



## Article

# PoET<sup>2</sup>S: Folding a digital twin into validator selection for blockchain consensus in peer-to-peer energy trading

Zainab S Idan<sup>1</sup>, Hussein Akeel Hussein Alaasam<sup>2</sup>, Firas H. Neamah<sup>3\*</sup>, Yahya Mahdi Hadi Al-Mayali<sup>1</sup>, Hussein Alrammahi<sup>4</sup>

<sup>1</sup>Department of Computer Engineering Technology, College of Technical Engineering, University of Alkafeel, Najaf, 54001, Iraq

<sup>2</sup>College of Basic Education, University of Kufa, Najaf, Iraq

<sup>3</sup>Department of Cybersecurity Engineering, Al-Maaqal University, Basra, Iraq

<sup>4</sup>Department of Software Engineering, Faculty of Engineering and Architecture, Altınbaş University, Mahmutbey Dilmenler Caddesi No. 26, Bağcılar, Istanbul, 34217, Türkiye

## ARTICLE INFO

## Article history:

Received 07 May 2026

Received in revised form

02 October 2026

Accepted 19 October 2026

## Keywords:

Blockchain consensus, Digital twin, Peer-to-peer energy trading, Validator selection, Committee integrity, Smart grid

\*Corresponding author

Email address:

[frs.hammodi@almaaqa.edu.iq](mailto:frs.hammodi@almaaqa.edu.iq)

DOI: 10.55670/fpll.futech.5.4.29

## ABSTRACT

Blockchain platforms for peer-to-peer energy trading often add a digital twin that forecasts each meter's generation and demand. In most designs the twin only raises alarms, while validators are still chosen by stake or computation. The proposed scheme, PoET<sup>2</sup>S, incorporates the twin's evidence into validator selection: a node's weight is a stake-and-energy term multiplied by its Twin-Trust Score raised to  $\gamma = 2.5$ , so nodes that contradict the physical forecast quickly lose committee seats. A discrete-event simulator compares PoET<sup>2</sup>S (500 nodes, 15% Byzantine, 30 paired seeds) with seven classical baselines and three design points from recent work: a binary veto, an additive energy-and-trust score, and a linear ( $\gamma = 1$ ) reputation weight. Detection is a near-tie: PoET<sup>2</sup>S catches 91.0% of attackers and the veto 91.7%. PoET<sup>2</sup>S raises more false alarms per catch by default (13.9 against 9.7) but matches the veto's catch rate at equal false-alarm cost. The malicious committee share is 10.8% under the veto, 3.8% under the linear weight, and 2.6% under PoET<sup>2</sup>S, and on 100 seeds the gain over the linear weight is 37% (Holm-adjusted  $p = 7.2 \times 10^{-9}$ ). At 45% Byzantine nodes, the one-third fault bound fails in 64.4% of veto blocks and 2.4% of PoET<sup>2</sup>S blocks. Simulated commit latency (30.2 ms) is higher than for Proof of Stake and Delegated Proof of Stake, lower than for Practical Byzantine Fault Tolerance, and flat up to 4,000 nodes; an optional fast path lowers it to 26.8 ms. Against Sybil attacks, top-K selection needs certified meter identities.

## 1. Introduction

Peer-to-peer energy markets let households with rooftop photovoltaic (PV) panels sell surplus electricity to their neighbors without routing every trade through a utility [1,2]. Without a central operator, households that do not trust one another must still agree on who produced what, who owes whom, and which trades are valid. Blockchains are the usual answer to this agreement problem [3]. A second line of work adds a digital twin, a software model of each meter and feeder that predicts what a node should produce and consume at a given time [4]. A household claiming to export 4 kW at midnight is easy to flag against such a forecast, so the twin acts as a lie detector for false-data injection (FDI) [5]. In most published systems, the twin and the consensus protocol run

side by side. The twin checks data and raises alarms, and the blockchain chooses validators by stake or computation, as it would for a cryptocurrency [6,7]. The decision about who writes the next block therefore ignores what the twin knows about the physical grid. This paper studies the case where that knowledge informs validator selection. PoET<sup>2</sup>S (Proof of Energy and Twin-Trust Score) places the twin inside validator selection. Each node carries a Twin-Trust Score (TTS) that rises when its reports match the twin forecast and falls when they do not. The score multiplies the selection weight and is raised to an exponent, so a node that keeps disagreeing with the physical grid is flagged and also loses its chance to validate. A fixed committee of the highest-weighted nodes

runs a three-phase commit, keeping the message count per block independent of network size.

This paper's comparison is designed to isolate that design choice. Besides the classical baselines (Proof of Work, Proof of Stake, Practical Byzantine Fault Tolerance (PBFT), Delegated Proof of Stake and twin-augmented versions of the last three), PoET<sup>2</sup>S is compared with three stylized design points that share its committee, commit protocol, message pattern and twin, and differ only in how the twin's evidence reaches selection: a binary veto, an additive energy-and-trust score and a linear reputation weight. The design points borrow one mechanism each from recent work [8,9] and from reputation-weighted selection (Table 1) but are not re-implementations of those systems; Section 2.4 lists the differences. The linear weight is the  $\gamma = 1$  case of the PoET<sup>2</sup>S weight, so the gap between the two measures the effect of the exponent. The results do not show PoET<sup>2</sup>S ahead on every axis. On raw detection, the veto is marginally ahead. The difference appears in committee integrity, that is, in how many Byzantine nodes end up in the validating committee. This quantity is not captured by aggregate detection rates, yet it bounds what an adversary can do once it holds seats. The objectives of this study are:

- to define a validator-selection rule that uses the twin's trust signal as a continuous weight, with every symbol and update rule stated explicitly (Sections 4 and 5);
- to build and validate a discrete-event simulator in which each protocol is a message-passing state machine and commit latency is measured, not computed from a formula (Section 7);
- to compare PoET<sup>2</sup>S, on the same attack realizations, with seven classical baselines and three twin-based design points, reporting detection, false-positive cost, committee infiltration and latency with dispersion and corrected significance tests (Section 8);
- to test whether the committee-integrity result holds across Byzantine fractions, adaptive attackers, parameter changes, network sizes, a Sybil identity-splitting attack and a matched false-positive comparison, and to measure how often each committee loses the one-third fault bound (Sections 8.5–8.11).

The corresponding contributions are a trust-weighted selection rule, PoET<sup>2</sup>S (Section 5); an open discrete-event simulator with per-seed outputs (Section 7 and the Data availability statement); a like-for-like comparison showing that detection is a near-tie while committee infiltration falls from 10.8% (veto) and 3.8% (linear) to 2.6% (Section 8.3); and a set of robustness studies, including a safety-violation analysis (Section 8.5), a Sybil study with tested countermeasures (Section 8.10) and a comparison at matched false-positive cost (Section 8.11). The study is limited to consensus and validator selection for a peer-to-peer (P2P) energy platform. The simulator does not clear a market or compute prices; market clearing is treated elsewhere [10]. The "profit" reported in Section 8.6 proxies the energy over-reported by malicious nodes while they hold validator seats.

## 2. Related work

### 2.1 Consensus for peer-to-peer energy markets

Energy-market blockchains have mostly adopted general-purpose consensus [11]. Proof of Work (PoW) is slow and energy-intensive, and Proof of Stake and its delegated form tie influence to wealth, so a large stakeholder can dominate validation regardless of the accuracy of its meter readings [12]. PBFT gives deterministic finality, but its all-to-

all voting costs  $O(V^2)$  messages for  $V$  validators [13]. None of these protocols consults the physical grid when it chooses who validates.

### 2.2 Digital twins as anomaly detectors

Twins are used in smart grids to forecast generation and demand and to detect FDI [4,5]. Recent work attaches a twin to a blockchain so that data are checked before they are committed. Suggala et al. [6] protect smart-grid twin data with a blockchain layer and an interpretable generalized additive neural network. Dutta et al. [7] combine a twin with contrastive and causal learning to detect intrusions in blockchain-based Internet of Things (IoT) systems. In both cases, the twin is a detector: it can reject data or a node, but it does not decide how much influence the remaining nodes have in consensus.

### 2.3 Reputation and trust in consensus

Trust and reputation have been used to bias block generation in permissioned chains. Li et al. [14] used a consortium blockchain with pre-selected nodes for energy trading in industrial IoT, and Kang et al. [15] chose miners by reputation computed from interaction history. Three recent systems are closest to the present work. Weerasinghe et al. [8] propose Proof-of-Monitoring (PoM), a consensus for managing 5G service-level agreements (SLAs) in which monitoring nodes verify SLA violations before a block is accepted. Saleh and Cevik [9] propose an energy-aware consensus (EACM) for smart-grid IoT that ranks validators by residual battery energy and a behavioral trust score. Deng et al. [16] partition a deep neural network across a blockchain-enabled industrial IoT (IIoT) twin and adjust each node's Proof-of-Work difficulty based on its reputation. In all of these, trust comes from monitoring, device state, or interaction history. In PoET<sup>2</sup>S, the trust signal is physical: it measures agreement with an energy forecast that a node cannot change without changing what it reports.

### 2.4 Design points used for comparison

None of the three systems above can be run unchanged in an energy-trading simulator. PoM monitors SLAs rather than meter readings, EACM ranks by battery energy, and Deng et al. modify PoW difficulty. We therefore extract one selection mechanism from each and implement it as a stylized design point on the PoET<sup>2</sup>S committee machinery. Table 1 lists each design point, its source mechanism, and the deviations from that source. Section 8 compares mechanisms, not the published systems, whose reported figures come from different testbeds.

## 3. System and threat model

### 3.1 Participants and the twin trust model

The network has  $N$  nodes: prosumers with PV panels (45% of nodes), consumers without PV (40%), and nodes registered as validators (15%), which also own PV. Each node's revenue meter signs its generation and demand report for every 12-s slot with a device key; a physically unclonable function (PUF) can bind that key to the meter hardware [17]. The twin is operated by the distribution system operator or an accredited aggregator, not by the nodes. It runs in a trusted execution environment (TEE) such as Intel SGX or Arm TrustZone [18,19], receives the signed reports, and knows each node's registered PV capacity and demand profile. After each slot the twin writes every node's TTS and verified energy term  $e_i$  into the next block, so these values become part of the committed state that all nodes can read. A node observes only its own reports, the committed blocks and the published TTS and  $e$  values; it does not see the twin's internal forecast.

**Table 1.** Design points used for comparison and their deviations from the systems that inspired them. All run on the same committee (K = 21), commit protocol, and twin as PoET<sup>2</sup>S.

| Design point        | Mechanism borrowed                            | Selection rule here                                                                   | Main deviations from source                                                           |
|---------------------|-----------------------------------------------|---------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| Veto                | Monitor gates acceptance (PoM [8])            | Nodes with anomaly $a > 0.5$ excluded; top-K of the rest by base weight               | Twin anomaly replaces SLA monitoring; no monitor stake or reward                      |
| Additive            | Energy plus trust score with floor (EACM [9]) | $0.5 \cdot e_i / e_{\max} + 0.5 \cdot \text{TTS}$ , eligible if $\text{TTS} \geq 0.4$ | Twin-verified energy replaces battery energy; TTS replaces behavioral trust; no stake |
| Linear              | Reputation scales influence [15], [16]        | Base weight $\times \text{TTS}$ ( $\gamma = 1$ )                                      | Weight replaces PoW difficulty; identical to PoET <sup>2</sup> S except $\gamma$      |
| PoET <sup>2</sup> S | This work                                     | Base weight $\times \text{TTS}^{2.5}$                                                 | —                                                                                     |

The energy term is verified, not self-reported: the twin caps reported generation at 1.2 times its physical envelope before the value enters  $e_i$  (Section 5.1). TEEs have known weaknesses. Transient-execution attacks such as Foreshadow can extract enclave secrets [20], and an operator can roll an enclave back to an older state unless rollback protection is used [21]. Publishing the trust state on-chain after every block limits rollback, because an old state would contradict committed blocks. A fully compromised twin is out of scope, as it is for every twin-based scheme compared here.

### 3.2 Adversary

In the main experiment, 15% of nodes (75 of 500) are Byzantine, and the fraction is swept from 10% to 45% in Section 8.5. Malicious nodes are chosen at random and hold stake from the same distribution as honest nodes. Each is assigned one of four report-manipulation families, following the FDI patterns surveyed in [5,22]: generation inflation (reports 1.6–2.4 times the true value), fabricated generation from a spoofed identity (the larger of the true value and 3 kW, plus a uniform 0.5–2.5 kW), demand under-reporting (0.2–0.5 times the true value), and a combined exploit (generation 2.0–3.2 times and demand 0.4 times the true value). Identity splitting, the Sybil attack [23], is studied separately in Section 8.10. Attacks in the main experiment are static; two adaptive strategies are tested in Section 8.6. The kill-chain view of cyber-physical attacks [24] places these attacks at the data-integrity stage; attacks on the network layer (eclipse, delay) are not modeled.

### 4. Digital twin and trust score

For each node  $i$  and slot  $t$  the twin forecasts generation and demand. The generation forecast blends a clear-sky physical envelope  $\varphi_i(t)$  with the node's previous report; the demand forecast is the node's calibrated load profile  $\psi_i(t)$ . Both are evaluated at the hour  $h$  of slot  $t$ :

$$\begin{aligned} f_i^g(t) &= \beta \varphi_i(t) + (1 - \beta) r_i^g(t-1), \\ f_i^d(t) &= \psi_i(t) \end{aligned} \quad (1)$$

where  $\varphi_i(t) = 0.85 \cdot C_i \cdot \max(0, \sin(\pi(h-6)/13))$  for hours  $6 \leq h \leq 19$  and zero otherwise,  $C_i$  is the registered PV capacity in kW,  $r_i^g$  and  $r_i^d$  are the reported generation and demand in kW,  $\psi_i(t) = D_i (1 + 0.6 e^{-(h-19)^2/5}) (1 + 0.4 e^{-(h-14)^2/8})$  is an evening- and midday-peaked profile around the node's mean demand  $D_i$ , and  $\beta = 0.65$ .

In the first slot, when no report exists,  $f_i^g = \varphi_i$ . Because the blend uses the node's reported history and never the true value, a persistent liar can pull 35% of its forecast towards its lie but cannot move the physical part. The twin has no irradiance data; cloud cover therefore appears as forecast error for honest nodes as well (Section 9.2). The per-slot deviation  $\delta_i(t)$  is a weighted relative error, and the anomaly  $a_i(t) \in (0, 1)$  passes it through a logistic dead-band so that ordinary measurement noise is not counted:

$$\begin{aligned} \delta_i(t) &= 0.55 \frac{|r_i^g(t) - f_i^g(t)|}{\max\{f_i^g(t), 0.3 C_i + 0.3\}} + 0.45 \frac{|r_i^d(t) - f_i^d(t)|}{\max\{f_i^d(t), 0.3\}}, \\ a_i(t) &= \frac{1}{1 + \exp[-\kappa(\delta_i(t) - \delta_0)]} \end{aligned} \quad (2)$$

with dead-band centre  $\delta_0 = 0.35$  and slope  $\kappa = 6$ . The floors in the denominators (in kW) keep  $\delta$  finite at night and for consumers with no PV. The Twin-Trust Score is an exponentially weighted moving average (EWMA) of  $1 - a$ , in the sense of the EWMA control chart [25]:

$$\begin{aligned} u_i(t) &= (1 - \eta) \text{TTS}_i(t-1) + \eta(1 - a_i(t)), \\ \text{TTS}_i(t) &= \min\{1, \max\{0.01, u_i(t)\}\}, \\ \text{TTS}_i(0) &= 1 \end{aligned} \quad (3)$$

where  $u_i(t)$  is the unclipped update and  $\eta = 0.22$ . The score is dimensionless and bounded to  $[0.01, 1]$ . The lower bound keeps  $\text{TTS}_i$  strictly positive, so a node that has lost trust can recover by reporting truthfully; since  $a_i < 1$ , the bound is reached only after a long run of large anomalies. Honest nodes do not sit at 1: forecast error from weather keeps their steady-state TTS at a mean of 0.71 (interquartile range 0.66–0.78). All symbols and default values are listed in Table 2.

## 5. PoET<sup>2</sup>S consensus

### 5.1 Validator weight

The selection weight of node  $i$  combines normalized stake, the twin-verified energy term, and the trust score:

$$\begin{aligned} W_i &= \left( \omega_s \frac{s_i}{\bar{s}} + \omega_e (e_i + 1) \right) \text{TTS}_i^\gamma, \\ e_i(t) &= (1 - \lambda_e) e_i(t-1) + \lambda_e \frac{\hat{g}_i(t)}{\bar{c}}, \\ \hat{g}_i(t) &= \min\{r_i^g(t), 1.2 \varphi_i(t)\} \end{aligned} \quad (4)$$

Here  $s_i$  is the stake of node  $i$  and  $\bar{s}$  the mean stake;  $e_i$  is a dimensionless EWMA ( $\lambda_e = 0.2$ ,  $e_i(0) = 0$ ) of the verified generation  $\hat{g}_i(t)$  in kW, normalized by the mean PV capacity  $\bar{C} = 6.5$  kW, so  $e_i$  is 0 for a consumer and of order 1 for a prosumer around midday. The cap 1.2  $\varphi_i$  in  $\hat{g}_i$  is the twin's verification: generation above the physical envelope plus a 20% clear-sky allowance does not count. The term  $\omega_e(e_i + 1)$  gives every node a base of  $\omega_e$ ; Section 8.10 tests whether removing this per-identity constant helps against identity splitting. The weights are  $\omega_s = 0.35$  and  $\omega_e = 0.45$ , and the part in brackets is called the base weight.

**Table 2.** Notation and default parameters

| Symbol               | Meaning                                                  | Value / unit                               |
|----------------------|----------------------------------------------------------|--------------------------------------------|
| $r_i^g, r_i^d$       | Reported generation, demand                              | kW                                         |
| $\phi_i, \psi_i$     | Physical generation envelope, load profile               | kW                                         |
| $C_i, \bar{C}$       | PV capacity (0 for consumers), mean capacity             | Max (3, $N(6.5, 2.8)$ ) kW [26], 6.5 kW    |
| $D_i, \text{CoV}$    | Mean household demand, per-slot coefficient of variation | Max (0.3, $N(1.3, 0.4)$ ) kW, 0.35 [26,27] |
| $\beta$              | Physics / report blend                                   | 0.65                                       |
| $\delta_0, \kappa$   | Dead-band centre, logistic slope                         | 0.35, 6                                    |
| $\eta$               | Trust EWMA rate                                          | 0.22                                       |
| $s_i$                | Stake                                                    | log-normal(3.0, 0.6)                       |
| $u_i$                | Unclipped trust update, Eq. (3)                          | dimensionless                              |
| $e_i, \lambda_e$     | Verified energy term, its EWMA rate                      | dimensionless, 0.2                         |
| $\hat{g}_i$          | Twin-verified generation, Eq. (4)                        | kW                                         |
| $\omega_s, \omega_e$ | Stake and energy weights                                 | 0.35, 0.45                                 |
| $\gamma$             | Trust exponent                                           | 2.5                                        |
| $K, q$               | Committee size, quorum                                   | 21, $\lceil 2K/3 \rceil = 14$              |
| $\tau$               | Detection threshold                                      | 0.35                                       |

The exponent  $\gamma = 2.5$  sets how strongly trust changes the weight. A node at  $TTS = 0.5$  keeps  $0.5^{2.5} \approx 0.18$  of the weight it would have at full trust, against 0.5 under the linear rule. A node whose anomaly never crosses a hard threshold is therefore still moved down the ranking. Section 8.4 shows how  $W_i$  evolves for a single node as its lie grows.

## 5.2 Committee, finalization and ranking consistency

In each slot, the  $K = 21$  nodes with the highest  $W_i$  form the committee, and the proposer is drawn from the committee with probability proportional to  $W_i$ . The committee runs a three-phase commit (pre-prepare, prepare, commit) in the manner of PBFT [13]. Every quorum is counted at each receiving replica: a replica sends its commit after it has received  $q = \lceil 2K/3 \rceil = 14$  prepares, commits locally after 14 commits, and the block is final when 14 members have committed locally. Because  $K$  is fixed, the message count per block does not depend on  $N$ .

PoET<sup>2</sup>S can also use an optional two-step fast path. A committee of  $K \geq 5f + 1$  members can decide in two message steps while at most  $f$  of them are Byzantine [28,29]. With  $K = 21$ , this allows  $f = 4$ . In the fast path, the proposer multicasts its proposal, every honest member multicasts an accept, and a member commits locally once it holds  $\lceil (K + 3f + 1)/2 \rceil = 17$  accepts, the learning quorum of Fast Byzantine (FaB) Paxos [28]. As before, the block is final when 14 members have committed locally. In the simulator, Byzantine members withhold their accepts, whereas in the three-phase commit they vote, as in all other experiments of this paper. Section 8.2 also reports the three-phase commit with withholding. When fewer than 17 honest members remain, the fast quorum cannot form; the members time out after 50 ms and run the three-phase commit. The fast path changes only the commit step: committee selection, the proposer draw, and detection are the same, so every result other than latency and message count is unchanged. Section 8.2 measures the latency gain.

All nodes must agree on the committee. Since the  $TTS$  and  $e_i$  values after block  $b$  are part of block  $b$ , every node can compute the same top- $K$  list for slot  $b + 1$  from the committed state; ties are broken by node identifier, and the proposer draw can be seeded with the hash of block  $b$ , or with a verifiable random function as in Algorand [30]. Two honest nodes can disagree about the committee only if they disagree about block  $b$ , which the commit rule excludes while fewer than one third of committee members are faulty.

Ranking all  $N$  nodes costs  $O(N \log N)$  operations per slot and is done off the latency-critical path. The simulator uses a global view of this state; the cost of disseminating it is part of the block and is not modelled separately.

## 5.3 Sybil considerations

Because  $W_i$  scales with  $TTS_i^\gamma$ , a node at trust  $TTS_s$  needs

$$F = \left( \frac{1}{TTS_s} \right)^\gamma \quad (5)$$

times the base weight of a node at full trust to reach the same selection weight; at  $TTS_s = 0.5$  and  $\gamma = 2.5$ ,  $F \approx 5.66$ . This factor compares two nodes with the same base weight only. It is not a Sybil bound. It says nothing about an attacker that splits its stake over many new identities, each of which starts at  $TTS = 1$  under Eq. (3) while honest nodes sit near 0.71, and each of which receives the per-identity constant  $\omega_e$ . Section 8.10 measures this case and tests three countermeasures: a newcomer trust ramp, a base weight without the per-identity constant, with energy offset 0 instead of 1 in Eq. (4), and committee selection by weighted sortition.

## 6. Baselines and design points

### 6.1 Classical baselines and their detectors

The seven classical baselines are Proof of Work (PoW), Proof of Stake (PoS), PBFT, Delegated Proof of Stake (DPoS), and twin-augmented versions of the last three (PoS+Twin, PBFT+Twin, DPoS+Twin), implemented from their textbook descriptions [12,13]. PoS elects a stake-weighted proposer and a stake-weighted committee of 16 attestors (11 attestations commit). DPoS rotates the proposer among the seven highest-stake delegates (four acknowledgments commit). PBFT runs among the nodes registered as validators; each node takes this role with probability 0.15, so  $V$  is about 15% of  $N$  and varies with the seed ( $V = 85$  at  $N = 500$  for seed 42). PoW uses an exponential block-finding time with a mean of 8 s and uniform hash power.

Detection is evaluated each slot on an audit sample of 150 nodes drawn with the seed's random generator, so all schemes audit the same nodes. Table 3 gives each detector. The classical detectors compare a node's report with the median report in the same slot from the audit sample; they use a single slot, keep no history, and differ only in their thresholds. PoW validates hashes, not meter data, so it has no content check and never flags a node; its 0% detection rate is a property of the protocol, not a tuning choice.

**Table 3.** Detector used by each scheme.  $\text{Dev}_g, \text{Dev}_d = |r - m| / \max(m, 0.5)$  for the audit-sample median  $m$  of generation and demand (absolute difference when  $m \leq 0.1$  kW); window = one slot.

| Scheme                      | Statistic                                   | Flag if                                                     |
|-----------------------------|---------------------------------------------|-------------------------------------------------------------|
| PoW                         | none                                        | never                                                       |
| PoS / DPoS / PBFT           | $\text{Dev}_g, \text{Dev}_d$ vs peer median | either $> 0.60 / 0.50 / 0.40$                               |
| +Twin variants              | baseline flag OR twin anomaly               | baseline flag or $a > 0.5$                                  |
| Veto                        | twin anomaly, current slot                  | $a > 0.5$                                                   |
| Additive                    | TTS and anomaly                             | $\text{TTS} < 0.4$ or $a > 0.6$                             |
| Linear, PoET <sup>2</sup> S | TTS and anomaly                             | $\text{TTS} < 1 - \tau$ or $a > 1 - \tau$ ( $\tau = 0.35$ ) |
| PoET <sup>2</sup> S hybrid  | TTS and anomaly                             | $\text{TTS} < 1 - \tau$ or $a > 0.5$                        |

**Table 4.** Messages per block, order in  $N$  and measured count ( $N = 500$  unless stated).

| Scheme                                      | Messages per block                      | Order in $N$ | Measured                                                                |
|---------------------------------------------|-----------------------------------------|--------------|-------------------------------------------------------------------------|
| PoW                                         | $N - 1$                                 | $O(N)$       | 499                                                                     |
| PoS                                         | $2C$ ( $C = 16$ )                       | $O(1)$       | 32                                                                      |
| DPoS                                        | $2D$ ( $D = 6$ non-proposing delegates) | $O(1)$       | 12                                                                      |
| PBFT                                        | $2V(V - 1)$                             | $O(N^2)$     | $11,313 \pm 2,509$ ( $N = 500$ );<br>$172,284$ ( $N = 2,000, V = 294$ ) |
| Veto, Additive, Linear, PoET <sup>2</sup> S | $2K(K - 1)$                             | $O(1)$       | 840                                                                     |
| PoET <sup>2</sup> S fast path               | $(K - 1)(1 + h)$ , $h$ honest members   | $O(1)$       | $\leq 440$                                                              |

The +Twin detectors are a pure logical OR of the baseline flag and the twin flag  $a_i > 0.5$ . In the classical baselines, a flag is recorded for scoring but does not remove the node from validation, as in the side-channel twin used in the literature reviewed in Section 2.2. Table 4 gives the message count per block. The committee schemes send  $2K(K - 1) = 840$  messages regardless of the value of  $N$ , while PBFT follows  $2V(V - 1)$ .

## 6.2 Twin-based design points

The three design points of Table 1 reuse the PoET<sup>2</sup>S committee size, commit protocol, message pattern, twin, and trust update of Eq. (3). The veto excludes, for the current slot, every node whose instantaneous anomaly exceeds 0.5 and fills the committee with the top  $K$  of the remaining nodes by base weight; trust history plays no part. The additive design point ranks nodes with  $\text{TTS} \geq 0.4$  by  $0.5 \cdot e_i / e_{\max} + 0.5 \cdot \text{TTS}_i$ , where  $e_{\max}$  is the largest  $e_i$  in the network in that slot, and ignores stake. The linear design point uses the base weight times  $\text{TTS}_i$ . Because the linear design point and PoET<sup>2</sup>S also share a detector, their detection results are identical, and any difference in committee composition is caused by  $\gamma$ .

## 7. Experimental setup

### 7.1 Simulator

All results come from a discrete-event simulator written in Python. A priority-queue scheduler advances simulated time, and a network module delivers explicit messages, each with a source, destination, type, and size in bytes. Every protocol is a state machine that sends messages and waits for them, so latency is the simulated time from the start of a round to finality, including propagation, uplink queueing, and signing and verification costs. We wrote a dedicated simulator rather than using ns-3 [31] or OMNeT++ [32] so that the consensus state machines and the grid model share one event loop. The trade-off is that the network model is our own; we validate it in Section 7.3, and we state its limits in Sections 9.2 and 9.3.

### 7.2 Network model

The overlay is a complete graph. Each message draws a round-trip time from a Pareto distribution ( $\alpha = 2$ ,  $x_m = 10$  ms), and the one-way delay is half of it; heavy-tailed delays of this kind are typical of wide-area traffic [33]. Delays are independent across messages. Each node has a 50 Mbit/s uplink modeled as a first-in, first-out queue, so a node that sends many messages in a burst waits for its own transmissions. Before sending a vote or proposal, a node spends 0.10 ms signing it, in line with ed25519 benchmarks [34], plus 0.05 ms verifying the message that triggered it, and every delivery adds 0.1 ms of processing at the receiver. These CPU costs are fixed delays and are not queued at a node. Proposals carry a 1 kB header and votes 256 B; transaction bodies are assumed to be disseminated separately, as in mempool-based designs such as Narwhal [35], and Section 8.2 repeats the latency measurement with full 30 kB block bodies.

### 7.3 Validating the network model

Two checks compare the simulator with independent results. The first is PBFT message complexity: Castro and Liskov's protocol sends  $V - 1$  pre-prepares,  $(V - 1)^2$  prepares from the backups, and  $V(V - 1)$  commits,  $2V(V - 1)$  messages in total [13]. Running our PBFT from  $N = 60$  to  $N = 2,000$  ( $V = 5$  to 294) gives measured counts equal to  $2V(V - 1)$  at every size, with an empirical log-log slope of 2.05 (Figure 1). The second is the delay sampler: 200,000 one-way samples have a mean of 9.978 ms and a median of 7.067 ms, within 0.2% of the analytical values of 10.0 ms and 7.071 ms. These checks confirm the message pattern and the delay distribution; they do not validate absolute end-to-end latency against a packet-level simulator.

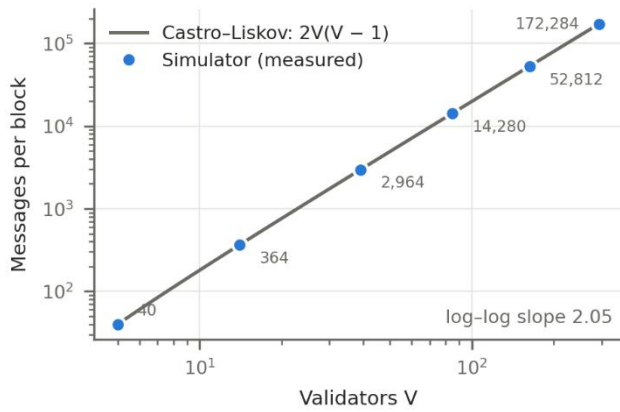


**Figure 1.** Network-model validation: measured PBFT messages per block (points) against the Castro-Liskov count  $2V(V - 1)$  (line). The log-log slope is 2.05.

#### 7.4 Workload, attacks and metrics

Generation and demand in each slot are sampled from profiles calibrated to Pecan Street Dataport [26] and UMass Smart\* [27]: a sinusoidal clear-sky profile scaled by a weather multiplier (clear with probability 0.75, multiplier  $N(1.0, 0.10)$ ; otherwise  $U(0.25, 0.65)$ ), drawn independently for each node and slot, and an evening- and midday-peaked load profile with a coefficient of variation of 0.35. The traces are time-compressed: each 12-s consensus slot samples the next hour of the daily profile, so a run of 60 slots covers two and a half days of solar and load variation. Table 5 lists the settings. Each run has 60 slots at  $N = 500$ . The main experiment, the adaptive and hybrid studies, the Byzantine sweep, the parameter sweep, and the operating-point sweep use 30 seeds (42–71); the comparison with the linear rule is repeated on 100 seeds (42–141), the Sybil study uses 10 seeds, the latency breakdown 5 seeds, and the scaling study 3 seeds. The single-node trajectory of Section 8.4 runs for 72 slots.

**Table 5.** Simulator settings

| Parameter          | Value                                                   |
|--------------------|---------------------------------------------------------|
| Network size $N$   | 500 (100–4,000 in Section 8.9)                          |
| Byzantine fraction | 15% (10–45% in Section 8.5)                             |
| Link delay         | Pareto round trip, $\alpha = 2$ , $x_m = 10$ ms, halved |
| Uplink bandwidth   | 50 Mbit/s per node, FIFO queue                          |
| Message sizes      | proposal header 1 kB, vote 256 B                        |
| Block content      | 120 transactions of 250 B (30 kB body)                  |
| Slot length        | 12 s (one hour of the daily profile per slot)           |
| Slots per run      | 60                                                      |
| Seeds              | 30 (42–71); 100 (42–141) in Section 8.3; 10 for Sybil   |
| Audit sample       | 150 nodes per slot                                      |

The metrics are: commit latency (ms) per block; attackers caught, the share of malicious nodes flagged at least once; false positives, the number of honest node-slots flagged; false positives per caught attacker (FP/caught); committee infiltration, the malicious share of each block's validating set averaged over blocks (the proposer for PoW, proposer and attesters for PoS, delegates for DPoS, all validators for PBFT and the committee otherwise); proposer capture, the share of blocks with a malicious proposer; safety violations, the share

of blocks whose validating set is at least one-third Byzantine, so that the fault bound of BFT commit [13] no longer holds; and messages per block. Throughput in transactions per second is not used as a metric. Dividing block size by commit latency yields only a rescaled latency, not a sustained rate: with one block of 120 transactions per 12-s slot, any scheme that finalizes within the slot sustains 10 transactions per second. Throughput is therefore not reported.

#### 7.5 Statistics

All schemes run on the same seeds, and an instrumented test in the code base confirms that a given seed produces identical attack realizations (the same malicious nodes, attack families and reported values) under every scheme; the comparisons are therefore paired. Pairwise differences are tested with the two-sided Wilcoxon signed-rank test over the 30 seed pairs [36]. Differences are first rounded to  $10^{-9}$ , so that values equal up to floating-point error count as ties. The test uses its exact null distribution when there are no ties or zero differences and the normal approximation otherwise. We chose it over the paired t-test because several differences are bounded at zero and skewed (infiltration and profit have many near-zero seeds); the paired t-test is also reported in the released statistics and leads to the same conclusions. Across the 35 tests of the main study, p-values are adjusted with the Holm procedure [37]; the 20 tests of the 100-seed and safety-violation analyses form a second Holm family. Tables give mean  $\pm$  standard deviation (SD) over seeds; 95% confidence intervals (CIs) for mean differences are percentile bootstrap intervals with 5,000 resamples of seed pairs [38].

### 8. Results

#### 8.1 PoET<sup>2</sup>S against the classical baselines

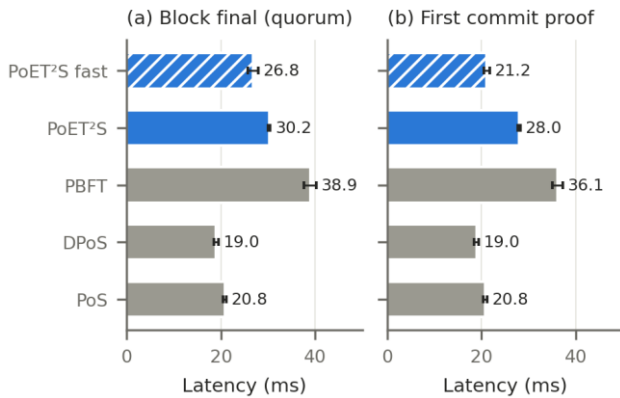
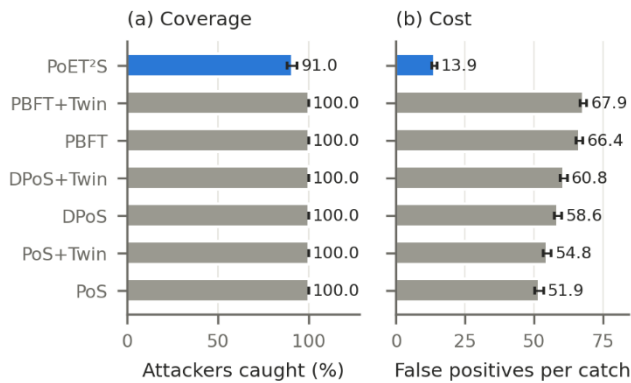
Table 6 reports the classical baselines and PoET<sup>2</sup>S at  $N = 500$ . On latency, PoET<sup>2</sup>S is not the fastest. Its commit latency of 30.2 ms is 9.4 ms higher than PoS (95% CI 9.3–9.6 ms) and 11.3 ms higher than DPoS, and 8.7 ms lower than PBFT (Figure 2); every seed agrees on the direction, and all six comparisons have Holm-adjusted  $p = 6.5 \times 10^{-8}$ . PoS and DPoS are faster because they finish in two message hops with small committees, whereas PoET<sup>2</sup>S needs three sequential phases with a quorum of 14 (Section 8.2). PoW is included for completeness; its latency is set by its 8-s mean block time. With the optional two-step fast path of Section 5.2, PoET<sup>2</sup>S commits in  $26.8 \pm 1.1$  ms, 3.4 ms faster than its three-phase commit on all 30 seeds, but still 6.0 ms slower than PoS and 7.8 ms slower than DPoS (Figure 2a).

On detection, the baselines catch every attacker, and PoET<sup>2</sup>S catches  $91.0\% \pm 2.7\%$  (Figure 3a). The baselines pay for full coverage with false alarms: their single-slot peer-median detectors flag between 3,889 and 5,091 honest node-slots per run, 51.9 to 67.9 per caught attacker, against 13.9 for PoET<sup>2</sup>S (Figure 3b). PoET<sup>2</sup>S has lower FP/caught than every baseline on all 30 seeds (Holm-adjusted  $p = 6.5 \times 10^{-8}$ ). Adding the twin as an OR-flag raises the false-positive count of each baseline because the twin flags some honest nodes that the peer median does not.

The largest difference is in committee infiltration (Figure 4). In every classical baseline, the malicious share of the validating set stays close to the 15% population share, from 13.3% for DPoS to 15.1% for PBFT and 15.2% for PoS, because their selection ignores the twin and flags do not remove nodes. PoET<sup>2</sup>S keeps it at  $2.6\% \pm 1.9\%$ . The large SD for DPoS (11.8%) reflects its seven fixed delegates: a seed either places one or more attackers among them or none.

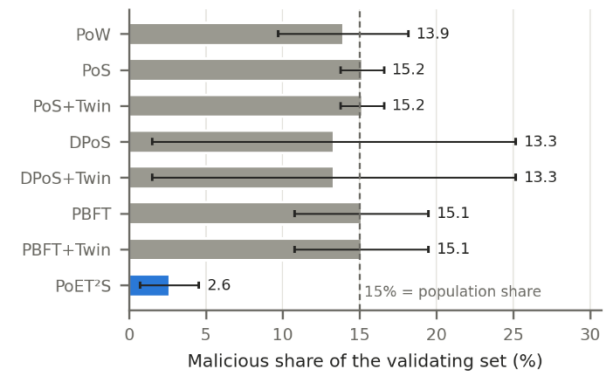
**Table 6.** Mean  $\pm$  SD over 30 seeds at N = 500. Infil. = committee infiltration; Prop. = proposer capture

| Scheme              | Latency (ms)    | Caught (%)     | FP/caught      | Infil. (%)      | Prop. (%)       |
|---------------------|-----------------|----------------|----------------|-----------------|-----------------|
| PoW                 | 8,033 $\pm$ 988 | 0.0            | —              | 13.9 $\pm$ 4.2  | 13.9 $\pm$ 4.2  |
| PoS                 | 20.8 $\pm$ 0.4  | 100.0          | 51.9 $\pm$ 1.6 | 15.2 $\pm$ 1.4  | 14.4 $\pm$ 4.5  |
| PoS+Twin            | 20.8 $\pm$ 0.4  | 100.0          | 54.8 $\pm$ 1.4 | 15.2 $\pm$ 1.4  | 14.4 $\pm$ 4.5  |
| DPoS                | 19.0 $\pm$ 0.5  | 100.0          | 58.6 $\pm$ 1.3 | 13.3 $\pm$ 11.8 | 13.3 $\pm$ 12.1 |
| DPoS+Twin           | 19.0 $\pm$ 0.5  | 100.0          | 60.8 $\pm$ 1.2 | 13.3 $\pm$ 11.8 | 13.3 $\pm$ 12.1 |
| PBFT                | 38.9 $\pm$ 1.3  | 100.0          | 66.4 $\pm$ 1.2 | 15.1 $\pm$ 4.4  | 15.3 $\pm$ 4.6  |
| PBFT+Twin           | 38.9 $\pm$ 1.3  | 100.0          | 67.9 $\pm$ 1.2 | 15.1 $\pm$ 4.4  | 15.3 $\pm$ 4.6  |
| PoET <sup>2</sup> S | 30.2 $\pm$ 0.2  | 91.0 $\pm$ 2.7 | 13.9 $\pm$ 0.9 | 2.6 $\pm$ 1.9   | 2.7 $\pm$ 2.9   |

**Figure 2.** Commit latency at N = 500 (mean  $\pm$  SD, 30 seeds): (a) time until the block is final (14 committee members, or the PBFT quorum, have committed locally); (b) time until the first node holds a commit proof. PoS and DPoS decide at the proposer, so both panels show the same value for them. PoET<sup>2</sup>S fast is the optional two-step path of Section 5.2. PoW (8,033 ms) is omitted.**Figure 3.** Detection at N = 500 (mean  $\pm$  SD, 30 seeds): (a) share of attackers caught; (b) false positives per caught attacker

## 8.2 Latency accounting

Table 7 decomposes the latency by removing one cost at a time. For PoET<sup>2</sup>S at N = 500, propagation dominates: without it, latency falls from 30.4 ms to 4.0 ms; removing signing and verification saves 0.7 ms, and removing uplink queueing saves 2.7 ms. The phase timestamps show how the time accumulates. The proposal reaches a quorum of members after 10.6 ms, the prepare quorum is complete after 21.0 ms, and finality follows a third hop later; each hop waits for the 14th-fastest of 20 heavy-tailed delays. PoS and DPoS need two hops.

**Figure 4.** Malicious share of the validating set (mean  $\pm$  SD, 30 seeds). The dashed line is the 15% population share

PBFT grows from 39.7 ms at N = 500 to 77.8 ms at N = 2,000 because of the primary's uplink. With V = 294 validators, the primary sends 293 pre-prepares sequentially; at 50 Mbit/s, each 1 kB message takes 0.16 ms to transmit, so the last one leaves about 48 ms after the first. The proposal quorum is reached only after 42.3 ms, and removing queueing brings the total back to 28.0 ms, close to the 28.2 ms at N = 500. With full 30 kB bodies, serialization cost grows for every scheme that pushes the body to many receivers, most so in PBFT (293.9 ms). All latencies are simulator-internal and should be read as relative comparisons between schemes under one network model.

Part of the gap to PoS and DPoS comes from what is measured. For PoS and DPoS, a block counts as committed when the proposer holds its quorum of attestations, whereas for PBFT and PoET<sup>2</sup>S we wait until 14 members, or the PBFT quorum, have committed locally. Figure 2b applies the PoS definition to every scheme: the time at which the first node holds a commit proof. On this measure, the three-phase PoET<sup>2</sup>S takes 28.0 ms and the fast path 21.2  $\pm$  0.6 ms, against 20.8 ms for PoS (PoS lower on 21 of 30 seeds; Wilcoxon p = 0.020), 19.0 ms for DPoS (lower than the fast path on all 30 seeds) and 36.1 ms for PBFT. The fast path also halves the message count, to at most 440 per block, but it has two costs. It waits for 17 of 20 heavy-tailed accepts, so its tail is longer: the 95th and 99th percentiles of block latency are 33.8 and 39.8 ms, against 33.2 and 35.2 ms for the three-phase commit. With full 30 kB block bodies, it is slower (103.7 against 95.8 ms), because the proposer's uplink then dominates and the fast quorum is larger. If Byzantine members also withhold their votes in the three-phase commit, its latency rises only slightly, from 30.2 to 31.0  $\pm$  0.6 ms for PoET<sup>2</sup>S, and to 34.6  $\pm$  2.8 ms for the veto.

**Table 7.** Latency breakdown (ms; 5 seeds  $\times$  20 blocks at  $N = 500$ , 3 seeds  $\times$  5 blocks at  $N = 2,000$ ). Each column removes one cost. tprop and tprep are the times at which a quorum has received the proposal and has prepared. Last column: full 30 kB block body instead of a 1 kB header.

| Scheme (N)                       | Full | No crypto | No queue | No prop. | t <sub>prop</sub> | t <sub>prep</sub> | 30 kB |
|----------------------------------|------|-----------|----------|----------|-------------------|-------------------|-------|
| PoS (500)                        | 20.8 | 20.3      | 19.3     | 2.3      | —                 | —                 | 72.7  |
| DPOS (500)                       | 18.5 | 18.0      | 17.9     | 1.1      | —                 | —                 | 37.8  |
| PBFT (500)                       | 39.7 | 39.0      | 28.2     | 14.5     | 17.4              | 28.9              | 293.9 |
| PoET <sup>2</sup> S (500)        | 30.4 | 29.7      | 27.7     | 4.0      | 10.6              | 21.0              | 95.8  |
| PoET <sup>2</sup> S fast (500)   | 26.8 | 26.4      | 24.5     | 3.8      | —                 | —                 | 103.7 |
| PBFT (2,000)                     | 77.8 | 77.2      | 28.0     | 53.4     | 42.3              | 60.2              | —     |
| PoET <sup>2</sup> S (2,000)      | 30.8 | 30.2      | 28.1     | 4.0      | 11.3              | 21.3              | —     |
| PoET <sup>2</sup> S fast (2,000) | 27.1 | 26.6      | 25.3     | 3.8      | —                 | —                 | —     |

**Table 8.** Twin-based schemes, mean  $\pm$  SD over 30 seeds at  $N = 500$ . All share committee, commit protocol, messages, and latency ( $30.2 \pm 0.2$  ms).

| Scheme              | Caught (%)     | FP/caught      | Infil. (%)     | Prop. (%)      | Profit proxy (kWh) |
|---------------------|----------------|----------------|----------------|----------------|--------------------|
| Veto                | 91.7 $\pm$ 3.1 | 9.7 $\pm$ 0.6  | 10.8 $\pm$ 3.6 | 10.2 $\pm$ 4.6 | 98.4 $\pm$ 46.9    |
| Additive            | 75.0 $\pm$ 4.8 | 6.0 $\pm$ 0.6  | 9.4 $\pm$ 3.5  | 8.7 $\pm$ 5.8  | 163.9 $\pm$ 96.0   |
| Linear              | 91.0 $\pm$ 2.7 | 13.9 $\pm$ 0.9 | 3.8 $\pm$ 2.9  | 4.4 $\pm$ 4.8  | 14.1 $\pm$ 33.6    |
| PoET <sup>2</sup> S | 91.0 $\pm$ 2.7 | 13.9 $\pm$ 0.9 | 2.6 $\pm$ 1.9  | 2.7 $\pm$ 2.9  | 3.2 $\pm$ 13.2     |

The fast path is safe only when a committee has at most four Byzantine members, and committee integrity matters again here. Under PoET<sup>2</sup>S, the fast path decides 100% of blocks at 10–15% Byzantine nodes, 98.8% at 30% and 85.2% at 45%; the remaining blocks fall back to the three-phase commit, and latency rises from 26.8 ms at 15% to 42.9 ms at 45%. Under the veto, the fast path already fails in 3.9% of blocks at 15% and in 46.8% at 30%, and the fallbacks make the veto's fast-path latency (36.1 ms at 15%) higher than its three-phase latency (30.2 ms). Because fallbacks occur exactly when a committee holds more than four Byzantine members, these shares are also the fraction of blocks for which a deployment running the fast path without fallback would lose its safety guarantee. The linear rule is in between (27.8 ms at 15%).

### 8.3 PoET<sup>2</sup>S against the twin-based design points

Table 8 and Figure 5 compare the four twin-based schemes, which share all machinery except the selection rule. Detection is a near-tie between the veto and PoET<sup>2</sup>S: 91.7% against 91.0% of attackers caught, a difference that is not significant (PoET<sup>2</sup>S higher on 10 of 30 seeds, Holm-adjusted  $p = 0.53$ ). The veto has the lower false-positive cost, 9.7 against 13.9 per catch, and the additive design point the lowest, 6.0, but it catches only 75.0% of attackers. The linear design point and PoET<sup>2</sup>S share a detector and have identical detection.

Committee infiltration separates the schemes. The veto admits 10.8% malicious committee members, the additive design point 9.4%, the linear design point 3.8%, and PoET<sup>2</sup>S 2.6%. Relative to the veto, PoET<sup>2</sup>S has lower infiltration on all 30 seeds (76% lower; Holm-adjusted  $p = 2.8 \times 10^{-5}$ ), and relative to the additive rule on 28 of 30 seeds (72% lower; Holm-adjusted  $p = 3.1 \times 10^{-5}$ ). Relative to the linear rule, which differs only in  $\gamma$ , PoET<sup>2</sup>S is lower on 18 seeds, higher on 11, and tied on 1 (31% lower on average; mean difference  $-1.21$  percentage points, 95% CI  $-1.94$  to  $-0.55$ ; Holm-adjusted  $p = 0.032$ ). Figure 6 shows the per-seed pairs, and Table A1 (Appendix A) lists them.

Proposer capture is 10.2%, 8.7%, 4.4%, and 2.7%; the difference between the linear rule and PoET<sup>2</sup>S is not significant at 30 seeds (Holm-adjusted  $p = 0.50$ ). Table 9 lists the tests.

Because the gain over the linear rule isolates  $\gamma$  and was least certain at 30 seeds, we repeated the comparison at 100 seeds (42–141). PoET<sup>2</sup>S has 3.15% infiltration against 4.99% for the linear rule, 37% lower; it is lower on 70 seeds, tied on 4, and higher on 26 (mean difference  $-1.85$  percentage points, 95% CI  $-2.35$  to  $-1.37$ ; Holm-adjusted  $p = 7.2 \times 10^{-9}$ ). With 100 seeds, the difference in proposer capture also becomes significant (3.0% against 5.2%, Holm-adjusted  $p = 3.3 \times 10^{-4}$ ). Against the veto, PoET<sup>2</sup>S is lower on 98 of 100 seeds (3.15% against 11.17%).

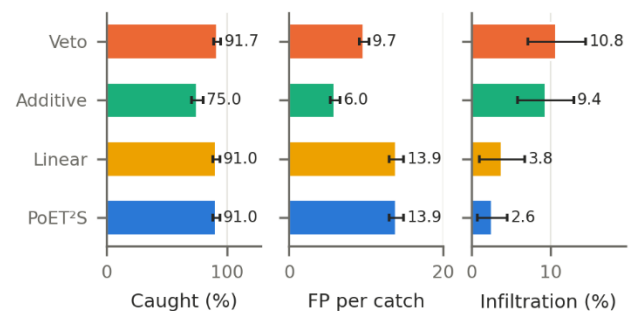


**Figure 5.** Twin-based schemes (mean  $\pm$  SD, 30 seeds): detection coverage, false-positive cost and committee infiltration

### 8.4 Weight trajectory of a single node

Figure 7 follows one prosumer (6.5 kW of PV, 1.3 kW mean demand, stake equal to the network mean) that reports truthfully for 12 slots and then multiplies its generation by a lie factor of 1.1, 1.25, 1.5, or 2. Its TTS falls with the size of the lie (panel a). The verified energy term still rises because the cap at 1.2  $\phi_i$  lets some of the lie through, especially in cloudy slots where the true output is far below the envelope; at a factor of 2, it reaches 1.32 times the honest value (panel b).

**Table 9.** Paired tests, PoET<sup>2</sup>S minus comparator, 30 seeds. Wins = seeds on which PoET<sup>2</sup>S is better. CI = bootstrap 95% interval of the mean difference; pp = percentage points; p = Wilcoxon signed-rank, Holm-adjusted over 35 tests. The hybrid rows compare the hybrid PoET<sup>2</sup>S detector with the veto.

| Comparator       | Metric                | Wins  | Mean diff. [95% CI]     | p (Holm)             |
|------------------|-----------------------|-------|-------------------------|----------------------|
| Veto             | Infiltration (pp)     | 30/30 | -8.15 [-9.55, -6.80]    | $2.8 \times 10^{-5}$ |
| Veto             | Proposer capture (pp) | 30/30 | -7.50 [-9.17, -6.00]    | $2.8 \times 10^{-5}$ |
| Veto             | Caught (pp)           | 10/30 | -0.67 [-1.47, +0.09]    | 0.532                |
| Veto             | FP/caught             | 0/30  | +4.16 [+3.97, +4.37]    | $2.8 \times 10^{-5}$ |
| Additive         | Infiltration (pp)     | 28/30 | -6.76 [-8.15, -5.36]    | $3.1 \times 10^{-5}$ |
| Additive         | Proposer capture (pp) | 23/30 | -6.00 [-8.56, -3.61]    | $3.2 \times 10^{-3}$ |
| Additive         | Caught (pp)           | 30/30 | +16.00 [+14.62, +17.42] | $2.8 \times 10^{-5}$ |
| Additive         | FP/caught             | 0/30  | +7.93 [+7.69, +8.17]    | $2.8 \times 10^{-5}$ |
| Linear           | Infiltration (pp)     | 18/30 | -1.21 [-1.94, -0.55]    | 0.032                |
| Linear           | Proposer capture (pp) | 14/30 | -1.72 [-3.33, -0.33]    | 0.501                |
| Linear           | Caught (pp)           | 0/30  | +0.00 [+0.00, +0.00]    | 1.000                |
| Linear           | FP/caught             | 0/30  | +0.00 [+0.00, +0.00]    | 1.000                |
| Veto (vs hybrid) | Caught (pp)           | 28/30 | +2.27 [+1.87, +2.71]    | $3.2 \times 10^{-5}$ |
| Veto (vs hybrid) | FP/caught             | 0/30  | +7.14 [+6.99, +7.30]    | $6.5 \times 10^{-8}$ |

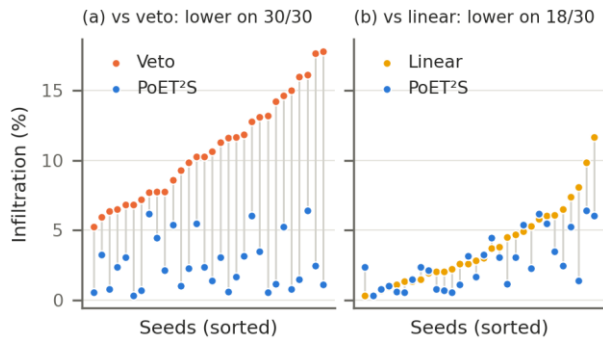


**Figure 6.** Per-seed committee infiltration of PoET<sup>2</sup>S against (a) the veto and (b) the linear rule. Each vertical segment joins the two values for one seed.

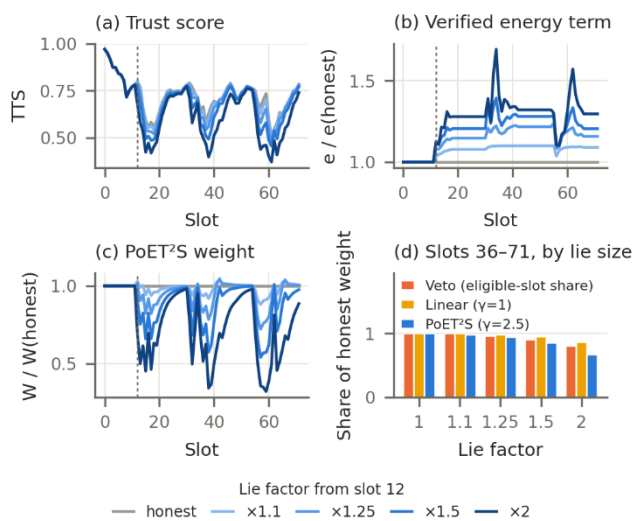


**Figure 7.** One prosuser lying from slot 12 by the factor shown: (a) Twin-Trust Score; (b) verified energy term relative to honest; (c) PoET<sup>2</sup>S weight relative to honest; (d) mean over slots 36–71 of the veto-eligible-slot share, the linear weight, and the PoET<sup>2</sup>S weight, each relative to the honest node.

The PoET<sup>2</sup>S weight relative to the same node reporting honestly is 0.98, 0.94, 0.85, and 0.68 of the honest value, averaged over slots 36–71, against 1.00, 0.99, 0.95, and 0.87 under the linear rule (panels c and d). The veto keeps the node eligible in 1.00, 0.97, 0.91, and 0.81 of the slots in which the honest node is eligible, so on average it penalizes a single liar somewhat less than PoET<sup>2</sup>S, but the difference is far smaller than the committee gap. The committee gap in Section 8.3 comes from how the weight is applied. The veto acts slot by slot: in a slot where the liar is not vetoed, it competes for a top-21 seat at full base weight, including its inflated energy term. PoET<sup>2</sup>S applies its discount in every slot, so a liar with a high base weight is kept below the committee cut-off consistently. Top-K selection from 500 nodes depends on the upper tail of the weight distribution, which is where this difference matters.

## 8.5 Robustness across Byzantine fractions

Figure 8 repeats the comparison for Byzantine fractions from 10% to 45% for all four twin-based schemes (30 seeds each). PoET<sup>2</sup>S has the lowest infiltration at every fraction: 3.2% against 7.2% (veto), 7.0% (additive), and 4.7% (linear) at 10%, and 13.7% against 35.0%, 31.2%, and 18.7% at 45% (Figure 8a). Detection stays a near-tie: the veto catches 91.3–92.8% of attackers, the linear rule and PoET<sup>2</sup>S 89.9–91.2%, and the additive rule 74.5–76.2%.

Figure 8b shows the share of blocks whose committee is at least one-third Byzantine, the point at which BFT commit loses its safety guarantee. PoET<sup>2</sup>S stays below 3% at every fraction (0.33% at 30%, 2.4% at 45%). The veto crosses the bound in 12.3% of blocks at 30% and 64.4% at 45%, the additive rule in 18.3% and 47.1%, and the linear rule in 2.0% and 10.7%. The additive rule already violates the bound in 4.4% of blocks at 20%, because its eligibility floor admits attackers whose trust has not yet fallen. PoET<sup>2</sup>S has fewer violations than the veto at every fraction from 25% upward (Holm-adjusted  $p \leq 0.014$ ) and fewer than the linear rule from 35% upward (Holm-adjusted  $p < 0.03$ ). Below 25%, the differences between the veto, the linear rule, and PoET<sup>2</sup>S are not significant after correction.

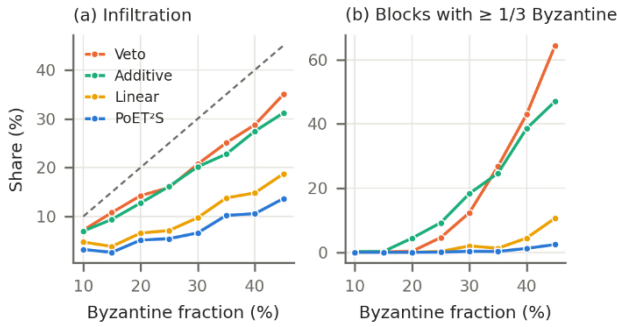


**Figure 8.** Byzantine-fraction sweep (mean over 30 seeds): (a) committee infiltration, with the population share dashed; (b) share of blocks whose committee is at least one-third Byzantine.

### 8.6 Adaptive adversaries

Two adaptive strategies were run on the same 30 seeds. A lie-and-recover attacker watches its own TTS: it inflates its generation 1.6–2.4 times while  $TTS > 0.55$ , lies by 5–25% while  $0.40 < TTS \leq 0.55$ , and reports truthfully below 0.40 to rebuild trust. A stealth attacker inflates by 8–18% in every slot, keeping its anomaly within the honest band and below the veto threshold. Neither strategy defeats either detector: the veto catches 92.0% and 90.9% and PoET<sup>2</sup>S 91.1% and 90.7% of attackers under the two strategies, and none of these differences is significant after Holm correction.

The committee-integrity result is unchanged by either strategy. Under lie-and-recover, infiltration is 2.9% for PoET<sup>2</sup>S against 10.8% for the veto, and under stealth it is 3.4% against 11.1%. PoET<sup>2</sup>S is lower on all 30 seeds in both cases (Holm-adjusted  $p = 6.5 \times 10^{-8}$ ). The profit proxy, the energy over-reported by malicious members of the validating set summed over the run (in kWh, since each slot samples one hour), is 2.8 against 70.0 and 4.2 against 64.9 (Figure 9). The adaptive strategies tested here are fixed rules; an attacker that optimizes its lie schedule against Eq. (3) is not covered.

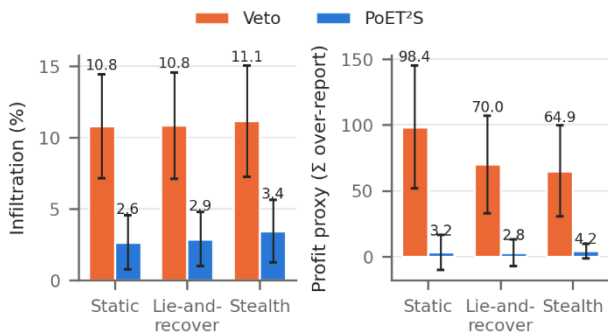


**Figure 9.** Static and adaptive attackers (mean  $\pm$  SD, 30 seeds): committee infiltration and profit proxy for the veto and PoET<sup>2</sup>S.

### 8.7 Parameter sensitivity

Each of the four PoET<sup>2</sup>S parameters was varied with the others at their defaults (30 seeds, Figure 10). The committee size  $K$  changes latency and infiltration but not detection: latency rises from 28.3 ms at  $K = 7$  to 34.9 ms at  $K = 51$ , and infiltration from 2.4% to 3.1%, because a larger committee reaches further down the ranking. The exponent  $\gamma$  changes only infiltration: 3.85% at  $\gamma = 1$ , 3.12% at 1.5, 2.64% at 2.5, 2.66% at 3.5, and 2.79% at 5. The gain flattens beyond  $\gamma \approx 2.5$ , and the small rise at larger  $\gamma$  is well within the seed-to-seed SD (1.9 percentage points at the default).

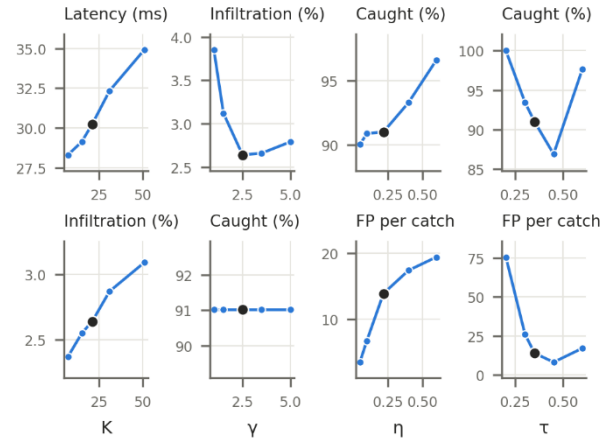


**Figure 10.** Sensitivity of PoET<sup>2</sup>S to  $K$ ,  $\gamma$ ,  $\eta$ , and  $\tau$  (30 seeds). The large dot marks the default. The upper row shows the metric each parameter mainly controls; the lower row shows a second metric.

The trust rate  $\eta$  and the threshold  $\tau$  set the detection trade-off. Raising  $\eta$  from 0.05 to 0.6 raises the catch rate from 90.0% to 96.6% and the false-positive cost from 3.4 to 19.4 per catch, while infiltration stays between 2.6% and 3.4%. The threshold  $\tau$  acts on both arms of the detector,  $TTS < 1 - \tau$  and  $a > 1 - \tau$ , so its effect is not monotonic. From  $\tau = 0.20$  to 0.45, the trust arm becomes less strict, and both the catch rate (100% to 86.9%) and the cost (75.6 to 8.1 per catch) fall. At  $\tau = 0.60$ , the anomaly arm fires at  $a > 0.40$ , which is inside the range of ordinary lies, and both rise again (97.6% at 17.0 per catch). Because  $\tau$  enters detection only, infiltration is 2.6% at every value.  $\eta = 0.10$  gives almost the same coverage as the default (90.9%) at 6.7 false positives per catch; we kept the default used throughout and report this better operating point in Section 8.11.

### 8.8 An optional hybrid detector

The default PoET<sup>2</sup>S detector relies mainly on the smoothed TTS, which reacts to sustained deviation but can miss short spikes; the veto uses only the current slot. Combining a smoothed statistic with a point trigger is common in time-series anomaly detection [39,40]. The hybrid detector flags a node if  $TTS < 1 - \tau$  or  $a > 0.5$ , the veto's own threshold. It changes detection only, so committee selection, infiltration and latency are unchanged. The hybrid catches 94.0% of attackers, more than the veto on 28 of 30 seeds (Holm-adjusted  $p = 3.2 \times 10^{-5}$ ), at a cost of 16.9 false positives per catch, against 13.9 for the default and 9.7 for the veto (Table 10). It is an operating point for deployments that value coverage over precision.

**Table 10.** Default and hybrid PoET<sup>2</sup>S against the veto (mean  $\pm$  SD, 30 seeds).

| Configuration                 | Caught (%)     | FP/caught      | Infil. (%)     | Latency (ms)   |
|-------------------------------|----------------|----------------|----------------|----------------|
| Veto                          | 91.7 $\pm$ 3.1 | 9.7 $\pm$ 0.6  | 10.8 $\pm$ 3.6 | 30.2 $\pm$ 0.2 |
| PoET <sup>2</sup> S (default) | 91.0 $\pm$ 2.7 | 13.9 $\pm$ 0.9 | 2.6 $\pm$ 1.9  | 30.2 $\pm$ 0.2 |
| PoET <sup>2</sup> S (hybrid)  | 94.0 $\pm$ 2.6 | 16.9 $\pm$ 0.9 | 2.6 $\pm$ 1.9  | 30.2 $\pm$ 0.2 |

### 8.9 Scaling with network size

Figure 11 shows latency and messages per block from  $N = 100$  to  $N = 4,000$  (3 seeds, 5 blocks per seed; 3 blocks at  $N = 4,000$ ). PoET<sup>2</sup>S stays between 30.7 and 30.9 ms with 840 messages per block. PoS stays near 20–21 ms and DPoS near 19.5–21 ms. PBFT grows from 30.5 ms at  $N = 100$  to 127.2 ms at  $N = 4,000$ , with about 692,000 messages per block at the largest size, because its validator set grows with  $N$  (Section 8.2).

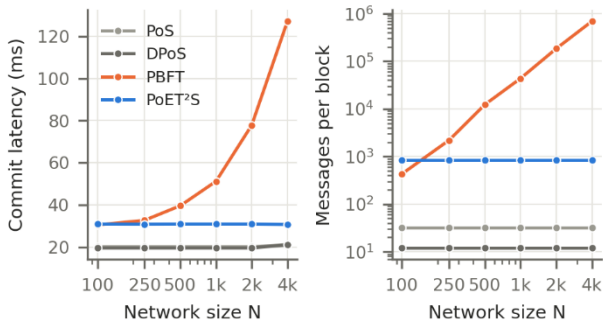


**Figure 11.** Commit latency (left) and messages per block (right, log scale) against network size.

### 8.10 Sybil identity splitting

Section 5.3 notes that Eq. (5) is not a Sybil bound, so identity splitting is tested directly. An attacker holds 1%, 2% or 5% of total stake and the same share of total PV capacity, and splits both evenly over  $m \in \{1, 2, 5, 10, 20, 50\}$  new identities; each identity's energy term is that of an honest producer of the same capacity. The identities join a network whose 500 nodes have run for 60 slots and are at their steady-state trust. The Sybil identities start at TTS = 1.0, 0.8, 0.6, or 0.4. We count the committee seats they win out of 21 (10 seeds). Table 11 shows the 5% budget.

The outcome depends on the starting trust of new identities. When they start at full trust, PoET<sup>2</sup>S is weaker than the linear rule: 20 identities at TTS = 1 win 19.7 of 21 seats, against 5.2 under the linear rule. The reason is that honest nodes sit near TTS = 0.71, so the exponent discounts every honest node by about  $0.71^{2.5} \approx 0.42$ , while a fresh identity is not discounted at all; in addition, each identity receives the constant  $\omega_e$ , so splitting adds base weight. Even with a 1% budget, 10 identities at full trust win 6.7 seats. When new identities start at TTS = 0.4, PoET<sup>2</sup>S gives a split attacker nothing for any  $m \geq 2$ , whereas the linear rule still gives 5 seats at  $m = 5$ .

We then tested three countermeasures (Table 12) and a stake bond. A newcomer trust ramp, in which every new identity starts at TTS = 0.4 and gains trust only by reporting truthfully, removes the fresh-identity advantage. It does not stop an attacker that keeps its identities honest through the probation period: once they reach the median honest TTS, a 5% budget split over 10 identities wins 8.6 seats under PoET<sup>2</sup>S and 10.0 under the linear rule. Removing the per-identity constant, i.e., setting the energy offset in Eq. (4) to 0, does not help (9.8 seats), and a stake bond at the 10th percentile of honest stake, not shown in Table 12, binds only for small budgets (at 1%, splits beyond 10 identities become infeasible). The cause is deterministic top-K selection: any identity whose weight exceeds the 21st-highest honest weight wins a seat, whatever the trust rule. Drawing the committee by weighted sortition,  $K$  members sampled without replacement with probability proportional to  $W_i$  as a verifiable lottery would [30] bound the attacker to at most 2.4 seats on average, and at least 7 seats in no more than 1.3% of draws for any seed, split, and starting trust tested. Sortition, however, costs committee integrity. Rerunning the main experiment (30 seeds) with offset 0 and top-K selection leaves PoET<sup>2</sup>S almost unchanged (infiltration  $2.9\% \pm 2.2\%$  against  $2.6\% \pm 1.9\%$ ; detection identical).

**Table 11.** Committee seats (of 21) won by Sybil identities with a 5% stake budget split over  $m$  identities, by starting TTS (mean over 10 seeds). Seven or more seats break the one-third fault bound.

| $m$ | PoET <sup>2</sup> S TTS 1.0 | PoET <sup>2</sup> S TTS 0.6 | PoET <sup>2</sup> S TTS 0.4 | Linear TTS 1.0 | Linear TTS 0.6 | Linear TTS 0.4 |
|-----|-----------------------------|-----------------------------|-----------------------------|----------------|----------------|----------------|
| 1   | 1.0                         | 1.0                         | 1.0                         | 1.0            | 1.0            | 1.0            |
| 2   | 2.0                         | 2.0                         | 0.0                         | 2.0            | 2.0            | 2.0            |
| 5   | 5.0                         | 5.0                         | 0.0                         | 5.0            | 5.0            | 5.0            |
| 10  | 10.0                        | 0.0                         | 0.0                         | 10.0           | 0.0            | 0.0            |
| 20  | 19.7                        | 0.0                         | 0.0                         | 5.2            | 0.0            | 0.0            |
| 50  | 6.8                         | 0.0                         | 0.0                         | 0.0            | 0.0            | 0.0            |

**Table 12.** Seats (of 21) won by Sybil identities with a 5% stake budget under PoET<sup>2</sup>S ( $\gamma = 2.5$ ), 10 seeds. Ramp: identities start at TTS 0.4. Honest: identities have passed probation and hold the median honest TTS. Offset 0: base weight without the per-identity constant. Sortition: expected seats under weighted sortition with offset 0.

| $m$ | Top-K, TTS 1.0 | Top-K, ramp | Top-K, honest | Top-K, honest, offset 0 | Sortition, TTS 1.0 | Sortition, honest |
|-----|----------------|-------------|---------------|-------------------------|--------------------|-------------------|
| 1   | 1.0            | 1.0         | 1.0           | 1.0                     | 0.93               | 0.71              |
| 2   | 2.0            | 0.0         | 2.0           | 2.0                     | 1.45               | 0.89              |
| 5   | 5.0            | 0.0         | 5.0           | 5.0                     | 2.02               | 1.06              |
| 10  | 10.0           | 0.0         | 8.6           | 9.8                     | 2.23               | 1.09              |
| 20  | 19.7           | 0.0         | 0.0           | 0.0                     | 2.33               | 1.14              |
| 50  | 6.8            | 0.0         | 0.0           | 0.0                     | 2.41               | 1.18              |

With offset 0 and sortition, infiltration rises to  $7.7\% \pm 1.1\%$ , still below the veto ( $10.0\%$ ) and the linear rule ( $10.7\%$ ) under the same sortition, but most of the top-K advantage is lost because top-K selection is what keeps down-weighted liars out entirely. A deployment therefore has two options: keep top-K selection and admit only certified identities, one per registered meter with a PUF-bound key (Section 3.1), which makes identity splitting require physical meters; or accept a smaller integrity gain in exchange for a committee that is Sybil-bounded by weight share. We keep top-K selection with the newcomer ramp and certified meter identities as the recommended configuration.

### 8.11 Comparison at matched false-positive cost

At default settings, PoET<sup>2</sup>S pays more false positives per catch than the veto, so some of its advantage may be tuning. We therefore swept the threshold of each twin-based scheme (veto threshold 0.30–0.70, additive trust floor 0.30–0.70, PoET<sup>2</sup>S threshold  $\tau$  0.20–0.60 and trust rate  $\eta$  0.05–0.40; 30 seeds per setting) and compared the schemes at equal false-positive cost (Figure 12). Between 6 and 10 false positives per catch, PoET<sup>2</sup>S with  $\eta = 0.10$  catches 90.9% of attackers at 6.7 false positives per catch and 2.7% infiltration, the veto at its default threshold 91.7% at 9.7 and 10.8%, and the additive rule with floor 0.6 90.7% at 8.3 and 5.6%. At 17.0 false positives per catch, the veto (threshold 0.40) and PoET<sup>2</sup>S ( $\tau = 0.60$ ) flag exactly the same nodes, because both then fire at  $a > 0.40$ , and catch 97.6% of attackers; their infiltration is 8.1% and 2.6%. At the low-cost end, PoET<sup>2</sup>S with  $\eta = 0.05$  catches 90.0% at 3.4 false positives per catch, where the veto (threshold 0.70) catches 69.4% at 2.8. Across the sweep, the veto never goes below 5.6% infiltration, and the additive rule comes closest to PoET<sup>2</sup>S (3.2%) only at 26.3 false positives per catch. The committee-integrity gain is therefore not an artifact of the default thresholds.

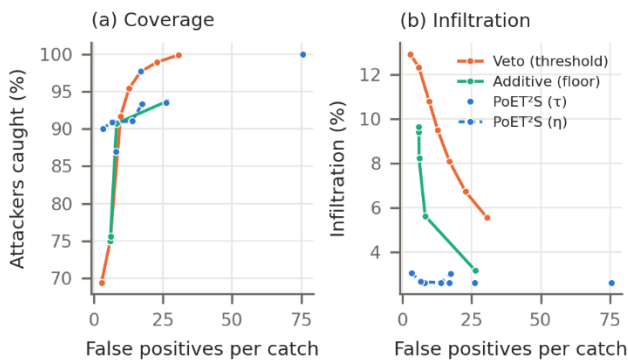


**Figure 12.** Operating points of the twin-based schemes as their thresholds vary (30 seeds per point): (a) attackers caught and (b) committee infiltration against false positives per catch. PoET<sup>2</sup>S points for  $\tau$  are shown without a line because  $\tau$  does not change infiltration.

### 8.12 Composite ranking

A composite score depends on which metrics it includes. Metrics shared by the four committee schemes (latency, messages) would favor them jointly, so the composite is restricted to the four twin-based schemes, which share all machinery, and to the three metrics on which they differ: attackers caught, FP/caught and infiltration. Each metric is min–max normalized to  $[0, 1]$  with 1 best, and scores are computed with equal weights ( $1/3$  each) and with the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) [41]. Table 13 gives the normalized matrix, the scores, a leave-one-metric-out analysis, and the share of

the weight simplex (1,326 weight vectors on a 0.02 grid) on which each scheme ranks first.

PoET<sup>2</sup>S ranks first with equal weights (0.65) and with TOPSIS (0.61). The ranking depends on infiltration: when that metric is dropped, the veto ranks first (0.76), and when attackers caught is dropped, the additive rule ranks first (0.59 against 0.50). PoET<sup>2</sup>S ranks first on 57.1% of the weight simplex, the additive rule on 23.4% and the veto on 19.5%; the linear rule never ranks first because PoET<sup>2</sup>S matches it on detection and improves on infiltration. The composite therefore adds little to the individual metrics: PoET<sup>2</sup>S is preferable when committee infiltration is given substantial weight.

## 9. Discussion and threats to validity

### 9.1 What the comparison shows

Using the twin as a continuous weight does not catch more attackers than using it as a veto, and at the default settings it costs more false positives per catch; at matched false-positive cost the two catch the same share of attackers (Section 8.11). It keeps Byzantine nodes out of the committee: 2.6% against 10.8% for the veto, and against 3.8% for the linear rule that differs only in  $\gamma$ . PoET<sup>2</sup>S is also slower than PoS and DPoS in our network model; the optional fast path narrows the gap to 6–8 ms, and to 0.4–2.2 ms when all schemes are measured to the first commit proof. The case for it rests on committee integrity, which bounds what an attacker can do after it passes the detector.

### 9.2 Ranking consistency, topology and weather

The simulator computes the committee from a global view of TTS,  $e$  and stake. Section 5.2 argues that committing this state to each block lets every node compute the same committee, but the dissemination cost and any delay in trust updates are not modeled. The overlay is a complete graph with independent Pareto delays; there is no feeder topology, no correlation between delays, and no network partition. Weather is drawn independently for each node and slot, and the twin has no irradiance data. A cloud front that covers a whole feeder would lower the TTS of many honest nodes together. Because selection is relative, a uniform drop would leave the ranking largely unchanged, but a partial one would favor nodes outside the cloud, and a detector with fixed thresholds would raise more false positives. Adding irradiance data to the twin and testing correlated weather are needed before deployment.

### 9.3 A single simulator

All results come from one simulator. Its message pattern and delay distribution are validated (Section 7.3), but absolute latencies have not been compared run for run with ns-3 or OMNeT++. The latency values should be read as comparisons between schemes under one network model, not as predictions for a deployment.

### 9.4 Baselines are our implementations

PBFT, PoS and DPoS follow their textbook descriptions and are not tuned production systems such as HotStuff [42] or Tendermint [43]. The twin-based design points are stylised mechanisms, not re-implementations of PoM, EACM, or the system of Deng et al. (Table 1). The comparisons that support the main claim, PoET<sup>2</sup>S against the design points, are less exposed to this concern because all four share one implementation.

**Table 13.** Composite over the twin-based schemes (30-seed means). Caught, FP/c, Infil.: normalised metrics (1 = best). EW, TOP.: equal-weight and TOPSIS scores. -C, -F, -I: equal-weight score with caught, FP/caught or infiltration left out. First: share of the weight simplex on which the scheme ranks first.

| Scheme              | Caught | FP/c | Infil. | EW   | TOP. | -C   | -F   | -I   | First |
|---------------------|--------|------|--------|------|------|------|------|------|-------|
| Veto                | 1.00   | 0.52 | 0.00   | 0.51 | 0.27 | 0.26 | 0.50 | 0.76 | 19.5% |
| Additive            | 0.00   | 1.00 | 0.17   | 0.39 | 0.44 | 0.59 | 0.09 | 0.50 | 23.4% |
| Linear              | 0.96   | 0.00 | 0.85   | 0.60 | 0.57 | 0.43 | 0.91 | 0.48 | 0.0%  |
| PoET <sup>2</sup> S | 0.96   | 0.00 | 1.00   | 0.65 | 0.61 | 0.50 | 0.98 | 0.48 | 57.1% |

## 9.5 Scope

The study covers consensus and validator selection. It does not clear a market, so effects on prices and welfare are not measured, and the profit proxy counts over-reported energy, not money. The Sybil study shows that the protection of top-K selection depends on identity admission. It is expected to hold if each identity is a certified meter that must pass the newcomer ramp, which this study assumes but does not test; without certified identities, only sortition bounds a stake-rich attacker, at a high cost in committee integrity.

## 10. Conclusion

This paper placed a digital twin's trust signal inside validator selection and compared the results, under identical attack realizations, with classical protocols and three stylized twin-based design points. Detection is a near tie with the veto (91.0% vs. 91.7%); the veto has the lower false-positive cost at default settings, but at matched cost, both catch the same share of attackers. Committee infiltration falls from 10.8% under the veto and 3.8% under the linear rule to 2.6% under PoET<sup>2</sup>S; the gain over the linear rule comes from the exponent and holds on 100 seeds (37%, Holm-adjusted  $p = 7.2 \times 10^{-9}$ ). At 45% Byzantine nodes, PoET<sup>2</sup>S committees lose the one-third fault bound in 2.4% of blocks, against 64.4% for the veto and 10.7% for the linear rule. The advantage also holds for two adaptive strategies. PoET<sup>2</sup>S is slower than PoS and DPOS, faster than PBFT, and flat in N in the simulated network; its low committee infiltration makes a two-step fast path usable, which lowers latency to 26.8 ms with every other result unchanged, whereas the same fast path slows the veto down. Top-K selection needs certified identities and a newcomer trust ramp to resist identity splitting; weighted sortition removes that requirement but gives up most of the integrity gain. PoET<sup>2</sup>S is a reasonable choice when keeping adversaries out of the validating committee matters more than raw latency, in a permissioned setting where each identity is a registered meter. Four directions follow from the limitations. First, look for a selection rule between top-K and weighted sortition, for example, sortition on a sharpened weight, that keeps most of the integrity gain while bounding identity splitting. Second, add irradiance data, a feeder topology, and correlated weather to the twin and network models. Third, compare absolute latencies run-for-run with a packet-level simulator and replace the textbook baselines with HotStuff and Tendermint implementations. Fourth, study an adaptive attacker that optimizes its lie schedule against trust dynamics, and couple the consensus layer to a market-clearing model so that effects on prices can be measured.

## Ethical issue

The authors are aware of and comply with best practices in publication ethics, specifically regarding authorship (avoidance of guest authorship), dual submission, manipulation of figures, competing interests, and compliance with research ethics policies. The authors adhere to publication requirements that the submitted work is original and has not been published elsewhere. All survey participants provided informed consent before participating, and the research team strictly maintained the anonymity and confidentiality of all respondents throughout the research process.

## Use of AI tools

The authors used an AI assistant (Claude, Anthropic) to help restructure the simulator code.

## Data availability statement

The manuscript contains all the data. However, additional data will be provided by the corresponding author upon reasonable request. The simulator source code, the scripts that produce every table and figure, and the per-seed outputs are available at <https://github.com/firasmutar/PoET2S>.

## Conflict of interest

The authors declare no potential conflict of interest.

## References

- [1] E. Mengelkamp, J. Gärtner, K. Rock, S. Kessler, L. Orsini, and C. Weinhardt, "Designing microgrid energy markets: A case study: The Brooklyn Microgrid," *Appl. Energy*, vol. 210, pp. 870–880, Jan. 2018, doi: 10.1016/j.apenergy.2017.06.054.
- [2] T. Sousa, T. Soares, P. Pinson, F. Moret, T. Baroche, and E. Sorin, "Peer-to-peer and community-based markets: A comprehensive review," *Renew. Sustain. Energy Rev.*, vol. 104, pp. 367–378, Apr. 2019, doi: 10.1016/j.rser.2019.01.036.
- [3] M. Andoni, V. Robu, D. Flynn, S. Abram, D. Geach, D. Jenkins, P. McCallum, and A. Peacock, "Blockchain technology in the energy sector: A systematic review of challenges and opportunities," *Renew. Sustain. Energy Rev.*, vol. 100, pp. 143–174, Feb. 2019, doi: 10.1016/j.rser.2018.10.014.
- [4] A. Q. Al-Shetwi, I. E. Atawi, M. A. El-Hameed, and A. Abuelrub, "Digital twin technology for renewable energy, smart grids, energy storage and vehicle-to-grid integration: Advancements, applications, key players, challenges and future perspectives in modernising sustainable grids," *IET Smart Grid*, vol. 8, no. 1, art. no. e70026, 2025, doi: 10.1049/stg2.70026.
- [5] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Trans. Smart Grid*, vol. 8, no. 4,

- pp. 1630–1638, Jul. 2017, doi: 10.1109/TSG.2015.2495133.
- [6] R. K. Suggala, J. Kumar, P. Jain, and B. K. Kumar, "Blockchain technology for digital twin security in smart grids using interpretable generalized additive neural networks," *Peer-to-Peer Netw. Appl.*, vol. 18, no. 4, art. no. 172, 2025, doi: 10.1007/s12083-025-01929-6.
  - [7] A. K. Dutta, M. Anjum, H. Min, Y. I. Daradkeh, J. T. Seo, and S. Shahab, "Digital twin-assisted blockchain IoT security model using contrastive and causal learning techniques," *Sci. Rep.*, vol. 16, art. no. 15732, Apr. 2026, doi: 10.1038/s41598-026-47104-6.
  - [8] N. Weerasinghe, R. Mishra, P. Porambage, M. Liyanage, and M. Ylianttila, "Proof-of-Monitoring (PoM): A novel consensus mechanism for blockchain-based secure service level agreement management," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 3, pp. 2783–2803, Sep. 2023, doi: 10.1109/TNSM.2023.3234862.
  - [9] O. A. Saleh and M. Cevik, "Blockchain-integrated secure authentication framework for smart grid IoT using energy-aware consensus mechanisms," *Sensors*, vol. 25, no. 21, art. no. 6622, 2025, doi: 10.3390/s25216622.
  - [10] W. Tushar, T. K. Saha, C. Yuen, D. Smith, and H. V. Poor, "Peer-to-peer trading in electricity networks: An overview," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3185–3200, Jul. 2020, doi: 10.1109/TSG.2020.2969657.
  - [11] E. Mengelkamp, B. Notheisen, C. Beer, D. Dauer, and C. Weinhardt, "A blockchain-based smart grid: Towards sustainable local energy markets," *Comput. Sci. Res. Dev.*, vol. 33, no. 1–2, pp. 207–214, 2018, doi: 10.1007/s00450-017-0360-9.
  - [12] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "A survey of distributed consensus protocols for blockchain networks," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1432–1465, 2020, doi: 10.1109/COMST.2020.2969706.
  - [13] M. Castro and B. Liskov, "Practical Byzantine fault tolerance," in *Proc. 3rd USENIX Symp. Operating Systems Design and Implementation (OSDI)*, New Orleans, LA, USA, 1999, pp. 173–186. [Online]. Available: <https://pmg.csail.mit.edu/papers/osdi99.pdf>
  - [14] 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. Ind. Informat.*, vol. 14, no. 8, pp. 3690–3700, Aug. 2018, doi: 10.1109/TII.2017.2786307.
  - [15] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled Internet of Vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2906–2920, Mar. 2019, doi: 10.1109/TVT.2019.2894944.
  - [16] X. Deng, J. Li, L. Shi, K. Wei, M. Ding, Y. Shao, W. Chen, and S. Jin, "Trustworthy DNN partition for blockchain-enabled digital twin in wireless IIoT networks," *Sci. China Inf. Sci.*, vol. 67, no. 11, art. no. 219303, 2024, doi: 10.1007/s11432-024-4159-2.
  - [17] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proc. 44th ACM/IEEE Design Automation Conf. (DAC)*, San Diego, CA, USA, 2007, pp. 9–14, doi: 10.1145/1278480.1278484.
  - [18] V. Costan and S. Devadas, "Intel SGX explained," *IACR Cryptology ePrint Archive*, Rep. 2016/086, 2016. [Online]. Available: <https://eprint.iacr.org/2016/086>
  - [19] S. Pinto and N. Santos, "Demystifying Arm TrustZone: A comprehensive survey," *ACM Comput. Surv.*, vol. 51, no. 6, art. no. 130, 2019, doi: 10.1145/3291047.
  - [20] J. Van Bulck, M. Minkin, O. Weisse, D. Genkin, B. Kasikci, F. Piessens, M. Silberstein, T. F. Wenisch, Y. Yarom, and R. Strackx, "Foreshadow: Extracting the keys to the Intel SGX kingdom with transient out-of-order execution," in *Proc. 27th USENIX Security Symp.*, Baltimore, MD, USA, Aug. 2018, pp. 991–1008. [Online]. Available: <https://www.usenix.org/conference/usenixsecurity18/presentation/bulck>
  - [21] S. Matetic, M. Ahmed, K. Kostianen, A. Dhar, D. Sommer, A. Gervais, A. Juels, and S. Capkun, "ROTE: Rollback protection for trusted execution," in *Proc. 26th USENIX Security Symp.*, Vancouver, BC, Canada, Aug. 2017, pp. 1289–1306. [Online]. Available: <https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/matetic>
  - [22] S. Tan, D. De, W.-Z. Song, J. Yang, and S. K. Das, "Survey of security advances in smart grid: A data driven approach," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 397–422, 2017, doi: 10.1109/COMST.2016.2616442.
  - [23] J. R. Douceur, "The Sybil attack," in *Proc. 1st Int. Workshop Peer-to-Peer Systems (IPTPS)*, ser. Lecture Notes in Computer Science, vol. 2429. Berlin, Germany: Springer, 2002, pp. 251–260, doi: 10.1007/3-540-45748-8\_24.
  - [24] A. Hahn, R. K. Thomas, I. Lozano, and A. Cardenas, "A multi-layered and kill-chain based security analysis framework for cyber-physical systems," *Int. J. Crit. Infrastruct. Prot.*, vol. 11, pp. 39–50, Dec. 2015, doi: 10.1016/j.ijcip.2015.08.003.
  - [25] S. W. Roberts, "Control chart tests based on geometric moving averages," *Technometrics*, vol. 1, no. 3, pp. 239–250, 1959, doi: 10.1080/00401706.1959.10489860.
  - [26] Pecan Street Inc., "Dataport," Austin, TX, USA. [Online]. Available: <https://www.pecanstreet.org/dataport/> (accessed Sep. 2026).
  - [27] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht, "Smart\*: An open data set and tools for enabling research in sustainable homes," in *Proc. KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, Beijing, China, Aug. 2012. [Online]. Available: <https://lass.cs.umass.edu/papers/pdf/sustkdd12-smart.pdf>

- [28] J.-P. Martin and L. Alvisi, "Fast Byzantine consensus," *IEEE Trans. Dependable Secure Comput.*, vol. 3, no. 3, pp. 202–215, Jul.–Sep. 2006, doi: 10.1109/TDSC.2006.35.
- [29] P. Kuznetsov, A. Tonkikh, and Y. X. Zhang, "Revisiting optimal resilience of fast Byzantine consensus," in *Proc. ACM Symp. Principles of Distributed Computing (PODC)*, Jul. 2021, pp. 343–353, doi: 10.1145/3465084.3467924.
- [30] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, "Algorand: Scaling Byzantine agreements for cryptocurrencies," in *Proc. 26th Symp. Operating Systems Principles (SOSP)*, Shanghai, China, 2017, pp. 51–68, doi: 10.1145/3132747.3132757.
- [31] G. F. Riley and T. R. Henderson, "The ns-3 network simulator," in *Modeling and Tools for Network Simulation*, K. Wehrle, M. Güneş, and J. Gross, Eds. Berlin, Germany: Springer, 2010, pp. 15–34, doi: 10.1007/978-3-642-12331-3\_2.
- [32] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *Proc. 1st Int. Conf. Simulation Tools and Techniques (SIMUTools)*, Marseille, France, 2008, art. no. 60, doi: 10.4108/ICST.SIMUTOOLS2008.3027.
- [33] M. E. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic: Evidence and possible causes," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 835–846, Dec. 1997, doi: 10.1109/90.650143.
- [34] D. J. Bernstein, N. Duif, T. Lange, P. Schwabe, and B.-Y. Yang, "High-speed high-security signatures," *J. Cryptogr. Eng.*, vol. 2, no. 2, pp. 77–89, 2012, doi: 10.1007/s13389-012-0027-1.
- [35] G. Danezis, L. Kokoris-Kogias, A. Sonnino, and A. Spiegelman, "Narwhal and Tusk: A DAG-based mempool and efficient BFT consensus," in *Proc. 17th European Conf. Computer Systems (EuroSys)*, Rennes, France, 2022, pp. 34–50, doi: 10.1145/3492321.3519594.
- [36] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, Dec. 1945, doi: 10.2307/3001968.
- [37] S. Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Statist.*, vol. 6, no. 2, pp. 65–70, 1979. [Online]. Available: <https://www.jstor.org/stable/4615733>
- [38] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*. New York, NY, USA: Chapman & Hall/CRC, 1993, doi: 10.1201/9780429246593.
- [39] V. Christodoulou and Y. Bi, "A combination of CUSUM-EWMA for anomaly detection in time series data," in *Proc. IEEE Int. Conf. Data Science and Advanced Analytics (DSAA)*, Paris, France, 2015, pp. 1–8, doi: 10.1109/DSAA.2015.7344899.
- [40] Z.-G. Zhou and P. Tang, "Improving time series anomaly detection based on exponentially weighted moving average (EWMA) of season-trend model residuals," in *Proc. IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS)*, Beijing, China, 2016, pp. 3414–3417, doi: 10.1109/IGARSS.2016.7729882.
- [41] C.-L. Hwang and K. Yoon, *Multiple Attribute Decision Making: Methods and Applications (Lecture Notes in Economics and Mathematical Systems, vol. 186)*. Berlin, Germany: Springer, 1981, doi: 10.1007/978-3-642-48318-9.
- [42] M. Yin, D. Malkhi, M. K. Reiter, G. G. Gueta, and I. Abraham, "HotStuff: BFT consensus with linearity and responsiveness," in *Proc. ACM Symp. Principles of Distributed Computing (PODC)*, Toronto, ON, Canada, 2019, pp. 347–356, doi: 10.1145/3293611.3331591.
- [43] E. Buchman, J. Kwon, and Z. Milosevic, "The latest gossip on BFT consensus," *arXiv:1807.04938*, 2018, doi: 10.48550/arXiv.1807.04938.



This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## Appendix

**Table A1.** Per-seed committee infiltration (%), N = 500, 15% Byzantine.

| Seed | PoET <sup>2</sup> S | Linear | Veto  | Additive |
|------|---------------------|--------|-------|----------|
| 42   | 2.38                | 1.51   | 10.24 | 10.32    |
| 43   | 0.79                | 2.06   | 15.00 | 9.37     |
| 44   | 0.56                | 2.22   | 5.24  | 11.03    |
| 45   | 0.56                | 1.35   | 13.18 | 6.43     |
| 46   | 3.10                | 3.81   | 6.83  | 11.11    |
| 47   | 3.49                | 6.11   | 13.10 | 10.56    |
| 48   | 0.71                | 2.06   | 7.22  | 7.22     |
| 49   | 1.51                | 1.35   | 15.95 | 7.38     |
| 50   | 1.11                | 2.62   | 17.78 | 13.33    |
| 51   | 6.03                | 11.67  | 12.78 | 6.27     |
| 52   | 6.43                | 9.84   | 16.11 | 13.25    |
| 53   | 1.67                | 2.86   | 11.67 | 12.70    |
| 54   | 5.24                | 7.38   | 14.60 | 11.75    |
| 55   | 4.44                | 3.73   | 7.78  | 6.27     |
| 56   | 2.30                | 5.32   | 9.84  | 5.95     |
| 57   | 3.25                | 3.02   | 5.95  | 8.25     |
| 58   | 5.48                | 6.03   | 10.24 | 7.06     |
| 59   | 3.17                | 2.62   | 11.82 | 8.41     |
| 60   | 2.14                | 1.98   | 7.78  | 2.06     |
| 61   | 1.43                | 8.10   | 10.63 | 16.51    |
| 62   | 3.10                | 4.68   | 11.27 | 6.35     |
| 63   | 1.19                | 4.52   | 14.21 | 8.81     |
| 64   | 0.32                | 0.32   | 6.83  | 12.06    |
| 65   | 0.79                | 0.71   | 6.35  | 13.33    |
| 66   | 5.40                | 4.92   | 8.57  | 10.87    |
| 67   | 1.03                | 0.95   | 9.29  | 11.27    |
| 68   | 2.46                | 6.51   | 17.62 | 2.46     |
| 69   | 0.64                | 1.11   | 11.59 | 4.60     |
| 70   | 6.19                | 5.79   | 7.70  | 12.54    |
| 71   | 2.38                | 0.32   | 6.51  | 14.52    |