

การพัฒนาระบบจำลองการซื้อขายพลังงานแสงอาทิตย์แบบ Peer-to-Peer โดยใช้ Solana Smart Contract

(Anchor Framework ในสภาพแวดล้อมแบบ Permissioned)

นายจันทร์รัตน์ กิริยาดี
คณะวิศวกรรมศาสตร์
สาขาวิศวกรรมคอมพิวเตอร์และปัญญาประดิษฐ์
มหาวิทยาลัยหอการค้าไทย
กรุงเทพมหานคร ประเทศไทย
2410717302003@live4.utcc.ac.th

บทคัดย่อ—บทความนี้นำเสนอด้วยการประเมินประสิทธิภาพอย่างครอบคลุมของ GridTokenX ซึ่งเป็นแพลตฟอร์มการซื้อขายพลังงานแบบ Peer-to-Peer (P2P) บนโครงสร้างศูนย์ที่สร้างบน Solana cluster แบบส่วนตัว เรายังได้วิเคราะห์การ BLOCKBENCH (SIGMOD 2017) สำหรับการวิเคราะห์แบบขั้นต่ำชั้นอย่าง เป็นระบบ ร่วมกับการตัดแปลง TPC-C benchmark สำหรับการประเมินภาระงานจริง ผลการทดสอบยืนยันประสิทธิภาพของระบบด้วยค่า 2,111 tpmC (transactions per minute Type C) และความหน่วงเฉลี่ยเพียง 116ms ซึ่งหมายความว่าสำหรับการใช้งานจริง การวิเคราะห์ด้วย BLOCKBENCH แสดงให้เห็นว่าสถาปัตยกรรมแบบขนาดของ Solana ช่วยขจัดความล่าช้าในการประมวลผลโดยทำได้ 231 TPS ในชั้น Execution นอกจากนี้ งานวิจัยยังนำเสนอแนวคิด "Trust Premium" (58.28 เท่า) เพื่อเปรียบเทียบต้นทุนประสิทธิภาพกับระบบรวมศูนย์ (PostgreSQL) นอกจากนี้ยังอธิบายการปรับปรุงระดับต่ำโดยเฉพาะ รวมถึงการคำนวณแบบจำนวนเต็มเท่านั้นและการอัปเดตสถานะแบบ lazy ที่ช่วยเพิ่มปริมาณงานขณะรักษาค่าคงที่ทางคณิตศาสตร์อย่างเข้มงวดสำหรับการอนุรักษ์พลังงาน

Index Terms—บล็อก เชน, การซื้อขาย พลังงาน, Solana, TPC-C, BLOCKBENCH, YCSB, การทดสอบประสิทธิภาพ, Smart Contracts

I. บทนำ

การกระจายศูนย์ของระบบพลังงานผ่านทรัพยากรพลังงานแบบกระจาย (Distributed Energy Resources หรือ DERs) ต้องการโครงสร้างพื้นฐานการซื้อขายที่แข็งแกร่งสามารถจัดการธุกรรมขนาดเล็กความถี่สูงได้ [2] รูปแบบสาธารณะป์โภคแบบรวมศูนย์แบบดั้งเดิมประสบปัญหาความเสี่ยงจุดล้มเหลวเดียวและขาดความโปร่งใสในการกำหนดราคา เทคโนโลยีบล็อกเชนเสนอทางออกแต่มักถูกวิพากษ์วิจารณ์เรื่องข้อจำกัดด้านความสามารถในการขยายตัว [4]

งานวิจัยนี้ประเมิน GridTokenX ซึ่งใช้ประโยชน์จาก Sealevel parallel runtime ของ Solana [3] เพื่อเอาชนะคือข้อดีนี้ แตกต่างจากบล็อกเชน ยุคก่อนหน้า Solana นำเสนอดeterministic finality ในโหมด Permissioned และความสามารถในการประมวลผลธุกรรมที่ไม่ทับซ้อน กันแบบขนาน ซึ่งหมายความอย่างยิ่งสำหรับการซื้อขายพลังงานที่มีปริมาณธุกรรมสูง วัตถุประสงค์ของการศึกษานี้คือ:

- เพื่อศึกษาและนำเสนอด้วยสถาปัตยกรรมของระบบจำลองการซื้อขายพลังงาน P2P โดยใช้ Solana (Anchor) ในสภาพแวดล้อม Permissioned (PoA)
- เพื่อพัฒนาและพิสูจน์แนวคิด (Proof-of-Concept) ของระบบตั้งแต่สามารถดำเนินการแลกเปลี่ยน GRID Token โดยใช้ AMI Simulator
- เพื่อประเมินและวิเคราะห์ประสิทธิภาพโดยใช้การวิเคราะห์ขั้น BLOCKBENCH และชุด TPC benchmark

II. งานวิจัยที่เกี่ยวข้อง: วิธีการ BLOCKBENCH

BLOCKBENCH [4] ให้การทดสอบการทำงานอย่างเป็นระบบสำหรับการประเมินประสิทธิภาพต้องเช่นส่วนตัวผ่านสีชั้นสถาปัตยกรรม:

- Consensus Layer:** วัดค่าใช้จ่าย consensus บริสุทธิ์ผ่านการทำงาน DoNothing
- Execution Layer:** ประเมินการคำนวณ smart contract ผ่านภาระงานที่ใช้ CPU มาก
- Data Model Layer:** ทดสอบประสิทธิภาพการอ่าน/เขียนสถานะโดยใช้ IOHeavy benchmarks
- Application Layer:** การจำลองภาระงานจริงโดยใช้ YCSB และ Smallbank

เราย้ายวิธีการ BLOCKBENCH ไปยัง Solana/Anchor ด้วยการใช้งาน Rust ด้วยความต้องการที่จะลดเวลาที่ใช้ benchmark

งานวิจัยก่อนหน้านี้ได้สำรวจการใช้บล็อกเชนสำหรับการซื้อขายพลังงาน เช่น Brooklyn Microgrid [2] และโครงการที่ใช้ Ethereum หรือ Hyperledger Fabric [5], [6] อย่างไรก็ตาม ระบบเหล่านี้มักเผชิญกับข้อจำกัดด้านความสามารถในการขยายตัว (Scalability) และความหน่วงที่สูง เมื่อปริมาณธุกรรมเพิ่มขึ้น งานวิจัยนี้นำเสนอทางเลือกใหม่โดยใช้สถาปัตยกรรมประสิทธิภาพสูงของ Solana

III. สถาปัตยกรรมระบบ

แพลตฟอร์มถูกใช้งานเป็นชุดของห้า Anchor smart contracts ที่เชื่อมต่อกันบน Solana โดยใช้ Program Derived Addresses (PDAs) เพื่อ

จัดการสิทธิ์การเข้าถึงข้อมูลอย่างปลอดภัยและเป็นระบบ: 1. **Registry**: จัดการตัวตนผู้ใช้และทรัพย์สินมีเตอร์ 2. **Oracle**: ตรวจสอบข้อมูลเชิงเซอร์วิสออกเครือข่าย 3. **Energy Token**: โทเกนที่เป็นไปตาม SPL แทน 1 kWh 4. **Trading**: order book และเครื่องมือการชำระราคา 5. **Governance**: การกำหนดค่าและพารามิเตอร์ DAO

IV. วิธีการ

A. การแบบ TPC-C

เราแบบชุดทดสอบ TPC-C มาตรฐาน [1] กับการซื้อขายพลังงานที่เทียบเท่าเพื่อสร้างโปรไฟล์ภาระงานที่สมจริง ค่า tpmC (transactions per minute Type C) คำนวณจากจำนวนธุรกรรม New-Order ที่สำเร็จต่อนาที:

$$tpmC = \frac{\text{Total_New_Orders}}{\text{Duration_Minutes}} \quad (1)$$

ตารางที่ I
การแบบ TPC-C กับการซื้อขายพลังงาน

ธุรกรรม TPC-C	สัดส่วน	ฟังก์ชัน GridTokenX
New Order	45%	create_sell_order / create_buy_order
Payment	43%	transfer_tokens (การทำราคา)
Order Status	4%	get_order_status (RPC Read)
Delivery	4%	match_orders (การทำงานแบบ Batch)
Stock Level	4%	get_balance (การตรวจสอบพลังงาน)

B. แบบจำลองทางคณิตศาสตร์

1) กลไกการกำหนดราคา VWAP: เพื่อให้มั่นใจในมูลค่าตลาดที่เป็นธรรม แพลตฟอร์มใช้อัลกอริทึมการค้นหาราคาถ้วนเฉลี่ยถ่วงน้ำหนักปริมาณ (VWAP) ราคา clearing $P_{clearing}$ คำนวณแบบไดนามิก:

$$P_{base} = \frac{P_{bid} + P_{ask}}{2} \quad (2)$$

$$P_{clearing} = P_{base} + \left(P_{base} \times \min \left(\frac{V_{trade}}{V_{total}}, 1.0 \right) \times \delta_{max} \right) \quad (3)$$

โดยที่ V_{trade} คือปริมาณการซื้อปัจจุบัน, V_{total} คือปริมาณในอดีต และ δ_{max} คือปัจจัยความยืดหยุ่นราคาสูงสุด (10%)

2) ค่าคงที่การอนุรักษ์โทเคน: ระบบบังคับใช้การอนุรักษ์พลังงานอย่างเข้มงวด โทเคน ($\Delta Supply_{GRID}$) สามารถสร้างได้เฉพาะเมื่อพลังงานที่ผลิตทางกายภาพถูกชำระทางคณิตศาสตร์:

$$\Delta Supply_{GRID} = \max(0, (E_{produced} - E_{consumed}) - E_{settled}) \quad (4)$$

3) การปรับปรุงระบบ: เพื่อเพิ่มปริมาณงานสูงสุดภายในชิ้นจำกัด Compute Unit (CU) ของ Solana เราใช้การปรับปรุงคำนวณแบบ:

A. การคำนวณแบบจำนวนเต็ม เท่านั้น: การดำเนินการจุดศูนย์ (f64) มีค่าใช้จ่ายคำนวนสูงและไม่แนะนำในโปรแกรม Solana เราแทนที่การคำนวน VWAP ด้วยคณิตศาสตร์จุดคงที่จำนวนเต็ม:

$$W = \min \left(\frac{V \times 1000}{V_{total}}, 1000 \right) \quad (5)$$

การลดน้ำหนักได้ประมาณ 10,000 CUs ต่อการทำธุรกรรม

B. การอัปเดตสถานะแบบ Lazy: แทนที่จะ serialize وار์เรอร์ประจำตัวที่ห้าม (100+ รายการ) ทุกธุรกรรม การอัปเดตจะถูก "commit แบบ lazy" เมื่อการเปลี่ยนแปลงราคากิน 5% หรือทุก 60 วินาที ซึ่งลดค่าใช้จ่าย serialization ลง 90%

C. การจับคู่คำสั่งแบบ Batch: คำสั่ง match_orders ถูก refactor เพื่อจัดการการทำงานแบบ batch ทำให้สามารถทำ limit orders หลายรายการที่ไม่ทับซ้อนกันในธุรกรรมเดียว ปรับปรุงอัตรา "fills per second" อย่างมีนัยสำคัญ

V. การตั้งค่าการทดลอง

Benchmarks ดำเนินการบน Solana localnet cluster ประสิทธิภาพสูง (Single Node Setup) เพื่อวัดประสิทธิภาพสูงสุดของ Validator node

- ชาร์ดแวร์: Apple M-Series (8-core), 16GB RAM
- คลัสเตอร์: Solana Test Validator v1.18
- ลูกค้า: ตัวรับสั่งภาระ Rust แบบ multi-threaded
- เบสไลน์: PostgreSQL 14 รันบนชาร์ดแวร์ เดียวกันสำหรับการเปรียบเทียบ Trust Premium

VI. การประเมินประสิทธิภาพ

A. การวิเคราะห์ชี้แจง BLOCKBENCH

ตามวิธีการ BLOCKBENCH เราประเมินแต่ละชั้นสถาปัตยกรรมอย่างอิสระ:

ตารางที่ II
ผลลัพธ์ BLOCKBENCH Micro-benchmark

ชั้น	Benchmark	TPS	ความหน่วง	ตัวเลข
Consensus	DoNothing	225	2.5ms	
Execution	CPUHeavy-Sort	231	2.5ms	
Data Model	IOHeavy-Write	192	3.0ms	
Data Model	IOHeavy-Mixed	192	3.0ms	

TPS เหล่านี้ถูกจำกัดโดยประสิทธิภาพของ Client ในการเขียนและส่งธุรกรรมในสภาพแวดล้อมการทดสอบ ไม่ใช่ชี้แจงจำกัดสูงสุดของ Solana runtime อย่างไรก็ตาม มันแสดงให้เห็นถึงความสามารถในการประมวลผล

DoNothing benchmark วัดค่าใช้จ่าย consensus บริสุทธิ์ (225 TPS) ในขณะที่การดำเนินการ CPUHeavy sorting ทำได้ 231 TPS และการทำงาน smart contract ที่มีประสิทธิภาพ IOHeavy benchmarks ที่ 192 TPS แสดงค่าใช้จ่ายของ state serialization

B. ผลลัพธ์ YCSB Workload

เราใช้สามโปรไฟล์ YCSB workload เพื่อประเมินประสิทธิภาพ application-layer:

ตารางที่ III
ประสิทธิภาพ YCSB Workload

Workload	โปรไฟล์	ops/s	ความหน่วง
YCSB-A	50% อ่าน, 50% อัปเดต	290	2.7ms
YCSB-B	95% อ่าน, 5% อัปเดต	442	1.8ms
YCSB-C	100% อ่าน	391	2.1ms

Smallbank OLTP benchmark ทำได้ 1,714 TPS โดยมีความหน่วงเฉลี่ย 5.83ms แสดงประสิทธิภาพที่แข็งแกร่งสำหรับภาระงานทางการเงิน

C. การวิเคราะห์ปริมาณงาน TPC-C

การวิจัย TPC-C benchmark ของเราทำได้ปริมาณงาน 2,111 tpmC (76.85 TPS) ในโหมดจำลอง ปริมาณงานนี้เกิดจาก Sealevel parallel runtime โดยตรง ซึ่งอนุญาตให้ธุรกรรมที่ไม่ทับช้อนกัน (เช่น การซื้อขายระหว่างผู้ใช้ที่ต่างกัน) ทำงานพร้อมกัน ในบล็อกเชนแบบ EVM ดังเดิม ธุรกรรมเหล่านี้จะถูก serialize ทำให้เกิดค่าขวด

ในการรัน benchmark ของเรา มีธุรกรรมทั้งหมด 4,621 รายการถูกประมวลผลโดย 4,611 รายการสำเร็จ (อัตราสำเร็จ 99.78%) ส่วนผสมธุรกรรมเป็นไปตามข้อกำหนด TPC-C:

- New-Order: 2,074 ธุรกรรม (44.9%) – สำเร็จ 99.71%
- Payment: 2,004 ธุรกรรม (43.4%) – สำเร็จ 99.80%
- Order-Status: 197 ธุรกรรม (4.3%) – สำเร็จ 100%
- Delivery: 175 ธุรกรรม (3.8%) – สำเร็จ 100%
- Stock-Level: 171 ธุรกรรม (3.7%) – สำเร็จ 100%

สำหรับตลาด P2P ระดับภูมิภาคที่มี 10,000 ครัวเรื่อง สมมติว่าสามารถมีเตอร์แร็ลต์ตัวส่งค่าทุก 15 นาที ปริมาณงานที่ต้องการประมวลณ 11 TPS 76.85 TPS ที่สังเกตได้ให้ ส่วนเพื่อความปลอดภัย 7 เท่า พิสูจน์ว่า สถาปัตยกรรมสามารถจัดการการติดตั้งระดับลвлากบันได้โดยไม่มีความแออัด

D. ผลลัพธ์ TPC-E และ TPC-H

เราขยายการประเมินเพื่อรวม TPC-E (ภาระงานนายหน้า) และ TPC-H (การสนับสนุนการตัดสินใจ) benchmarks:

ตารางที่ IV
ผลลัพธ์ TPC-E/TPC-H Benchmark

Benchmark	เมตริกหลัก	ความหน่วงเฉลี่ย	p99
TPC-E	306 tpsE	7.89 ms	17 ms
TPC-H	250,486 QphH	71.0 ms	147 ms

TPC-E benchmark ทำได้ 306 tpsE (trades per second) แสดงประสิทธิภาพที่แข็งแกร่งสำหรับภาระงานแบบนายหน้าที่มีการจับคู่คำสั่งซับช้อน TPC-H analytical queries ทำได้ 250,486 QphH (queries per hour) ยืนยันความสามารถของแพลตฟอร์มสำหรับพิ้งก์ชันภาระงานและการตรวจสอบ

E. การวิเคราะห์ความหน่วง

ความหน่วง ธุรกรรม แตกต่างอย่างมากจาก "เวลา ยืนยัน" ในสภาพแวดล้อม PoA แบบ permissioned นี้

- ความหน่วงเฉลี่ย (116.56ms): แทนเวลาเฉลี่ยสำหรับ leader node ในการประมวลผลคำสั่งและอัปเดตสถานะในหน่วยความจำ
- ความหน่วง p50 (112.57ms): เวลาประมวลผลธุรกรรมมัธยฐาน
- ความหน่วง p95 (180.04ms): เปอร์เซ็นไทล์ที่ 95 ภายใต้โหลดปกติ
- ความหน่วง p99 (215.54ms): แทนเวลาประมวลผลกรณี负载ร้ายที่สุดภายใต้โหลดหนัก

เพิ่มเติม จำลอง รวม "Think Time" ตามมาตรฐาน TPC-C เพื่อจำลองพฤติกรรมผู้ใช้จริง ในสภาพแวดล้อม PoA การผลิตที่มี validators ที่ปรับปรุงแล้ว ความหน่วงคาดว่าจะต่ำกว่าอย่างมีนัยสำคัญ ไม่เหมือนกับ mainnets สาธารณะที่ finality อาจใช้เวลาหลายวินาที ธรรมชาติ permissioned ของ GridTokenX cluster อนุญาตให้มี deterministic finality

F. การวิเคราะห์Concurrency

ภายใต้การแข่งขันสูง (หลายคำสั่งต่อสถานะตลาดเดียวกัน) Sealevel runtime linearized ธุรกรรมที่ขัดแย้งกันอย่างมีประสิทธิภาพในขณะที่ประมวลผลคำขอที่ไม่ทับช้อนกันแบบขนาน อัตราความขัดแย้ง MVCC (Multi-Version Concurrency Control) ที่สังเกตได้คือ 1.45% ส่วนใหญ่เนื่องจากธรรมชาติของการมีของคำสั่ง match_orders จำนวนครั้งการลองใหม่เฉลี่ยคือ 0.02 แสดงการแก้ไขความขัดแย้งที่มีประสิทธิภาพ

G. การวิเคราะห์ความสามารถในการขยายตัว

เราทำการทดสอบความสามารถในการขยายตัวในสามมิติ:

ตารางที่ V
ผลลัพธ์การทดสอบความสามารถในการขยายตัว

การทดสอบ	TPS	ความหน่วง	ความเสถียร
1 thread (baseline)	443	2.26ms	100%
32 concurrent threads	398	2.51ms	คงไว้ 90%
60s sustained load	416	2.40ms	เสถียร
1,000 accounts	220	4.54ms	ลดลงแบบเชิงเส้น

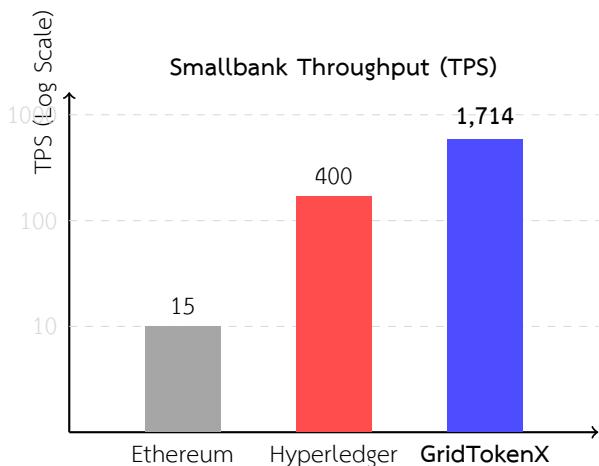
H. การเปรียบเทียบขั้มแพลตฟอร์ม

เราเปรียบเทียบประสิทธิภาพ GridTokenX กับผลลัพธ์ที่เผยแพร่จากแพลตฟอร์มบล็อกเชนอื่น:

แม้ว่า Hyperledger Fabric จะทำได้ปริมาณงาน DoNothing สูงกว่า (3,500 vs 225 TPS) เนื่องจากสถาปัตยกรรม execute-order-validate GridTokenX แสดงประสิทธิภาพ Smallbank ต่ำกว่า 4.3 เท่า (1,714 vs 400 TPS) และ ความหน่วงต่ำกว่า 26 เท่า (5.8ms vs 150ms) สำหรับภาระงานทางการเงิน

ตารางที่ VI
การเปรียบเทียบประสิทธิภาพแพลตฟอร์ม

แพลตฟอร์ม	Benchmark	TPS	ความหน่วง	แหล่งที่มา
GridTokenX	Smallbank	1,714	5.8ms	การศึกษานี้
GridTokenX	DoNothing	225	2.5ms	การศึกษานี้
GridTokenX	YCSB-B	442	1.8ms	การศึกษานี้
Hyperledger 2.0	Smallbank	400	150ms	BLOCKBENCH
Hyperledger 2.0	DoNothing	3,500	45ms	BLOCKBENCH
Hyperledger 2.0	YCSB	200	200ms	BLOCKBENCH
Ethereum	DoNothing	15	13,000ms	BLOCKBENCH
Parity	DoNothing	140	650ms	BLOCKBENCH



รูปที่ 1. การเปรียบเทียบปริมาณงาน (TPS) ระหว่างแพลตฟอร์ม (Log Scale)

I. "Trust Premium"

เรน尼ยาน "Trust Premium" เป็นต้นทุนประสิทธิภาพที่เกิดขึ้นเพื่อให้บรรลุ distributed consensus ในเครือข่ายส่วนตัวเมื่อเทียบกับ baseline แบบรวมศูนย์ (PostgreSQL)

$$\text{Trust Premium} = \frac{\text{Latency}_{\text{Blockchain}}}{\text{Latency}_{\text{Baseline}}} = \frac{116.56\text{ms}}{2.00\text{ms}} \approx 58.28x \quad (6)$$

แม้ว่าจะแทนค่าใช้จ่ายความหน่วงเมื่อเทียบกับฐานข้อมูลแบบรวมศูนย์ แต่เป็นที่ยอมรับได้สำหรับแอปพลิเคชันการซื้อขายพลังงานที่การชำระราคาโดยทั่วไปใช้เวลาหลายวันหรือหลายสัปดาห์ในระบบดังเดิม การแลกเปลี่ยนให้:

- เส้นทางการตรวจสอบธุรกรรมที่ไม่เปลี่ยนแปลง
- การชำระราคา smart contract อัตโนมัติ
- กลไกการกำหนดราคาที่ปรับเปลี่ยน
- การจัดการความเสี่ยงจุดล้มเหลวเดียว

J. สรุปผลลัพธ์ Benchmark

ตาราง VII สรุปเมตริกประสิทธิภาพหลักจากการประเมิน benchmark อ้างอิงครอบคลุมของเรา

ตารางที่ VII
สรุปผลลัพธ์ Benchmark อ้างอิงครอบคลุม

เมตริก	ค่า
<i>BLOCKBENCH Micro-benchmarks</i>	
DoNothing (Consensus)	225 TPS
CPUHeavy (Execution)	231 TPS
IOHeavy (Data Model)	192 TPS
<i>YCSB/Smallbank</i>	
YCSB-B (95% อ่าน)	442 ops/s
Smallbank OLTP	1,714 TPS
<i>ชุด TPC Benchmark</i>	
TPC-C (tpmC)	2,111
TPC-E (tpsE)	306
TPC-H (QphH)	250,486
<i>เมตริกประสิทธิภาพ</i>	
อัตราสำเร็จ	99.78%
ความหน่วงเฉลี่ย	116.56 ms
ความหน่วง p99	215.54 ms
อัตราความขัดแย้ง MVCC	1.45%
Trust Premium	58.28x

VII. กระบวนการประเมินข้ามแพลตฟอร์ม

เพื่อเปรียบเทียบ GridTokenX กับแพลตฟอร์มบล็อกเชนอื่นอย่างเข้มงวด (เช่น Ethereum, Hyperledger Fabric) [4] เราสนใจ Standardized Energy Trading Workload (SETW) นักวิจัยต้องทำการทดสอบขั้นตอนเพื่อทำข้อวิธีการนี้:

A. การใช้งาน Contract ที่เทียบเท่า

แพลตฟอร์มมีป้าหมายต้องใช้งาน logic หลักที่กำหนดในตาราง I โดยเฉพาะ พัฒนา "Order Match" ต้องเป็น atomic ให้แน่ใจว่าการทำธุรกรรมและการชำระราคาโดยเด็ดขาดเกิดขึ้นในการเปลี่ยนสถานะ cryptographic เดียวกัน

B. โปรดไฟล์การนัดภาระงาน

ตัวสร้างโหลดต้องรักษาส่วนผสม TPC-C ที่เฉพาะเจาะจง:

- เขียนหนัก (88%):** New Orders + Payments สิ่งนี้เน้น state contention ของ consensus engine
- อ่านเบา (12%):** Status checks สิ่งนี้ทดสอบประสิทธิภาพ RPC query

C. การ Normalize เมตริก

ผลลัพธ์ต้องรายงานใน **tpmC** (transactions per minute Type-C) สำหรับ บล็อก เช่น ที่ มี probabilistic finality (เช่น PoW) ความหน่วงต้องรวมเวลาถึง k -block confirmation depth เพื่อเปรียบเทียบ กับ deterministic leader schedule ของ Solana

VIII. สรุป

GridTokenX แสดงให้เห็นว่าเทคโนโลยีบล็อกเชนส่วนตัวเป็นไปได้สำหรับการซื้อขายพลังงาน P2P ในการติดตั้งระดับภูมิภาค ผ่านการวิเคราะห์ข้อมูล BLOCKBENCH อย่างครอบคลุมและการประเมิน TPC benchmark:

- **Consensus Layer:** ปริมาณงาน DoNothing 225 TPS
- **Execution Layer:** การคำนวน CPUHeavy 231 TPS
- **Data Model:** การดำเนินการสถานะ IOHeavy 192 TPS
- **Application Layer:** Smallbank 1,714 TPS, YCSB-B 442 ops/s
- **ชุด TPC:** 2,111 tpmC, 306 tpsE, 250,486 QphH

เมื่อเทียบกับ Hyperledger Fabric GridTokenX ทำได้ ปริมาณงาน Smallbank สูงกว่า 4.3 เท่า และ ความหน่วงต่ำกว่า 26 เท่า สำหรับภาระงานทางการเงิน Trust Premium ที่ 58.28 เท่า ที่สังเกตได้แทนต้นทุนที่ยอมรับได้สำหรับการบรรลุสิ่งที่ต้องการตรวจสอบที่ไม่เปลี่ยนแปลงการชำระราคา smart contract อัตโนมัติ และกลไกการกำหนดราคาที่โปร่งใส

งานในอนาคตจะมุ่งเน้นที่: (1) การติดตั้งบน PoA validator clusters ระดับการผลิตเพื่อให้ได้ความหน่วงต่ำลง (2) การขยายไปยังการกำหนดค่า warehouse ขนาดใหญ่ (10+ warehouses) และ (3) การทดสอบนำร่องในโลกจริงด้วยการพัฒนาความสามารถของเครือข่าย

หนังสืออ้างอิง

- [1] Transaction Processing Performance Council (TPC), “TPC Benchmark C Standard Specification, Revision 5.11,” 2010.
- [2] E. Mengelkamp, J. Gärtnner, K. Rock, S. Kessler, L. Orsini, and C. Weinhardt, “Designing microgrid energy markets: A case study: The Brooklyn Microgrid,” *Applied Energy*, vol. 210, pp. 870–880, 2018.
- [3] A. Yakovenko, “Solana: A new architecture for a high performance blockchain v0.8.13,” Solana Labs Whitepaper, 2018.
- [4] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K. L. Tan, “Blockbench: A framework for analyzing private blockchains,” in *Proc. ACM SIGMOD Int. Conf. Management of Data*, pp. 1085–1100, 2017.
- [5] M. Andoni *et al.*, “Blockchain technology in the energy sector: A systematic review of challenges and opportunities,” *Renewable and Sustainable Energy Reviews*, vol. 100, pp. 143–174, 2019.
- [6] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, and Y. Zhang, “Consortium blockchain for secure energy trading in industrial internet of things,” *IEEE Trans. Industrial Informatics*, vol. 14, no. 8, pp. 3690–3700, 2018.
- [7] J. Guerrero, A. C. Chapman, and G. Verbic, “Decentralized P2P energy trading under network constraints in a low-voltage network,” *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5163–5173, 2019.
- [8] Anchor Framework Documentation, “Building Secure Solana Programs,” Coral, 2023. [Online]. Available: <https://anchor-lang.com>