Trellises for Linear Block Codes

DESPOINA GEORGIADOU

Chania 2008
Outline

1 Introduction
   • Block Codes and Trellis Representation
   • The generator matrix $G$

2 Construction of a bit-level trellis
   • Trellis Construction With Generator Matrix

3 Trellis Complexity

4 Parallel decomposition
The FSM Model

For every block code we know that:

- it has finite memory, that stores input information for a certain finite time interval
- symbols stored in memory and the current input affect the output code symbols according to a certain encoding rule
- at any encoding time symbols stored in memory specify a state of the encoder at that time instant
- since encoder’s memory has finite size, the allowable states are also finite
- when new symbols are shifted in memory, some old are shifted out of it causing a state transition

So, an encoder can be modeled as a finite-state machine (FSM).
This dynamic behaviour of a encoder can be graphically represented by a trellis diagram!
Let \( a_i \) be the encoder's input bit at time-\( i \), \( u_j \) the output bit at time-\( i \) and \( g_j \) the rows of the generator matrix, then:

\[
(a_0, a_1, \ldots, a_{k-1})^* \begin{pmatrix}
g_0 \\
g_1 \\
\vdots \\
g_{k-1}
\end{pmatrix} = (a_0 g_0 + a_1 g_1 + \ldots + a_{k-1} g_{k-1})
\]

\[
= \begin{bmatrix}
a_0 g_{00} & a_0 g_{01} & \ldots & a_0 g_{0(n-1)} \\
a_1 g_{10} & a_1 g_{11} & \ldots & a_1 g_{1(n-1)} \\
\vdots & \vdots & \ddots & \vdots \\
a_{k-1} g_{(k-1)0} & a_{k-1} g_{(k-1)1} & \ldots & a_{k-1} g_{(k-1)(n-1)}
\end{bmatrix}
\]

\[
= \sum_{i=0}^{n-1} a_{k-1} g_{(k-1)i} + \sum_{i=0}^{n-1} a_{k-1} g_{(k-1)(n-1)}
\]

\[
= (u_0, u_1, \ldots, u_{n-1})
\]
About trellis

Last time we talked about states (or nodes or vertices), branches and their labels.

Today we define:

\[ \Gamma = \{0, 1, 2, \ldots, n\} \]: the entire encoding interval (span), is a sequence of all encoding time instants

state space \( \sum_i(C) \): all allowable states at a given time instant \( i \)

state space dimension at time-\( i \): \( p_i(C) = \log_2|\sum_i(C)| \)

state space complexity profile: the sequence \( \{ |\sum_0(C)|, |\sum_1(C)|, \ldots, |\sum_n(C)| \} \)

state space dimension profile: the sequence \( (p_0, p_1, \ldots, p_n) \)
The generator matrix $G$

**Trellis Oriented form of $G$**

A generator matrix $G$ is in **Trellis Oriented Form (TOF)** when for each row $g \in \{g_0, g_1, \ldots, g_{k-1}\}$

- The **leading 1**$^a$ appears in a column before the leading 1 of any row below it.
- No two rows have their **trailing 1**$^b$ in the same column.

---

$a$ the first nonzero component of a row
$b$ the last nonzero component of a row

For $G_{TOGM}$ (trellis oriented generator matrix) we have:

1. **digit (or bit) span** of $g$, $\phi(g) = \{i, \ldots, j\}$, is the smallest index interval containing all nonzero components of $g$.
2. **time span** of $g$, $\tau(g) = \{i, \ldots, j+1\}$, same as bit span in terms of time
3. **Active time span**, $\tau_a(g) = [i+1, j]$, for $j > i$. 
The generator matrix $G$

**Examples**

Every generator matrix can be put in TOF (by elementary row operations) which is not necessarily systematic form.

Matrix $G = \begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1
\end{pmatrix}$ is not in TOF, whereas $G_{TOGM} = \begin{pmatrix}
g_0 \\
g_1 \\
g_2 \\
g_3
\end{pmatrix} = \begin{pmatrix}
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1
\end{pmatrix}$ is. The $G_{TOGM}$ matrix is the $G$ with the second and the fourth rows interchanged and the fourth row added to all the others.

**Bit span an active time span example**

- $\phi(g_3) = [4, 7]$
- $\tau_a(g_3) = [5, 7]$
- $\tau(g_3) = [4, 8]$
The generator matrix \( G \)

**Partitioning matrix \( G_{TOGM} \) of a \((n,k)\) linear code \( C \)**

At time-\( i \), \( 0 \leq i \leq n \) we partition the rows of \( G_{TOGM} \) into:

1. \( G^{p}_i \): rows with bit span in the interval \([0,i-1]\)
2. \( G^{f}_i \): rows with bit span in the interval \([i,n-1]\)
3. \( G^{s}_i \): rows whose active time spans contain time-\( i \)

We can respectively partition information bits \( a_0, a_1, ..., a_{k-1} \) into:

1. \( A^{p}_i \): bits that don’t affect the encoder output after time-\( i \)
2. \( A^{f}_i \): bits that affect the encoder output only after time-\( i \)
3. \( A^{s}_i \): bits that affect the output both before and after time-\( i \)

So \( A^{s}_i \) is the encoder’s memory (or state) at time-\( i \) and \( |A^{s}_i| = p_i \)!
The generator matrix $G$

Partitioning matrix $G_{TOGM}$ of a $(n,k)$ linear code $C$ - Example

So, for $G_{TOGM} = \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$ we have:

<table>
<thead>
<tr>
<th>Time</th>
<th>$G_i^p$</th>
<th>$G_i^f$</th>
<th>$G_i^s$</th>
<th>$p_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\phi$</td>
<td>$g_0,g_1,g_2,g_3$</td>
<td>$\phi$</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>$\phi$</td>
<td>$g_1,g_2,g_3$</td>
<td>$g_0$</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>$\phi$</td>
<td>$g_2,g_3$</td>
<td>$g_0,g_1$</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>$\phi$</td>
<td>$g_3$</td>
<td>$g_0,g_1,g_2$</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>$g_0$</td>
<td>$g_3$</td>
<td>$g_1,g_2$</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>$g_0$</td>
<td>$\phi$</td>
<td>$g_1,g_2,g_3$</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>$g_0,g_2$</td>
<td>$\phi$</td>
<td>$g_1,g_3$</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>$g_0,g_1,g_2$</td>
<td>$\phi$</td>
<td>$g_3$</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>$g_0,g_1,g_2,g_3$</td>
<td>$\phi$</td>
<td>$\phi$</td>
<td>0</td>
</tr>
</tbody>
</table>
We also mention that:

- $g^*$: the row in $G^f_i$ whose leading 1 is at bit position $i$
  - its uniqueness is guaranteed, but the existence is not.
- $a^*$: information bit that corresponds to row $g^*$ (current input information bit)

The output code bit generated between time-$i$ and time-$(i+1)$ is:

$$u_i = a^* + \sum_{l=1}^{p_i} (a^{(i)}_l \ast g^{(i)}_{l,i})$$

- $a^*$, if $g^*$ exists
- $\sum_{l=1}^{p_i} (a^{(i)}_l \ast g^{(i)}_{l,i})$, if $g^*$ doesn’t exist

---

$g^{(i)}_{l,i}$ is the $i$th component of $g^{(i)}_l$ in $G^s_i$

In this case we can put $a^* = 0$ (dummy information bit)
The generator matrix $G$

**Output code bit - example**

For $G_{TOGM} = \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$ we have:

- $G_2^s = \{g_0, g_1\}$ and $A_2^s = \{a_0, a_1\}$ ($a_0, a_2$ define the encoder's state at time-2)
- $g^* = g_2$ and $a^* = a_2$ ($a_2$ is the current input)
- $u_2 = a^* + \sum_{i=1}^{p_2} (a_i^{(2)} \ast g_i^{(2)}) = a_2 + a_0 g_{02} + a_1 g_{12} = a_2 + a_0 \ast 1 + a_1 \ast 0 = a_2 + a_0$
We also mention that:

- $g^0_i$ : the row in $G^s_i$ whose trailing 1 is at bit position $i$
  - $g^0_i$ is the last nonzero component of $g^0$ (if it exists)

- $a^0_i$ : information bit in $A^s_i$ that corresponds to row $g^0$ (current input information bit)
  - it is the oldest bit information in memory at time-$i$

At time-$(i+1)$ we have:

- $G^s_{i+1} = (G^s_i \setminus \{g^0\}) \cup g^*$, if $g^0$ exists
- $A^s_{i+1} = (A^s_i \setminus \{a^0\}) \cup a^*$, if $a^0$ exists
State labeling
In a code trellis, each state is labeled based on the information set that defines the state space at a particular encoding time instant.
The label \( l(s) \) of a state \( s \) is set to zero except for the components at the positions corresponding to the information bits in \( A^s_i = \{ a^i_1, a^i_2, ..., a^i_{p_i} \} \).
Thus, the label of the state \( s_i \) is: \( l(s_i) = (a^i_1, a^i_2, ..., a^i_{p_i}) \).

Labeling Example
If at time \( i=4 \) we have \( A^s_4 = \{ a_1, a_2 \} \), then \( l(s_4) = (0, a_1, a_2, 0) \).
Now we are ready to construct a trellis diagram:

To construct the bit-level trellis diagram, all we need is:

- $G_i^{s+1}$
- $A_i^s$
- $A_i^{s+1}$
- $A_{i+1}^s$
Construction of a bit-level trellis with generator matrix

The construction steps are:

1. Determine $G_{i+1}^s$ and $A_{i+1}^s$.
2. Form the state space $\sum_{i+1}(C)$ at time-(i+1).
3. For each state $s_i \in \sum_i(C)$, determine its transition(s) based on the change from $A_i^s$ to $A_{i+1}^s$ (bits $a^0$ and $a^*$).
4. Connect $s_i$ to its adjacent state(s) in $\sum_{i+1}(C)$ by branches.
5. For each transition, determine the output code bit $u_i$.
6. Use this $u_i$ to label the corresponding branch.
Construction of a bit-level trellis

So for $G_{TOGM} = \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$ we have:

<table>
<thead>
<tr>
<th>Time</th>
<th>$G^s_i$</th>
<th>$a^*$</th>
<th>$a^0$</th>
<th>$A^s_i$</th>
<th>State Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\phi$</td>
<td>$a_0$</td>
<td>-</td>
<td>$\phi$</td>
<td>(0000)</td>
</tr>
<tr>
<td>1</td>
<td>$g_0$</td>
<td>$a_1$</td>
<td>-</td>
<td>$a_0$</td>
<td>(a0000)</td>
</tr>
<tr>
<td>2</td>
<td>$g_0,g_1$</td>
<td>$a_2$</td>
<td>-</td>
<td>$a_0,a_1$</td>
<td>(a0a100)</td>
</tr>
<tr>
<td>3</td>
<td>$g_0,g_1,g_2$</td>
<td>-</td>
<td>$a_0$</td>
<td>$a_0,a_1,a_2$</td>
<td>(a0a1a20)</td>
</tr>
<tr>
<td>4</td>
<td>$g_1,g_2$</td>
<td>$a_3$</td>
<td>-</td>
<td>$a_1,a_2$</td>
<td>(0a1a20)</td>
</tr>
<tr>
<td>5</td>
<td>$g_1,g_2,g_3$</td>
<td>-</td>
<td>$a_2$</td>
<td>$a_1,a_2,a_3$</td>
<td>(0a1a3a3)</td>
</tr>
<tr>
<td>6</td>
<td>$g_1,g_3$</td>
<td>-</td>
<td>$a_1$</td>
<td>$a_1,a_3$</td>
<td>(0a10a3)</td>
</tr>
<tr>
<td>7</td>
<td>$g_3$</td>
<td>-</td>
<td>$a_3$</td>
<td>$a_3$</td>
<td>(000a3)</td>
</tr>
<tr>
<td>8</td>
<td>$\phi$</td>
<td>-</td>
<td>-</td>
<td>$\phi$</td>
<td>(0000)</td>
</tr>
</tbody>
</table>
Computing all $u_i$ for labeling the branches, we come to the end!

So, the trellis we obtain is:
A trellis’ complexity is:

- **the branch complexity**: the number of branches
- **the state complexity**: the maximum state space dimension $p_{max}(C)$

A minimal trellis

The trellis $T$ is said to be minimal if for any other $n$-section trellis $T'$ for $C$ with state space dimension profile $(p'_0, p'_1, p'_2, ..., p'_n)$ the following inequality holds: $p_i \leq p'_i$, for $0 \leq i \leq n$.

The minimal trellis has the minimal total number of states and is also a minimal branch trellis (with smallest branch complexity).
Parallel Decomposition For Minimal Trellis

Why?
A minimal code trellis is generally densely connected. So there is a difficulty in implementation.

To address that problem, we can decompose the trellis into parallel and structurally identical subtrellises of smaller dimensions without cross-connections between them.
Decomposition of minimal trellis into minimal parallel subtrellises

We define the following index set:
\( I_{\text{max}}(C) = \{ i : p_i(C) = p_{\text{max}}(C), \text{ for } 0 \leq i \leq n \} \)

**Theorem**

If there exists a row \( g \) in the \( G_{\text{TOGM}} \) for an \( (n,k) \) linear code \( C \) such that \( \tau_a(g) \supseteq I_{\text{max}}(C) \), then the subcode \( C_1 \) of \( C \) generated by \( G_{\text{TOGM}} \setminus \{ g \} \) has a minimal trellis \( T_1 \) with the maximum state space dimension \( p_{\text{max}}(C_1) = p_{\text{max}}(C) - 1 \), and
\( I_{\text{max}}(C_1) = I_{\text{max}}(C) \cup \{ i : p_i(C) = p_i(\text{max}) - 1, \text{ i not in } \tau_a(g) \} \).

If \( G \) is in TOF, \( G_1 = G \setminus \{ g \} \) is also in TOF. If theorem holds, then we can decompose \( C \) into two parallel and structurally identical subtrellises, one for \( C_1 \) and the other for the coset \( C_1 \oplus g \).
Parallel Decomposition-Example

For $G_{TOGM} = \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$ we have: $I_{max}(C)=[3,5]$ as state space dimension is $(0,1,2,3,2,3,2,1,0)$ and $p_{max}(C)=3$. The only row whose active time span $\tau_a$ contains $I_{max}(C)$ is $g_1$ where $\tau_a(g_1)=[2,6]$. So, $G_1 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$
Parallel Decomposition-Example

Constructing the trellis for both $C_1$ and $g_1 \oplus C_1$ we get the following parallel decomposition: