

# การพัฒนาระบบจำลองการซื้อขายพลังงานแสงอาทิตย์แบบ Peer-to-Peer โดยใช้ Solana Smart Contract

(Anchor Framework ในสภาพแวดล้อมแบบ Permissioned)

นายจันทร์ พิริยาดี  
คณะวิศวกรรมศาสตร์  
สาขาวิศวกรรมคอมพิวเตอร์และปัญญาประดิษฐ์  
มหาวิทยาลัยหอการค้าไทย  
กรุงเทพมหานคร ประเทศไทย  
2410717302003@live4.utcc.ac.th

บทคัดย่อ—บทความนี้นำเสนอการประเมินประสิทธิภาพอย่างครอบคลุมของ GridTokenX ซึ่งเป็นแพลตฟอร์มการซื้อขายพลังงานแบบ Peer-to-Peer (P2P) บนโครงสร้างสูญญ์ที่สร้างบน Solana cluster แบบส่วนตัว เราใช้วิธีการ BLOCKBENCH (SIGMOD 2017) สำหรับการวิเคราะห์แบบขั้นตอนย่อของระบบ ร่วมกับการตัดแปลง TPC-C benchmark สำหรับการประเมินภาระงานจริง ผลการทดสอบแสดงให้เห็นว่าแพลตฟอร์มสามารถทำได้ 2,111 tpmC (transactions per minute Type C) โดยมีค่าเฉลี่ยความหน่วง 116ms และระบุ "Trust Premium" ที่ 58.28 เท่า เมื่อเทียบกับระบบสูญญ์ BLOCKBENCH micro-benchmarks แสดงริมा�ณของ consensus layer ที่ 225 TPS, execution layer ที่ 231 TPS และ data model layer ที่ 192 TPS การวิเคราะห์ YCSB workload และ 442 ops/s สำหรับภาระงานที่อ่านมาก นอกจากนี้ยังอธิบายการปรับปรุงระดับต่ำโดยเฉพาะ รวมถึงการคำนวณแบบจำนำเงินเดือนเท่านั้นและการอัปเดตสถานะแบบ lazy ที่ช่วยเพิ่มปริมาณงานขณะรักษาค่าคงที่ทางคณิตศาสตร์อย่างเข้มงวดสำหรับการอนุรักษ์พลังงาน

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

## I. บทนำ

การกระจายสูญญ์ของระบบพลังงานผ่านห่วงโซ่การผลิตแบบกระจาย (Distributed Energy Resources) ต้องการโครงสร้างพื้นฐานการซื้อขายที่แข็งแกร่งสามารถจัดการธุรกรรมขนาดเล็กความถี่สูงได้ [2] รูปแบบสาธารณูปโภคแบบรวมสูญญ์แบบดั้งเดิมประสบปัญหาความเสี่ยงจากล้มเหลวเดียวและขาดความมั่นคงเมื่อไปร์เซ็นต์ต่ำกว่า 1% [3] เทคโนโลยีบล็อกเชนเสนอทางออกแต่แม้กุกุกวิพากษ์วิจารณ์เรื่องของจำกัดด้านความสามารถในการขยายตัว [4]

งานวิจัยนี้ประเมิน GridTokenX ซึ่งใช้ประโยชน์จาก Sealevel parallel runtime ของ Solana [3] เพื่อออกแบบขอขายพลังงานที่มีความต่อเนื่องและเชื่อมโยงกันได้

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

## II. งานวิจัยที่เกี่ยวข้อง: วิธีการ BLOCKBENCH

BLOCKBENCH [4] ให้กรอบการทำงานอย่างเป็นระบบสำหรับการประเมินประสิทธิภาพบล็อกเชนส่วนตัวผ่านเครื่องทดสอบ:

- 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 เรขาคณิตวิธีการ BLOCKBENCH ไปยัง Solana/Anchor ด้วยการใช้งาน Rust ด้วยตัวมือของแต่ละหมวดหมู่ benchmark

## III. สถาปัตยกรรมระบบ

แพลตฟอร์มถูกใช้งานเป็นชุดของห้า Anchor smart contracts ที่เชื่อมต่อกันบน Solana:  
1. **Registry:** จัดการวัตถุที่ใช้และทรัพย์สินมิเตอร์ 2. **Oracle:** ตรวจสอบข้อมูลเชื่อมต่อของเครือข่าย 3. **Energy Token:** โทเคนที่เป็นไปตาม SPL แทน 1 kWh 4. **Trading:** order book และเครื่องมือการชำระราคา 5. **Governance:** การกำหนดค่าและพารามิเตอร์ DAO

## IV. วิธีการ

### A. การแบ่ง TPC-C

รายการ	TPC-C	มาตรฐาน	[1]
กับการซื้อขายพลังงานที่เทียบเท่าเพื่อสร้างไปร์เซ็นต์ภาระงานที่สมจริง:			

รายการ	TPC-C	มาตรฐาน	ตารางที่ 1
การแบ่ง TPC-C กับการซื้อขายพลังงาน			
ผู้ใช้งาน	GridTokenX	ผู้ใช้งาน	
New Order	45%	create_sell_order / create_buy_order	
Payment	43%	transfer_tokens (การชำระเงิน)	
Order Status	4%	get_order_status (RPC Read)	
Delivery	4%	match_orders (การดำเนินการ Batch)	
Stock Level	4%	get_balance (การตรวจสอบคงเหลือ)	

### B. แบบจำลองทางคณิตศาสตร์

1) กลไกการกำหนดราคา **VWAP:** เพื่อให้มั่นใจในมูลค่าตลาดที่เป็นธรรม แพลตฟอร์มใช้อัลกอริズึมการค้นหาราคาอันเฉียบแหลม (VWAP) ราคา clearing  $P_{clearing}$  คำนวณแบบด้านนี้:

$$P_{base} = \frac{P_{bid} + P_{ask}}{2} \quad (1)$$

$$P_{clearing} = P_{base} + \left( P_{base} \times \min \left( \frac{V_{trade}}{V_{total}}, 1.0 \right) \times \delta_{max} \right) \quad (2)$$

โดยที่  $V_{trade}$  คือปริมาณการซื้อขายปัจจุบัน,  $V_{total}$  คือปริมาณในอดีต และ  $\delta_{max}$  คือปัจจัยความผันผวนราคางาน (10%)

2) ค่าคงที่การอนุรักษ์ໂທเด็น: ระบบบังคับใช้การอนุรักษ์พลังงานอย่างเข้มงวด โທเด็น ( $\Delta SupplyGRID$ ) สามารถสร้างได้เฉพาะเมื่อพลังงานที่ผลิตทางกายภาพถูกำไรทางคณิตศาสตร์:

$$\Delta SupplyGRID = \max(0, (E_{produced} - E_{consumed}) - E_{settled}) \quad (3)$$

3) การปรับปรุงระบบ: เพื่อเพิ่มปริมาณงานสูงสุดภายใต้จำกัด Compute Unit (CU) ของ Solana เราใช้การปรับปรุงสำหรับสามประการ:

A. การคำนวณแบบจำนำวนitemที่ต้องการ: การดำเนินการจุดที่นิยม (f64)  
มีค่าใช้จ่ายคำนวณสูงและไม่แนะนำในโปรแกรม Solana เราแทนที่การคำนวณ VWAP ด้วยคณิตศาสตร์จุดคงที่ที่มีจำนวนเต็ม:

$$W = \min\left(\frac{V \times 1000}{V_{total}}, 1000\right) \quad (4)$$

การลดต้นทุนประจำตัวต่อ 10,000 CU's ต่อการทำธุรกรรม

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

C. การจับคู่คำสั่งแบบ Batch: คำสั่ง **match\_orders** ถูก refactor เพื่อจัดการการทำงานแบบ batch ทำให้สามารถจำกัด limit orders หลายรายการที่ไม่ต้องหักห้ามกันในธุรกรรมเดียว ปรับปรุงอัตรา "fills per second" อย่างมีนัยสำคัญ

## V. การตั้งค่าการทดสอบ

Benchmarks ดำเนินการบน Solana localnet cluster ประเมินประสิทธิภาพสูงเช่นตัวแปรความหน่วงอินเทอร์เน็ต

- ชาร์ดแวร์:** Apple M-Series (8-core), 16GB RAM
- Cluster:** Solana Test Validator v1.18
- Client:** ตัวสร้างภาระงาน Rust แบบ multi-threaded

## VI. การประเมินประสิทธิภาพ

### A. การวิเคราะห์ชั้น BLOCKBENCH

ตามวิธีการ BLOCKBENCH เราประเมินแต่ละชั้นสถาปัตยกรรมอย่างอิสระ:

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

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

### B. ผลลัพธ์ YCSB Workload

เราใช้สามไฟล์ YCSB workload เพื่อประเมินประสิทธิภาพ application-layer:

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

ตารางที่ III  
ประสิทธิภาพ YCSB Workload

Workload	โปรดักส์	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

### C. การวิเคราะห์ประสิทธิภาพ TPC-C

การใช้งาน TPC-C benchmark ของเรามีตัวเลข 2,111 tpmC (76.85 TPS) ในโหมดจำลอง ประเมินงานนี้ก็จาก Sealevel parallel runtime โดยตรง ซึ่งอนุญาตให้ธุรกรรมที่ไม่เก็บข้อมูลกัน (เช่น การซื้อขายระหว่างคู่ผู้ใช้ที่ต่างกัน) ทำงานพร้อมกันในบล็อกเชนแบบ EVM ดังเดิม ธุรกรรมเหล่านี้จะถูก serialize ทำให้เกิดข้อ等候

ในการรัน benchmark ของเรามีธุรกรรมทั้งหมด 4,621 รายการถูกประมวลผลโดย 4,611 รายการสำเร็จ (อัตราสำเร็จ 99.78%) สำหรับธุรกรรมที่เหลือไปตามข้อกำหนด 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 ครัวเรือน สมมติว่าสามารถมีตัวตนแต่ละตัวส่งค่าทุก 15 นาที ประเมินงานที่ต้องการประมาณ 11 TPS 76.85 TPS ที่สูงเกิดได้ที่ くなไปเมื่อความปลอดภัย 7 เท่า พิสูจน์ว่าสถาปัตยกรรมสามารถจัดการติดต่อระดับโลกบ้านได้โดยไม่มีความแออัด

### D. ผลลัพธ์ TPC-E และ TPC-H

รายงานการประเมินเพื่อร่วม 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) แสดงประสิทธิภาพที่แข็งแกร่งสำหรับภาระงานแบบนายหน้าที่มีการจับคู่สำหรับชั้นห้อง TPC-H analytical queries ทำได้ 250,486 QphH (queries per hour) บีบบับความสามารถของแพลตฟอร์มสำหรับพื้นที่นักการรายงานและการตรวจสอบ

### E. การวิเคราะห์ความหน่วง

ความหน่วงธุรกรรมแตกต่างอย่างมากจาก "เวลาเย็นยัน" ในสภาพแวดล้อม PoA แบบ permissioned นี้

- ความหน่วงเฉลี่ย (115.6ms): แทนเวลาเฉลี่ยสำหรับ leader node ในการประมวลผลคำสั่งและอัปเดตสถานะใหม่โดยความจำ
- ความหน่วง p50 (112.57ms): เวลาประมวลผลธุรกรรมมีอยู่ฐาน
- ความหน่วง p95 (180.04ms): เปรอร์เซ็นไทล์ที่ 95 ภายใต้โหลดปกติ
- ความหน่วง p99 (215.54ms): แทนเวลาประมวลผลกรณีล่าวร้ายที่สุดภายใต้โหลดหนัก โดยคำนึงถึงความล่าช้าที่ยอมให้ก่ออุบัติกรรมบนบล็อกเชนที่สมจริง ในสภาพแวดล้อม PoA การผลิตที่มี validators ที่ปรับปรุงแล้ว ความหน่วงคาดว่าจะต่ำกว่าอย่างมีนัยสำคัญ ไม่เหมือนกับ mainnets สาธารณะที่ finality อาจใช้เวลาหลายวินาที ระยะเวลาต้องการของ GridTokenX cluster อนุญาตให้มี deterministic finality

#### F. การวิเคราะห์ Concurrency

ภายใต้การแข่งขันสูง (หลักคำสั่งต่อสถานะต่อกัน) Sealevel runtime linearized ธุรกรรมที่ตัดเบี้ยนกันอ่าวยีมีประสิทธิภาพในขณะที่ประมวลผลคำขอที่ไม่ทับซ้อนกันแบบขนาน อัตราความขัดแย้ง MVCC (Multi-Version Concurrency Control) ที่สังเกตได้คือ 1.45% ส่วนใหญ่มาจากธรรมชาติของโมดูลิกของคำสั่ง **match\_orders** จำนวนครั้งการลองใหม่เฉลี่ยคือ 0.02 แสดงการแก้ไขความขัดแย้งที่มีประสิทธิภาพ

#### G. การวิเคราะห์ความสามรถในการขยายตัว

เราทำการทดสอบความสามรถในการขยายตัวในสามมิติ:

ตารางที่ V

ผลลัพธ์การทดสอบความสามรถในการขยายตัว

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	ลดลงแบบเรื่อยๆ

#### H. การเปรียบเทียบข้ามแพลตฟอร์ม

เราเปรียบเทียบประสิทธิภาพ

กับผลลัพธ์ที่เผยแพร่จากแพลตฟอร์มบล็อกเชนอื่น:

#### J. สรุปผลลัพธ์ Benchmark

ตาราง VII สรุปเมตริกประสิทธิภาพหลักจากการประเมิน อย่างครอบคลุมของเรา benchmark

ตารางที่ VII

สรุปผลลัพธ์ Benchmark อย่างครอบคลุม

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

ตารางที่ VI

การเปรียบเทียบประสิทธิภาพแพลตฟอร์ม

แพลตฟอร์ม	Benchmark	TPS	ความหน่วง	แหล่งที่มา
GridTokenX	Smallbank	1,714	5.8ms	การศึกษา
GridTokenX	DoNothing	225	2.5ms	การศึกษา
GridTokenX	YCSB-B	442	1.8ms	การศึกษา
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

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

#### I. "Trust Premium"

เรานิยาม "Trust Premium" เป็นต้นทุนประสิทธิภาพที่เกิดขึ้นเพื่อให้บรรลุ distributed consensus ในเครือข่ายส่วนตัวเมื่อเทียบกับ baseline แบบรวมศูนย์ (PostgreSQL)

$$Trust\ Premium = \frac{Latency_{Blockchain}}{Latency_{Baseline}} = \frac{116.56ms}{2.00ms} \approx 58.28x \quad (5)$$

แม้ว่าจะแทนค่าใช้จ่ายความหน่วงเมื่อเทียบกับฐานข้อมูลแบบรวมศูนย์

แต่เป็นที่ยอมรับได้สำหรับแอปพลิเคชันการซื้อขายพลังงานที่การชำระราคาโดยทั่วไปใช้เวลาหลายวันหรือหลายเดือน ผู้ให้บริการต้องรับรองความถูกต้องของข้อมูลที่ได้รับ ผ่านการประเมิน BLOCKBENCH อย่างครอบคลุมและการประเมิน TPC benchmark:

- เสน่ห์ของการตรวจสอบธุรกรรมที่ไม่เปลี่ยนแปลง
- การชำระราคา smart contract อัตโนมัติ
- กติกาการกำหนดราคาที่ปรับเปลี่ยนได้
- การซัพพอร์ตความเสี่ยงด้วยตัวเลือกเดียว
- Consensus Layer: บริมาณงาน DoNothing 225 TPS
- Execution Layer: การคำนวณ CPUHeavy 231 TPS
- Data Model: การดำเนินการสถานะ IOHeavy 192 TPS
- Application Layer: Smallbank 1,714 TPS, YCSB-B 442 ops/s

#### VII. กระบวนการประเมินข้ามแพลตฟอร์ม

เพื่อเปรียบเทียบ GridTokenX กับแพลตฟอร์มบล็อกเชนอื่นอย่างเชิงมวงด (เช่น Ethereum, Hyperledger Fabric) [4] เราเสนอ Standardized Energy Trading Workload (SETW) นักวิจัยต้องดำเนินการตามสามขั้นตอนเพื่อทำข้าวสารนี้:

##### A. การใช้งาน Contract ที่เทียบเท่า

แพลตฟอร์มเป้าหมายต้องใช้งาน logic หลักที่ทำหน้าที่ในตาราง I โดยเฉพาะ ฟังก์ชัน "Order Match" ต้องเป็น atomic ให้แน่ใจว่าการทำธุรกรรมและการคำนวณต้องเกิดขึ้นในกระบวนการเปลี่ยนสถานะ cryptographic เดียวกัน

##### B. โปรไฟล์การคัดกรองงาน

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) สำหรับบล็อกเชนที่มี probabilistic finality (เช่น PoW) ความหน่วงต้องรองรับมาตรฐาน k-block confirmation depth เพื่อเปรียบเทียบกับ deterministic leader schedule ของ Solana

#### VIII. สรุป

GridTokenX แสดงให้เห็นว่าเทคโนโลยีบล็อกเชนที่น่าเชื่อถือสามารถดำเนินการซื้อขายพลังงานได้ดีกว่าระบบเดิมๆ ผ่านการวิเคราะห์ชั้น BLOCKBENCH อย่างครอบคลุมและการประเมิน TPC benchmark:

- Consensus Layer: บริมาณงาน DoNothing 225 TPS
- Execution Layer: การคำนวณ 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 ที่สังเกตได้แทนดันทุนที่ยอมรับได้สำหรับลูกค้าทางการตรวจสอบที่เปลี่ยนแปลง การชำระราคา smart contract อัตโนมัติ และกลไกการกำหนดราคาที่ปรับเปลี่ยนในอนาคตจะมุ่งเน้นที่:
  - (1) การติดตั้งบน PoA validator clusters
  - ระดับการผลิตเพื่อให้ได้ความหน่วงต่ำลง (2) การขยายไปยังการกำหนดค่า warehouse ขนาดใหญ่ (10+ warehouses) และ (3) การทดสอบนำร่องในโครงการตัวอย่างการผลิตและสมาร์ทมิเตอร์

พัฒนาด้วยภาษา Python

- [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>