

# 周报

2025 年 3 月 11 日

- XMSS 树结构的并行实现优化
- 论文的写作

# XMSS 树并行实现优化

XMSS 树是 HT 签名的基本组件，是签名中最耗时的部分。

## 两层次并行技术

- 节点级并行
  - 同一层次的节点并行计算
  - 二级并行实现 [WDC<sup>+</sup>25]
- WOTS 级并行
  - 单个节点内的 WOTS 链并行计算
  - 三级并行实现 [WDC<sup>+</sup>25]

## 动态调度与 GPU 优化

- 运行期间动态调整线程数
- 结合 GPU 特性进行性能优化

# 实验结果

操作	执行时间	性能提升
PKGGEN 串行实现 [WDC+25]	31.382 ms	基准
PKGGEN 二级并行 [WDC+25]	3.844 ms	8.16x
PKGGEN 二级并行 + 动态调度	3.822 ms	8.21x
PKGGEN 二三级并行 [WDC+25]	0.220 ms	142.65x
PKGGEN 二三级并行 + 动态调度 + GPU 优化	0.197 ms	159.30x

## 关键发现:

- 二级并行中动态调度提升有限 (8.16x  $\rightarrow$  8.21x)
- 三级并行中动态调度效果显著 (142.65x  $\rightarrow$  159.30x)
- 执行时间优化: 0.220ms  $\rightarrow$  0.197ms (10.9%提升)

- 核心理论基础: 动态调度作为性能提升的理论依据
  - 关键发现: 不同核函数  $g$  存在最优线程数  $t$  使性能达到最大
  - 数学模型:
    - 签名过程表示为运行序列  $((g_1, t_1), \dots, (g_n, t_n))$
    - 目标: 构建映射函数  $F: G \rightarrow T$
    - 为每个核函数  $g_i$  分配最优线程配置  $t_i$
- ① 最优线程配置映射函数  $F$ 
    - 基于核函数特性自动确定最优线程数
    - 减少人工调优需求, 提高通用性
  - ② 完整调度实现机制
    - 运行时线程分配与管理
    - 针对不同签名组件的专用优化
    - 最大化 GPU 资源利用率

继续加快推进，特别写作

反复打磨论文写作

## 下周计划

- ① 最优线程配置映射函数  $F$  写作
- ② 拓展动态调度实现到 FORS 等组件



Ziheng Wang, Xiaoshe Dong, Heng Chen, Yan Kang, and Qiang Wang.

Cuspx: Efficient gpu implementations of post-quantum signature sphincs<sup>+</sup>.

*IEEE Transactions on Computers*, 74(1):15–28, 2025.