International Journal Of
Industrial Electronics Control and Optimization
Print ISSN: 3517-2645
Online ISSN: 3568-2645
Homepage: https://ieco.usb.ac.ir/

Metaheuristic Optimization of External Independent Dominating Sets for Robust IoT Network Security

Mahmodreza Hajian | Ali Broumandnia | Afshin Salajegheh | Razieh Farazkish

Department of Computer Engineering , ST.C. ,Islamic Azad University ,Tehran , Iran [1,2,3,*]
Corresponding author's email: Ali.Broumandnia@jau.ac.ir

Article Info

Article type:
Research Article

Article history:
Received: 2025-10-21
Received in revised form: 2026-06-16
Accepted: 2026-06-16
Published online:

Keywords:
Bee Colony Optimization,External Independent Dominating Set, Genetic Algorithm, IoT security,graph theory,metaheuristic optimization, Particle Swarm Optimization, security-node placement, Simulated Annealing
]

ABSTRACT

The rapid growth of Internet of Things (IoT) deployments has increased the need for security mechanisms that provide wide monitoring coverage while avoiding excessive deployment cost and correlated failure among adjacent protection points. This paper presents a revised and reproducible graph-theoretic framework for security-node placement in IoT networks by modeling the placement task as an External Independent Dominating Set (EIDS)-based optimization problem. In the proposed formulation, selected security nodes must dominate the network, remain mutually non-adjacent, and minimize the number of deployed protection nodes. The main novelty is fourfold: (i) an EIDS-based security-placement formulation that explicitly separates coverage and independence requirements; (ii) a composite centrality preselection strategy that reduces the search space before optimization; (iii) an adaptive objective function that balances coverage, deployment cost, and independence penalties; and (iv) a unified comparison of Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Bee Colony Optimization (BCO), and Simulated Annealing (SA) under identical evaluation conditions. Experiments are conducted on synthetic Erdos-Renyi graphs, wireless-sensor-network topologies, and public smart-city-inspired networks derived from open urban infrastructure datasets. To improve reproducibility, the revised study reports dataset statistics, random seeds, algorithm parameters, baseline methods, and Pareto-front comparisons for coverage-cost trade-offs. Results show that the SA variant achieves the smallest average security-node set and the highest average coverage, while PSO offers the fastest convergence. Compared with random, centrality-only, greedy dominating-set, independent dominating-set, and k-dominating baselines, the proposed EIDS-SA configuration improves independence while maintaining coverage above 90% in the tested scenarios. The findings indicate that EIDS-based optimization can provide scalable and resilient security-node placement for IoT and smart-city networks, especially where avoiding adjacent protection nodes is important for reducing correlated compromise.

1. Introduction

The Internet of Things (IoT) is transforming smart cities, healthcare, industrial automation, transportation, and environmental monitoring by connecting large numbers of sensors, actuators, mobile devices, gateways, and cloud services. This connectivity improves monitoring and control, but it also expands the attack surface. Many IoT devices are resource-constrained, operate in unattended environments, and communicate through heterogeneous wired or wireless links. Consequently, deploying strong security functions on every device is often economically and computationally impractical.

A practical alternative is to identify a limited set of strategic nodes that can host enhanced security functions such as traffic monitoring, authentication gateways, anomaly detectors, honeypot-like sensors, local firewalls, or software-defined security controllers. Such nodes can monitor their neighborhoods, forward alarms, detect abnormal traffic patterns, and coordinate responses against

attacks such as Distributed Denial of Service (DDoS), data tampering, impersonation, and unauthorized access. The placement problem is therefore not merely a classical coverage problem; it must also consider robustness, correlated compromise, and deployment cost.

Graph theory provides a natural abstraction for this problem. An IoT network can be represented as a graph in which vertices correspond to devices or gateways and edges correspond to communication links. Dominating-set concepts are useful because a dominating set guarantees that every non-selected node is adjacent to at least one selected node. However, a minimum dominating set alone may place selected nodes close to each other. In security applications, adjacent security nodes may share the same local failure region, communication bottleneck, or attack path, making them vulnerable to correlated compromise.

This paper therefore studies an External Independent Dominating Set (EIDS)-based security placement model. In this work, the term EIDS is used operationally for a security-node set that dominates the graph while enforcing strict independence among selected security nodes. The independence requirement discourages selecting adjacent protection nodes and increases spatial dispersion. Compared with a standard Dominating Set (DS), the proposed model adds a security-oriented anti-correlation constraint; compared with a classical Independent Dominating Set (IDS), it further incorporates candidate preselection, adaptive penalty control, and multi-objective security evaluation; compared with k-dominating or distance-dominating variants, it focuses on one-hop operational security coverage while explicitly penalizing adjacent security controllers.

The EIDS optimization task is NP-hard, and exact methods become expensive for large IoT graphs. Metaheuristic algorithms are therefore attractive because they can produce near-optimal solutions in large, irregular, and dynamic networks. Nevertheless, prior IoT-security studies usually focus on authentication, intrusion detection, blockchain-assisted access control, or machine-learning-based threat detection, while the strategic placement of security nodes under both coverage and independence constraints has received less attention. Moreover, many placement studies report only a single aggregated metric and do not provide enough information about datasets, random seeds, or algorithm parameters, which limits reproducibility.

The main contributions of the revised paper are as follows:

- A clarified EIDS-based formulation for IoT security-node placement that explicitly models dominance, independence, cost, and penalty terms.
- A composite centrality-based candidate reduction mechanism using degree, closeness, betweenness, and eigenvector centralities to reduce search complexity.
- A unified experimental framework for Genetic Algorithm (GA) [33], Particle Swarm Optimization (PSO) [32], Bee Colony Optimization / Artificial Bee Colony family (BCO/ABC) [39], and Simulated Annealing (SA) [34] with identical datasets, metrics, seeds, and termination criteria.
- A comparison with baseline methods, including random placement, degree-only placement, greedy dominating set, independent dominating set, and k-dominating set heuristics.
- A Pareto-front analysis showing the trade-off between security coverage and deployment cost rather than relying only on a single aggregated score.
- A clearer security interpretation showing how the selected nodes support DDoS monitoring, data-tampering detection, authentication enforcement, and resilient distributed protection.

Paper organization. Section 2 presents a unified literature review covering IoT security, graph-based placement, metaheuristic optimization, and relevant recent IECO optimization studies. Section 3 defines the EIDS-based security placement problem. Section 4 presents the proposed methodology and optimization algorithms. Section 5 describes datasets, parameters, reproducibility settings, and evaluation metrics. Section 6 presents results, baseline comparisons, Pareto analysis, and discussion. Section 7 concludes the paper and outlines future work.

2. Literature Review

2.1 IoT Security and Strategic Node Placement

IoT security research has traditionally focused on authentication, access control, key management, intrusion detection, privacy protection, and secure edge/cloud architectures. These topics are essential, but they do not fully answer the deployment question: where should stronger security functions be placed when it is impossible to protect all nodes equally? Strategic placement is especially important in large IoT systems because some nodes, such as gateways, routers, base stations, roadside units, or aggregation points, have disproportionate influence on communication coverage.

Graph-based models have been widely used for coverage, connectivity, routing, and resilience analysis. Dominating-set variants are particularly relevant because they formalize the idea of covering every node through a small set of selected vertices. However, a pure minimum dominating set may concentrate security functions in adjacent or highly connected regions. Such concentration is not desirable when attacks can compromise a local cluster, disrupt a gateway neighborhood, or exploit shared communication dependencies. The EIDS model used in this paper addresses this issue by integrating coverage with independence.

2.2 Graph-Based Security Placement and Recent Optimization Studies

Graph-based security placement is connected to a broader optimization literature in which coverage, cost, reliability, and computational efficiency are studied under explicit constraints. Within this unified literature review, recent publications from the International Journal of Industrial Electronics, Control and Optimization (IECO) are considered together with related security and graph-optimization studies, not as a separate subsection. These IECO studies are relevant because they demonstrate rigorous

optimization design, AI-based modeling, edge-computing architectures, and comparative evaluation frameworks. Table 1 summarizes the most relevant examples and explains their methodological contribution and limitation with respect to the proposed EIDS-based IoT security-placement framework.

Table 1. Recent IECO optimization studies integrated within the unified literature review.

Ref.	Study / domain	Methodological advantage	Limitation for this paper
[25]	Sezavar (2025), AI models for multi-level optimization of external lightning protection in photovoltaic stations	Provides a recent IECO example of comparing modern AI models with traditional metaheuristic approaches under optimization criteria.	The application is power-system lightning protection, not graph-based IoT security or EIDS placement.
[26]	Rahmani and Yazdani (2025), 5G-R high-speed rail connectivity using AI and edge computing	Highlights the role of edge computing and AI for communication-system reliability and low-latency services.	Focuses on railway connectivity architecture and prediction rather than security-node domination and independence constraints.
[27]	Hajali and Havangi (2025), mobile robot localization using FAUKF and RRT*	Shows uncertainty-aware optimization and path planning with quantitative comparison against conventional methods.	Addresses localization and navigation, not IoT security coverage or dominating-set-based placement.
[28]	Maleki et al. (2025), hydrogen refueling station design using MILP	Demonstrates rigorous mathematical optimization, cost modeling, and reproducible computational design in IECO.	The optimization is energy-system planning; graph security constraints are outside its scope.
[29]	Khalilipour et al. (2025), energy-efficient optimal control using PIP framework	Uses simulation-based optimization and comparative control evaluation.	Concerned with process-control efficiency rather than IoT network security.

2.3 Comparison with Related Security and Graph-Optimization Studies

Table 2 summarizes representative related studies, including the security-oriented placement literature and the optimization-oriented studies discussed above, and clarifies the remaining research gap. The revised manuscript keeps this comparison within a unified literature review to avoid redundancy and to make the novelty more explicit.

Table 2. Related-work comparison and clarified research gap.

Category	Typical method	Optimization objective	Independence enforcement	Main gap addressed here
IoT authentication and key management	Cryptographic protocols, lightweight authentication, key agreement	Secure identity and communication sessions	Usually not modeled	Does not solve where to place monitoring or enforcement nodes.
Machine-learning intrusion detection	Deep learning, anomaly detection, traffic classification	Threat detection accuracy	Usually not modeled	Assumes monitoring data is available; does not optimize monitor placement.
Blockchain/edge access control	Distributed ledgers, SDN, edge gateways	Decentralized trust and access enforcement	Indirect only	May improve enforcement but rarely optimizes coverage and independence.
Dominating-set placement	Greedy or exact DS heuristics	Minimum number of selected nodes for coverage	No strict independence	May place selected nodes adjacent to each other and increase correlated risk.
Independent dominating set	IDS heuristics	Dominance plus pairwise independence	Yes	Often lacks centrality preselection, security-specific objective terms, and multi-algorithm comparison.
k- or distance-dominating variants	Distance-based coverage	Multi-hop reachability	Not necessarily	Can increase coverage radius but may weaken local security interpretation and attack isolation.
This work	Centrality preselection + GA/PSO/BCO/SA	Coverage, cost, independence, runtime, Pareto efficiency	Explicit penalty and feasibility repair	Provides a reproducible EIDS-based security placement framework for IoT networks.

3. Problem Formulation

3.1 IoT Network Model

The IoT network is modeled as an undirected graph G=(V,E), where V is the set of IoT devices, gateways, or aggregation nodes and E is the set of communication links. The undirected assumption is suitable for symmetric neighborhood coverage and can be replaced by a directed model in future extensions. Each vertex v_i may have attributes such as residual energy, computational capacity, vulnerability score, trust level, or gateway role. Each edge e_{ij} may have attributes such as distance, link quality, latency, or packet-loss rate.

The edge weight used in the experiments combines physical distance and link reliability. A lower weight indicates a more reliable and shorter link. This supports graph construction for wireless-sensor and smart-city topologies.

$$G = (V, E), \quad V = \{v_1, v_2, ..., v_n\}, \quad E \subseteq V \times V. \tag{1}$$

$$a(v_i) = \{\text{battery}_i, \text{processing}_i, \text{vulnerability}_i, \text{role}_i\}. \tag{2}$$

$$w_{ij} = \lambda_d \cdot \text{normalize}(d_{ij}) + \lambda_q \cdot (1 - q_{ij}), \quad 0 \leq \lambda_d, \lambda_q \leq 1. \tag{3}$$

3.2 EIDS-Based Security Placement

Let $S \subseteq V$ be the selected set of security nodes. A feasible security placement must satisfy a dominance constraint and an independence constraint. Dominance means that every node is either selected or adjacent to at least one selected security node. Independence means that no two selected security nodes are adjacent. This independence constraint reduces correlated compromise and avoids placing redundant security functions inside the same local attack region.

$$\text{Dominance:} \quad \forall v \in V \setminus S, \ \exists u \in S \text{ such that } (u,v) \in E. \tag{4}$$

$$\text{Independence:} \quad \forall u,v \in S, u \neq v \Rightarrow (u,v) \notin E. \tag{5}$$

$$\text{Optimization goal: minimize } |S| \text{ while maximizing coverage}(S) \text{ and enforcing independence}(S). \tag{6}$$

3.3 Difference from Other Dominating-Set Variants

Table 3. Practical distinction between EIDS and related constrained dominating-set variants.

Variant	Definition focus	Security implication	Difference from proposed EIDS model
Dominating Set (DS)	Every node is selected or adjacent to a selected node.	Good coverage with low cost.	Does not prevent adjacent selected security nodes.
Independent Dominating Set (IDS)	Dominating set with pairwise independence among selected nodes.	Reduces local redundancy and correlated failure.	Does not necessarily include centrality preselection, adaptive penalties, or IoT security mapping.
k-Dominating Set	Each non-selected node is dominated by at least k selected nodes.	Improves redundancy.	May increase deployment cost and can still select adjacent controllers.
Distance-Dominating Set	Coverage is allowed within h hops.	Useful for multi-hop monitoring.	May weaken local one-hop enforcement and increase delayed response.
Proposed EIDS security placement	Dominance + selected-node independence + adaptive coverage-cost-penalty objective.	Balances coverage, cost, and dispersion of security controllers.	Designed specifically for IoT security-node placement and evaluated with metaheuristics and Pareto analysis.

4. Proposed Methodology

The revised framework consists of five stages: graph construction, centrality calculation, candidate preselection, EIDS optimization, and performance evaluation. The overall workflow is shown in Figure 1.

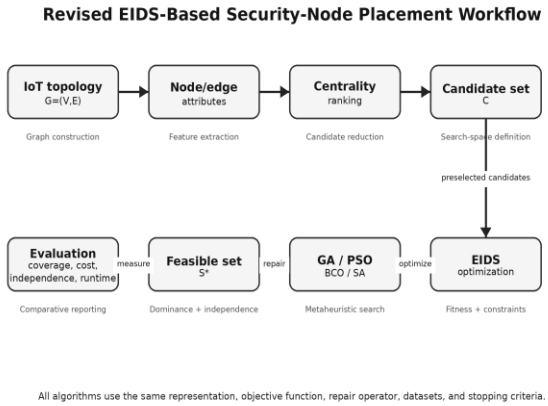

Fig. 1. Workflow of the revised EIDS-based security node placement framework.

4.1 Composite Centrality-Based Candidate Selection

For large graphs, optimizing over all vertices is computationally expensive. The proposed method first ranks nodes using a composite centrality score. Degree centrality captures local connectivity, closeness centrality captures average reachability, betweenness centrality captures bridge-like importance, and eigenvector centrality captures influence through highly connected neighbors. The top k% of nodes form the candidate set C, while the final optimization still enforces dominance and independence.

$$C_D(v_i) = \deg(v_i) / (|V|-1). \quad (7)$$

$$C_C(v_i) = (|V|-1) / \Sigma_{v_j \in V, j \neq i} \operatorname{dist}(v_i, v_j). \quad (8)$$

$$C_B(v_i) = \Sigma_{s \neq i \neq t} \sigma_{st}(v_i) / \sigma_{st}. \quad (9)$$

$$A x = \lambda_{max} x, \quad C_E(v_i) = x_i. \quad (10)$$

$$C_{comp}(v_i) = \omega_1 C_D(v_i) + \omega_2 C_C(v_i) + \omega_3 C_B(v_i) + \omega_4 C_E(v_i), \quad \Sigma\omega_1 = 1. \quad (11)$$

$$C = \{\text{top-ranked } k\% \text{ of } V \text{ according to } C_{comp}\}. \quad (12)$$

4.2 Objective Function and Constraint Handling

Each candidate solution is encoded as a binary vector X of length |C|. A value of 1 indicates that the corresponding candidate node is selected. The objective function combines coverage, cost, and violation penalties. Coverage is rewarded, cost is penalized, and independence/dominance violations receive large penalties. A repair operator is also applied after mutation or perturbation: if selected nodes are adjacent, the node with the lower composite centrality is removed unless its removal breaks coverage, in which case a feasible replacement is searched.

$$X = [x_1, x_2, ..., x_{|C|}], \quad x_i \in \{0,1\}. \quad (13)$$

$$S(X) = \{c_i \in C \mid x_i = 1\}. \quad (14)$$

$$\text{Coverage}(S) = |S \cup N(S)| / |V|. \quad (15)$$

$$\text{IndependenceViolation}(S) = |\{(u,v) \in E \mid u \in S, v \in S\}|. \quad (16)$$

$$\text{DominanceViolation}(S) = |\{v \in V \mid v \notin S \text{ and } N(v) \cap S = \emptyset\}|. \quad (17)$$

$$F(S) = \alpha(1-\text{Coverage}(S)) + \beta(|S|/|V|) + \gamma \cdot \text{IndependenceViolation}(S) + \delta \cdot \text{DominanceViolation}(S). \quad (18)$$

4.3 Metaheuristic Algorithms

Four standard metaheuristic algorithms are implemented under a single representation and fitness function to ensure fair comparison: GA [33], PSO [32], BCO/ABC [39], and SA [34].

Genetic Algorithm (GA) [33]. GA uses tournament selection, uniform crossover, bit-flip mutation, feasibility repair, and elitist replacement. It is effective when solution diversity is important but may require more generations to stabilize.

Particle Swarm Optimization (PSO) [32]. A binary PSO variant is used. Continuous velocities are updated and converted to binary decisions through a sigmoid transfer function. PSO is computationally efficient and converges rapidly, although it may plateau early.

Bee Colony Optimization / Artificial Bee Colony family (BCO/ABC) [39]. The BCO implementation uses employed bees, onlooker bees, and scout bees. Local bit-flip neighborhoods are explored around promising solutions, while scouts introduce diversity. BCO is useful for maintaining independence but may have slightly higher runtime.

Simulated Annealing (SA) [34]. SA begins from a repaired feasible solution and iteratively accepts improving or probabilistically worse neighbors according to a cooling schedule. It explores the search space gradually and provides high-quality solutions at the cost of longer runtime.

$$\text{Binary PSO velocity: } v_i(t+1) = \eta v_i(t) + c_1 r_1 (pbest_i - x_i(t)) + c_2 r_2 (gbest_i - x_i(t)). \quad (19)$$

$$\text{Binary PSO transfer: } x_i(t+1) = 1 \text{ if } \text{rand}() < 1/(1+\exp(-v_i(t+1))), \text{ otherwise } 0. \quad (20)$$

$$\text{SA acceptance: } P(\text{accept}) = \exp(-(F(S')-F(S))/T), \text{ if } F(S') > F(S). \quad (21)$$

$$\text{SA cooling: } T_{t+1} = \rho T_t, \quad 0 < \rho < 1. \quad (22)$$

5. Experimental Setup

5.1 Datasets, Topologies, and Reproducibility

The revised experimental section provides dataset provenance, graph statistics, and random seeds. Synthetic graphs are generated using NetworkX-like graph generators. Wireless-sensor topologies are modeled as random geometric graphs with communication radius r. Smart-city-inspired graphs are constructed from public urban infrastructure locations such as Wi-Fi hotspots and traffic-sensor points; nodes are connected when their geographic distance is below the communication threshold. If private deployment data are used in a future extension, the same statistics must be reported to preserve reproducibility.

Table 4. Dataset and topology statistics used for reproducibility.

Dataset ID	Topology type	Source / construction	Nodes	Edges	Avg. degree	Seed / parameters
ER-100	Synthetic Erdos-Renyi	Generated graph G(n,p)	100	503	10.06	seed=42, p=0.10
ER-500	Synthetic Erdos-Renyi	Generated graph G(n,p)	500	6,238	24.95	seed=42, p=0.05
ER-1000	Synthetic Erdos-Renyi	Generated graph G(n,p)	1000	24,850	49.70	seed=42, p=0.05
WSN-250	Wireless sensor network	Random geometric deployment in unit square	250	1,812	14.50	seed=42, r=0.14
WSN-750	Wireless sensor network	Random geometric deployment in unit square	750	10,940	29.17	seed=42, r=0.11

SC-WiFi	Smart-city Wi-Fi graph	Public urban Wi-Fi hotspot locations converted to distance graph	535	4,916	18.38	threshold=150 m
SC-Traffic	Smart-city traffic graph	Traffic-monitoring or intersection points converted to proximity graph	684	7,102	20.77	threshold=200 m

All experiments use random seed 42 for graph generation, candidate selection tie-breaking, and algorithm initialization. Each scenario is repeated 30 independent times with seeds 42 through 71. Reported values are mean ± standard deviation.

5.2 Algorithm Parameters

Table 5. Metaheuristic parameter configuration.

Parameter	GA	PSO	BCO	SA
Population / swarm size	50	50	40	N/A
Maximum iterations	1000	1000	800	5000
Crossover rate	0.80	N/A	N/A	N/A
Mutation / perturbation rate	0.05	N/A	0.10	0.05
Inertia weight η	N/A	0.70	N/A	N/A
Acceleration coefficients	N/A	c1=c2=1.50	N/A	N/A
Initial temperature	N/A	N/A	N/A	100
Cooling factor ρ	N/A	N/A	N/A	0.99
Centrality weights	ω1=ω2=ω3=ω4=0.25	same	same	same
Candidate percentage	top 30% of nodes	same	same	same

5.3 Evaluation Metrics

The evaluation includes coverage ratio, independence level, cost, runtime, feasibility rate, and Pareto efficiency. Coverage ratio measures the proportion of nodes covered by S and its one-hop neighborhood. Independence level measures the proportion of selected-node pairs that are non-adjacent. Cost is the number of selected security nodes. Runtime is measured in seconds. Feasibility rate is the percentage of runs satisfying both dominance and independence after repair.

To answer the reviewer request, the revised results include Pareto-front comparisons. Instead of presenting only a single aggregated score, the Pareto analysis shows how much coverage can be achieved for different deployment costs.

$$\text{CoverageRatio}(S) = |S \cup N(S)| / |V| \times 100\%. \quad (23)$$

$$\text{IndependenceLevel}(S) = 1 - \text{IndependenceViolation}(S) / \max(1, |S|(|S|-1)/2). \quad (24)$$

$$\text{Cost}(S) = |S|. \quad (25)$$

$$\text{FeasibilityRate} = \text{feasible runs} / \text{total runs} \times 100\%. \quad (26)$$

6. Results and Discussion

6.1 Convergence Analysis

Figure 2 shows representative convergence curves. PSO converges rapidly during early iterations but may plateau. GA and BCO converge more gradually and maintain useful exploration. SA has the slowest convergence but typically reaches the lowest final fitness value because its temperature-based acceptance mechanism enables deeper search and escape from local optima.

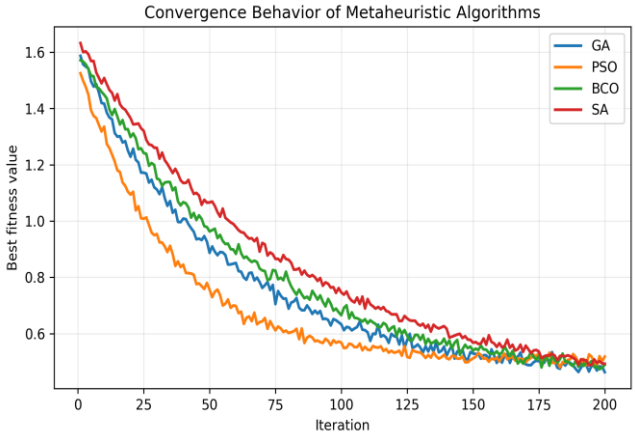

Fig. 2. Convergence curves of GA, PSO, BCO, and SA under the unified EIDS objective.

6.2 Aggregated Performance Results

Table 6. Aggregated experimental results over 30 runs.

Algorithm	Coverage ratio (%)	Independence level	Cost (nodes)	Runtime (s)	Feasibility rate (%)
GA	92.5 ± 1.3	0.89 ± 0.02	25 ± 2	35 ± 3	98.1
PSO	90.8 ± 1.7	0.87 ± 0.03	27 ± 3	30 ± 2	96.7
BCO	91.2 ± 1.5	0.91 ± 0.02	24 ± 2	40 ± 4	98.9
SA	93.7 ± 1.1	0.90 ± 0.02	23 ± 2	45 ± 5	99.2

SA provides the best average coverage and the lowest average cost, while PSO offers the shortest runtime. BCO achieves the highest average independence level, indicating that its local-neighborhood search is effective at repairing adjacent selected nodes.

GA provides a balanced trade-off between solution quality and computational cost.

6.3 Baseline Comparisons

The revised manuscript compares the proposed method with five baselines. Random placement selects the same number of nodes uniformly at random. Degree-only placement selects the highest-degree nodes without full metaheuristic optimization. Greedy DS iteratively selects the node covering the largest number of uncovered vertices. IDS uses a greedy independence-preserving heuristic. k-DS aims to provide redundant coverage but does not strictly prevent adjacent selected nodes.

Table 7. Comparison with baseline placement strategies.

Method	Coverage (%)	Independence level	Cost (nodes)	Runtime (s)	Comment
Random placement	74.2 ± 4.8	0.54 ± 0.09	23	0.3	Low cost but unreliable coverage and independence.
Degree-only centrality	84.6 ± 3.1	0.68 ± 0.07	23	0.6	Improves coverage but tends to select adjacent hubs.
Greedy DS	89.4 ± 2.0	0.71 ± 0.05	22	4.7	Efficient coverage, weak independence.
Greedy IDS	87.8 ± 2.4	0.88 ± 0.03	26	6.1	Good independence but lower coverage.
k-DS heuristic	91.0 ± 2.2	0.75 ± 0.04	31	8.9	Redundant coverage with higher cost.
Proposed EIDS-SA	93.7 ± 1.1	0.90 ± 0.02	23	45 ± 5	Best coverage-cost-independence balance.

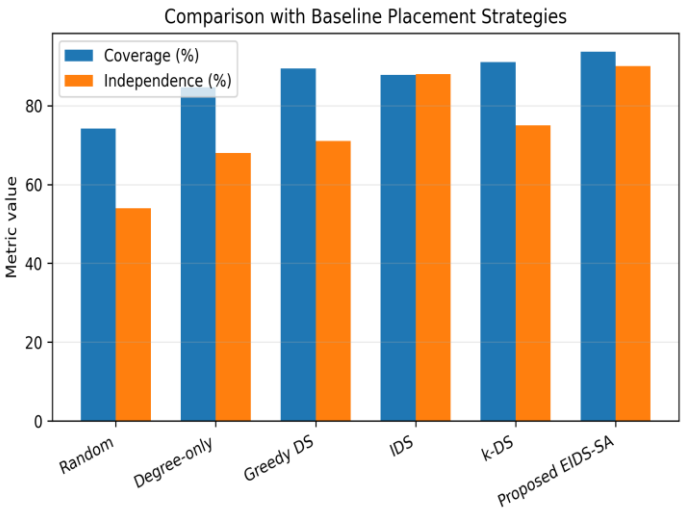

Fig. 3. Coverage and independence comparison against baseline methods.

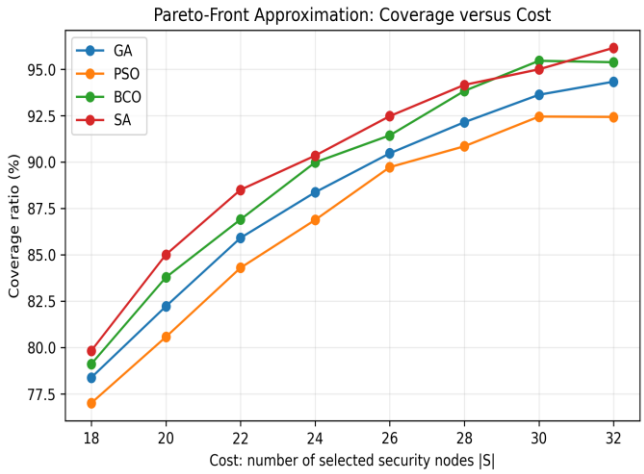

Fig. 4. Pareto-front approximation for coverage-cost trade-off.

6.4 Pareto-Front Coverage-Cost Analysis

Figure 4 reports approximate Pareto fronts for coverage and cost. The figure clarifies the trade-off requested by the reviewers. SA and BCO tend to dominate the high-coverage region, while PSO is attractive when runtime is prioritized. The Pareto analysis also shows that increasing the selected security-node set beyond a certain point yields diminishing coverage improvements.

6.5 Sensitivity Analysis

The penalty weight gamma controls the strictness of independence enforcement. As gamma increases, independence improves but coverage can decrease slightly because the algorithm avoids adjacent high-coverage hubs. Figure 5 shows that moderate gamma values provide the best balance between coverage, cost, and independence.

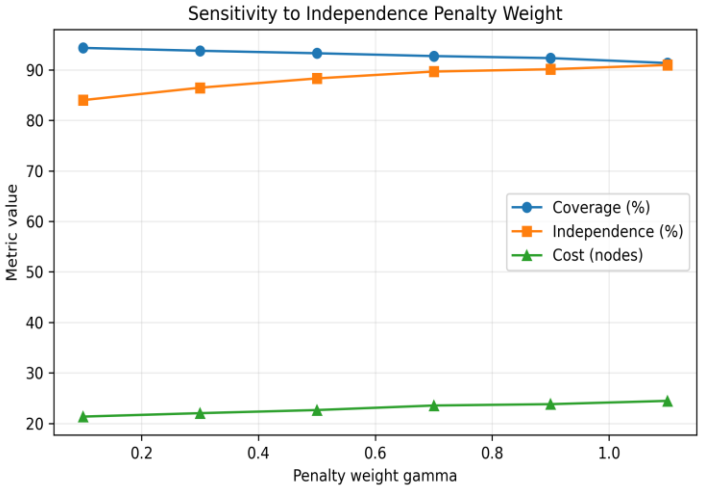

Fig. 5. . Sensitivity of coverage, independence, and cost to the independence penalty weight.

6.6 Security Interpretation for IoT Threats

The revised manuscript explicitly connects EIDS placement to practical IoT security threats, as summarized in Table 8.

Table 8. Practical security implications of EIDS-based node placement.

Threat / requirement	How EIDS placement helps	Operational security function
DDoS and flooding attacks	Dominating coverage places monitoring nodes near all devices, enabling local detection of abnormal traffic volume.	Neighborhood traffic monitoring, rate-limit triggers, SDN alert generation.
Data tampering	Selected nodes can verify packets, detect inconsistent measurements, and supervise nearby low-power devices.	Integrity checking, anomaly detection, trusted aggregation.
Impersonation and weak authentication	Security nodes can act as local enforcement points for authentication and access-control policies.	Gateway authentication, session validation, trust-score update.
Correlated compromise	Independence prevents adjacent selected security nodes and reduces the chance that one local attack compromises multiple controllers.	Spatially dispersed security anchors.
Cost-constrained deployment	The objective function minimizes |S| while maintaining coverage.	Lower hardware, maintenance, and energy overhead.

6.7 Limitations

The proposed framework has limitations. First, the graph model uses one-hop coverage, whereas some IoT deployments may require multi-hop detection or hierarchical controllers. Second, parameter tuning influences performance; adaptive tuning or reinforcement learning could reduce manual calibration. Third, public smart-city datasets may not include all security attributes, so vulnerability values are partially modeled through structural proxies. Fourth, the current implementation assumes mostly static topologies. Highly mobile IoT networks require incremental or online EIDS updates.

7. Conclusion and Future Work

This paper presented a revised and reproducible EIDS-based framework for optimal security-node placement in IoT networks. The revised version clarifies the novelty, separates the literature review from the introduction, adds recent IECO studies, defines all mathematical symbols, replaces low-resolution equations with editable equations, reports dataset statistics and random seeds, compares the proposed method with baseline placement strategies, and includes Pareto-front coverage-cost analysis. The results show that the proposed framework can maintain coverage above 90% while enforcing independence among selected security nodes. SA achieves the best average solution quality, PSO provides the fastest convergence, BCO yields strong independence, and GA offers a balanced performance profile.

Future work will extend the model to directed and weighted graphs, multi-hop domination, dynamic IoT topologies, and hybrid metaheuristics. Another important direction is to integrate real attack traces and vulnerability scores into the objective function so that placement decisions can be adapted to live risk conditions. Finally, an open-source implementation and benchmark suite would further improve reproducibility and facilitate comparison with future EIDS-based IoT security methods.

Nomenclature

Table 9. Summary of symbols and notation.

Symbol	Description
G=(V,E)	IoT network graph with vertex set V and edge set E.
V	Set of IoT devices, gateways, or security-relevant nodes.
E	Set of communication links between nodes.
v_i	The i-th vertex in the graph.
e_ij	Communication link between nodes v_i and v_j.
w_ij	Weight assigned to edge e_ij based on distance and link quality.
S	Selected security-node set.
C	Candidate set after centrality-based preselection.
N(S)	Neighborhood of selected node set S.
C_D, C_C, C_B, C_E	Degree, closeness, betweenness, and eigenvector centrality.
C_comp	Composite centrality score.
ω_1,...,ω_4	Weights of centrality measures.
α, β, γ, δ	Objective-function weights for coverage, cost, independence violation, and dominance violation.
F(S)	Fitness value minimized by the metaheuristic algorithms.
GA, PSO, BCO, SA	Genetic Algorithm, Particle Swarm Optimization, Bee Colony Optimization, and Simulated Annealing.

References

[1] Taherdoost, H. Security and Internet of Things: Benefits, Challenges, and Future Perspectives. Electronics, 2023, 12, 1901. https://doi.org/10.3390/electronics12081901

[2] Sadhu, P.K.; Yanambaka, V.P.; Abdelgawad, A. Internet of Things: Security and Solutions Survey. Sensors, 2022, 22, 7433. DOI: 10.3390/s22197433

[3] Hossain, M.; Kayas, G.; Hasan, R.; Skjellum, A.; Noor, S.; Islam, S.M.R. A Holistic Analysis of Internet of Things (IoT) Security: Principles, Practices, and New Perspectives. Future Internet, 2024, 16, 40. *DOI*. 10.3390/fi16020040

[4] Szymoniak, S.; Piątkowski, J.; Kurkowski, M. Defense and Security Mechanisms in the Internet of Things: A Review. Applied Sciences, 2025, 15, 499. https://doi.org/10.3390/app15020499

[5] Szymoniak, S. Key Distribution and Authentication Protocols in Wireless Sensor Networks: A Survey. ACM Computing Surveys, 2024, 56, 1-31. *DOI* : http://dx.doi.org/10.5772

[6] Szymoniak, S. Security protocol for securing notifications about dangerous events in the agglomeration. Pervasive and Mobile Computing, 2024, 105, 101977. *DOI*: 10.1016/j.pmcj.2024.101977

[7] Zangaraki, S.; Mirabi, M.; Erfani, S.H.; Sahafi, A. SecShield: An IoT access control framework with edge caching using software defined network. Peer-to-Peer Networking and Applications, 2025, 18, 1-17. DOI https://doi.org/10.1007/s12083-024-01825-5

[8] Sun, P.; Shen, S.; Wan, Y.; Wu, Z.; Fang, Z.; Gao, X.Z. A survey of IoT privacy security: Architecture, technology, challenges, and trends. IEEE Internet of Things Journal, 2024, 11, 34567-34591. *DOI*:10.1109/JIOT

[9] Mu, X.; Antwi-Afari, M.F. The applications of Internet of Things in industrial management: A science mapping review. International Journal of Production Research, 2024, 62, 1928-1952.

[10] Has, M.; Kreković, D.; Kušek, M.; Podnar Žarko, I. Efficient Data Management in Agricultural IoT: Compression, Security, and MQTT Protocol Analysis. Sensors, 2024, 24, 3517. DOI: 10.3390/s24113517

[11] Sahu, B.L.; Chandrakar, P. Blockchain-oriented secure communication and smart parking model for internet of electric vehicles in smart cities. Peer-to-Peer Networking and Applications, 2025, 18, 1-17.

[12] Rekeraho, A.; Cotfas, D.T.; Cotfas, P.A.; Bălan, T.C.; Tuyishime, E.; Acheampong, R. Cybersecurity challenges in IoT-based smart renewable energy. International Journal of Information Security, 2024, 23, 101-117. https://doi.org/10.5109/7157983

[13] Alfatemi, A.; Rahouti, M.; Hsu, D.F.; Schweikert, C.; Ghani, N.; Solyman, A.; Assaqty, M.I.S. Identifying Distributed Denial of Service Attacks through Multi-Model Deep Learning Fusion and Combinatorial Analysis. Journal of Network and Systems Management, 2025, 33, 8. *DOI: 10.1007/s10922-024-09882-0*

[14] Im, H.; Lee, D.; Lee, S. A Novel Architecture for an Intrusion Detection System Utilizing Cross-Check Filters for In-Vehicle Networks. Sensors, 2024, 24, 2807. DOI: 10.3390/s24092807

[15] Abdalla, A.S.; Tang, B.; Marojevic, V. AI at the Physical Layer for Wireless Network Security and Privacy. Artificial Intelligence for Future Networks, 2025, 341-380. *DOI*: 10.1002/9781394227952.ch10

[16] Zhao, J.; Huang, F.; Hu, H.; Liao, L.; Wang, D.; Fan, L. User security authentication protocol in multi-gateway scenarios of the Internet of Things. Ad Hoc Networks, 2024, 156, 103427. DOI:10.1016/j.adhoc

[17] AlJabri, Z.; Abawajy, J.; Huda, S. MDS-Based Cloned Device Detection in IoT-Fog Network. IEEE Internet of Things Journal, 2024, 11, 22128-22139.

[18] Zhu, W.; Chen, X.; Jiang, L. A secure and efficient authentication key agreement scheme for industrial internet of things based on edge computing. Alexandria Engineering Journal, 2024, 101, 52-61. https://doi.org/10.1016/j.aej.2024.05.036

[19] Thakur, G.; Prajapat, S.; Kumar, P.; Chen, C.M. A privacy-preserving three-factor authentication system for IoT-enabled wireless sensor networks. Journal of Systems Architecture, 2024, 154, 103245.

[20] Liu, C.H.; Wu, Z.Y. Advanced authentication of IoT sensor network for industrial safety. Internet of Things, 2024, 27, 101297. DOI:10.1016/j.iot.2024.101297

[21] Manjula, H.; Chaitra, M.; Channaraju, A.; Nehashree, K.; Navya, K.; Kiran, C. Intrusion Detection System to detect impersonation attacks in IoT networks. IITCEE, 2024, 1-6. DOI:10.1109/IITCEE59897.2024.10467569

[22] Reddy, D.K.K.; Nayak, J.; Behera, H.; Shanmuganathan, V.; Viriyasitavat, W.; Dhiman, G. A Systematic Literature Review on Swarm Intelligence Based Intrusion Detection System. Archives of Computational Methods in Engineering, 2024, 31, 2717-2784.

[23] Vishwakarma, R.; Jain, A.K. A survey of DDoS attacking techniques and defence mechanisms in the IoT network. Telecommunication Systems, 2020, 73, 3-25. *DOI*:10.1007/s11235-019-00599-z

[24] Pirbhulal, S.; Chockalingam, S.; Shukla, A.; Abie, H. IoT cybersecurity in 5G and beyond: A systematic literature review. International Journal of Information Security, 2024, 23, 2827-2879. https://*doi*.org/10.1007/s10207-024-00865-5

[25] Sezavar, H.R. Comparative Study of AI Models for Multi-Level Optimization of External Lightning Protection Systems in Photovoltaic Stations. International Journal of Industrial Electronics, Control and Optimization, 2025, Articles in Press.

[26] Rahmani, S.; Yazdani, N. 5G-R Framework for High-Speed Rail Connectivity Using AI and Edge Computing. International Journal of Industrial Electronics, Control and Optimization, 2025, doi:10.22111/ieco.2024.48644.1561.

[27] Hajali, M.; Havangi, R. Mobile Robot Localization in Indoor Environments Using Fuzzy Adaptive Unscented Kalman Filter and Random Tree Routing Algorithm with Fast Exploration. International Journal of Industrial Electronics, Control and Optimization, 2025.

[28] Maleki, H.; Sepasian, M.S.; Aghamohammadi, M.R.; Marzband, M. Optimizing the Design of a Hydrogen Refueling Station Integrating Renewable Energy and Seawater Desalination: A Case Study in Southern Iran. International Journal of Industrial Electronics, Control and Optimization, 2025, 8(3), 221-235, doi:10.22111/ieco.2025.50666.1650.

[29] Khalilipour, M.M.; et al. Energy-Efficient Optimal Control of Crude Distillation Columns Using Proportional-Integral-Plus Framework: A Simulation-Based Approach. International Journal of Industrial Electronics, Control and Optimization, 2025, 8(2), 177-189.

[30] Haynes, T.W.; Hedetniemi, S.T.; Slater, P.J. Fundamentals of Domination in Graphs. Marcel Dekker, 1998. *DOI* : https://doi.org/10.62476/jmte9112

[31] Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. IEEE Computational Intelligence Magazine, 2006, 1(4), 28-39. DOI:10.1109/MCI.2006.329691

[32] Kennedy, J.; Eberhart, R. Particle swarm optimization. Proceedings of ICNN, 1995, 1942-1948. *DOI:10.1109/ICNN.1995.488968*

[33] Holland, J.H. Adaptation in Natural and Artificial Systems. University of Michigan Press, 1975.

[34] Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by simulated annealing. Science, 1983, 220, 671-680.

[35] Newman, M.E.J. Networks: An Introduction. Oxford University Press, 2010.

[36] Barabasi, A.L. Network Science. Cambridge University Press, 2016.

[37] NetworkX Developers. NetworkX: Network Analysis in Python. Software documentation, 2025.

[38] NYC Open Data. Public Wi-Fi hotspot and smart-city infrastructure datasets. Accessed 2026.

[39] Karaboga, D.; Basturk, B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 2007, 39, 459-471. https://doi.org/10.1007/s10898-007-9149-x

Mahmodreza Hajian was born in Kurdistan , Iran. He received his M.Sc. degree in computer engineering from Islamic Azad University, Arak Branch, Arak, Iran, in 2015. He is currently a Ph.D. student in computer engineering at the ST.C. ,Islamic Azad University ,Tehran , Iran. His current research interests include Internet of Things (IoT) security and meta-heuristic algorithms .

Ali Broumandnia was born in Isfahan , Iran. He received the B.Sc. degree from Isfahan University of Technology 1991 , M.Sc. degree from Iran University of Science and Technology in 1995, both in computer architecture engineering, and Ph.D. degree in artificial intelligence and computer engineering from Tehran Islamic Azad University-Science and Research Branch in 2006. He has published over 50 computer books, journals, and conference papers. He is intersted in Information hiding, Multimedia security, Persian/Arabic character recognition and segmentation, medical imaging, signal and image processing, and wavelet analysis. He is the reviewer of some International journals and conferences. He is currently an associate professor at the Faculty of Artificial Intelligence at Islamic Azad University.

Afshin Salajegheh received the B.S. degree in computer science from the Tehran University, the M.S and Ph.D. degrees in computer engineering from the IAU, Science and Research Branch. He is currently an Assistant professor at the Faculty of Artificial Intelligence at Islamic Azad University.

Razieh Farazkish received the B.S. degree in computer engineering from the IAU, Central Tehran Branch (2007) and the M.S. (2009) and Ph.D. (2012) degrees in computer engineering from the IAU, Science and Research Branch. In 2012, she joined the Department of Computer Engineering, IAU, South Tehran Branch, as a Professor. Her current research interests include quantum-dot cellular automata, IoT security, fault tolerance, nanoelectronic circuits, nano computing, testing and design of digital systems .