Original Article
ISSN (Online): 3048-9245

ShodhAI: Journal of Artificial Intelligence
July-December 2025 2(2), 57–64

TRUST-AWARE ENVIRONMENT-SENSITIVE DEEP REINFORCEMENT LEARNING FOR ADAPTIVE CLUSTER HEAD SELECTION IN MOBILE AD HOC NETWORKS

Dr. Haridas S. [1]

[1] Associate Professor, Department of Computer Science, Government First Grade College, Tumkur, Karnataka, India

Received 21 July 2025
Accepted 25 August 2025
Published 30 September 2025

Corresponding Author

Dr. Haridas S., Haridas.S@outlook.com

DOI
10.29121/ShodhAI.v2.i2.2025.109

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ABSTRACT

Mobile Ad hoc Networks (MANETs) suffer from performance degradation as node count and mobility increase, since every node must independently manage routing without fixed infrastructure. Clustering is a well-established remedy: nodes are grouped under an elected cluster head (CH) that coordinates intra- and inter-cluster communication, which reduces routing-table size and control overhead. Most existing CH-selection schemes score candidate nodes using a fixed combination of residual energy, node degree, mobility and distance, but they treat trust as a secondary or static attribute and re-evaluate it too infrequently to catch adaptive insider attacks such as wormhole or grey-hole routing. This paper extends a reward-optimized Deep Q-Learning (RoDQL) clustering formulation by folding a continuously-updated, feedback-driven trust score into the state representation and reward function of the learning agent, alongside energy, mobility, node-degree and distance. A guard-node subsystem audits packet-forwarding behaviour updates each node's trust value on every routing cycle, and feeds this into the agent so that a node's suitability as CH is re-assessed as its behaviour changes, not only at cluster-formation time. The resulting framework, referred to here as Trust-Augmented RoDQL (TA-RoDQL), is presented together with its Markov Decision Process formulation, state/action/reward design, and a two-stage algorithm for cluster-head election and trust-triggered re-election. The expected benefits — more balanced cluster sizes, lower energy depletion at CHs, and faster detection and isolation of malicious nodes — are discussed against the underlying reward-shaping mechanics, and a simulation protocol is laid out so the approach can be validated empirically against classical Q-learning, DQN-only and static-weight clustering baselines.

Keywords: Manet, Clustering, Deep Reinforcement Learning, Cluster Head Selection, Trust Management, Network Security, Energy Efficiency

1. INTRODUCTION

Mobile Ad hoc Networks are self-organizing, infrastructure-less collections of wireless nodes that must jointly discover routes, forward packets and adapt to continuous topology change caused by node mobility. As the number of participating nodes grows, the cost of maintaining routing state at every node grows with it, and network throughput degrades. Clustering addresses this by partitioning the network into logical groups, each led by a cluster head that absorbs the bulk of

the coordination and forwarding responsibility on behalf of its members. A well-chosen CH extends network lifetime and keeps routing overhead low; a poorly chosen one is quickly drained of energy, and its failure fragments the cluster and destabilizes routing.

Classical CH-selection rules combine a handful of static criteria — residual energy, node degree, distance to neighbours and node mobility — into a weighted score, and elect the highest-scoring node. These rules are computationally cheap but brittle: the weights are fixed in advance and cannot adapt to a network whose traffic pattern, mobility profile or threat level changes over time. Reinforcement learning reframes CH selection as a sequential decision problem, in which an agent observes the network state and learns, through trial and reward, which nodes make durable cluster heads under the conditions actually being experienced. Deep Q-Networks (DQN) make this tractable at scale by approximating the action-value function with a neural network rather than an explicit Q-table, which removes the scalability ceiling that limits tabular Q-learning in large or high-mobility networks.

A gap remains in how trust is handled. Several DQN-based clustering schemes note that a CH is well placed to observe its members' forwarding behaviour and can therefore help detect attacks such as wormhole or grey-hole routing, but they typically treat trust as an outcome of clustering — computed after a CH is already active — rather than as an input the learning agent uses when it decides who to elect in the first place. This paper folds trust directly into the state and reward design of a reward-optimized DQN clustering agent (RoDQL), so that a node's trustworthiness, alongside its energy, degree, distance and mobility, shapes both its likelihood of being elected CH and the ongoing decision to keep or replace it. The resulting Trust-Augmented RoDQL (TA-RoDQL) framework is presented as an extension of, and builds directly on, the author's earlier RoDQL clustering formulation.

2. RELATED WORK

Prior clustering research for MANETs falls broadly into three groups. The first group optimizes cluster stability using bio-inspired or QoS-guided heuristics — for instance, Moth-Flame-based stabilized clustering elects a helper CH to shore up the primary CH before it deteriorates, and QoS-guided dynamic scheduling places workloads at near-optimal locations to bound latency. These methods are lightweight but rely on hand-tuned rules that do not adapt once deployed.

The second group applies reinforcement learning to routing and clustering. Early work relied on tabular Q-learning, which converges slowly once the action space grows with network size. Deep Q-Networks were subsequently used to steer routing in congested networks and to identify preferable links between node pairs, but these formulations generally assume a powerful central unit capable of computing actions for every node, which limits their applicability to fully distributed MANETs.

The third group targets security, most often by layering trust or certificate-based mechanisms on top of an already-formed cluster structure — for example, cluster-based certificate management with ECC-CRT key agreement, hexagonal clustered trust-based group key agreement, and multipart trust-based public key management. These schemes are effective at hardening an existing cluster but do not feed trust back into the CH-election decision itself, so a currently-trustworthy but soon-to-be-compromised node can still be re-elected before its misbehaviour is detected.

The gap this paper addresses sits at the intersection of the second and third groups: a distributed, DQN-based CH-election agent whose state representation and reward function are continuously informed by a live trust signal, so that election and re-election decisions reflect both resource suitability and behavioural trustworthiness at the same time.

3. PRELIMINARIES: DEEP Q-LEARNING FOR CLUSTERING

Reinforcement learning models CH selection as a Markov Decision Process (MDP) defined by the tuple (S, A, R, P), the sets of states, actions, rewards and state-transition probabilities respectively. An agent observes state s, takes action a, and receives reward r as the environment moves to state s′; over many episodes the agent learns an action-value function Q(s,a) that estimates the long-run return of taking action a in state s. Deep Q-Learning replaces the Q-table with a neural network Q(s,a;θ), which allows the same formulation to scale to the large state and action spaces that arise once energy, degree, distance, mobility and trust are all included as state variables. Training uses an experience-replay buffer that stores transition tuples e = (s, a, r, s′) and samples them in mini-batches, which decorrelates consecutive updates and stabilizes learning; a separate target network with parameters θ^{-}, periodically synchronized with the online network, further stabilizes the temporal-difference target.

4. PROPOSED METHODOLOGY: TA-RODQL

The proposed framework retains the core network-state variables of reward-optimized DQN clustering — residual energy, node relative degree, inter-node distance and node mobility — and adds a fifth, dynamically-maintained variable: node trust. A guard-node subsystem, distributed across the network, observes each node's packet-forwarding ratio, latency and route-consistency, and converts these observations into a trust score that is refreshed on every routing cycle rather than only at cluster-formation time.

4.1. STATE VARIABLES

1) Distance. The distance between two neighbouring nodes i and j is computed from their angular position (θ_2, θ_1) and radial coordinates (r_2, r_1):

$$d\_ij = \sqrt{(r_1^{\,2} + r_2^{\,2} - 2\, r_1\, r_2\, \cos(\theta_2 - \theta_1))}$$

2) Node mobility. Mobility is estimated from the average positional displacement of a node over a sliding time window T:

$$S\_ni = (1/T)\, \Sigma\_\{t=1..T\} \sqrt{((x(t)-x(t-1))^2 + (y(t)-y(t-1))^2 + (z(t)-z(t-1))^2)}$$

3) Residual energy. The residual energy of node i is the difference between its total available power and the power already spent on transmission:

$$RE\_ni = E\_P - E\_T$$

with total energy consumption for node i expressed as:

$$C\_E(i) = [\ T\_p \cdot (d\_s/d\_r) - R\_p \cdot (d\_s/d\_r)\ ] + i \cdot L_0$$

where T_p and R_p are the transmit and receive power draws, d_s is data size, d_r is data rate, and L_0 is the loss attributable to overhearing.

4) Node relative degree. The number of neighbours is expressed as a relative-degree measure:

$$D\_p = |\ d\_p - \sqrt{N}\ |, \quad d\_p = 1 \text{ if } 0 < D\_pq < R, \text{ else } 0$$

5) Trust score (new). Each guard node maintains a per-neighbour trust value that blends packet-forwarding success and routing consistency over a sliding observation window w:

$$Tr\_ni(t) = \alpha \cdot (P\_fwd / P\_recv) + (1 - \alpha) \cdot Tr\_ni(t - w)$$

where P_fwd / P_recv is the observed forwarding ratio in the current window, Tr_ni(t − w) is the node's trust value carried over from the previous window, and $\alpha \in (0,1)$ is a smoothing factor that controls how quickly the trust estimate reacts to new behaviour versus how much weight it gives to established history. A node whose trust value falls below a guard-defined threshold is flagged, and the network is notified so the node can be excluded from future CH elections and, where warranted, isolated from routing.

4.2. MDP FORMULATION

For each node N_i, i = {1, 2, …, n}, the MDP is defined as the tuple of states, actions, a transition model and a set of reward functions:

N_i = { States(S_i), Actions(A_i), Transition Model(T_i), Reward functions(R_i) }

State. At time t, the state of node N_i is the vector S_i,k = (RE_ni, Tr_ni, d_ni, Td_ni, D_ni) — residual energy, trust, distance, transmission distance and delay — which determines whether the node is offered as the next relay hop for a given packet.

Action. The action set is the space of residual-energy-indexed choices available to the agent at a given step:

$$A\_i = \{\ RE\_ni,k \mid RE\_ni,k \in \{0, \delta\_k, 2\delta\_k, \ldots, RE\_max,k\}\ \}$$

where δ_k is the step size. At every decision point the agent either selects a node from the candidate set as the next hop / CH, or terminates the path (i.e. declines to route through the candidate).

Transition model. The transition probability from state S_i,k to S_i+1,k is:

$$T\_i = S\_i \times A\_i \times S\_i \rightarrow [0,1]$$

Reward function. The reward signal ρ_i, which drives the CH-selection policy π_k, is defined as:

$$\rho\_i = S\_i \times A\_i \times S\_i \rightarrow R, \quad \rho\_i,k = \pi\_k(S\_i,k)$$

The reward is shaped to be jointly increasing in residual energy and trust, and decreasing in distance, mobility and delay, so that the agent is pushed toward electing nodes that are simultaneously energy-rich, well connected, stable, and behaviourally trustworthy — rather than optimizing any one criterion at the expense of the others. π_k is evaluated through the action-value function Q_k^π(S_i,k, ρ_i,k), and the optimal policy π_k* is the one whose value function dominates all others across every state, with optimal action-value Q_k*(S_i,k, ρ_i,k).

4.3. TRUST-TRIGGERED RE-ELECTION

Because trust is refreshed every routing cycle rather than only at initial cluster formation, a currently-serving CH whose trust score drops below threshold mid-cycle triggers an out-of-cycle re-election rather than waiting for the next scheduled cluster-formation round. This is the principal behavioural difference from the base RoDQL formulation, where CH suitability is effectively re-scored only at fixed intervals: TA-RoDQL allows the guard-node subsystem to interrupt that cycle whenever a live trust breach is detected, which shortens the exposure window during which a compromised or misbehaving node can continue acting as CH.

5. ALGORITHM

Algorithm 1 summarizes the reward-optimized deep Q-learning procedure used to train the CH-election policy, extended with the trust-refresh and re-election check.

Algorithm 1: Trust-Augmented Reward-Optimized Deep Q-Learning (TA-RoDQL)

Input: tuple of (A, S, R, P); guard-node trust stream Tr_ni(t)

Output: π(s) — elected cluster head, refreshed on trust breach

1) Set replay-buffer capacity to N.
2) Initialize action-value function Q with random weight θ.
3) Copy model parameters to build the target network: Q, $\theta^{-} = \theta$.
4) for episode = 1: M do
5) Observe initial state s_1 = (RE, Tr, d, Td, D) for all candidate nodes.
6) for step = 1 : T do
7) Choose action a_t with probability ε (explore), or the current best action with probability 1 − ε (exploit).
8) a_t = argmax_a Q*(φ(s_t), a; θ)
9) Execute a_t, transition to s_(t+1), receive reward r_(t+1).
10) Update Tr_ni for all guard-monitored neighbours from the current observation window.
11) if Tr_ni of the active CH falls below threshold then trigger out-of-cycle re-election.
12) Store {φ_t, a_t, r_t, φ_(t+1)} in replay buffer D.

13) Sample a random mini-batch {φ_j, a_j, r_j, φ_(j+1)} from D.

14) Compute target: y = r_(j+1) if φ_(j+1) is terminal, else r_(j+1) + γ·max_a Q(φ_(j+1), a; θ).

15) Update θ by stochastic gradient descent on $(y - Q(\varphi_j, a_j; \theta))^2$.

16) Every C steps, synchronize target network: $\theta^- \leftarrow \theta$.

17) end for

18) end for

6. SIMULATION PROTOCOL AND EXPECTED OUTCOMES

Because this paper's contribution is a formulation-level extension of the RoDQL clustering model, its performance claims should be validated empirically before publication; the following protocol is proposed for that purpose.

Baselines for comparison: (i) static weighted-sum CH selection (energy, degree, distance, mobility only); (ii) tabular Q-learning clustering; (iii) plain DQN clustering without trust; (iv) the proposed TA-RoDQL.

Metrics to record: average residual energy per round, network lifetime (rounds until first / 50% node death), cluster-size balance (standard deviation of member count across clusters), routing overhead (control packets per data packet delivered), and attack-detection rate / detection latency under injected wormhole and grey-hole behaviour.

On the basis of the reward shaping in Section 4.2, TA-RoDQL is expected to (a) maintain higher average residual energy at cluster heads than the static and tabular baselines, because the learned policy adapts CH selection to changing energy state rather than applying a fixed weighting; (b) detect wormhole-style misbehaviour faster than schemes that layer trust on after clustering, because trust is part of the state the agent observes on every step rather than a downstream audit; and (c) produce more evenly balanced cluster sizes than static weighting, since the node-degree term in the state is continuously re-evaluated rather than fixed at formation time. These are the specific, falsifiable expectations the simulation protocol above is designed to test.

7. SECURITY DISCUSSION

The guard-node subsystem gives the trust term in Section 4.1 an independent source of evidence: because guard nodes observe forwarding behaviour rather than relying on self-reported node state, a compromised node cannot directly inflate its own trust score. When a node's behaviour deviates enough to fail the trust threshold, the network is notified and the node is excluded from subsequent CH elections and, where the deviation is consistent with an active attack such as wormhole routing, isolated from the routing path altogether. Folding this signal into the reward function (Section 4.2) additionally discourages the learning agent from re-selecting nodes whose trust has degraded, closing the loop between detection and election rather than treating them as separate stages.

8. CONCLUSION AND FUTURE WORK

This paper set out a trust-augmented extension of reward-optimized Deep Q-Learning clustering for MANETs, in which a continuously-updated trust score, maintained by a guard-node subsystem, is folded into the state representation and reward function alongside residual energy, node degree, distance and mobility. The

extension keeps the MDP formulation and training procedure of the base RoDQL model but adds a fifth state variable and an out-of-cycle re-election trigger, so that CH suitability is reassessed as node behaviour changes rather than only at fixed cluster-formation intervals. A simulation protocol with concrete baselines and metrics has been laid out to validate the expected gains in energy balance, cluster-size stability and attack-detection latency. Future work includes running the protocol above across varied mobility models and attacker fractions, extending the trust model to cover collusive multi-node attacks, and exploring a multi-agent formulation in which neighbouring CHs share partial state to coordinate inter-cluster routing decisions.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

Alowish, M., Shiraishi, Y., Takano, Y., Mohri, M., and Morii, M. (2020). Stabilized clustering enabled V2V communication in an NDN-SDVN environment for content retrieval. IEEE Access, 8, 135138–135151.

Desai, A. M., and Jhaveri, R. H. (2018). Secure routing in mobile ad hoc networks: A predictive approach. International Journal of Information Technology.

Feng, Y., Teng, G. F., Wang, A. X., and Yao, Y. M. (2007). Chaotic inertia weight in particle swarm optimization. In 2007 Second International Conference on Innovative Computing, Information and Control (ICICIC 2007) (p. 475). IEEE.

Haridas, S. (2026). Environment-aware rewards optimized deep-Q-learning for cluster head selection in MANET. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), 15(8), 363–368.

Haridas, S., and Rama Prasath, A. (2020). Enhancement of network lifetime in MANET: Improved particle swarm optimization for delayless and secured geographic routing. Journal of Advanced Research in Dynamical and Control Systems, 12(07-Special Issue).

Janani, V. S., and Manikandan, M. S. (2020). Hexagonal clustered trust based distributed group key agreement scheme in mobile ad hoc networks. Wireless Personal Communications, 1–20.

Janani, V. S., and Manikandan, M. S. K. (2017). Enhanced security using cluster based certificate management and ECC-CRT key agreement schemes in mobile ad hoc networks. Wireless Personal Communications, 97(4), 6131–6150.

Kanagasundaram, H., and A, K. (2018). EIMO-ESOLSR: Energy efficient and security-based model for OLSR routing protocol in mobile ad-hoc network. IET Communications.

Kavitha, G. (2018). QoS improvement based security enhancement for link activity monitoring service in mobile ad hoc network. Cluster Computing.

Kavitha, T., and Muthaiah, R. (2018). A lightweight FFT based enciphering system for extending the lifetime of mobile ad hoc networks. Cluster Computing.

Khan, B. U. I., Anwar, F., Olanrewaju, R. F., Pampori, B. R., and Mir, R. N. (2020). A game theory-based strategic approach to ensure reliable data transmission with optimized network operations in futuristic mobile adhoc networks. IEEE Access, 1–1.

Krishnan, R. S., Julie, E. G., Robinson, Y. H., Kumar, R., Son, L., Tuan, T. A., and Long, H. V. (2020). Modified zone based intrusion detection system for security enhancement in mobile ad hoc networks. Wireless Networks, 26, 1275–1289.

Kumar, K. V., Jayasankar, T., Eswaramoorthy, V., and Nivedhitha, V. (2020). SDARP: Security based data aware routing protocol for ad hoc sensor networks. International Journal of Intelligent Networks, 1, 36–42.

Mehrkanoon, S., Alzate, C., Mall, R., Langone, R., and Suykens, J. A. (2014). Multiclass semisupervised learning based upon kernel spectral clustering. IEEE Transactions on Neural Networks and Learning Systems, 26(4), 720–733.

Ochola, E. O., Mejaele, L., Eloff, M. M., and Poll, J. (2017). MANET reactive routing protocols node mobility variation effect in analysing the impact of black hole attack.

Robinson, Y. H., and Julie, E. G. (2019). MTPKM: Multipart trust based public key management technique to reduce security vulnerability in mobile ad-hoc networks. Wireless Personal Communications.