目录

论文 1 小结：（ToN-review）A Trusted Threshold Cluster Group PBFT Consensus Algorithm for IoT Blockchain

（一）阅读思想方面：

1.阅读论文时，**从解决问题的角度出发去寻找思路，再去找到方法，而不是被论文从底层逐步往上的行文思路牵走了**——从顶部全局俯瞰一篇文章这样的角度能更好帮助我们发现文章中的优点和问题，比如文章解决问题的思路可以不同；创新方式保守还是大胆；使用的方法新颖还是老旧；是否有正确性和有效性（effectiveness）；方法的实验验证是否规范可行全面

2.接下来阅读论文，需要什么学什么，不需要一个全方位的学习，实在太浪费时间了，因为知识太多是学不完的，产出要向纵深挖掘。

3.相关密码学数学知识，先有个印象即可；有关方法，以后能够想起来可以解决某个问题。

（二）论文技术方面：

1. 基础知识学习：联盟链经典的共识机制 PBFT 流程（纸质笔记）

⭐ Level 1：最基础

- 区块链基本结构
- 共识机制概念
- 拜占庭问题
- PBFT 三阶段流程

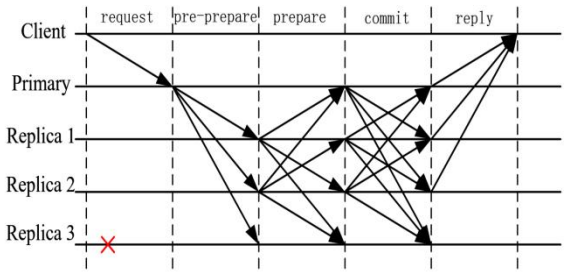


Fig. 1. The classical three-stage process of the Consistency protocol.

P.S.相关概念

（1）IoT:物联网（Internet of thing)

IoT 特点：

节点数量超大（成千上万）

设备性能弱（带宽差、CPU 弱）

网络不稳定（丢包、延迟）

节点易离线

导致传统 PBFT：

$O(n^2)$ 消息量 → 崩溃

View-change 开销大

大规模网络无法扩展

IoT 节点负载过高

（2）BLS:一种基于椭圆曲线（ECC）的聚合签名方式（若干分散碎片合成一个钥匙）

（3）TPS：吞吐量

2. 论文创新知识小结：

2.2.1 重点 or 常用

⭐ Level 2：论文核心（理解论文必须）

- IoT 区块链特点
- 节点信誉模型（α/β/γ）
- 分组（cluster-based）PBFT
- 门限群签名（感性理解即可）

文章有三个主要创新点：

（1）节点信誉模型：
把节点按信誉 Ci 分为三类（表 II）：
α 类：Ci > Cp，可作为主节点；
β 类：Cn ⩽ Ci ⩽ Cp，只能做副本；
γ 类：Ci ⩽ Cn，不能参与共识，仅作候选。
初始全球信誉值 Ci 设为 1/N（所有节点平等起点），之后通过交易/消息交互数据不断更新。

（2）分层的思想：
Consensus stage within the cluster（集群内部阈值一致性阶段）
α Consensus group consensus achievement stage（主节点层级的最终共识阶段）

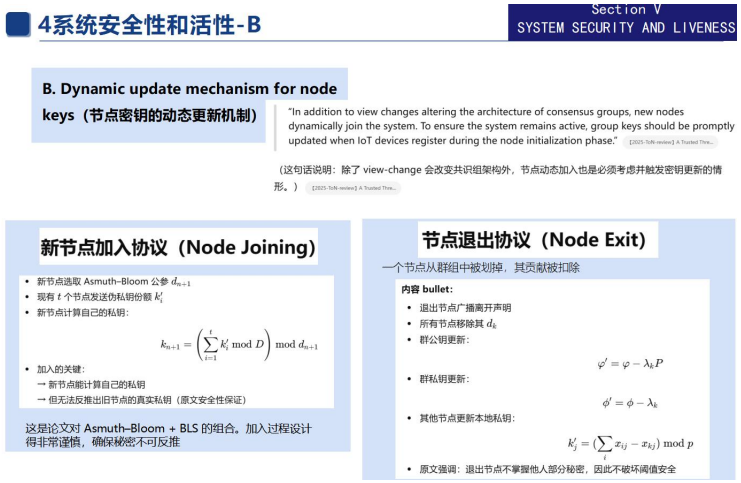
（3）t-of-n 的门限阈值签名方式+动态密钥更新机制：
特别的，T-threshold consensus 是 cluster 内部的门限确认机制，它允许 cluster 在只收集到 t 个 replica 的一致签名碎片后，就生成可验证的 BLS 群签名，代表整个 cluster 已达成一致。

2.2.2 复杂 or 简略

⭐ **Level 3：深度理解论文（可选）**

- BLS 双线性映射原理
- Lagrange 插值的 t-of-n 门限结构
- 动态密钥更新协议

动态密钥更新机制：



3. 实验思路小结：

实验 A：Security Testing and Analysis（安全性测试与分析）

1) Node trust evaluation:不同节点类型的信任值变化

2) View-Change probability test:视图切换概率对比

实验 B：Communication Time Complexity（通信时间复杂度）

实验 C. Scalability Testing 可扩展性：

1) Consensus latency 时间延迟

2) Throughput testing 吞吐量

（三）论文行文方面

1. 总体行文思路应该是：

1.1 论文八股式：

Abstract

1. INTRODUCTION（引言）

2. BACKGROUND AND IoT SECURITY PROBLEM（背景与 IoT 安全问题）

3. RELATED WORK（相关研究：可单独成块）

4. TCBFT CONSENSUS ALGORITHM（TCBFT 共识算法：建模）

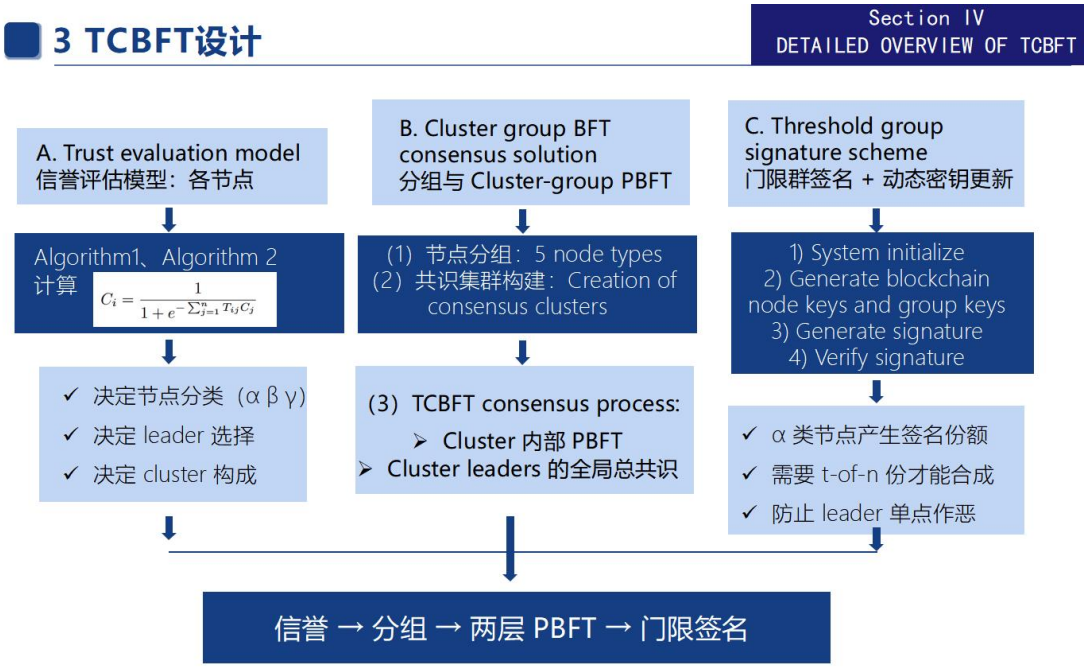
5. SYSTEM SECURITY AND LIVENESS（系统安全性与活性：数学分析）

6. PERFORMANCE EVALUATION（性能评估：实验验证）

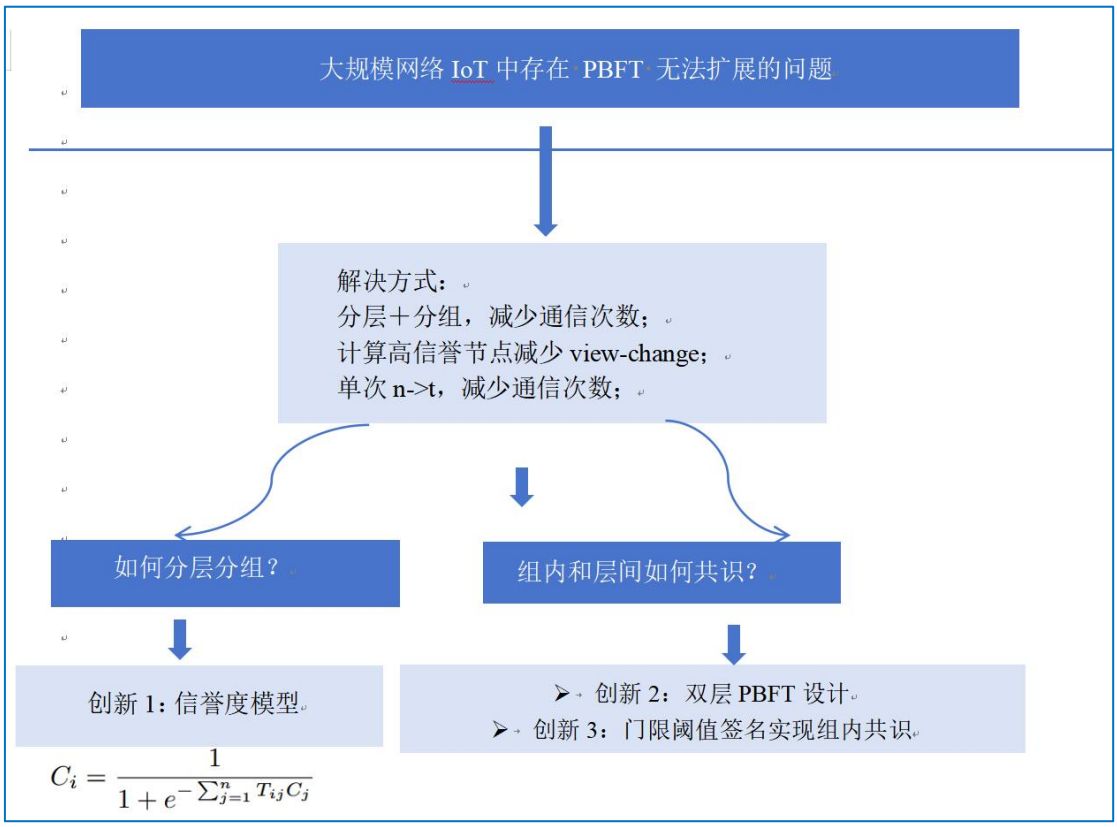
7. CONCLUSIONS AND FUTURE WORK（结论与未来工作）

Reference

1.2 核心部分 Sect4 行文：



1.3 从解决问题的视角看：



大规模网络 IoT 中存在 PBFT 无法扩展的问题
解决方式：
分层＋分组，减少通信次数；
计算高信誉节点减少 view-change；
单次 n->t，减少通信次数；
如何分层分组？
组内和层间如何共识？
创新 1：信誉度模型
$C_i = \frac{1}{1+e^{-\sum_{j=1}^{n} T_{ij}C_j}}$
➢ 创新 2：双层 PBFT 设计
➢ 创新 3：门限阈值签名实现组内共识

2. Abstract

写得不错，层次分明，可学习。

3. Introduction

不要太长，；

related work 单独放出来，浓缩版可以放在 intro 里面。

4. 图表这一块

让人能看懂，对比明显，解释清晰：

3.2.1 流程图的循环部分清晰易懂，写清楚层次和循环出口

并不好的一个示范：

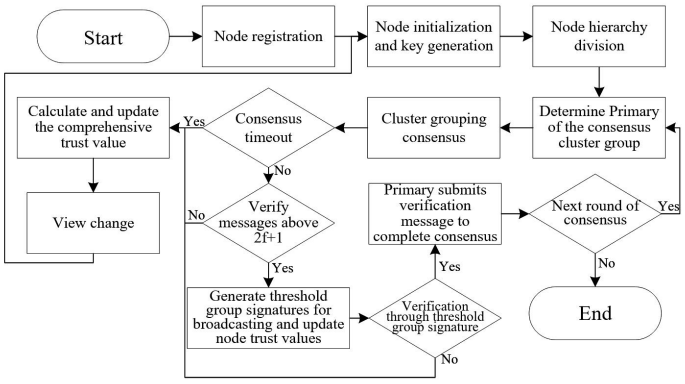


Fig. 4. The flow chart for the process of the TCBFT.

3.2.2 三维热力图注意坐标方向和颜色设计

C. Scalability Testing

1) **Consensus latency**

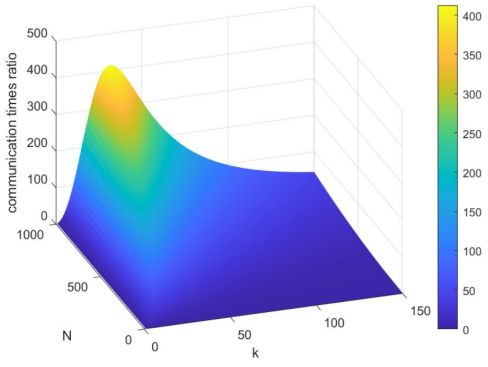


Fig. 8. Surface graph of communication frequency ratio in consensus process.

3.2.3 折线图可以学习配色方式，红蓝突出 this work

TABLE V
COMPARISON OF AVERAGE TIME BETWEEN
DOBLE-LAYER PBFT-ECDSA AND THIS WORK.

Number of nodes participating in consensus	Doble-Layer PBFT(ecdsa)		This work		Improvement
	sign	Verify	sign	Verify	
100	18.87ms	18.29ms	35.61ms	18.70ms	46.15%slower
200	40.97ms	94.78ms	75.61ms	34.62ms	18.79%faster
300	56.66ms	142.46ms	225.63ms	36.71ms	10.93%faster
400	120.60ms	238.35ms	268.80ms	60.34ms	8.30%faster
500	103.97ms	358.02ms	352.30ms	34.71ms	16.22%faster

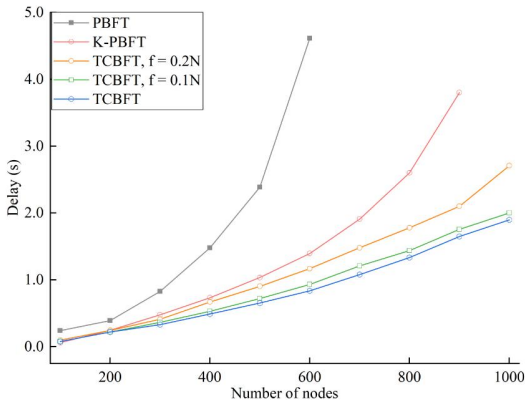
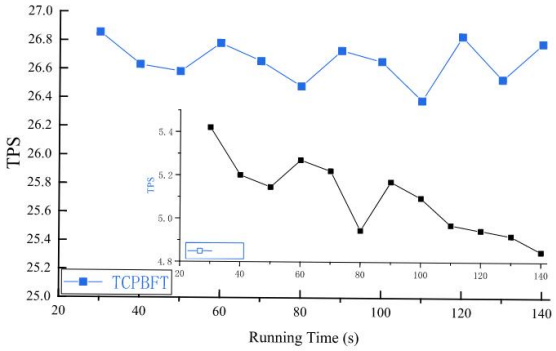
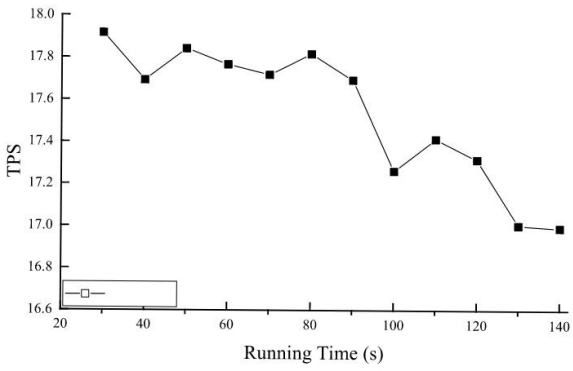


Fig. 10. The relationship between consensus latency and the number of nodes in the system.

3.2.4 大小图嵌套的方式对比波动情况



(a) Average throughput of TCBFT and PBFT



(b) Average throughput of K-PBFT

Fig. 11. The average throughput of node consensus.