

1.1.1.1 Tail biting veterbi 的译码

LTE 的 verterbi 编码是 tail biting 的，没有尾比特，所以译码的时候要对 branch metric 数据做前后扩展(如下图)。译码厚的结果要截取中间部分的 hardbits。下面是一个例子：

假设编码的比特数是 56，编码速率是 1/3，编码器约束是 7 (6 级延迟寄存器)

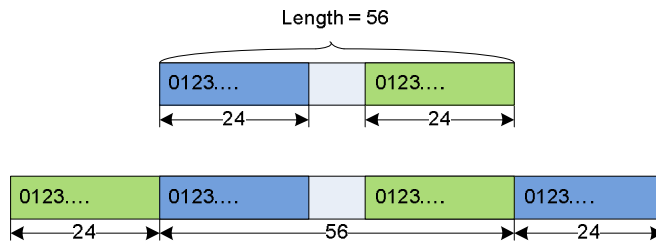
编码比特流是 0x12, 0x34, 0x56, 0x78, 0x9a, 0xbc, 0xde,

计算出来的 branch metric 的 size 是 56×4 。

前扩展的 size 是: $4(\text{BM values per stage}) \times 6(\text{延迟寄存器}) \times 4(4 \text{ 倍约束})$

前扩展的 size 是: $4(\text{BM values per stage}) \times 6(\text{延迟寄存器}) \times 5(5 \text{ 倍约束})$

如下图：



编码器约束是7，编码速率是1/3，所以每个stage的branchmetric有4个softbits
所以patch的时候首尾各加上 $(7-1) \times 4 = 24$

译码的输出如下图：

```
0x00813EED vcpHdBuf
0x00813EED 9A BC DE 12 34 56 78 9A BC DE 12 34 57
```

除掉前扩展的 24 个比特，结果就是 0x12, 0x34, 0x56, 0x78, 0x9a, 0xbc, 0xde