组会报告

徐益

2018 年 8 月 27 日

1 工作内容

1. 实现与原 fixed-layered-OMS 误码一致的 avx2-OMS 译码模块，并测试吞吐量。
2. 尝试提高吞吐量。
3. 尝试解决 fixed-OMS 译码曲线出现的“平台”现象。

2 实现 avx2-OMS 译码模块

表 1: 不同译码算法的吞吐量 ($SNR = 1.6dB(BER < 10^{-5}), N = 25344, R = 0.5, I_{max} = 10$)

scheduling	BP	float-OMS	fixed-OMS	avx2-OMS
Throuput	0.190Mbps	0.572Mbps	0.4160Mbps	6.744Mbps

3 尝试提高吞吐量

加入判决机制:

表 2: 不同译码算法的吞吐量 ($SNR = 1.6dB(BER < 10^{-5}), N = 25344, R = 0.5, I_{max} = 10$)

scheduling	avx2-OMS	avx2-OMS with judgment
Throuput	6.744Mbps	5.573Mbps

4 尝试解决 fixed-OMS 译码曲线出现的“平台”现象

4.1 消除 float 转 int8_t 时 inf 造成的影响

原方案：

```
for (i = 0; i < Nd; i++)
{
        temp_llr = (int32_t)llr[i] * FACTOR_BETA;
        if (temp_llr > MAX_FIXED_MSG)
                llr_fixed[i] = MAX_FIXED_MSG;
        else if (temp_llr < MIN_FIXED_MSG)
                llr_fixed[i] = MIN_FIXED_MSG;
        else
                llr_fixed[i] = (int8_t)(llr[i]*FACTOR_BETA);
}
```

现方案：

```
for (i = 0; i < Nd; i++)
{
        if (llr[i] >= 4)
                llr_fixed[i] = MAX_FIXED_MSG;
        else if (llr[i] <= -4)
                llr_fixed[i] = MIN_FIXED_MSG;
        else
                llr_fixed[i] = (int8_t)(llr[i]*FACTOR_BETA);
}
```

4.2 使饱和后的 cn_msg 保持不变

原方案：

```
omin_llr = VECTOR_MAX(VECTOR_SUB(min_llr, vbeta), VECTOR_ZERO);
osubmin_llr = VECTOR_MAX(VECTOR_SUB(submin_llr, vbeta), VECTOR_ZERO);
```

现方案：

```
min_temp = VECTOR_MAX(VECTOR_SUB(min_llr, vbeta), VECTOR_ZERO);
submin_temp = VECTOR_MAX(VECTOR_SUB(submin_llr, vbeta), VECTOR_ZERO);
/* if(min_llr==max_msg) omin_llr=vmax_msg; else omin_llr=min_temp */
omin_llr = VECTOR_CMOV(min_llr, vmax_msg, vmax_msg, min_temp);
osubmin_llr = VECTOR_CMOV(submin_llr, vmax_msg, vmax_msg, submin_temp);
```

4.3 误码性能对比

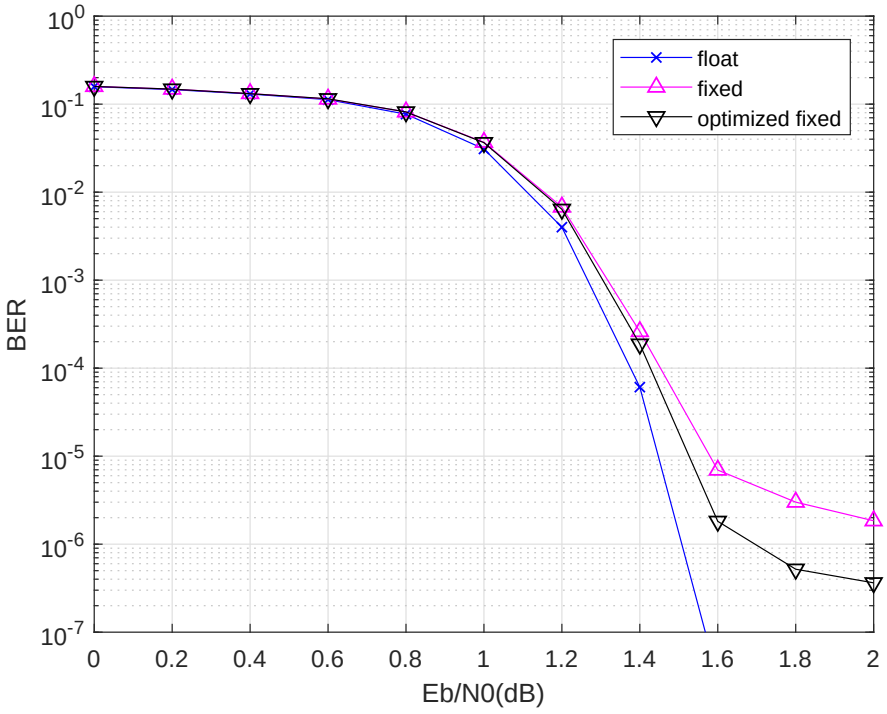

图 1: 优化前后误码性能对比 ($N = 25344, R = 0.5, I_{max} = 10$)

4.4 吞吐量对比

表 3: 不同译码算法的吞吐量 ($SNR = 1.6dB(BER < 10^{-5}), N = 25344, R = 0.5, I_{max} = 10$)

scheduling	avx2-OMS	optimized avx2-OMS
Throuput	6.744Mbps	6.689Mbps

5 下阶段计划

1. 进一步优化 avx2-oms 算法
2. 尝试 avx2-nms 算法的实现