l]_{\text{un}} \to \infty$$

- Therefore

$$\lim_{l \to \infty} = \min \left\{ \lim_{l \to \infty} [d_l]_{\text{re}}, \lim_{l \to \infty} [d_l]_{\text{un}} \right\} = d_{\text{free}}$$