Implementation of Peer-to-Peer Energy Trading via Solana Smart Contracts Permissioned Environment

Chanthawat Kiriyadee[1], and Suwannee Adsavakulchai[1*]

[1]Department of Computer Engineering and Artificial Intelligence, School of Engineering, University of the Thai Chamber of Commerce (UTCC), Bangkok, Thailand

Contact Email: 2410717302003@live4.utcc.ac.th, Suwannee_ads@utcc.ac.th

Thailand's three state utilities EGAT, MEA, and PEA have advanced a National Energy Trading Platform (NETP) since 2017 to facilitate Peer-to-Peer (P2P) trading among rooftop-solar prosumers, envisaging a blockchain-based settlement layer with a dedicated digital currency. However, remains in development, and Thailand's regulated buy-back scheme still provides no direct mechanism for prosumers to exchange surplus generation within the local electrical grid. This paper presents a smart-contract-based on-chain settlement layer for P2P energy trading, realized as six modular Anchor programs deployed on a permissioned Solana runtime that utilize the Sealevel parallel-execution engine. By routing every high-frequency write to a per-entity program-derived account (PDA), the system permits transactions with disjoint write sets to execute in parallel, with transaction ordering provided by Solana's Proof-of-History (PoH). The evaluate market clearing price framework couples a continuous double auction (CDA) for real-time trades with a uniform-price auction that clears aggregated prosumer bids into a single settlement price over fifteen-minute epochs. We simulate energy consumption, generation, and power flow for a network of M meters over D days at fifteen-minute intervals within a low-voltage grid topology, of which 15% of meters are prosumers. The full token life-cycle minting, settlement, and burning is evaluated on a single-node validator (Apple M2, Agave 3.1.10). We report the resulting throughput, energy-conservation, and net-revenue outcomes, each independently re-derived and audited from on-chain state.

1. Introduction

Thailand's state utilities have advanced a National Energy Trading Platform (NETP) since 2017 to facilitate P2P trading among rooftop-solar prosumers, using a Tendermint-based consortium blockchain [1]. This article present a Solana-based architecture as an alternative execution layer, motivated by NETP's own finding that Tendermint outperformed Ethereum and Hyperledger Fabric in throughput testing suggesting further gains may be available from Solana's parallel execution model. The Smart Contract is built with six programs on Anchor Framework, a Solana official framework using Rust programming language complied to SBF bytecode and executed on the Solana Virtual Machine (SVM). By routing high-frequency writes to per-entity program-derived accounts (PDAs), transactions with disjoint write sets never contend for the same account lock and therefore execute in parallel under Solana's Sealevel runtime.

2. Methodology

The contribution of this work are:

- The smart-contract layer is built on the Solana Anchor framework [2] (anchor-lang and anchor-spl, version 1.0.0) and comprises five core programs **energy-token, governance, oracle, registry, trading and treasury** program, all written in Rust and compiled to SBF bytecode. Anchor supports a modular program architecture in which one program invokes instructions of another through cross-program invocation (CPI); authority for such calls is delegated to program-derived addresses (PDAs), which sign CPIs without any private key existing. The deployment targets a permissioned environment realised as a private network of `solana-test-validator`. All experiments reported here were executed on a single-node validator on local hardware (Apple M2, Agave 3.1.10).
- Fleet datasets use the CINELDI 80-bus rural LV reference grid (SINTEF) [3] for network topology only the 80-node single-phase feeder with its V/A/W parameters. The grid is imported from MATPOWER, translated to a GridLAB-D (v5.3.0) model, and its AC power flow resolved with pandapower [4]. As CINELDI carries only hourly load and no generation, all time-series are simulator-produced: per-meter consumption and twelve prosumer nodes' (15%) rooftop PV are generated at fifteen-minute cadence, with solar modelled deterministically via pvlib [5] under a fixed seed.

3. Key Results and Discussion

All measurements use a single-node validator (Apple M2, Agave 3.1.10); throughput is client-observed confirmed goodput (confirmed transactions per wall-clock second, burst start to last confirmation), characterising execution-layer runtime and account locking rather than consensus throughput. Every non-confirmed transaction is attributed to one failure class (rejected at submission, failed an on-chain guard, or expired unconfirmed), separating delivery loss from validation working as intended. The community-month evaluation is grounded in the simulator setup: 80 meters (seeded solar/load physics), 12 prosumer sellers under a 10 kWh/day cap, 68 consumers, 30 days × 96 ticks = 230,400 readings; month energy 15,868.5 kWh generated, 144,879.8 kWh consumed, 10,386.7 kWh interval surplus.

3.1. Equations and Data

Table 1: The simulated community-month dataset (seed-deterministic; no field data).

Quantity	Value
fleet	80 meters (seeded solar/load physics): 12 prosumer sellers, 68 consumers
market policy	sell cap $C = 10$ kWh/day per prosumer; 0.1 kWh dust floor
horizon / readings	30 days × 96 ticks (Δt = 15 min) = 230,400 readings, integer Wh
month energy	15,868.5 kWh generated; 144,879.8 kWh consumed; 10,386.7 kWh interval surplus
oracle-accepted surplus	8,253.502 kWh; 919 readings gate-rejected

The outcomes replays in 1,394.7 s (1,858× real time) with zero delivery loss; all 919 rejected readings are deterministic anomaly-gate rejections. Telemetry sustains ≈213 readings·s^{-1}, order bursts ≈53 TPS, and each Ed25519-verified settlement costs 107 k CU. The closure identities of hold exactly on chain (15/15 audit assertions), and they hold under all three price rules; net revenue orders uniform 2.290 > buyback 2.200 > CDA 2.065 ฿/kWh, so wheeling policy sets the P2P participation threshold.

4. Conclusion

The measurements support the central design claim with hot-path state partitioned into per-entity PDAs, the residual scaling limit is consensus and lock serialization rather than computation.

4. Acknowledgment

The authors thank the University of the Thai Chamber of Commerce (UTCC) for institutional support, and the organizers, the UEC ASEAN Research Center (UARC) and Multimedia University (MMU), for hosting this workshop.

4. References

[1] Metropolitan Electricity Authority (MEA) and Electricity Generating Authority of Thailand (EGAT) and Provincial Electricity Authority (PEA), "Development of the National Energy Trading Platform (NETP): Research and Development Project Report and Digital Platform Roadmap for Thailand's Electricity Authorities," technical report, Dec. 2019.

[2] Coral / Anchor contributors, "Anchor: Solana's Sealevel runtime framework." [Online]. Available: https://www.anchor-lang.com/

[3] L. M. Engan, S. Ekrheim, S. Bjarghov, J. Klemets, G. Kjølle, and I. Schytte, Reference dataset for semi-urban and rural Norwegian low voltage distribution system. (Dec. 2024). Zenodo. doi: 10.5281/zenodo.14528192.

[4] L. Thurner et al., "pandapower—An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems," IEEE Transactions on Power Systems, vol. 33, no. 6, pp. 6510–6521, 2018.

[5] W. F. Holmgren, C. W. Hansen, and M. A. Mikofski, "pvlib python: A python package for modeling solar energy systems," Journal of Open Source Software, vol. 3, no. 29, p. 884, 2018, doi: 10.21105/joss.00884.