

Location Optimization Using Wireless Sensor Network with Blockchain: A Trust-Aware, Multi-Objective Framework

Vishal Kumar Sah¹, Mrs. Richa Pandey²

¹MCA Student, School of Computer Application, BBD University, Lucknow, India

²Assistant Professor, School of Computer Application, BBD University, Lucknow, India

doi.org/10.64643/IJIRT209487

Abstract—The performance of a Wireless Sensor Network (WSN), including its coverage area, operational lifetime, and ability to be trusted, depends upon the physical location of the nodes and the accuracy of their reported positions. While previous research has focused on using swarm intelligence and hybrid meta-heuristic algorithms to optimize node placement, and separately has explored securing localization and trust with blockchain, the two lines of work have rarely been brought together. In particular, placement optimization algorithms typically assume that nodes have known, reliable positions, while blockchain-based localization schemes typically assume that the network is already deployed.

This study proposes a trust-aware, multi-objective framework that connects the two. A swarm-based optimizer determines where to place nodes based on objectives such as coverage, connectivity, energy consumption, and localization error. Additionally, a permissioned blockchain, which is operated by edge gateways, maintains a tamper-proof record of trust scores for each node, position claims made by nodes, and deployment decisions. Localization utilizes trust-weighted contributions from anchor nodes to determine the location of other nodes. In doing so, anchor nodes that are deemed untrustworthy are given less weight in contributing to position estimates. The authors formally define the problem, develop a placement algorithm and smart-contract logic, analyze the complexity and security of the system, and establish a reproducible evaluation protocol that includes baselines drawn from 2025–2026 literature. Figures from prior studies are used only as background information; no new experimental results are presented.

Index Terms—Blockchain, Energy Efficiency, Localization, Node Placement, Particle Swarm Optimization, PBFT, Trust Management, Wireless Sensor Networks.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) enable environmental monitoring, disaster response, healthcare delivery, and critical-infrastructure protection. Two location-related issues are central to designing WSNs.

First, placement: how should nodes be deployed within an area so that a maximum number of areas are covered by a minimum number of nodes, the network remains connected throughout its lifetime, and it consumes minimal energy? Second, localization: how can each node's position be estimated when most nodes lack GPS receivers, and how much can that estimate be relied upon when some anchors are hostile or faulty? False position claims negatively affect coverage, routing, and all location-tagged readings; therefore, placement and localization should ideally be addressed simultaneously.

Finding an optimal placement is NP-hard. Recent studies have used various forms of Particle Swarm Optimization (PSO), Grey Wolf Optimization (GWO), Whale Optimization Algorithm (WOA), and hybrid Genetic Algorithm–Particle Swarm Optimization (GA–PSO) to solve this problem [6]–[10]. A 2026 study extends this line of research to dynamic coverage optimization [12], and another study addresses boundary-node detection and localization in anisotropic networks [11].

In parallel with placement optimization research, blockchain has been investigated as a means to enhance WSNs' resistance to insider attacks. A 2025 framework calculates behavior-based trust locally and commits aggregate trust scores to a permissioned chain through PBFT at edge gateways [2]. Other

researchers have investigated combining blockchain with DV-Hop localization and deep learning to reduce DDoS effects and improve node positioning [3], and a 2026 study anchors uncertainty-aware Wi-Fi localization evidence on a permissioned ledger while keeping raw telemetry off-chain [4].

These two research areas have remained largely independent. Placement optimization algorithms typically assume that nodes have known reliable positions, while blockchain-based localization schemes typically assume that the network is already deployed. Consequently, few approaches integrate trust and localization quality into determining where nodes are placed. This paper addresses that shortfall.

Our contributions are:

A joint formulation that incorporates localization error and node trust into the placement objective along with coverage, connectivity, and energy consumption.

A hybrid on-chain/off-chain architecture that employs a lightweight smart contract to record trust scores, position claims, and deployment epochs, while maintaining raw telemetry off-chain.

A trust-weighted localization estimator and a trust-aware hybrid swarm placement algorithm that includes complexity and security analyses.

A reproducible evaluation protocol that specifies metrics, baselines, and statistical tests drawn from 2025–2026 studies.

The remaining sections of this paper are organized as follows. Section II reviews related work. Section III defines the problem. Section IV describes the proposed framework. Section V analyzes security aspects of the framework. Section VI presents the evaluation methodology. Section VII discusses tradeoffs and limitations of the framework. Section VIII concludes the paper.

II. RELATED WORK

A. Node Placement and Coverage Optimization

A 2026 study proposed an improved PSO-based deployment method that moves nodes dynamically to maximize blanket coverage while limiting node count and preserving connectivity and power efficiency [6]. An enhanced hybrid PSO with spatial position encoding was developed to balance exploration and exploitation to increase coverage while decreasing the number of deployed sensors [7]. A PSO-based framework with role assignment and adaptive maintenance was compared against nine deployment strategies in simulation and reported average coverage of approximately 91.4% with lifetime exceeding 3,400 rounds [8]. A GA–PSO hybrid was reported to require 15–25% fewer nodes while maintaining at least 95% coverage and connectivity relative to standalone GA or PSO [9].

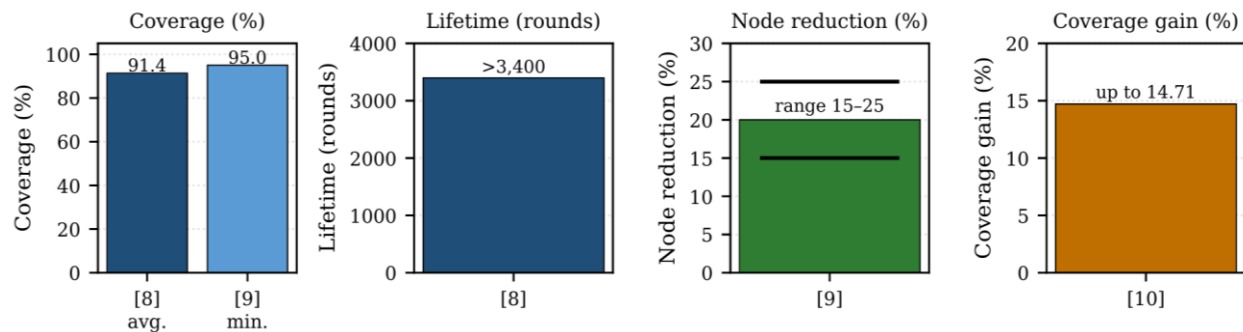


Fig. 1. Performance figures reported by recent placement studies [8]–[10], plotted as quoted. Simulators, baselines and parameters differ, so the panels are not directly comparable.

Explanation of Fig. 1. Fig. 1 collects the headline results of the placement studies that serve as our reference baselines. The first panel shows the coverage reported by [8] (average 91.4%) and [9] (minimum 95.0%); the second shows the network lifetime of [8] (more than 3,400 rounds); the third shows the node reduction of [9] (15–25%, drawn as a

range); and the fourth shows the coverage gain of [10] (up to 14.71%).

The values are reproduced exactly as published. Because each study used a different simulator, baseline set and parameter configuration, the bars must not be compared with one another. Their purpose

is to indicate the performance level that the proposed protocol in Section VI should be measured against.

B. Blockchain for Secure Localization, Trust, and Energy

Blockchain-based trust management for WSNs assesses packet forwarding behavior, energy consumption patterns, and behavioral consistency among nodes and aggregates and stores these assessments using PBFT among edge gateways [2]. A deep-learning and blockchain approach combined with DV-Hop was investigated to mitigate DDoS attacks and enhance localization accuracy [3]. The passive Wi-Fi framework in [4] generates a posterior spatial distribution with calibrated uncertainty and links outputs to a permissioned ledger; its authors note that practical deployment depends on antenna geometry calibration and mobile or UAV-assisted collection, and they report a 40 m operational validity boundary. Blockchain has also been utilized for decentralized energy swapping between sensor nodes [5], and for secure, energy-efficient IoT-WSN operation in smart cities [1]. Recent 2026 studies continue to examine blockchain-based trusted routing [13], zero-trust architectures for IoT-enabled WSNs [14], blockchain-driven trust management and AI computing [15], and voting-based trustworthy localization [16].

TABLE I POSITIONING AGAINST REPRESENTATIVE 2025–2026 WORK

Ref.	Placement	Trusted localization	Blockchain
[2]	No	Trust only	Yes (PBFT, edge)
[3]	DV-Hop	Yes	Yes
[4]	No	Uncertainty-aware	Yes (permissioned)
[5]	No	No	Yes (energy)
[6]–[10]	Yes (PSO/hybrids)	No	No
This paper	Yes	Yes, in objective	Yes

C. Research Gap

Table I provides an overview of how representative studies address placement, trusted localization, and blockchain. The placement studies [6]–[10] do not consider trust, and the blockchain studies [2]–[5] do not optimize where nodes are placed; the proposed framework addresses all three aspects concurrently.

Explanation of Table I. Table I positions this work against the representative studies discussed above. Each row is a study (or group of studies) and each column is one of the three capabilities required for trustworthy node deployment. The placement studies [6]–[10] optimize where nodes are put but ignore trust and blockchain, whereas the blockchain studies [2]–[5] secure trust, localization or energy trading but do not decide node positions. No existing row has “Yes” in all three columns; the last row shows that the proposed framework does, and that it places trusted localization inside the optimization objective itself. This empty intersection is the research gap addressed in this paper.

III. PROBLEM FORMULATION

Assume a monitored region $A \subset \mathbb{R}^2$ contains N sensor nodes with positions $X = \{x_1, \dots, x_N\}$, sensing radius r_s , communication radius r_c , and M anchors (either a subset of nodes or fixed beacons) with known claimed positions.

A small set of K edge gateways acts as blockchain validators. Fig. 2 illustrates this scenario.

Coverage: Using a disk sensing model $D(x, r_s)$:

$$C(X) = |A \cap \bigcup_i D(x_i, r_s)| / |A| \quad (1)$$

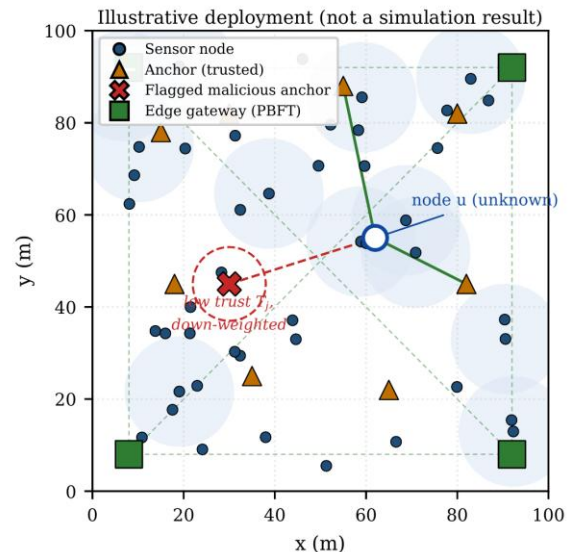


Fig. 2. Illustrative deployment (not a simulation result): sensor nodes, anchors and four edge gateways. Node u down-weights the range from a trust-flagged anchor when it estimates its position.

Shaded discs show the sensing range r_s .

Connectivity: Let $G(X)$ be the graph formed when two nodes i and j are connected if $\|x_i - x_j\| \leq r_c$; define $Q(X) \in [0,1]$ as the fraction of nodes in the largest component connected to a gateway.

Energy: Based on the first-order radio model, transmitting l bits over distance d costs $E_{tx} = l \cdot E_{elec} + l \cdot \epsilon \cdot d^n$, and receiving costs $E_{rx} = l \cdot E_{elec}$. Let $\hat{E}(X)$ be the normalized mean per-round energy consumption including the blockchain-related uplink overhead.

Trust: Each node i has a trust score in $[0,1]$:

$$T_i = \alpha \cdot F_i + \beta \cdot R_i + \gamma \cdot S_i, \quad \alpha + \beta + \gamma = 1 \quad (2)$$

where F_i represents packet-forwarding ratio, R_i represents residual-energy consistency, and S_i represents behavioral consistency, similar to the types of factors used in [2].

Trust-weighted Localization: Node u estimates its position from anchors j using distance estimates $\hat{d}_{uj}$ (range-based) or hop-derived distances (DV-Hop style [3]):

$$\hat{p}_u = \arg \min_p \sum_j T_j \cdot (\|p - a_j\| - \hat{d}_{uj})^2 \quad (3)$$

The expected localization error $L(X)$ is defined as $(1/N) \sum_u E[\|\hat{p}_u - x_u\|]$ over the anchor-attack model. Fig. 3 illustrates how higher-trust anchors contribute more significantly to the position estimate.

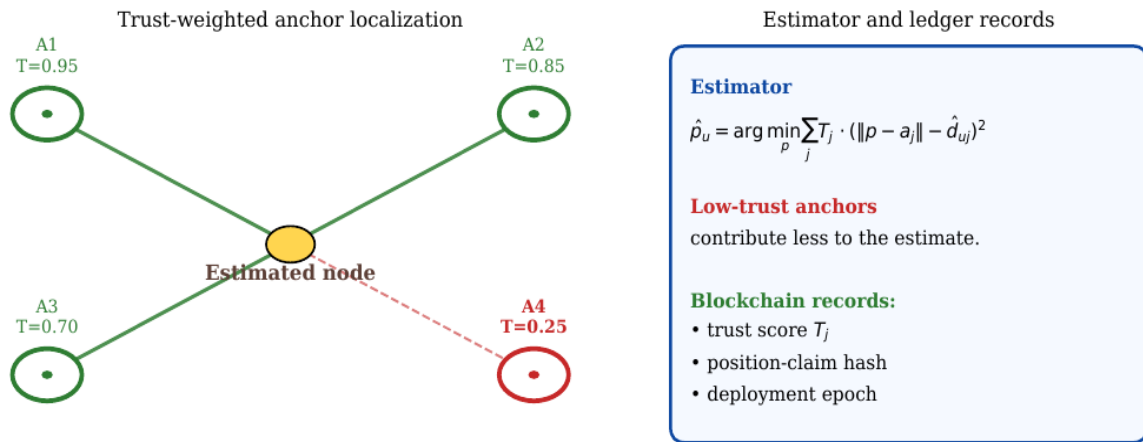


Fig. 3. Trust-weighted anchor localization: higher-trust anchors contribute more strongly to the position estimate.

Explanation of Fig. 2. Fig. 2 is a schematic of the system model in Section III on a $100 \text{ m} \times 100 \text{ m}$ region; it is an illustration, not a simulation output. Circles are sensor nodes, triangles are trusted anchors, squares are the four edge gateways that act as PBFT validators, and the red cross marks an anchor flagged as malicious. The unknown node u (white circle) estimates its position from the ranges to nearby anchors. The range from the flagged anchor (dashed red line) is down-weighted by its low trust score, while the ranges from trusted anchors (solid green lines) dominate the estimate. The shaded discs show the sensing radius r_s of each node, so overlaps and uncovered gaps in the region are visible, which relates directly to the coverage term $C(X)$.

Explanation of Fig. 3. Fig. 3 shows how (3) behaves for a single node. Four anchors A1–A4 have trust scores 0.95, 0.85, 0.70 and 0.25. Since each squared range residual is multiplied by the anchor’s trust score,

A4 contributes very little to the estimate, and its false or inconsistent range cannot pull the result far from the true position, while A1–A3 carry most of the weight. The box on the right lists what is stored on the blockchain for each anchor: its trust score, a hash of its position claim, and the deployment epoch.

Joint Objective:

$$\min_x F(X) = w_1(1-C) + w_2(1-Q) + w_3\hat{E} + w_4\hat{L} \quad (4)$$

s.t. $x_i \in A$, $N \leq N_{\max}$, where weights sum to one and $\hat{L}$ is normalized by r_c . Weights are assigned based on application requirements; alternatively, this problem can be solved as a Pareto front.

IV. PROPOSED FRAMEWORK

A. Architecture

Fig. 4 illustrates the architecture: sensors transmit data to edge gateways, which validate trust data and execute optimization algorithms, and a permissioned

ledger records the outcome. Sensor nodes lack the resources to take part in consensus, so validation is handed to K edge gateways mirroring the edge-layer permissioned design presented in [2]. Raw telemetry is kept off-chain; the ledger only stores compact

records (trust scores, position claims with hashes of supporting evidence, and epoch deployment plans) consistent with the on-chain/off-chain split proposed in [4]. This approach minimizes storage and energy costs.

Hybrid on-chain/off-chain architecture for trust-aware WSN placement and localization

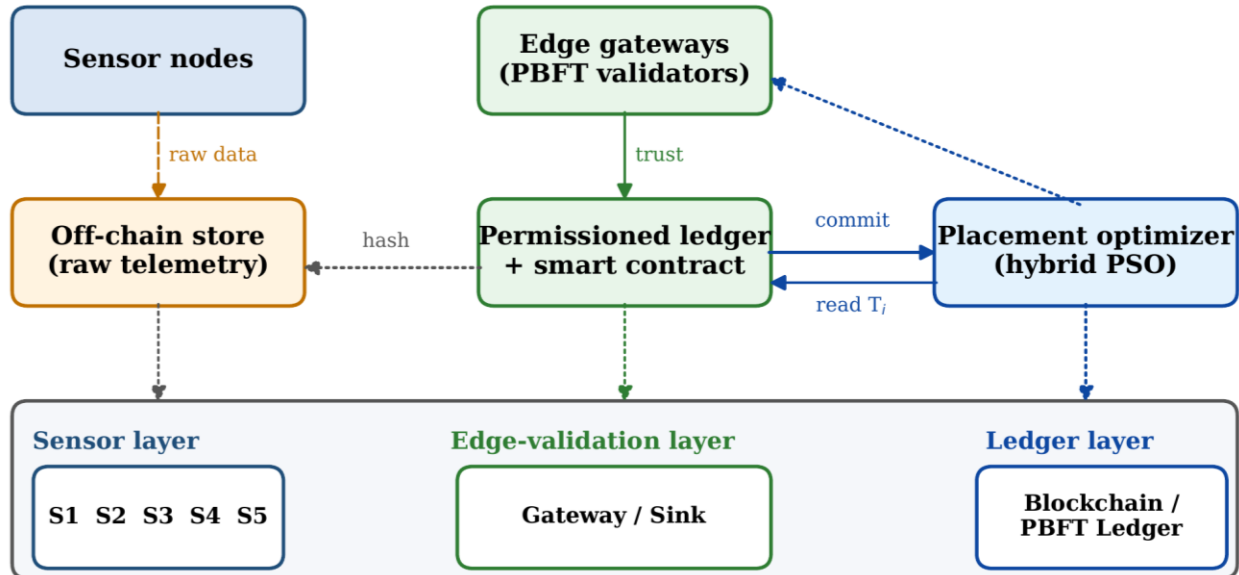


Fig. 4. Hybrid on-chain/off-chain architecture for trust-aware WSN placement and localization. Top: data and control flow between sensor nodes, edge gateways, ledger, off-chain store and placement optimizer. Bottom: layered view of the sensor, edge-validation and ledger layers.

Explanation of Fig. 4. Fig. 4 shows the overall architecture. In the top diagram, sensor nodes send raw data to an off-chain store and trust-related inputs to the edge gateways. The gateways, acting as PBFT validators, write only compact records (trust scores and hashes of the supporting evidence) to the permissioned ledger and smart contract. The placement optimizer (hybrid PSO) reads trust scores T_i from the ledger and commits the approved placement plan back to it, while the hash link ensures that the off-chain evidence can later be checked against the ledger. The bottom diagram groups the same components into three layers: the sensor layer, the edge-validation layer and the ledger layer. Consensus work is therefore kept away from the resource-constrained sensors.

B. Smart Contract Logic

The contract provides four functions: *registerNode* (identity and initial position claim), *submitTrust*

(gateway-signed aggregate trust scores per epoch), *flagAnchor* (an anchor is removed from consideration when its trust score falls below τ or when its position claim contradicts a quorum of neighbor observations), and *commitPlan* (records the approved placement for the epoch). All state changes require signatures from at least $2f + 1$ of K validators to tolerate $f = \lfloor (K - 1)/3 \rfloor$ Byzantine gateways. Fig. 5 illustrates this commit flow. Fig. 6 illustrates the per-epoch placement workflow that reads and writes these records.

Explanation of Fig. 5. Fig. 5 is a sequence diagram of one PBFT round. (1) Sensor nodes' signed trust inputs reach the primary gateway. (2) The primary broadcasts a PRE-PREPARE message containing the request and its digest. (3) The replica gateways answer with signed PREPARE messages. (4) After collecting $2f + 1$ PREPAREs, the gateways send COMMIT messages. (5) After $2f + 1$ COMMITs, the smart contract call is executed and a new hash-chained block is appended.

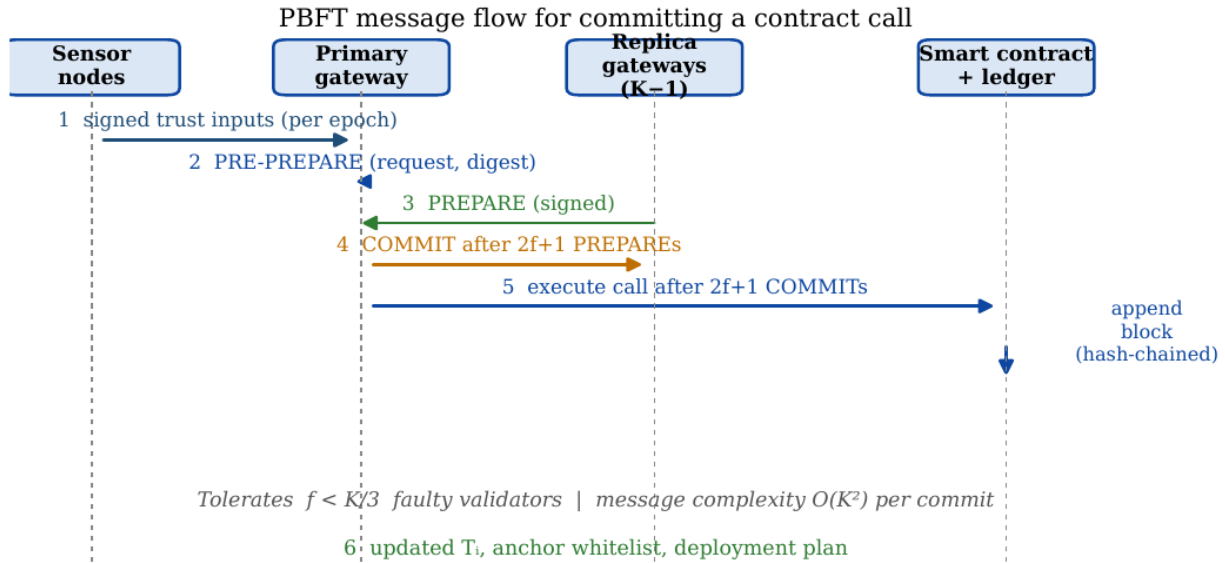


Fig. 5. Message flow for committing a contract call (submitTrust, flagAnchor or commitPlan) through PBFT among K edge gateways.

(6) The updated trust scores, anchor whitelist and deployment plan become available. The protocol tolerates $f < K/3$ faulty validators, and its message complexity is $O(K^2)$ per commit.

C. Trust-Aware Swarm Placement

We utilize a hybrid PSO with GA-style mutation due to the documented advantages of hybrids: rapid convergence from PSO and exploration from GA [9], and adaptive exploration–exploitation control with position encoding [7]. Each particle encodes N coordinates; the fitness function is $F(X)$ defined in Section III, with T_i values retrieved from the ledger. Placement is reevaluated during each epoch or when a trust event (an anchor flagged or a node depleted) exceeds a predefined threshold. For static networks, the plan guides initial deployment or supplemental node drops; for mobile or UAV-assisted deployments as discussed in [4], the plan guides waypoints.

TABLE II TRUST-AWARE HYBRID PSO PLACEMENT (PER EPOCH)

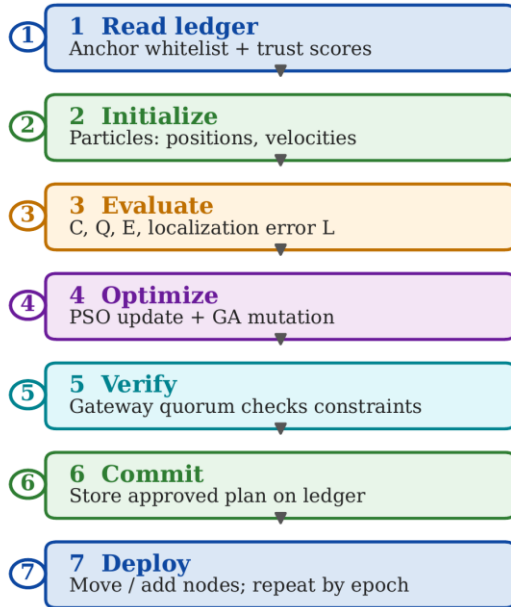
Step	Operation
1	Read anchor whitelist and trust scores T from ledger
2	Initialize P particles (positions X_p , velocities V_p) inside A
3	for $g = 1 \dots G$ do
4	for each particle p :
5	Estimate C , Q , E from X_p (geometry + radio)

6	Estimate L via trust-weighted localization
7	$\text{fitness}_p = w_1(1-C) + w_2(1-Q) + w_3\hat{E} + w_4\hat{L}$
8	Update personal and global best
9	Update V_p , X_p with adaptive inertia; clip to A
10	With prob. p_m mutate worst 20% particles
11	end for
12	Gateway quorum verifies best plan
13	$\text{commitPlan}(\text{best})$ on ledger; deploy / move nodes

Explanation of Table II. Table II gives the algorithm that is executed in every deployment epoch. Steps 1–2 read the current anchor whitelist and trust scores from the ledger and randomly initialize P particles, each representing one complete candidate placement of N nodes.

Steps 3–11 are the optimization loop: for every particle, coverage C , connectivity Q and energy E are estimated from the geometry and radio model, the localization error L is computed with the trust-weighted estimator of (3), and the fitness of (4) is evaluated.

Personal and global bests are updated, velocities and positions are modified using an adaptive inertia weight, and the worst 20% of particles are mutated with probability p_m (the GA-style step that preserves diversity). Steps 12–13 let the gateway quorum verify the best plan and commit it to the ledger, after which the nodes are deployed or moved.

Trust-aware hybrid PSO placement workflow
(each deployment epoch)Fig. 6. Trust-aware hybrid PSO placement workflow
used for each deployment epoch.

Explanation of Fig. 6. Fig. 6 presents the seven stages of Table II as a flowchart: (1) read the ledger; (2) initialize particles; (3) evaluate C , Q , E and L ; (4)

optimize with the PSO update and GA mutation; (5) have the gateway quorum verify the constraints; (6) commit the approved plan on the ledger; and (7) deploy or move nodes. The cycle repeats at every epoch, or earlier if a trust event occurs, so the placement always reflects the latest trust state stored on the ledger.

D. Complexity

Each fitness evaluation requires $O(N^2)$ for connectivity plus $O(N \cdot |G_A|)$ for coverage on a grid G_A ; the optimizer runs in $O(P \cdot G \cdot (N^2 + N|G_A|))$ offline at gateways or the sink. PBFT message complexity is $O(K^2)$ per commit, which explains why K is typically small (4–10) rather than proportional to N . Sensor-side cost is limited to calculating local trust inputs and one signed uplink per epoch.

V. SECURITY ANALYSIS

The framework does not protect against a coordinated majority of anchors that are consistently and plausibly wrong. The estimator degrades gracefully but not without limit, and the analysis assumes that less than half of the total anchor weight is adversarial. Table III summarizes the main threats and the corresponding mitigations.

TABLE III THREATS AND MITIGATIONS

Threat	Effect	Mitigation
Malicious anchor (false position)	Biased localization	Trust-weighted estimator; quorum conflict check; de-listing via <i>flagAnchor</i>
Sybil / replay of claims	Fake coverage or trust	Registered identities; epoch nonces; signed records
Compromised gateway	Forged trust or plan	PBFT tolerates $f < K/3$ faulty validators
DDoS on nodes / gateways	Data loss, energy drain	Anomaly-driven trust penalty (cf. [3]); redundant gateways
Ledger tampering	Evidence falsification	Hash-chained blocks; off-chain evidence hashes anchored on-chain

Explanation of Table III. Table III links each attack to its effect on the system and to the defence used against it. A malicious anchor that reports a false position biases localization, so its influence is reduced by trust weighting and it is removed by *flagAnchor* when its claim contradicts a quorum of neighbors. Sybil and replay attacks, which create fake identities or reuse old messages, are blocked by registered identities, per-epoch nonces and signed records. A compromised gateway cannot forge a trust score or plan because the PBFT quorum tolerates up to $f < K/3$ faulty validators.

DDoS attacks are countered by an anomaly-driven trust penalty and redundant gateways, and tampering with stored evidence is detected through hash-chained blocks and on-chain evidence hashes. Each threat therefore has at least one dedicated defence layer.

VI. EVALUATION METHODOLOGY

We propose a simulation study (e.g., MATLAB, Python/NS-3) using the configuration specified in Table IV. No numeric results are presented here;

however, the protocol is designed so that results can be reproduced and directly compared with coverage and lifetime figures published in [8]–[10].

TABLE IV PROPOSED SIMULATION PARAMETERS (TO BE TUNED)

Parameter	Value / range
Region	100×100 m and 500×500 m, with and without obstacles
Nodes N	50, 100, 200; anchors 10–20%
r_s / r_c	10–20 m / $2 \times r_s$
Radio	First-order model; $E_{elec} = 50$ nJ/bit; initial energy 0.5–2 J
Adversary	0, 10, 20, 30% malicious anchors (position offset, collusion)
Validators K	4, 7, 10 gateways; PBFT
Optimizer	$P = 40$ –60, $G = 200$ –500; 30 independent runs

Explanation of Table IV. Table IV lists the parameters of the proposed simulation study. Two region sizes, with and without obstacles, test both small and larger deployments; node counts (50–200) and anchor ratios (10–20%) vary the network density; and the sensing and communication radii and the first-order radio model with its initial energy define coverage and energy behavior. The adversary row raises the share of malicious anchors from 0 to 30% to test how far trust weighting protects localization, and the validator count $K = 4, 7, 10$ tests how consensus size affects latency. The optimizer settings (P , G and 30 independent runs) provide enough repetitions for statistical testing. All values are ranges that can be tuned once the simulations begin.

Metrics: Coverage rate; connectivity ratio; network lifetime (rounds until first node failure and until 50% depletion); mean localization error (m) and its 95th percentile; commit latency and throughput of the ledger; per-node energy overhead due to blockchain; false-positive/negative rate of malicious-anchor detection.

Baselines: (i) Random deployment; (ii) plain PSO and GA; (iii) improved PSO from [6], EHPSO [7], PSO framework from [8], and GA-PSO [9]; (iv) DV-Hop without blockchain and blockchain-DV-Hop [3] for localization.

Statistics: Report mean $\pm$ standard deviation over 30 runs with Wilcoxon signed-rank tests at $\alpha = 0.05$ for pairwise comparisons, and ablations that remove (a) trust weighting, (b) localization error from the objective function, and (c) the ledger.

VII. DISCUSSION

Expected trade-offs: Incorporating localization error into the objective function may result in a slight decrease in raw coverage compared with optimizers solely focused on coverage, since node positions that yield strong anchor geometry are not necessarily optimal for coverage. Blockchain introduces latency (consensus rounds) and gateway workload; however, since it operates on an epoch-by-epoch basis rather than on a per-packet basis, its impact on real-time sensing should be minimal, although this needs to be empirically verified. These expectations are precisely what the protocol in Section VI aims to test.

Practical Considerations: As indicated in [4], successful field deployments depend on accurate calibration, antenna geometry, and a well-defined range of validity. Validating simulated models of range error with testbed data is also crucial. Energy-aware designs such as decentralized energy swapping [5] could potentially complement placement strategies by redistributing load among nodes.

Limitations: Simplifications made regarding disk sensing models and static obstacles do not reflect real-world complexities; additionally, the framework assumes that trustworthy gateways exist for initial identity registration; finally, results reported in [6]–[10] were generated using differing simulators and parameterizations, rendering comparisons between papers suggestive but not definitive.

VIII. CONCLUSION AND FUTURE WORK

In conclusion, this paper describes a framework that simultaneously addresses node placement and trustworthy localization as a single optimization problem supported by a permissioned blockchain operated by edge gateways. This framework combines a trust-aware hybrid PSO, a trust-weighted position estimator, and a compact smart contract; it is also accompanied by a reproducible evaluation protocol.

Because no experimental results are reported in this paper, confirmation of expected benefits awaits future implementation of the protocol. Future work will include conducting experiments based on the protocol outlined above and reporting results; expanding the model to encompass three-dimensional spaces and underwater environments; developing methods for learning trust weights using federated learning; investigating lightweight or post-quantum signature schemes for constrained nodes; and evaluating performance on a UAV-assisted testbed.

REFERENCES

- [1] H. Aldawsari, "A blockchain-based approach for secure energy-efficient IoT-based wireless sensor networks for smart cities," *Alexandria Eng. J.*, vol. 126, pp. 1–7, 2025.
- [2] "Secure wireless sensor networks with blockchain-based trust management," in *Proc. 4th Int. Conf. Internet of Things, Commun. Intell. Technol. (IoTICIT 2025)*, J. Dong and T. Zheng, Eds. Springer, 2025.
- [3] B. Kaur, D. Prashar, L. Masic, A. Almogren, A. U. Rehman, A. Altameem, and S. Hussein, "Enhancing the reliability and accuracy of wireless sensor networks using a deep learning and blockchain approach with DV-HOP algorithm for DDoS mitigation and node localization," *EURASIP J. Wireless Commun. Netw.*, vol. 2025, Art. no. 46, 2025, doi: 10.1186/s13638-025-02465-w.
- [4] "Blockchain-enabled uncertainty-aware passive Wi-Fi localization for secure critical infrastructure sensor networks," *PubMed Central*, PMC13165744, 2026.
- [5] U. Draz, T. Ali, S. Yasin, M. Hijji, M. Ayaz, and E.-H. M. Aggoune, "Decentralized energy swapping for sustainable wireless sensor networks using blockchain technology," *Mathematics*, vol. 13, no. 3, Art. no. 395, 2025, doi: 10.3390/math13030395.
- [6] "Dynamic deployment technique for WSN using improved particle swarm optimization to optimize blanket coverage," *IETE J. Res.*, 2026, doi: 10.1080/03772063.2026.2642875.
- [7] "Coverage optimization and node minimization in WSNs: An enhanced hybrid PSO approach with spatial position encoding," *Sci. Rep.*, 2025, doi: 10.1038/s41598-025-09849-4.
- [8] "Efficient node deployment for enhancing coverage and connectivity in wireless sensor networks," *Sci. Rep.*, 2025, doi: 10.1038/s41598-025-14252-0.
- [9] R. Mishra, S. K. Jha, et al., "nodeWSNsec: A hybrid metaheuristic approach for reliable security and node deployment in WSNs," arXiv:2508.16619, 2025.
- [10] "Sensor node deployment optimization for continuous coverage in WSNs," 2025, PMC12196809; PubMed 40573507.
- [11] S. Xing and R. Song, "Boundary node detection and localization optimization in anisotropic networks," *Ad Hoc Netw.*, vol. 188, Art. no. 104228, 2026, doi: 10.1016/j.adhoc.2026.104228.
- [12] "Dynamic coverage optimization in wireless sensor networks via hybrid Q-learning swarm intelligence," *Appl. Soft Comput.*, 2026, doi: 10.1016/j.asoc.2026.115878.
- [13] B. N. Sudheer and K. Sujatha, "Blockchain based randomized trusted node routing model in wireless sensor network," *Discover Comput.*, vol. 29, Art. no. 385, 2026, doi: 10.1007/s10791-026-10230-w.
- [14] S. Bhattacharya, "Implementing zero trust architecture (ZTA) in IoT enabled (wireless sensor nodes) WSNs creating a digital blockchain of things," *Discover Netw.*, vol. 2, Art. no. 16, 2026, doi: 10.1007/s44354-026-00030-5.
- [15] M. Alharbi et al., "Blockchain-driven trust management and AI computing for sensor networks optimization," *Sci. Rep.*, 2026, doi: 10.1038/s41598-026-41302-y.
- [16] S. Tomic, M. Beko, D. Vukobratovic, and S. Krco, "Voting mechanism for trustworthy localization in wireless sensor networks," *Front. Sensors*, vol. 7, Art. no. 1730414, 2026, doi: 10.3389/fens.2026.1730414.