

为什么 MoELSH 和 EPW 这种基于 MoE 的水印算法，在数学本质上也能享受“次线性衰减”(Sub-linear Decay) 的红利？

答案在于一个词：缓冲 (Buffering)。或者用数学术语说：Lipschitz 连续性 (Lipschitz Continuity)。

让我们像剥洋葱一样剥开这个问题。

## 1. 核心理论回顾：什么是“信号-攻击解耦”？

最初那篇论文的核心论点是：

- Token-Logit 水印 (如 KGW)：信号空间 = 攻击空间。你改一个词 (攻击)，哈希值全变 (信号丢失)。这是一对一的硬碰撞，衰减是线性的，甚至是指教级的。
- MoE 水印 (如 OKR 或 MoELSH)：攻击空间 (Token)  $\rightarrow$  Gating Network (缓冲层)  $\rightarrow$  信号空间 (Expert Weights/Choice)。

由于 Gating Network 是一个连续可导的函数 (通常是 Matrix Multiplication + Softmax)，它天然具有平滑性。攻击者对 Token 的离散修改，在经过 Gating 层时，被“平摊”成了连续空间上的微小扰动。

## 2. 为什么 MoELSH 也满足这个结论？

MoELSH 的工作流程是：Token 输入  $\rightarrow$  Router Weights (RW)  $\rightarrow$  LSH  $\rightarrow$  Green List

让我们看看攻击是如何在这个链条中衰减的：

### 第一层缓冲：Embedding 空间的几何稳定性

攻击者进行“释义攻击”(Paraphrase)。他把“开心”换成了“高兴”。在 Token ID 空间，这是剧变 (ID 从 1024 变成了 8975)。但在 Embedding 空间，向量  $\mathbf{e}_{\text{happy}}$  和  $\mathbf{e}_{\text{glad}}$  的夹角极小。结论：输入的变化被 Embedding 层降维打击了。

### 第二层缓冲：Gating 网络的 Lipschitz 约束

这是数学证明的核心。

$$\mathbf{rw} = \text{Softmax}(\mathbf{W} \cdot \mathbf{e})$$

只要权重矩阵  $W$  没有梯度爆炸（通常有正则化），这个变换就是 Lipschitz 连续的。

### 第三层缓冲：LSH 的角度容忍度

这是 MoELSH 特有的机制。它没有直接用  $\mathbf{rw}$ ，而是用了  $\text{sign}(\mathbf{R} \cdot \mathbf{rw})$ 。根据 LSH 的性质，两个向量哈希值碰撞（签名相同）的概率是：

$$P(\text{Signature Matches}) = 1 - \frac{\theta}{\pi}$$

其中  $\theta$  是  $\mathbf{rw}$  和  $\mathbf{rw}'$ （攻击后）的夹角。

由于第二层缓冲保证了  $\|\Delta \mathbf{rw}\|$  很小，所以夹角  $\theta$  也很小。 $\theta$  很小  $\rightarrow$  签名改变的概率极低  $\rightarrow$  Green List 保持不变。

## 3. 次线性衰减 ( $O(\sqrt{\gamma})$ ) 的数学来源

为什么是根号  $\sqrt{\gamma}$  而不是线性  $\gamma$ ? 这涉及到高维几何。

在论文中，攻击强度  $\gamma$  通常用 KL 散度或编辑距离衡量。但在高维空间（Embedding 维度通常是 4096+）中，一个随机的攻击向量（释义造成的扰动），大概率是与当前的 Gating 权重向量正交的。

想象一下，你在一个高维球面上。你随机推一下（攻击），你大概率是在沿着切线方向移动，而不是沿着半径方向（改变激活值）移动。因此，激活值（以及随后的 LSH 投影）的变化幅度，通常与攻击强度的平方根成正比。

MoELSH 的鲁棒性正是来源于此：**攻击者必须施加巨大的扰动 (Token 面目全非)**，才能在 Routing 权重上产生足够大的角度偏差，从而翻转 LSH 的比特位。

## 4. 对比：为什么 KGW 不满足？

在 KGW 算法，就不满足这个性质。

KGW 的公式： $\text{Seed} = \text{Hash}(\text{Token}_{\{t-1\}}, \text{Token}_{\{t-2\}} \dots)$

哈希函数 (Hash) 是非连续的、混沌的。

- **输入：** 改一个 Token。
- **输出：** 种子完全改变（雪崩效应）。

- **衰减：** 瞬间归零 (Step Function)，或者线性衰减 (取决于窗口大小)。

KGW 没有“Gating Network”这个缓冲层，也没有 LSH 的几何容忍度。它是硬碰硬。

## 5. 结论：殊途同归的架构智慧

MoELSH，虽然和文稿中的依据Gating实现路径不同（前者用于生成 Green List，后者用于直接路由），但它们利用了同一个物理学原理来抵抗攻击：

**利用 MoE 架构的稀疏激活特性和 Gating 网络的平滑性，将离散的、剧烈的 Token 级攻击，转化为连续的、微小的向量级扰动，从而实现信号的存活。**

\*\*