

การพัฒนาระบบจำลองการซื้อขายพลังงานแสงอาทิตย์แบบ Peer-to-Peer โดยใช้ Solana Smart Contract (Anchor Framework ในสภาพแวดล้อมแบบ Permissioned)

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

บทคัดย่อ—บทความนี้นำเสนอการประเมินประสิทธิภาพอย่างครอบคลุมของ GridTokenX ซึ่งเป็นแพลตฟอร์มการซื้อขายพลังงานแบบ Peer-to-Peer (P2P) แบบกระจายศูนย์ที่พัฒนาบน Solana Cluster แบบส่วนตัว (Private Cluster) งานวิจัยนี้ใช้วิธีการ BLOCKBENCH (SIGMOD 2017) สำหรับการวิเคราะห์ระบบที่ละชั้นอย่างเป็นระบบ ร่วมกับการประยุกต์ใช้ TPC-C Benchmark เพื่อประเมินการรองรับภาระงานในสถานการณ์จริง ผลการทดลองยืนยันประสิทธิภาพของระบบด้วยค่า 2,111 tpmC (Transactions Per Minute Type C) และมีความหน่วงเฉลี่ยเพียง 116 ms ซึ่งเหมาะสมสำหรับการใช้งานจริง การวิเคราะห์ด้วย BLOCKBENCH แสดงให้เห็นว่าสถาปัตยกรรมแบบขนานของ Solana ช่วยจัดคอขวดในการประมวลผล โดยสามารถรองรับได้ถึง 231 TPS ในขั้น Execution นอกจากนี้ งานวิจัยยังนำเสนอแนวคิด "Trust Premium" (58.28 เท่า) เพื่อเปรียบเทียบต้นทุนด้านประสิทธิภาพกับระบบรวมศูนย์ (PostgreSQL) พร้อมทั้งอธิบายการปรับปรุงประสิทธิภาพระดับล่าง (Low-level Optimizations) ได้แก่ การคำนวณด้วยเลขจำนวนเต็มเท่านั้น (Integer-only Arithmetic) และการอัปเดตสถานะแบบ Lazy Update ซึ่งช่วยเพิ่มปริมาณงานโดยยังคงรักษาความถูกต้องทางคณิตศาสตร์อย่างเคร่งครัดเพื่อการอนุรักษ์พลังงาน

Index Terms—บล็อกเชน, การซื้อขายพลังงาน, Solana, TPC-C, BLOCKBENCH, YCSB, การทดสอบประสิทธิภาพ, Smart Contracts

I. บทนำ

การกระจายศูนย์ระบบพลังงานผ่านทรัพยากรพลังงานแบบกระจาย (Distributed Energy Resources หรือ DERs) จำเป็นต้องมีโครงสร้างพื้นฐานการซื้อขายที่แข็งแกร่ง ซึ่งสามารถรองรับธุรกรรมขนาดเล็กที่มีความถี่สูงได้ [2] รูปแบบสาธารณูปโภคแบบรวมศูนย์ดั้งเดิมมักประสบปัญหาความเสี่ยงจากการล้มเหลวที่จุดเดียว (Single Point of Failure) และขาดความโปร่งใสในการกำหนดราคา เทคโนโลยีบล็อกเชนนำเสนอทางออกสำหรับปัญหานี้ แต่มักถูกวิพากษ์วิจารณ์เรื่องข้อจำกัดด้านความสามารถในการขยายระบบ (Scalability) [4]

งานวิจัยนี้ประเมินประสิทธิภาพของ GridTokenX ซึ่งใช้ประโยชน์จาก Sealevel Parallel Runtime ของ Solana [3] เพื่อแก้ไขปัญหาคอขวดเหล่านี้ ซึ่งแตกต่างจากบล็อกเชนรุ่นก่อนหน้า โดย Solana นำเสนอความแน่นอนของสถานะสุดท้าย (Deterministic Finality) ในโหมด Permissioned และความสามารถในการประมวลผลธุรกรรมที่ไม่ทับซ้อน

กันในรูปแบบขนาน ซึ่งเหมาะสมอย่างยิ่งสำหรับการซื้อขายพลังงานที่มีปริมาณธุรกรรมสูง วัตถุประสงค์ของการศึกษานี้คือ:

- 1) เพื่อศึกษาและนำเสนอสถาปัตยกรรมของระบบจำลองการซื้อขายพลังงาน P2P โดยใช้ Solana (Anchor) ในสภาพแวดล้อมแบบ Permissioned (PoA)
- 2) เพื่อพัฒนาและพิสูจน์แนวคิด (Proof-of-Concept) ของระบบต้นแบบที่สามารถจำลองการแลกเปลี่ยน GRID Token โดยใช้ AMI Simulator
- 3) เพื่อประเมินและวิเคราะห์ประสิทธิภาพโดยใช้การวิเคราะห์ตามลำดับขั้นด้วย BLOCKBENCH และชุดการทดสอบ TPC Benchmark

II. งานวิจัยที่เกี่ยวข้อง: วิธีการ BLOCKBENCH

BLOCKBENCH [4] นำเสนอกรอบการทำงานอย่างเป็นระบบสำหรับการประเมินประสิทธิภาพบล็อกเชนส่วนตัว (Private Blockchain) ผ่านสถาปัตยกรรม 4 ชั้น ได้แก่:

- 1) **Consensus Layer:** วัดค่าใช้จ่ายของกระบวนการฉันทามติ (Consensus Overhead) ผ่านการทำงานแบบ DoNothing
- 2) **Execution Layer:** ประเมินการประมวลผล Smart Contract ผ่านภาระงานที่ใช้หน่วยประมวลผลสูง (CPU-intensive)
- 3) **Data Model Layer:** ทดสอบประสิทธิภาพการอ่าน/เขียนสถานะโดยใช้ IOHeavy Benchmarks
- 4) **Application Layer:** การจำลองภาระงานในสถานการณ์จริงโดยใช้ YCSB และ Smallbank

งานวิจัยนี้ได้ขยายวิธีการของ BLOCKBENCH ไปยัง Solana/Anchor ด้วยการพัฒนา Benchmark แต่ละหมวดหมู่ด้วยภาษา Rust แบบเนทีฟ (Native Rust Implementation)

งานวิจัยก่อนหน้านี้ได้สำรวจการประยุกต์ใช้บล็อกเชนสำหรับการซื้อขายพลังงาน เช่น โครงการ Brooklyn Microgrid [2] และโครงการที่พัฒนาบน Ethereum หรือ Hyperledger Fabric [5], [6] อย่างไรก็ตาม ระบบเหล่านี้มักเผชิญกับข้อจำกัดด้านความสามารถในการขยายระบบ (Scalability)

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

และความหน่วงที่สูงเมื่อปริมาณธุรกรรมเพิ่มขึ้น งานวิจัยนี้นำเสนอทางเลือกใหม่โดยอาศัยสถาปัตยกรรมประสิทธิภาพสูงของ Solana

III. สถาปัตยกรรมระบบ

แพลตฟอร์มถูกพัฒนาขึ้นในรูปแบบชุดของ Anchor Smart Contracts จำนวน 5 สัญญา ที่เชื่อมต่อกันบนเครือข่าย Solana โดยอาศัย Program Derived Addresses (PDAs) เพื่อบริหารจัดการสิทธิ์การเข้าถึงข้อมูลอย่างปลอดภัย และเป็นระบบ ประกอบด้วย: 1. **Registry**: จัดการข้อมูลอัตลักษณ์ผู้ใช้และสินทรัพย์ประเภทมิเตอร์ 2. **Oracle**: ตรวจสอบความถูกต้องของข้อมูลจากเซ็นเซอร์ภายนอกเครือข่าย (Off-chain Sensors) 3. **Energy Token**: โทเค้นมาตรฐาน SPL ซึ่งแทนค่าพลังงาน 1 kWh 4. **Trading**: ระบบสมุดคำสั่งซื้อขาย (Order Book) และกลไกการชำระราคา (Settlement Engine) 5. **Governance**: การ ตั้ง ค่าระบบ และ พารามิเตอร์ของ DAO

IV. ระเบียบวิธีวิจัย

A. การเทียบเคียง TPC-C (TPC-C Mapping)

งานวิจัยนี้ได้ทำการเทียบเคียงธุรกรรมมาตรฐาน TPC-C [1] กับธุรกรรม การซื้อขายพลังงานที่เทียบเท่ากัน เพื่อสร้างโปรไฟล์ภาระงานที่มีความสมจริง ค่า tpmC (Transactions Per Minute Type C) คำนวณจาก จำนวนธุรกรรมคำสั่งซื้อใหม่ (New-Order) ที่สำเร็จต่อนาที:

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

B. แบบจำลองทางคณิตศาสตร์

1) **กลไกการกำหนดราคา VWAP**: เพื่อให้มั่นใจว่าราคาตลาดมีความยุติธรรม แพลตฟอร์มใช้อัลกอริทึมการค้นหาราคาเฉลี่ยถ่วงน้ำหนักตามปริมาณ (Volume-Weighted Average Price: VWAP) โดยราคาดุลยภาพ (Clearing Price) $P_{clearing}$ จะถูกคำนวณแบบไดนามิกดังนี้:

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

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

โดยที่ V_{trade} คือปริมาณการจับคู่ปัจจุบัน, V_{total} คือปริมาณรวมในอดีต และ δ_{max} คือค่าสัมประสิทธิ์ความยืดหยุ่นของราคาสูงสุด (10%)

2) **กฎการอนุรักษ์โทเค้น**: ระบบบังคับใช้กฎการอนุรักษ์พลังงานอย่างเคร่งครัด โดยโทเค้น ($\Delta Supply_{GRID}$) จะสามารถสร้างขึ้นได้ก็ต่อเมื่อพลังงานที่ผลิตจริงทางกายภาพได้รับการชำระบัญชีทางคณิตศาสตร์แล้วเท่านั้น:

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

3) **การปรับปรุงประสิทธิภาพระบบ**: เพื่อเพิ่มปริมาณงานสูงสุดภายใต้ขีดจำกัด Compute Unit (CU) ของ Solana เราได้ดำเนินการปรับปรุงประสิทธิภาพที่สำคัญ 3 ประการ:

A. การ คำนวณ ด้วย เลขจำนวน เต็ม เท่านั้น (Integer-only Arithmetic): การดำเนินการเลขทศนิยม (f64) ใช้ทรัพยากรการคำนวณสูงและไม่แนะนำให้ใช้ในโปรแกรม Solana เราจึงแทนที่การคำนวณ VWAP ด้วยระบบเลขจำนวนเต็มแบบจุดคงที่ (Fixed-point Arithmetic):

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

การปรับลดนี้ช่วยประหยัดทรัพยากรได้ประมาณ 10,000 CUs ต่อการทำธุรกรรม

B. การอัปเดตสถานะแบบ Lazy Update: แทนที่จะทำการ Serialize ออร์เรย์ประวัติราคาทั้งหมด (มากกว่า 100 รายการ) ในทุกธุรกรรม ระบบจะทำการบันทึกข้อมูลแบบ "Lazy Commit" เฉพาะเมื่อค่าความเบี่ยงเบนของราคาเกิน 5% หรือทุกๆ 60 วินาที ซึ่งช่วยลดภาระงานในการทำ Serialization ลงได้ถึง 90%

C. การจับคู่คำสั่งแบบ Batch: คำสั่ง match_orders ถูกปรับโครงสร้างโค้ด (Refactor) เพื่อรองรับการทำงานแบบ Batch ทำให้สามารถชำระ Limit Orders หลายรายการที่ไม่ทับซ้อนกันได้ภายในธุรกรรมเดียวแบบอะตอมิก ซึ่งช่วยปรับปรุงอัตรา "Fills Per Second" ได้อย่างมีนัยสำคัญ

V. การตั้งค่าการทดลอง

การทดสอบ Benchmark ดำเนินการบน Solana Localnet Cluster ประสิทธิภาพสูง (Single Node Setup) เพื่อวัด ประสิทธิภาพสูงสุด ของ Validator Node เดียว โดย ปราศจาก ปัจจัย ความหน่วง จาก เครือข่าย ภายนอก

- **ฮาร์ดแวร์**: Apple M-Series (8-core), 16GB RAM
- **Cluster**: Solana Test Validator v1.18
- **Client**: โปรแกรม สร้าง ภาระ งาน ด้วย ภาษา Rust แบบ Multi-threaded
- **Baseline**: PostgreSQL 14 รันบน ฮาร์ดแวร์เดียวกัน สำหรับการเปรียบเทียบ Trust Premium

VI. การประเมินประสิทธิภาพ

A. การวิเคราะห์ขั้น BLOCKBENCH

ตามวิธีการ BLOCKBENCH เราประเมินแต่ละขั้นสถาปัตยกรรมอย่างอิสระ:

ตารางที่ II
ผลลัพธ์ BLOCKBENCH Micro-benchmark

ชั้น	Benchmark	TPS	ความหน่วง
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

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

ตัวเลข TPS เหล่านี้ถูกจำกัดโดยประสิทธิภาพของ Client ในการลงนามและส่งธุรกรรมในสภาพแวดล้อมการทดสอบ ไม่ใช่ขีดจำกัดสูงสุดของ Solana Runtime อย่างไรก็ตาม ผลลัพธ์แสดงให้เห็นถึงความสม่ำเสมอของการประมวลผล

DoNothing Benchmark วัด ค่า ใช้ จ่าย ของ กระบวนการ ค้นหา (225 TPS) ในขณะที่ การ ดำเนิน การ CPUHeavy Sorting ทำได้ 231 TPS แสดง ถึง ประสิทธิภาพ การ ทำงาน ของ Smart Contract ส่วน IOHeavy Benchmarks ที่ 192 TPS แสดง ถึง ภาระ งาน จาก การ ทำ State Serialization

B. ผลลัพธ์ YCSB Workload

เราใช้โปรไฟล์ YCSB Workload 3 รูปแบบ เพื่อประเมินประสิทธิภาพในชั้น Application Layer:

Smallbank OLTP Benchmark ทำได้ **1,714 TPS** โดยมีความหน่วงเฉลี่ย 5.83 ms แสดง ถึง ประสิทธิภาพ ที่ แข็งแกร่ง สำหรับ ภาระ งาน ด้าน ธุรกรรมทางการเงิน

C. การวิเคราะห์ปริมาณงาน TPC-C

การใช้งาน TPC-C Benchmark ของเราทำได้ปริมาณงาน **2,111 tpmC** (76.85 TPS) ในโหมดการจำลอง ปริมาณงานนี้เกิดจาก Sealevel Parallel Runtime โดยตรง ซึ่งอนุญาตให้ธุรกรรมที่ไม่ทับซ้อนกัน (เช่น การซื้อขายระหว่างคู่ผู้ใช้ที่ต่างกัน) ทำงานพร้อมกันได้ ในขณะที่บล็อกเชนแบบ EVM ดั้งเดิม ธุรกรรมเหล่านี้จะถูกจัดลำดับ (Serialize) ทำให้เกิดคอขวด

ในการรัน Benchmark ของเรา มีธุรกรรมทั้งหมด 4,621 รายการถูกประมวลผล โดยสำเร็จ 4,611 รายการ (**อัตราสำเร็จ 99.78%**) สัดส่วนประเภทธุรกรรมเป็นไปตามข้อกำหนด 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%

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

สำหรับตลาด P2P ระดับภูมิภาคที่มี 10,000 คริวเรือน สมมติว่าสมาร์ทมิเตอร์แต่ละตัวส่งค่าทุก 15 นาที ปริมาณงานที่ต้องการจะอยู่ที่ประมาณ 11 TPS ซึ่งค่า 76.85 TPS ที่วัดได้นั้นให้ **ส่วนเผื่อความปลอดภัยถึง 7 เท่า** พิสูจน์ว่าสถาปัตยกรรมสามารถรองรับการติดตั้งใช้งานในระดับละเวกบ้านได้โดยไม่เกิดปัญหาความแออัด

D. ผลลัพธ์ TPC-E และ TPC-H

เราขยายการประเมินเพื่อรวม TPC-E (ภาระงานธุรกิจหลักทรัพย์) และ TPC-H (ระบบสนับสนุนการตัดสินใจ) Benchmarks:

TPC-E Benchmark ทำได้ **306 tpsE** (Trades Per Second) แสดงประสิทธิภาพที่แข็งแกร่งสำหรับภาระงานแบบนายหน้าที่มีการจับคู่คำสั่งที่ซับซ้อน ส่วน TPC-H Analytical Queries ทำได้ **250,486 QphH** (Queries Per Hour) ยืนยันความสามารถของแพลตฟอร์มสำหรับฟังก์ชันการรายงานและการตรวจสอบ

E. การวิเคราะห์ความหน่วง

ความหน่วงของธุรกรรมแตกต่างกันอย่างมากจาก "เวลาในการยืนยันธุรกรรม" ในสภาพแวดล้อม PoA แบบ Permissioned นี้

- 1) **ความหน่วงเฉลี่ย (116.56 ms):** แทนเวลาเฉลี่ยสำหรับ Leader Node ในการประมวลผลคำสั่งและอัปเดตสถานะในหน่วยความจำ
- 2) **ความหน่วง p50 (112.57 ms):** เวลาประมวลผลธุรกรรมค่ามัธยฐาน
- 3) **ความหน่วง p95 (180.04 ms):** ที่เปอร์เซ็นไทล์ที่ 95 ภายใต้โหลดปกติ
- 4) **ความหน่วง p99 (215.54 ms):** แทนเวลาประมวลผลในกรณีที่แย่ที่สุดภายใต้โหลดหนัก

โหมดการจำลองรวม "Think Time" ตามมาตรฐาน TPC-C เพื่อจำลองพฤติกรรมผู้ใช้จริง ในสภาพแวดล้อม PoA ระดับการผลิต (Production) ที่มี Validators ที่ได้รับการปรับปรุง ความหน่วงคาดว่าจะต่ำกว่านี้อย่างมีนัยสำคัญ ซึ่งแตกต่างจาก Mainnets สาธารณะที่ Finality อาจใช้เวลาหลายวินาที ธรรมชาติแบบ Permissioned ของ GridTokenX Cluster ช่วยให้ความแน่นอนของสถานะสุดท้าย (Deterministic Finality)

F. การวิเคราะห์ Concurrency

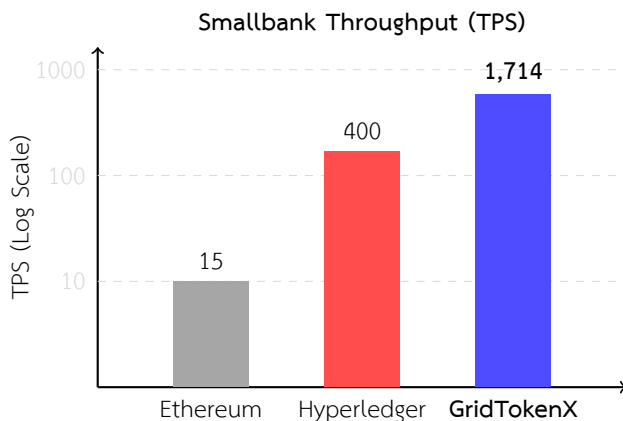
ภายใต้สภาวะการแข่งขันสูง (หลายคำสั่งต่อสถานะตลาดเดียวกัน) Sealevel Runtime สามารถจัดลำดับธุรกรรมที่ขัดแย้งกันให้เป็นเส้นตรง (Linearize) ได้อย่างมีประสิทธิภาพ ในขณะที่ประมวลผลคำขอที่ไม่ทับซ้อนกันในรูปแบบขนาน อัตราความขัดแย้ง MVCC (Multi-Version Concurrency Control) ที่สังเกตได้คือ **1.45%** ซึ่งส่วนใหญ่เกิดจากธรรมชาติแบบอะตอมิกของคำสั่ง match_orders จำนวนครั้งการลองใหม่เฉลี่ยคือ 0.02 แสดงถึงประสิทธิภาพในการจัดการความขัดแย้ง

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	ลดลงในลักษณะเชิงเส้น

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)

G. การวิเคราะห์ความสามารถในการขยายตัว

เราทำการทดสอบความสามารถในการขยายตัวใน 3 มิติ:

H. การเปรียบเทียบข้ามแพลตฟอร์ม

เราเปรียบเทียบประสิทธิภาพของ GridTokenX กับผลลัพธ์ที่เผยแพร่จากแพลตฟอร์มบล็อกเชนอื่น ดังแสดงในตารางที่ VI และรูปที่ 1:

แม้ว่า Hyperledger Fabric จะ ทำ ปริมาณ งาน DoNothing ได้ สูง กว่า (3,500 vs 225 TPS) เนื่องจากสถาปัตยกรรมแบบ Execute-Order-Validate แต่ GridTokenX แสดงให้เห็นถึง **ประสิทธิภาพ Smallbank ที่ดีกว่า 4.3 เท่า** (1,714 vs 400 TPS) และ **ความหน่วงต่ำกว่า 26 เท่า** (5.8 ms vs 150 ms) สำหรับภาระงานด้านธุรกรรมทางการเงิน

ตารางที่ VII
สรุปผลลัพธ์ Benchmark อย่างครอบคลุม

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

I. "Trust Premium"

เรานิยาม "Trust Premium" ว่าเป็นต้นทุนด้านประสิทธิภาพที่ต้องจ่ายเพื่อให้ได้มาซึ่งฉันทามติแบบกระจายศูนย์ (Distributed Consensus) ในเครือข่ายส่วนตัว เมื่อเปรียบเทียบกับเกณฑ์มาตรฐาน (Baseline) ของระบบรวมศูนย์ (PostgreSQL)

$$\begin{aligned}
 \text{Trust Premium} &= \frac{\text{Latency}_{\text{Blockchain}}}{\text{Latency}_{\text{Baseline}}} \\
 &= \frac{116.56 \text{ ms}}{2.00 \text{ ms}} \approx 58.28 \times
 \end{aligned}
 \tag{6}$$

แม้ว่าค่านี้จะแสดงถึงความหน่วงที่เพิ่มขึ้นเมื่อเทียบกับฐานข้อมูลแบบรวมศูนย์ แต่ถือว่าเป็นระดับที่ยอมรับได้สำหรับแอปพลิเคชันการซื้อขายพลังงาน ซึ่งในระบบดั้งเดิมกระบวนการชำระราคาอาจใช้เวลาหลายวันหรือหลายสัปดาห์ สิ่งที่ได้รับแลกเปลี่ยนมาคือ:

- ร่องรอยการตรวจสอบธุรกรรมที่ไม่สามารถแก้ไขได้ (Immutable Audit Trail)
- การชำระราคาอัตโนมัติด้วย Smart Contract
- กลไกการกำหนดราคาที่เป็นธรรม
- การขจัดความเสี่ยงจากการล้มเหลวที่จุดเดียว (Single Point of Failure)

J. สรุปผลลัพธ์ Benchmark

ตารางที่ VII สรุปเมตริกประสิทธิภาพหลักจากการประเมิน Benchmark อย่างครอบคลุมของเรา

VII. กรอบการประเมินข้ามแพลตฟอร์ม

เพื่อเปรียบเทียบ GridTokenX กับแพลตฟอร์มบล็อกเชนอื่นอย่างเข้มงวด (เช่น Ethereum, Hyperledger Fabric) [4] เราขอเสนอ *Standardized Energy Trading Workload (SETW)* ซึ่งนักวิจัยสามารถทำซ้ำวิธีการนี้ได้ โดยปฏิบัติตาม 3 ขั้นตอนดังนี้:

A. การพัฒนา Smart Contract ที่เทียบเท่า

แพลตฟอร์มเป้าหมายต้องมีการพัฒนาตรรกะหลัก (Core Logic) ตามที่กำหนดในตารางที่ I โดยเฉพาะฟังก์ชัน “Order Match” จะต้องมีความเป็นอะตอมิก (Atomic) เพื่อให้มั่นใจว่าการจับคู่คำสั่งและการชำระราคาโทเคนเกิดขึ้นภายในการเปลี่ยนสถานะทางรหัสวิทยา (Cryptographic State Transition) เดียวกัน

B. โพรไฟล์การป้อนภาระงาน (Workload Injection Profile)

โปรแกรมสร้างภาระงาน (Load Generator) ต้องรักษาสัดส่วนผสมของธุรกรรมตามมาตรฐาน TPC-C อย่างเคร่งครัด:

- **เน้นการเขียน (Write-heavy 88%):** ประกอบด้วย New Orders และ Payments เพื่อทดสอบการแย่งชิงสถานะ (State Contention) ของกลไกฉันทามติ
- **เน้นการอ่าน (Read-light 12%):** ประกอบด้วย Status Checks เพื่อทดสอบประสิทธิภาพการสอบถามข้อมูล (RPC Query)

C. การปรับฐานเมตริก (Metric Normalization)

ผลลัพธ์ต้องรายงานในหน่วย **tpmC** (Transactions Per Minute Type-C) สำหรับบล็อกเชนที่มีความแน่นอนของสถานะสุดท้ายแบบน่าจะเป็น (Probabilistic Finality) เช่น PoW การวัดความหน่วงจะต้องรวมเวลาจนถึงระดับความลึกของการยืนยันบล็อกที่ k : (k -block confirmation depth) เพื่อให้สามารถเปรียบเทียบกับตารางเวลาผู้นำแบบกำหนดได้ (Deterministic Leader Schedule) ของ Solana ได้อย่างถูกต้อง

VIII. บทสรุป

GridTokenX แสดงให้เห็นถึงความเป็นไปได้ในการนำเทคโนโลยีบล็อกเชนส่วนตัวมาประยุกต์ใช้กับระบบซื้อขายพลังงาน P2P ในระดับภูมิภาคผ่านผลการวิเคราะห์ด้วย BLOCKBENCH อย่างครอบคลุมและการประเมินด้วยชุดทดสอบ TPC Benchmark:

- **Consensus Layer:** ปริมาณงาน DoNothing 225 TPS
- **Execution Layer:** การประมวลผล 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

เมื่อเปรียบเทียบกับ Hyperledger Fabric พบว่า GridTokenX สามารถทำ ปริมาณงาน Smallbank ได้สูงกว่า 4.3 เท่า และมีความหน่วงต่ำกว่า 26 เท่า สำหรับภาระงานด้านธุรกรรมทางการเงิน ค่า Trust Premium ที่ 58.28 เท่า ที่สังเกตได้ ถือเป็นต้นทุนที่ยอมรับได้แลกกับการได้มาซึ่งร่องรอยการตรวจสอบที่ไม่สามารถแก้ไขได้ การชำระราคาอัตโนมัติด้วย Smart Contract และกลไกการกำหนดราคาที่โปร่งใส

งานในอนาคตจะมุ่งเน้นไปที่: (1) การติดตั้งใช้งานบนกลุ่ม PoA Validator ระดับการผลิต (Production Grade) เพื่อลดความหน่วงให้ต่ำลง (2) การขยายระบบเพื่อรองรับการตั้งค่าคลังข้อมูลขนาดใหญ่ (10+ Warehouses) และ (3) การทดสอบนำร่องในพื้นที่จริง (Real-world Pilot Testing) ด้วยการเชื่อมต่อกับสมาร์ทมิเตอร์

หนังสืออ้างอิง

- [1] Transaction Processing Performance Council (TPC), “TPC Benchmark C Standard Specification, Revision 5.11,” 2010.
- [2] E. Mengelkamp, J. Gärttner, 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>