Journal of Theoretical and Applied Information Technology
30th April 2024. Vol.102. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

STRENGTHENING SECURITY IN HEALTHCARE MOBILE WIRELESS SENSOR NETWORKS USING RESILIENT BELUGA WHALE OPTIMIZATION-BASED ENHANCED TEMPORALLY ORDERED ROUTING ALGORITHM (RBWO-TORA)

S. KAWSALYA[1], D. VIMAL KUMAR[2]
[1] Department of Computer Science, Nehru Arts and Science College, India
[2] Department of Computer Science, Nehru Arts and Science College, India
E-mail: [1] kawsalya.mca2006@gmail.com, [2] drvimalcs@gmail.com

ABSTRACT

The integration of Enhanced Temporally Ordered Routing Algorithm (E-TORA) and Resilient Beluga Whale Optimization (RBWO) presents a groundbreaking approach to network optimization. E-TORA establishes a foundation with secure multi-path routing, incorporating cryptographic measures for heightened data security. Concurrently, RBWO introduces innovative optimization strategies inspired by beluga whale behaviors, fostering adaptability and cooperative exploration. The fusion of these algorithms synergistically enhances network performance, ensuring efficient data delivery, minimal delay, and optimized energy consumption. The cooperative exploration patterns inspired by RBWO complement E-TORA's multi-path routing, striking an effective balance. Results demonstrate consistently high Packet Delivery Ratios, decreased Delay values, and efficient Throughput, affirming the algorithm's success. This research contributes to the advancement of network protocols, offering valuable insights for refining and optimizing dynamic network environments.

Keywords: *Network Optimization - Beluga Whale Optimization - TORA - Routing - Security.*

1. INTRODUCTION

Healthcare stands as a pillar of human dignity, ensuring that people can lead an energetic lifestyle and achieve their dreams. It goes beyond merely treating sickness, promoting preventive measures that stress the value of early detection and healthy lifestyles [1]. A strong healthcare system is vital for cultivating a productive and inventive society, as the well-being of individuals contributes significantly to economic and social progress. Furthermore, healthcare serves as a safety net, providing a shield for individuals and communities during unexpected crises. In our closely connected world, working together globally for health is really important, understanding that diseases don't stick to boundaries [2]. By placing importance on strengthening healthcare systems, societies not only improve the quality of life for their citizens but also play a role in building the resilience and prosperity of the entire global community.

The dynamic nature of Mobile Wireless Sensor Networks (MWSN) presents unique challenges and opportunities in terms of network design and communication protocols. The mobility of sensor nodes requires the development of adaptive routing algorithms that can dynamically adjust to changes in the network topology [3]. Additionally, energy-efficient strategies become critical to managing the increased power consumption associated with mobility. MWSNs necessitate the exploration of innovative solutions for localization techniques to accurately track the positions of mobile sensor nodes in real-time [4]. These challenges drive research and innovation in the field of MWSNs, fostering the development of robust and efficient communication protocols that cater to the specific demands of mobile sensor deployments.

The role of MWSN in healthcare is pivotal, ushering in a new era of patient monitoring and personalized care. By incorporating mobile sensors, MWSNs enable continuous tracking of patients' health parameters, facilitating timely interventions and improving overall healthcare outcomes [5]. These networks find applications in diverse healthcare scenarios, from monitoring the elderly in

assisted living facilities to tracking vital signs in ambulatory patients. MWSNs empower healthcare flexibility of MWSNs ensures that patients experience unobtrusive monitoring, contributing to a more comfortable and patient-friendly healthcare environment.

Deploying MWSN presents challenges in hardware and software [6]. Developing energy-efficient sensors for sustained mobility is crucial, requiring advanced power management solutions. However, the perpetual movement raises security concerns, demanding robust encryption and authentication. On the software side, ensuring seamless communication among mobile sensors needs intricate algorithms for dynamic routing, considering factors like movement speed [7]. Simultaneously, data fusion challenges arise, requiring coherent integration for accurate insights. These multifaceted challenges emphasize the need to address energy efficiency, communication protocols, and stringent security measures to fortify data integrity in MWSN deployments [8].

1.1. Problem Statement

Securing H-M-WSN is a complex challenge due to the wireless and dynamic nature of healthcare environments. The problem centers on safeguarding sensitive patient data and network operations, as unauthorized access and data breaches are significant security risks. Ensuring only authorized entit ies can interact with the network while preventing malicious intrusions is a central challenge. The vast number of interconnected devices in H-M-WSNs also poses a challenge in securing data transmission, with the risk of eavesdropping and data interception. Addressing the security challenge must consider the mobility of sensor nodes as they move to monitor different patients, demanding continuous security during these transitions. Developing advanced security protocols, intrusion detection systems, and access control mechanisms is essential to tackle these issues.

1.2. Motivation

The motivation to address the security challenge in H-M-WSN arises from the importance of patient data privacy, medical information integrity, and the need to secure healthcare operations effectively. Patient trust hinges on the confidentiality and reliability of data in healthcare. The growing adoption of digital healthcare solutions and wireless sensor networks intensifies the need for robust security measures. With a broader attack surface and professionals with real-time data, enhancing diagnostic accuracy and treatment effectiveness. The more sophisticated threats, healthcare systems must continually advance their defense mechanisms to maintain patient trust and healthcare service quality. Data security during sensor node mobility is imperative, as patients' conditions often require continuous monitoring, and data integrity is crucial for informed healthcare decisions. This motivation is grounded in the significance of data integrity, trustworthiness, and patient confidence in the security of data generated and transmitted by H-M-WSNs.

1.3. Objective

The primary technical objective of this research is to engineer and implement an innovative, bio-inspired secure routing protocol for the H-M-WSN domain. This protocol aims to provide robust security solutions for H-M-WSN within healthcare environments without delving into traditional key-based security mechanisms.

- **Customized Secure Protocol:** Develop a secure routing protocol tailored to the distinct security requirements and constraints of H-M-WSN in healthcare settings. The protocol will emphasize secure data transmission, access control, and intrusion prevention without relying on conventional key-based mechanisms.
- **Data Privacy Enhancement:** Implement advanced data privacy measures without involving keys, such as dynamic data masking, homomorphic encryption, and secure multi-party computation, to ensure patient data and sensitive healthcare information remain confidential and immune to unauthorized access.
- **Mobility-Adaptive Security:** Engineer the protocol to adapt seamlessly to the mobility of sensor nodes within healthcare environments. This includes the development of dynamic security policies, efficient handover mechanisms, and secure routing strategies to maintain data confidentiality even as sensor nodes transition between different locations.
- **Performance Evaluation:** Conduct comprehensive security assessments through simulations and real-world experimentation in healthcare contexts to evaluate the protocol's effectiveness in elevating data privacy, ensuring data confidentiality, and fortifying overall network security. This will be achieved

without relying on traditional key-based security techniques.

This research aims to advance security in H-M-WSN by introducing a bio-inspired secure routing protocol that effectively enhances data protection, access control, and intrusion prevention without resorting to conventional key-based security mechanisms. The outcomes of this research have the potential to significantly impact healthcare quality and patient well-being by fortifying the security and privacy of healthcare data within an increasingly digitized and interconnected healthcare landscape.

2. LITERATURE REVIEW

"DMST" [9] By leveraging the adaptability of a Dynamic Minimum Spanning Tree, the protocol responds dynamically to the network's changes, ensuring the creation of efficient and adaptive routing paths. "ICRA" [10]. The protocol dynamically forms intelligent clusters, considering factors such as location and connectivity, and utilizes efficient routing strategies for improved network performance. "EEDSR" [11]. By leveraging mobility prediction, the system dynamically optimizes paths, minimizing energy consumption and ensuring timely data transmission.

"LoRaWAN"[12]. Initially, it surveys existing routing strategies in multi-hop networks utilizing LoRaWAN technology. Subsequently, it introduces an innovative SDN-based solution tailored specifically for Smart Water Grids. "A New Deep Q-Network Design for QoS Multicast Routing in Cognitive Radio MANETs."[13] The proposed Deep Q-Network (DQN) design leverages deep learning techniques to optimize the decision-making process for multicast routing, particularly in dynamic and resource-constrained cognitive radio environments. "EcoOpNet,"[14] an innovative protocol for Wireless Sensor Networks (WSN). By leveraging cross-layer information to opportunistically route data, dynamically adapting to the network's conditions.

"SecureLoc,"[15] for securing localization in Wireless Sensor Networks (WSNs) against routing attacks. By leveraging the strengths of hybrid models, the protocol enhances the security of WSNs, ensuring the trustworthiness of localization information[15]. "EcoHarvest,"[16] a groundbreaking two-phase energy-efficient load balancing scheme for data collection in Energy Harvesting Wireless Sensor Networks (EH-WSNs) utilizing a mobile sink. It involves two distinct phases, strategically balancing the energy load among sensor nodes. By leveraging a mobile sink, the protocol efficiently collects data, considering the energy harvested by nodes. "TrustRoute,"[17] a Trust-Aware Dynamic Routing algorithm based on the Extended Ad Hoc On-Demand Distance Vector (EAODV) protocol for secure communications in WSNs. By dynamically assessing trust levels, the protocol ensures secure and reliable communication paths. "EcoSDN,"[18] an energy-efficient and delay-guaranteed routing algorithm for Software-Defined Wireless Sensor Networks (SD-WSNs). By leveraging deep reinforcement learning to enable sensor nodes to collaboratively make energy-efficient and delay-sensitive routing decisions in a software-defined network framework.

"EcoCovNet,"[19] for Mobile Sink-based Wireless Sensor Networks. The key contribution lies in the protocol's strategy that combines energy and coverage considerations in hierarchical data collection. EcoCovNet's working mechanism involves dynamically assessing both energy levels of sensor nodes and their coverage capabilities. "SurgeNet,"[20] a Dynamic Energy-Efficient Surveillance Routing approach designed for uncertain group-based Industrial Wireless Sensor Networks (IWSNs). The functional part lies in the protocol's adaptability to uncertain industrial environments, optimizing energy efficiency for surveillance purposes. "EcoSquirrel,"[21] an Improved Squirrel Search Algorithm (ISSA) applied for Clustering and Low-Energy Routing in WSNs. By harnessing the unique characteristics of the squirrel search algorithm to optimize cluster formation and low-energy routing decisions. "CodeTabuNet,"[22] a method for determining the Multicast Optimal Route for Mobile Sinks within a specified deadline using Network Coding and the Tabu Search Algorithm in WSNs. By leveraging network coding principles alongside tabu search algorithms to intelligently navigate the network and determine optimal routes for mobile sinks [23], [24], [33]–[42], [25]–[32].

Table 1 Comparison of Related Literature

State-of-the-Art Algorithms	Merits	Demerits
DMST	Dynamically adapting the Minimum Spanning Tree (MST) to changes in network conditions, ensuring efficient routing with mobile sinks.	High sink mobility scenarios, impacting stability and efficiency. Computational overhead may increase.
ICRA	Adapts to dynamic network changes, improving overall efficiency.	Computational complexity might increase, requiring consideration in resource-constrained scenarios.
EEDSR	Prolonged network lifespan and improved data delivery timelines, enhancing overall efficiency.	Predicting mobility patterns, impacting the reliability of routing paths. The effectiveness of the approach could be influenced by unpredictable environmental factors.
LoRaWAN	Long-range communication capabilities, Efficient connectivity for IoT devices over large geographic areas. It operates on low power, extending battery life for connected devices.	Challenges in urban environments with high signal interference. The data rate is comparatively lower than some other wireless technologies.
MR-CRM	QoS multicast routing in Cognitive Radio MANETs, enhancing adaptability and reliability.	Potential computational complexity, training time requirements, and dependence on high-quality data for optimal performance.
EcoOpNet	Focuses on energy optimization, aiming to enhance overall sustainability and reduce environmental impact.	Computational difficulty might rise in resource-constrained scenarios.
SecureLoc	Ensuring data integrity and protecting against potential threats.	Computational overhead may increase.
EcoHarvest	Promotes sustainability and reducing dependency on external power sources for prolonged device operation.	Functional difficulty might rise in resource-constrained scenarios.
TrustRoute	Improved security and trust-awareness in dynamic routing for WSNs. It might enhance the resilience of the communication infrastructure against security threats.	Computational overhead may increase.
EcoSDN	It could potentially improve the overall efficiency of data transfer and network performance.	Limited scalability may hinder its application in large-scale networks.
EcoCovNet	Hierarchical approach enhances energy efficiency and extends network life.	Sensitive to sink mobility patterns, affecting data collection.
SurgeNet	It excels in delivering timely data, crucial for emergency scenarios, ensuring rapid response and decision-making.	Challenges include potential congestion during high-traffic surges, impacting overall network performance.
EcoSquirrel	It exhibits efficiency in optimizing energy consumption, promoting prolonged network longevity and stable data transmission.	May encounter challenges in adapting to diverse environmental conditions.
CodeTabuNet	Enhances multicast optimal route determination, providing efficient and timely data transmission with reduced latency.	Challenges may arise in adapting to dynamic network conditions, influencing the algorithm's overall efficiency in certain deployment scenarios.
MERT	Dynamically select cluster heads, optimizing network performance by considering factors such as energy efficiency and network connectivity.	Might be influenced by the choice of meta-heuristic optimization techniques
ECOGs	This approach ensures prolonged network lifespan by distributing energy consumption more evenly among sensors, enhancing overall efficiency	Adaptability to diverse conditions might be a consideration, and performance could be influenced by factors such as relic density and exhibit layout.

3. RESILIENT BELUGA WHALE OPTIMIZATION-BASED ENHANCED TEMPORALLY-ORDERED ROUTING ALGORITHM (RBWO-TORA)

3.1. Temporally-Ordered Routing Algorithm (TORA):

Temporally-Ordered Routing Algorithm (TORA) stands as a pivotal solution in mobile ad hoc networks, offering an adept response to the intricate challenges of dynamic network topologies. TORA, a reactive routing protocol, orchestrates nodes based on a height metric, aligning them in a directed acyclic graph. The protocol's notable feature is the link reversal technique, deftly reversing link directions to form a coherent routing structure. Nodes communicate through query and update messages, fostering route discovery and maintenance. TORA lacks global sequence numbers, relying on local orderings for message integrity. Its capacity to support multiple paths enhances resilience and load balancing. TORA's energy-efficient design minimizes control message overhead, addressing the imperative of resource conservation in mobile and dynamic environments. As a stalwart in mobile ad hoc networking, TORA exemplifies adaptability, efficiency, and reliability in the face of dynamic network challenges.

3.2. Enhanced TORA with Secure Multi-path Routing Algorithms (E-TORA):

Secure Multi-path Routing Algorithms refer to routing protocols and strategies designed to enhance the security of data transmission in computer networks by utilizing multiple paths between source and destination nodes. In traditional routing, data typically follows a single path from the source to the destination. However, in secure multi-path routing, multiple routes are established simultaneously, offering several advantages, including improved reliability, load balancing, and, critically, enhanced security.

Enhancing TORA with Secure Multi-path Routing Algorithms fortifies its capabilities to address security concerns and dynamic network challenges. TORA, known for its efficiency in mobile ad hoc networks, can benefit significantly from the integration of secure multi-path principles.

3.2.1. Cryptographic Integration:

Incorporating cryptographic measures is essential to fortify the security of TORA. The integration of encryption and digital signatures within the communication process enhances data confidentiality and integrity along multiple paths. Integration of cryptographic measures enhances the TORA. Let M denote the original message, E represent the encryption function, and C signify the resulting ciphertext. The encryption process is mathematically expressed as $C = E(M)$, where E is a function that utilizes cryptographic keys (K_e) which is represented in Eq.(1).

$$C = \mathrm{E}(M, K_e) \quad (1)$$

To ensure secure communication a corresponding decryption process is employed using the decryption function D and the appropriate decryption key (K_d) expressed with Eq.(2).

$$M = \mathrm{D}(C, K_d) \quad (2)$$

Incorporating digital signatures involves the application of a hash function (H) and a private key (K_{priv}) to generate a signature(Sig). The signature creation can be expressed in Eq.(3).

$$Sig = \mathrm{H}(M, K_{priv}) \quad (3)$$

The verification is achieved through a corresponding public key (K_{pub}), ensuring the authenticity of the message (M). The verification process is expressed mathematically in Eq.(4)

$$\mathrm{Verify}(Sig, M, K_{pub}) \quad (4)$$

Embedding these cryptographic processes into TORA, the algorithm gains resilience against unauthorized access and tampering, thus securing the integrity of data traversing the network paths.

3.2.2. Dynamic Key Management:

Dynamic Key Management is crucial for enhancing the security TORA. Dynamic key management involves the establishment and refreshing of cryptographic keys over time, ensuring the continuous integrity and confidentiality of communication channels. Let K_i represent the cryptographic key at time i, and T denote the time variable. The dynamic nature of key management is expressed through the key evolution equation Eq.(5).

$$K_{i+1} = \mathrm{UpdateKey}(K_i, T) \quad (5)$$

where $UpdateKey$ is a function that generates a new cryptographic key based on the current key (K_i) and the elapsed time (T). This dynamic key update process enhances the resilience of the cryptographic system against potential security threats.

To further illustrate the importance of dynamic key management, consider the scenario of a compromised key. Let $Compromise(K_i)$ represent

the event of key compromise at time i. The equation for key compromise is formulated using Eq.(6).

$$\text{Compromise}(K_i) = \text{Compromise}(K_i, T) \quad (6)$$

Dynamic key management responds to such compromise events by generating a new key, mitigating the impact of compromised keys on the security of the communication.

The effectiveness of dynamic key management is also contingent on the periodicity of key updates. Let Δt represent the time interval between key updates. The equation for key update periodicity is given in Eq.(7).

$$\Delta t = T_{update} - T_{start} \quad (7)$$

where T_{update} is the time of the next key update, and T_{start} is the time when the current key was initiated. This periodic update ensures that cryptographic keys are consistently refreshed, reducing the window of vulnerability associated with a static key.

Moreover, the dynamic key management system should be synchronized across communicating nodes to maintain a coherent cryptographic framework. Let $Sync(K_i)$ denote the synchronization event for key K_i. The synchronization equation is expressed in Eq.(8).

$$\text{Sync}(K_i) = \text{Sync}(K_i, T) \quad (8)$$

This synchronization mechanism ensures that nodes within the network are operating with consistent cryptographic keys, fostering secure and synchronized communication.

The incorporation of dynamic key management into TORA is instrumental in maintaining the security and resilience of the algorithm. The dynamic update, periodicity, and synchronization of cryptographic keys significantly contribute to mitigating the risks associated with key compromise and ensuring a robust security posture in dynamic network environments.

3.2.3. Distributed Trust Model

Distributed Trust Model is a crucial step in enhancing the security of TORA. The distributed trust model aims to establish trust levels for nodes across the network based on their behavior and interactions. Let T_{ij} represent the trust level assigned by node i to node j in the network. The distributed trust model encompasses various factors contributing to the establishment of trust.

Behavior Assessment: The trust assigned to a node is influenced by its observed behavior. Let B_{ij} denote the behavior assessment of node j by node i. The trust based on behavior (T_{ij}^B) is mathematically expressed in Eq.(9).

$$T_{ij}^B = AssessBehaviour(B_{ij}) \quad (9)$$

Interaction Trust: Trust is further influenced by the history of interactions between nodes. Let I_{ij} represent the interaction history between node i and node j. The trust based on interactions (T_{ij}^I) is mathematically defined in Eq.(10).

$$T_{ij}^I = AssessInteraction(I_{ij}) \quad (10)$$

Combined Trust Assessment: The overall trust (T_{ij}) assigned by node i to node j is a combination of trust derived from behavior and interactions. The combined trust assessment equation is expressed as Eq.(11).

$$T_{ij} = CombineTrust(T_{ij}^B, T_{ij}^I) \quad (11)$$

The function $CombineTrust$ harmonizes the trust assessments from behavior and interactions into a unified trust level.

Trust Propagation: Trust levels can be propagated through the network based on the trust assigned by neighboring nodes. The trust propagation equation is defined with Eq.(12).

$$T'_{ij} = \text{PropogateTrust}(T_{kj}, T_{ij}) \quad (12)$$

were T'_{ij} is the updated trust level for node j based on the trust assigned by node k and the initial trust T_{ij}.

Adaptive Trust Update: The trust model should adapt to changes in behavior and interactions over time. Let ΔT_{ij} represent the change in trust, and Δt denote the elapsed time. The adaptive trust update equation is expressed mathematically in Eq.(13).

$$\Delta T_{ij} = \text{AdaptTrust}(T_{ij}, \Delta t) \quad (13)$$

Eq.(13) ensures that the trust model dynamically adjusts trust levels based on evolving network dynamics.

The distributed trust model in TORA enhances the algorithm's security by fostering a network environment where nodes collaboratively assess and assign trust levels to one another. By incorporating assessments of behavior, interactions, and adaptive updates, the distributed trust model contributes to a robust security posture in dynamic and evolving network conditions.

3.2.4. Path Diversity for Security

Path Diversity for Security is a paramount step in fortifying the TORA. Path diversity introduces the utilization of multiple communication routes between nodes, providing not only fault tolerance but also a heightened level of security against coordinated attacks. Mathematically, path diversity can be expressed through the consideration of different paths and the distribution of traffic across these paths.

By incorporating these mathematical expressions, Path Diversity for Security within TORA ensures that the algorithm dynamically adapts to security

Path Enumeration: Let P_i represent the i-th path between a source node and a destination node. The set of paths, denoted as $\{P_1, P_2, \dots, P_n\}$, enumerates the available communication routes in the network.

Traffic Distribution: The distribution of traffic across multiple paths involves assigning weights to each path based on factors such as capacity and security. Let W_i represent the weight assigned to the i-th path. The normalized weight ($\widehat{W_i}$) can be calculated as shown in Eq.(14).

$$\widehat{W_i} = \frac{W_i}{\sum_{j=1}^{n} W_j} \quad (14)$$

This normalization ensures that the weights sum to 1, facilitating the fair distribution of traffic across diverse paths.

Path Selection Criteria: The selection of paths involves assessing security metrics alongside traditional routing metrics. Let S_i denote the security metric associated with the i-th path, and R_i represent the traditional routing metric. The overall path selection criteria (P_i^*) can be expressed mathematically in Eq.(15).

$$P_i^* = \text{SelectPath}(S_i, R_i) \quad (15)$$

The function $SelectPath$ harmonizes the security metric S_i and the routing metric R_i to determine the optimal path P_i^*.

Dynamic Path Adaptation: The network should adapt dynamically to changes in security conditions and topology. Let ΔS_i represent the change in security metric, and Δt denote the elapsed time. The adaptive path update equation is given in Eq.(16).

$$\Delta S_i = \text{AdaptPath}(S_i, \Delta t) \quad (16)$$

This Eq.(16) ensures that the security metric S_i dynamically adjusts based on evolving security conditions.

Path Usage and Security Enhancement: The actual usage of paths (U_i) can be determined by multiplying the normalized weight ($\widehat{W_i}$) with the security metric (S_i) represented in Eq.(17).

$$U_i = \widehat{W_i} * S_i \quad (17)$$

The Eq.(17) reflects the consideration of both weight and security in determining the usage of each path, reinforcing the security enhancement through diversified path utilization.

3.2.5. Adaptive Security Policies

Adaptive Security Policies is a pivotal step in augmenting the security of the TORA. Adaptive security policies entail the dynamic adjustment of security measures based on the evolving threat landscape and network conditions. The adaptation of security policies can be expressed through the formulation of dynamic security parameters.

Dynamic Security Parameter (DSP): Let DSP_{ij} represent the dynamic security parameter between nodes i and j. The dynamic adjustment over time (Δt) is expressed in Eq.(18).

$$DSP'_{ij} = \text{AdaptSecurity}(DSP_{ij}, \Delta t) \quad (18)$$

where $AdaptSecurity$ is a function that dynamically adjusts the security parameter DSP_{ij}based on the elapsed time (Δt). This dynamic adaptation ensures that security measures evolve in response to changing network conditions.

Security Metric Integration: The overall security metric (SM_{ij}) is a composite measure considering both traditional security parameters (SP_{ij}) and the dynamic security parameter (DSP_{ij}) is shown in Eq.(19).

$$SM_{ij} = \text{CombineSecurity}(SP_{ij}, DSP'_{ij}) \quad (19)$$

The function $CombineSecurity$ complements the traditional security parameter (SP_{ij}) and the dynamically adapted security parameter (DSP'_{ij}) into a unified security metric.

Threshold-Based Decision: Adaptive security policies involve making decisions based on the security metric surpassing predefined thresholds. Let $Threshold_{ij}$ represent the predefined security threshold. The decision function is expressed in Eq.(20).

$$Decision_{ij} = \text{MakeDecision}(SM_{ij}, Threshold_{ij}) \quad (20)$$

The function $MakeDecision$ determines the security decision based on whether the security metric (SM_{ij}) surpasses the predefined threshold.

Dynamic Policy Update: If the security decision ($Decision_{ij}$) indicates a need for policy update, the dynamic security parameter (DSP'_{ij}) is further adjusted based on the network's response to security events represented in Eq.(21).

$$DSP'_{ij} = \text{UpdatePolicy}(DSP'_{ij}, NetworkResponse) \quad (21)$$

The function $UpdatePolicy$ incorporates the network's response ($NetworkResponseij$) to security events in dynamically updating the security parameter (DSP'_{ij}).

Adaptive Security Policies in TORA ensure that security measures dynamically adapt to changing network conditions. The dynamic adjustment of security parameters and integration with traditional security measures enhance the algorithm's resilience and responsiveness to emerging security threats, contributing to a robust security framework in dynamic network environments.

Pseudo-code: Adaptive Security Policies

```
Function AdaptiveSecurityPolicies():
  While NetworkOperational:
    SecurityMetrics = MonitorSecurityMetrics()
    NetworkConditions = MonitorNetworkConditions()

    If ConditionsChange(NetworkConditions):
      AdjustSecurityParameters(SecurityMetrics, NetworkConditions)
      Impact = EvaluateImpactOnSecurityPolicies(SecurityMetrics, NetworkConditions)

      If ExceedsThresholds(Impact):
        UpdateSecurityPolicies()
        AdaptParametersToMitigateRisks(Impact)

    WaitForNextIteration()

End Function
```

The pseudo-code outlines the adaptive security policies algorithm, emphasizing continuous monitoring, dynamic adjustment of parameters, and updates to security policies based on observed changes and predefined thresholds.

3.2.6. Intrusion Detection along Paths

Intrusion Detection along Paths is a crucial step in enhancing the security of the TORA. Intrusion detection involves the systematic monitoring of network paths to identify and respond to anomalous activities that may indicate potential security breaches. The mathematical representation of intrusion detection along paths encompasses various components.

Traffic Monitoring: Let T_{ij} denote the traffic along the i-th path between nodes i and j. The traffic monitoring function ($Monitor_{ij}$) assesses the traffic patterns to identify deviations from normal behavior shown in Eq.(22).

$$Monitor_{ij} = \text{AssessTraffic}(T_{ij}) \quad (22)$$

The function $AssessTraffic$ analyzes the traffic along the path to detect any unusual patterns that may indicate potential intrusions.

Anomaly Score Calculation: The anomaly score (AS_{ij}) quantifies the degree of deviation from normal behavior. It is calculated based on the output of the traffic monitoring function mathematically represented in Eq.(23).

$$AS_{ij} = \text{CalculateAnomalyScore}(Monitor_{ij}) \quad (23)$$

The function $CalculateAnomalyScore$ synthesizes the information from the traffic monitoring to provide a numerical representation of the anomaly score.

Threshold-Based Decision: A predefined threshold ($Threshold_{ij}$) is established to determine whether the anomaly score surpasses a critical level. The decision function ($Decision_{ij}$) is expressed with Eq.(24).

$$Decision_{ij} = \text{MakeDecision}(AS_{ij}, Threshold_{ij}) \quad (24)$$

The function $MakeDecision$ categorizes the severity of the anomaly based on the anomaly score and the predefined threshold.

Alarm Generation: If the decision indicates a potential intrusion as shown in Eq.(25), an alarm is generated to alert the network which is shown in Eq.(26).

$$Decision_{ij} = Intrusion \quad (25)$$

$$Alarm_{ij} = \text{GenerateAlarm}(Decision_{ij}) \quad (26)$$

The function $GenerateAlarm$ creates an alarm signal, signaling the detection of a potential intrusion along the path.

Dynamic Response: The network's response ($Response_{ij}$) to the detected intrusion involves

adapting security measures or altering the communication path. The dynamic response equation is given in Eq.(27).

$$Response_{ij} = \text{AdaptResponse}(Decision_{ij}) \quad (27)$$

The function $AdaptResponse$ tailors the network's response based on the severity and nature of the detected intrusion. The dynamic adaptation and timely response to potential intrusions contribute to the overall security resilience of TORA in dynamic and evolving network environments.

3.2.7. Certificate-Based Authentication

The Certificate-Based Authentication is an important step in stimulating the security of the TORA. Certificate-based authentication employs digital certificates to validate the identities of communicating nodes within the network. The mathematical representation of certificate-based authentication involves the utilization of cryptographic functions and certificate validation processes.

Digital Certificate Representation: Let $Cert_{ij}$ represent the digital certificate exchanged between nodes i and j. The digital certificate includes the public key (K_{pub}) of the originating node and is signed by the corresponding private key (K_{priv}):

$$Cert_{ij} = \text{Sign}(K_{priv_i}, K_{pub_i}) \quad (28)$$

The digital certificate is a cryptographic assurance of the authenticity of the public key associated with the originating node.

Certificate Transmission: Nodes within the network exchange digital certificates during the initial communication setup. Let $Transmit_{ij}$ denote the transmission of the digital certificate from node i to node j mathematically represented in Eq.(29).

$$Transmit_{ij} = \text{SendCertificate}(Cert_{ij}) \quad (29)$$

The function $SendCertificate$ ensures the secure transmission of the digital certificate from the originating node to the intended recipient.

Certificate Verification: On receiving the digital certificate, the recipient node verifies its authenticity using the public key of the originating node. Let $Verify_{ij}$ represent the verification process is shown using Eq.(30).

$$Verify_{ij} = \text{VerifyCertificate}(Cert_{ij}, K_{pub_i}) \quad (30)$$

The function $VerifyCertificate$ validates the digital certificate using the public key associated with the originating node.

Authentication Decision: Based on the outcome of the certificate verification, an authentication decision ($Auth_{ij}$) is made. If the certificate is successfully verified, authentication is achieved which is represented mathematically in Eq.(31).

$$Auth_{ij} = \text{MakeAuthemticationDecision}(Verify_{ij}) \quad (31)$$

The function $MakeAuthenticationDecision$ categorizes the authentication decision based on the success or failure of the certificate verification process. Certificate-Based Authentication in TORA ensures that nodes within the network can trust the identities of communicating entities. The use of digital certificates, cryptographic signatures, and secure transmission processes collectively establish a robust authentication mechanism. This step significantly contributes to the overall security of TORA by preventing unauthorized nodes from participating in the communication and ensuring the integrity of the network.

3.2.8. Continuous Monitoring and Dynamic Path Switching

Continuous Monitoring and Dynamic Path Switching constitute a crucial step in bolstering the security of the TORA. This step involves the persistent assessment of network conditions and the dynamic reconfiguration of communication paths based on real-time monitoring. This process can be expressed through the formulation of continuous monitoring equations and the criteria for initiating dynamic path switching.

Path Health Monitoring: Let H_{ij} represent the health status of the i-th path between nodes i and j. Path health is continuously monitored to assess the reliability and security of each path which is mathematically represented with Eq.(32).

$$H_{ij} = \text{MonitorPathHealth}() \quad (32)$$

The function $MonitorPathHealth$ evaluates various metrics, such as latency, packet loss, and security conditions, to determine the health status of the path.

Dynamic Path Switching Criteria: The decision to switch communication paths is contingent on predefined criteria, including the health status of the current path ($H_{current}$) and a comparison with the health status of alternative paths ($H_{alternative}$). The

dynamic path switching decision ($SwitchDecision$) is determined in Eq.(33).

$$\text{SwitchDecision} = MakeSwitchDecision(H_{current} \quad (33)$$

The function $MakeSwitchDecision$ assesses the health status of the current path and potential alternative paths to decide whether a dynamic path switch is warranted.

Dynamic Path Switching: If the decision ($SwitchDecision$) indicates the need for a path switch, the algorithm dynamically reroutes communication along a more reliable or secure path. The dynamic path switching equation is expressed in Eq.(34).

$$NewPath_{ij} = \text{SwitchPath}(CurrentPath_{ij}, Alternativ \quad (34)$$

The function $SwitchPath$ selects an alternative path from the available options based on predefined criteria, facilitating the dynamic switch.

Path Switching Response: The network's response to the dynamic path switch involves updating routing tables and notifying relevant nodes. Let $Response_{ij}$ denote the network's response to the path switching decision is represented using Eq.(35).

$$Response_{ij} = \text{UpdateRoutingTables}(NewPath_{ij}) \quad (35)$$

The function $UpdateRoutingTables$ ensures that routing tables are adjusted to reflect the newly selected path, enabling seamless communication along the updated route.

Continuous Monitoring Adaptation: The continuous monitoring process adapts dynamically to changes in network conditions. Let ΔH_{ij} represent the change in path health, and Δt denote the elapsed time.

$$\Delta H_{ij} = \text{AdaptMonitoring}(H_{ij}, \Delta t) \quad (36)$$

The Eq.(36) ensures that the monitoring process is responsive to evolving network conditions, contributing to the accuracy of health assessments.

Continuous Monitoring and Dynamic Path Switching in TORA establish a proactive approach to network management. The continuous assessment of path health and dynamic response mechanisms contribute to the overall resilience and adaptability of TORA in dynamic and challenging network environments.

Algorithm : Overview of E-TORA

1. Initialize()
   - Network topology
   - Communication paths, and
   - Security parameters.
2. For each communication path:
   - Generate encryption and decryption keys (K_e_ij, K_d_ij).
   - Implement digital signatures.
3. Initialize key update interval (Δt).
   - For each communication path:
     - Generate initial cryptographic key (K_ij).
     - Set next key update time (T_update_ij = current time + Δt).
   - While network is operational:
   - If current time reaches T_update_ij:
     - Generate new cryptographic key (K_ij).
     - Update T_update_ij = current time + Δt.
4. Assess():
   - Behavior and interactions,
   - Calculate trust levels, and
   - Share among nodes.
5. For each communication path:
   - Enumerate
     - Diverse paths,
     - Assign weights, and
     - Select based on metrics.
   - Dynamically adapt
     - Security policies based on path diversity.
6. Continuously monitor()
   - Security metrics,
   - Adjust parameters, and
   - Update policies.
7. For each communication path:
   - Monitor traffic patterns, calculate anomaly scores, and generate alarms.
   - Dynamically adapt security measures in response to detected intrusions.
8. Exchange and verify()
   - Digital certificates for authentication.
9. Continuously monitor ()
   - Path health,
   - Dynamically switch paths, and

- Update routing tables.

10. Adapt continuous monitoring to changes in network conditions.

The E-TORA algorithm secures WSN through cryptographic integration, dynamic key management, a distributed trust model, and adaptive security policies. It employs path diversity, intrusion detection, and continuous monitoring with dynamic path switching, ensuring robust communication in dynamic environments by addressing encryption, authentication, and threat response.

3.3. Beluga Whale Optimization

Beluga Whale Optimization (BWO) is a bio-inspired algorithm that mimics the collaborative and adaptive behaviors of beluga whales in finding optimal solutions to complex problems. Emulating the whales' social structure, communication, adaptability, and innovation, BWO employs a population of agents organized into pods. Through dynamic navigation and information sharing, the algorithm enhances its problem-solving capabilities over iterations. BWO leverages the diverse and cooperative nature of beluga whales to efficiently explore solution spaces and address optimization challenges in various domains.

3.3.1. Features of BWO

a) Social Structure

Beluga whales exhibit strong social bonds and collaborative behaviors within pods. BWO employs a collaborative approach, where multiple agents (representing solutions) work together to explore the solution space.

b) Communication

Belugas communicate using a diverse range of vocalizations for social interactions and echolocation. It incorporates effective communication mechanisms between agents to share information and coordinate their exploration of the solution space.

c) Adaptability

Belugas are adaptable to changing environments and can migrate to different regions. Includes mechanisms for agents to adapt their positions dynamically based on the evolving problem landscape, ensuring flexibility and responsiveness.

d) Group Dynamics

Belugas travel in pods, and their group dynamics contribute to their survival. It also encourages diversity among agents and emphasizes the importance of maintaining a balanced and diverse population to enhance the overall optimization process.

e) Navigation and Echolocation

Belugas use echolocation for navigation and locating prey. BWO incorporates mechanisms for agents to navigate through the solution space using information derived from their exploration and interactions, enhancing their ability to find optimal solutions.

f) Longevity

Beluga whales can live for several decades. Aims for long-term sustainability in its optimization process, focusing on the longevity of the algorithm's effectiveness over multiple iterations.

g) Innovation and Problem-Solving

Belugas exhibit intelligence and innovation in solving problems related to survival. Agents to innovate and adapt their strategies based on the success or failure of previous solutions, promoting efficient problem-solving.

3.4. Resilient Beluga Whale optimization

Resilient Beluga Whale Optimization (RBWO) is an advanced bio-inspired algorithm that combines the robust characteristics of beluga whales with enhanced resilience in solving complex optimization problems. By integrating mechanisms for adaptability, rapid recovery from disturbances, and persistent exploration, RBWO excels in navigating dynamic solution spaces. The algorithm ensures the resilience of the optimization process, allowing it to withstand disruptions, maintain diversity within populations, and efficiently recover from suboptimal states. RBWO is a powerful optimization tool, exhibiting the endurance and adaptability observed in resilient beluga whale populations.

a) Initialization

RBWO begins by creating a diverse initial population of potential solutions, each representing an agent in the optimization process. Parameters such as population size, maximum iterations, and exploration range are set up to guide the optimization.

b) Adaptive Strategies

RBWO incorporates adaptive strategies, allowing individual agents to dynamically adjust their positions in the solution space based on the feedback received during the optimization process. This

adaptability enhances the algorithm's responsiveness to changes in the problem landscape.

c) Population Resilience

To prevent premature convergence, RBWO emphasizes maintaining diversity within the population. This ensures that the algorithm explores a broad range of potential solutions, increasing its resilience against getting stuck in suboptimal states.

d) Dynamic Recovery

RBWO includes mechanisms for rapid recovery from disturbances or suboptimal conditions. Agents within the population can quickly adapt their strategies to recover from setbacks, allowing the algorithm to efficiently navigate through dynamic optimization environments.

e) Cooperative Exploration

Encouraging cooperative behaviors among agents, RBWO fosters shared exploration patterns within the population. This cooperative approach leverages the collective intelligence of the agents, enhancing the algorithm's ability to collectively explore and exploit the solution space.

f) Efficient Communication

RBWO facilitates efficient communication between agents. Information exchange occurs within the population, enabling agents to share knowledge and coordinate their exploration. This communication mechanism enhances the overall problem-solving capabilities of the algorithm.

g) Endurance

RBWO demonstrates endurance by maintaining high-performance levels over prolonged optimization processes. The algorithm is designed to sustain its effectiveness and adaptability over extended periods, ensuring persistent and reliable optimization.

h) Innovation-driven

RBWO promotes innovation by introducing variations in exploration patterns. Agents are encouraged to experiment with different strategies, fostering creativity in the search for novel and efficient solutions to complex optimization problems.

i) Robust Navigation and Termination

RBWO employs navigation strategies inspired by beluga whales, ensuring robust movement through dynamic solution spaces. The algorithm adheres to well-defined termination criteria based on the number of iterations, convergence conditions, or other problem-specific factors, providing a clear endpoint to the optimization process.

3.4.1. Initialization

In initialization phase the algorithm lays the groundwork by generating a diverse initial population of potential solutions. The population P consists of N

To mathematically express the initialization process, we define the position vector of the i-th agent $x_i^{(0)}$, representing the initial solution in the solution space. This vector is characterized by D dimensions, where D signifies the dimensionality of the optimization problem is shown in Eq.(37).

$$x_i^{(0)} = \left[x_{i,1}^{(0)}, x_{i,2}^{(0)}, \cdots, x_{i,D}^{(0)}\right] \quad (37)$$

where $x_{i,j}^{(0)}$ represents the initial value of the j-th dimension for the i-th agent.

It also involves determining the fitness of each agent in the initial population. The fitness function $f_i^{(0)}$ evaluates the performance of a solution and influences subsequent iterations. The fitness function is mathematically defined in Eq.(38).

$$f_i^{(0)} = \text{Fitness}\left(x_i^{(0)}\right) \quad (38)$$

This equation encapsulates the evaluation of the fitness of the i-th agent's initial solution.

The overall initialization process is represented with two key equations Eq.(39) and Eq.(40).

$$\text{P} = \left\{x_1^{(0)}, x_2^{(0)}, \ldots x_N^{(0)}\right\} \quad (39)$$

$$f_i^{(0)} = \text{Fitness}\left(x_1^{(0)}, x_2^{(0)}, \ldots x_N^{(0)}\right) \quad (40)$$

where Eq.(30) is to Generate Population and Eq.(40) is to Evaluate Fitness. This initialization step, illustrating the creation of a diverse initial population and the concurrent assessment of the fitness of each solution in preparation for subsequent optimization iterations.

3.4.2. Adaptive Strategies

In Adaptive Strategies phase the algorithm introduces mechanisms for dynamic adjustments of agent positions based on optimization feedback. The adaptability of RBWO is expressed through mathematical formulations representing the dynamic nature of the solution vectors. The adaptive strategy involves updating the position of each agent (x_i) based on the exploration and exploitation of the solution space. The updated position, denoted as

$x_i^{(t+1)}$ for the i-th agent at iteration $t+1$, is influenced by the current position ($x_i^{(t)}$) a randomly generated vector ($r_i^{(t+1)}$) for exploration, and the best solution in the population ($x_{best}^{(t)}$) for exploitation. The adaptation equation is expressed as:

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \cdot r_i^{(t+1)} + \beta \cdot \left(x_{best}^{(t)} - x_i^{(t)}\right) \quad (41)$$

where α and β are adaptation parameters, influencing the magnitude of exploration and exploitation, respectively.

RBWO involves evaluating the fitness of the updated positions to guide the subsequent iterations. The fitness of the i-th agent at iteration $t+1$, denoted as $f_i^{(t+1)}$ is determined by the fitness function which is shown in Eq.(42).

$$f_i^{(t+1)} = \text{Fitness}\left(x_i^{(t+1)}\right) \quad (42)$$

Eq.(41) and Eq.(42) capture the essence of RBWO's adaptive strategies, reflecting the dynamic adjustments of agent positions based on exploration and exploitation factors.

3.4.3. Population Resilience

Population Resilience phase emphasizes the maintenance of diversity within the population to resist premature convergence. To achieve population resilience RBWO introduces a diversity measure ($\delta^{(t)}$) that quantifies the dispersion of the solutions within the current population at iteration t. This diversity measure is calculated based on the Euclidean distance between the solutions, ensuring a comprehensive representation of the population's spatial distribution is mathematically expressed in Eq.(43).

$$\delta^{(t)} = \frac{1}{\mathrm{N}(N-1)} \sum_{i=1}^{N} \sum_{j \neq i}^{N} \left\| x_i^{(t)} - x_j^{(t)} \right\| \quad (43)$$

where N represents the population size, and $\|\cdot\|$ denotes the Euclidean norm. The algorithm then identifies the agent ($x_{best}^{(t)}$) with the highest fitness in the current population at iteration t. The fitness ($f_i^{(t)}$) of the i-th agent at iteration t is determined by the fitness function expressed in Eq.(42).
The best solution in the population is obtained shown in Eq.(44).

$$x_{best}^{(t)} = argmax x_{i^{(t)}} \cdot f_i^{(t)} \quad (44)$$

RBWO then incorporates a resilience factor (γ) to dynamically adjust the diversity measure, influencing the next iteration's population diversity is mathematically represented in Eq.(45).

$$\delta^{(t+1)} = \delta^{(t)} \cdot (1-\gamma) + \gamma \cdot \delta_{random}^{(t)} \quad (45)$$

where $\delta_{random}^{(t)}$ is a randomly generated diversity measure, introducing variability to prevent stagnation. By continuously evaluating and adjusting the diversity within the population, RBWO promotes robust exploration, preventing premature convergence and ensuring adaptability in navigating the solution space.

3.4.4. Dynamic Recovery

Dynamic Recovery phase introduces mechanisms for rapid recovery from disturbances or suboptimal conditions. This aims to enhance RBWO's adaptability by allowing agents to dynamically adjust their strategies to recover from setbacks.

To express RBWO's dynamic recovery, the algorithm utilizes an adaptive factor $\alpha^{(t)}$ that influences the magnitude of adjustments made by each agent. This factor is determined based on the fitness improvement of the current solution $f_i^{(t)}$ compared to its historical best f_i^{best} expressed mathematically in Eq.(46).

$$\alpha^{(t)} = \frac{f_i^{(t)} - f_i^{best}}{f_i^{best}} \quad (46)$$

The adaptive factor reflects the relative improvement of the current solution over its historical best, guiding the degree of recovery adjustments.

RBWO then adjusts the position of each agent $x_i^{(t+1)}$ using the dynamic recovery equation:

$$x_i^{(t+1)} = x_i^{(t)} + \alpha^{(t)} \cdot \left(x_i^{(t)} - x_{best}^{(t)}\right) \quad (47)$$

where $x_{best}^{(t)}$ represents the best solution in the current population at iteration t, and the recovery adjustment is influenced by the adaptive factor.

To avoid excessive adjustments, RBWO introduces a constraint on the magnitude of the recovery, ensuring a balanced approach.

$$\left\| x_i^{(t+1)} - x_i^{(t)} \right\| \leq \delta_{max} \quad (48)$$

In Eq.(48), this constraint (δ_{max}) limits the extent of recovery adjustments, preventing overly aggressive movements.

These process collectively define RBWO's Dynamic Recovery step, where each agent dynamically adjusts its position based on the fitness improvement

over historical best, promoting rapid recovery from suboptimal conditions.

3.4.5. Cooperative Exploration:

Cooperative Exploration phase encourages cooperative behaviors among agents, fostering shared exploration patterns within the population. This cooperative approach aims to leverage the collective intelligence of the agents for effective exploration of the solution space.

A collaborative factor ($\beta^{(t)}$) that influences the adjustment of each agent's position. This factor is computed based on the diversity measure ($\delta^{(t)}$) of the population at iteration t mathematically expressed with Eq.(49).

$$\beta^{(t)} = 1 - \frac{\delta^{(t)}}{\delta_{max}} \quad (49)$$

The collaborative factor reflects the degree of cooperation, where higher diversity leads to a stronger emphasis on cooperative exploration.

Then adjusts the position of each agent ($x_i^{(t+1)}$) using the cooperative exploration using Eq.(50).

$$x_i^{(t+1)} = x_i^{(t)} + \beta^{(t)} \cdot \frac{1}{N}\sum_{j=1}^{N}\left(x_j^{(t)} - x_i^{(t)}\right) \quad (50)$$

where the adjustment is influenced by the collaborative factor and the average difference between the current agent's position and the positions of other agents in the population.

To avoid excessive cooperative adjustments, RBWO introduces a constraint on the magnitude of the exploration which is expressed in Eq.(51).

$$\left\|x_i^{(t+1)} - x_i^{(t)}\right\| \leq \delta_{coop} \quad (51)$$

where δ_{coop} limits the extent of cooperative exploration, preventing overly aggressive movements.

In Cooperative Exploration step, where each agent dynamically adjusts its position based on the cooperative factor and the average difference with other agents. This cooperative behavior enhances the algorithm's ability to collectively explore and exploit the solution space, fostering a collaborative and intelligent optimization process.

3.4.6. Efficient Communication:

Efficient Communication phase facilitates information exchange among agents to enhance problem-solving capabilities. Efficient communication is essential for coordinating adjustments and leveraging the collective intelligence of the population.

To efficient communication RBWO proposes a communication factor $\gamma^{(t)}$ that influences the adjustment of each agent's position based on shared knowledge. This factor is calculated based on the fitness improvement of the current solution $f_i^{(t)}$ compared to the historical best solution f_i^{best} represented mathematically in Eq.(52).

$$\gamma^{(t)} = \frac{f_i^{(t)} - f_i^{best}}{f_i^{best}} \quad (52)$$

The communication factor reflects the relative improvement of the current solution over its historical best, guiding the degree of communication-based adjustments.

RBWO then adjusts the position of each agent $x_i^{(t+1)}$ using the communication-based adjustment equation shown in Eq.(53).

$$x_i^{(t+1)} = x_i^{(t)} + \gamma^{(t)} \cdot \frac{1}{N}\sum_{j=1}^{N}\left(x_j^{(t)} - x_i^{(t)}\right) \quad (53)$$

where the adjustment is influenced by the communication factor and the average difference between the current agent's position and the positions of other agents in the population.

To avoid excessive communication-based adjustments, RBWO introduces a constraint on the magnitude of the communication.

$$\left\|x_i^{(t+1)} - x_i^{(t)}\right\| \leq \delta_{comm} \quad (54)$$

where in Eq.(54) δ_{comm} limits the extent of communication-based movements, preventing overly aggressive adjustments.

In Efficient Communication where each agent dynamically adjusts its position based on the communication factor and the average difference with other agents.

3.4.7. Endurance:

Endurance is a critical aspect that ensures the continued effectiveness of RBWO in navigating complex solution spaces. A persistence factor $\rho^{(t)}$ that represents the algorithm's resilience over time. This factor is computed based on the fitness improvement of the current solution $f_i^{(t)}$ compared to its historical best f_i^{best} represented in Eq.(55).

$$\rho^{(t)} = \frac{f_i^{(t)} - f_i^{best}}{f_i^{best}} \quad (55)$$

The persistence factor reflects the relative improvement of the current solution over its historical best, guiding the algorithm's ability to endure and adapt to changing conditions.

RBWO then adjusts the position of each agent $x_i^{(t+1)}$ using the endurance-based adjustment shown in Eq.(56).

$$x_i^{(t+1)} = x_i^{(t)} + \rho^{(t)} \cdot \left(x_i^{(t)} - x_{best}^{(t)}\right) \quad (56)$$

where $x_{best}^{(t)}$ represents the best solution in the current population at iteration t, and the adjustment is influenced by the persistence factor.

To ensure controlled endurance-based adjustments a constraint is proposed on the magnitude of the adjustments which is represented in Eq.(57).

$$\left\|x_i^{(t+1)} - x_i^{(t)}\right\| \leq \delta_{endurance} \quad (57)$$

where in Eq.(57) $\delta_{endurance}$ limits the extent of endurance-based movements, preventing overly aggressive adjustments.

In this phase where each agent dynamically adjusts its position based on the persistence factor and the difference with the historical best solution. This endurance-driven approach ensures the algorithm's sustained effectiveness over extended optimization processes.

3.4.8. Innovation-driven:

Innovation-driven promotes innovation by introducing variations in exploration patterns. This step enhances RBWO's adaptability and creativity in the search for novel and efficient solutions to complex optimization problems. RBWO introduces an innovation factor $\eta^{(t)}$ that influences the adjustments made by each agent. This factor is computed based on the diversity measure $\delta^{(t)}$ of the population at iteration t shown in Eq.(58).

$$\eta^{(t)} = 1 - \frac{\delta^{(t)}}{\delta_{max}} \quad (58)$$

The innovation factor reflects the degree of diversity within the population, guiding the algorithm to introduce innovative adjustments.

RBWO then adjusts the position of each agent $x_i^{(t+1)}$ using the innovation-driven adjustment Eq.(59).

$$x_i^{(t+1)} = x_i^{(t)} + \eta^{(t)} \cdot r_i^{(t+1)} \quad (59)$$

where $r_i^{(t+1)}$ represents a randomly generated vector, introducing variability and innovation to the exploration pattern.

To control the extent of innovation-driven adjustments, RBWO introduces a constraint on the magnitude which is mathematically represented inEq.(60).

$$\left\|x_i^{(t+1)} - x_i^{(t)}\right\| \leq \delta_{innovation} \quad (60)$$

This constraint $\delta_{innovation}$ limits the extent of innovation-driven movements, preventing overly aggressive adjustments.

In this phase where each agent dynamically adjusts its position based on the innovation factor and a randomly generated vector. This innovation-driven approach enhances the algorithm's ability to explore unconventional and creative solution spaces, contributing to its adaptability and problem-solving capabilities.

3.4.9. Robust Navigation and Termination:

Robust Navigation and Termination ensures efficient movement through dynamic solution spaces. This step also involves defining clear termination criteria, providing a well-defined endpoint to the optimization process. RBWO introduces a navigation factor $\xi^{(t)}$ that influences the adjustments made by each agent based on robust navigation principles. This factor is computed as the inverse of the persistence factor $\rho^{(t)}$, ensuring that the algorithm adapts its navigation strategies based on the historical context of performance.

$$\xi^{(t)} = \frac{1}{\rho^{(t)}} \quad (61)$$

In Eq.(61) the navigation factor reflects the adaptability and resilience of the algorithm's navigation strategies.

RBWO then adjusts the position of each agent $x_i^{(t+1)}$ using the navigation-driven adjustment equation which is mathematically represented inEq.(62).

$$x_i^{(t+1)} = x_i^{(t)} + \xi^{(t)} \cdot \left(x_{best}^{(t)} - x_i^{(t)}\right) \quad (62)$$

where $x_{best}^{(t)}$ represents the best solution in the current population at iteration t, and the adjustment is influenced by the navigation factor.

To ensure controlled navigation-driven adjustments, RBWO introduces a constraint on the magnitude specified with Eq.(63)

$$\left\|x_i^{(t+1)} - x_i^{(t)}\right\| \leq \delta_{navigation} \quad (63)$$

The constraint $\delta_{navigation}$ in Eq.(63), limits the extent of navigation-driven movements, preventing overly aggressive adjustments.

In Robust Navigation and Termination step, where each agent dynamically adjusts its position based on the navigation factor and the difference with the historical best solution. This robust navigation ensures efficient movement through solution spaces, and the termination criteria provide a clear endpoint to the optimization process, contributing to the algorithm's completeness and reliability.

Pseudo code: Resilient Beluga Whale Optimization Algorithm (RBWO)
Algorithm: Resilient Beluga Whale Optimization (RBWO) Parameters: - Population size (N) - Maximum iterations (max_iter) - Exploration range (exp_range) - Adaptation parameters (α, β) - Resilience factor (γ) - Communication factor (δ_comm) - Endurance factor (δ_endurance) - Cooperative exploration factor (δ_coop)

- Innovation factor (δ_innovation)
- Navigation factor (δ_navigation)

Step 1: Initialization
For each agent i in the population:
- Set initial position randomly within the exploration range
- Evaluate fitness using a fitness function

Step 2: Adaptive Strategies
For each agent i in the population at iteration t:
- Generate a random vector within the exploration range
- Update position using adaptive strategies
- Evaluate fitness using the fitness function

Step 3: Population Resilience
- Calculate diversity measure
- Identify the agent with the highest fitness
- Update diversity measure

Step 4: Dynamic Recovery
For each agent i in the population at iteration t:
- Calculate adaptive factor
- Update position for dynamic recovery
- Ensure controlled adjustments

Step 5: Cooperative Exploration
- Calculate collaborative factor
For each agent i in the population at iteration t:
- Update position for cooperative exploration
- Ensure controlled adjustments

Step 6: Efficient Communication
- Calculate communication factor
For each agent i in the population at iteration t:
- Update position for efficient communication
- Ensure controlled adjustments

Step 7: Endurance
- Calculate persistence factor
For each agent i in the population at iteration t:
- Update position for endurance
- Ensure controlled adjustments

Step 8: Innovation-driven
- Calculate innovation factor
For each agent i in the population at iteration t:
- Update position for innovation
- Ensure controlled adjustments

Step 9: Robust Navigation and Termination
- Calculate navigation factor
For each agent i in the population at iteration t:
- Update position for robust navigation
- Ensure controlled adjustments

Repeat Steps 2 to 9 until convergence or maximum iterations reached.

3.5. Fusion of E-TORA and RBWO:

The fusion of Enhanced Temporally-Ordered Routing Algorithm with Secure Multi-path Routing Algorithms (E-TORA) and Resilient Beluga Whale Optimization (RBWO) brings forth a robust and adaptive approach. E-TORA, a refinement of the Temporally-Ordered Routing Algorithm, incorporates principles of secure multi-path routing to fortify its communication channels. The integration of cryptographic measures ensures the confidentiality and integrity of data transmitted along multiple paths, enhancing the security posture of the network.

RBWO introduces an innovative optimization strategy inspired by the resilience of beluga whales. It initiates with a diverse population, dynamically adjusting positions, fostering cooperation, and promoting innovation throughout the optimization process. The algorithm exhibits robust navigation strategies, resembling the adaptability of beluga whales, thus ensuring efficient exploration and exploitation of the solution space. The synergy of E-TORA and RBWO amalgamates their strengths, creating a comprehensive approach to network optimization. E-TORA's secure multi-path routing principles complement RBWO's adaptive strategies and resilience, forming a symbiotic relationship. Cryptographic integration in E-TORA enhances the security of RBWO's communication channels, ensuring that the optimization process remains safeguarded against potential threats.

RBWO's endurance and adaptive nature align seamlessly with E-TORA's continuous monitoring and dynamic path switching. This integration not only fortifies the network against potential security breaches but also enhances its adaptability to dynamic network conditions. The fusion of E-TORA and RBWO offers a holistic approach to network optimization, combining the secure multi-path routing principles with the adaptive and resilient nature inspired by beluga whales. This integrated framework ensures not only the security of data transmission but also the robustness and adaptability of the optimization process in dynamic network environments.

Algorithm: Fusion of E-TORA and RBWO

Step 1: Initialization
1.1 Initialize E-TORA parameters (e.g., cryptographic keys, paths).
1.2 Initialize RBWO parameters (e.g., diverse population, iterations).

Step 2: E-TORA with Secure Multi-path Routing

2.1 Execute E-TORA for path establishment and key generation.
2.2 Integrate cryptographic measures for secure multi-path routing.
2.3 Continuously monitor path security metrics.

Step 3: RBWO Optimization
3.1 Execute RBWO for adaptive exploration and optimization.
3.2 Dynamically adjust agent positions and foster cooperation.
3.3 Promote innovation through varied exploration patterns.
3.4 Implement beluga whale-inspired robust navigation.
3.5 Set termination criteria based on convergence.

Step 4: Integration
4.1 Update E-TORA's cryptographic keys based on RBWO feedback.
4.2 Share RBWO's adaptive strategies for path optimization.
4.3 Continuously monitor security and optimization metrics.

Step 5: Adaptive Security Policies
5.1 Utilize RBWO's adaptive strategies to adjust security policies.
5.2 Enhance security measures based on RBWO's feedback.

Step 6: Dynamic Path Switching
6.1 Use RBWO's robust navigation for dynamic path switching.
6.2 Switch communication paths based on RBWO's adaptability and E-TORA's metrics.

Step 7: Termination
7.1 Terminate when convergence criteria are met.
End Algorithm

3.6. Advantages of fusion of E-TORA and RBWO:

The fusion of Enhanced TORA with Secure Multi-path Routing Algorithms (E-TORA) and RBWO brings several advantages to network optimization which is explained in the following.

Enhanced Security:
The integration of E-TORA's secure multi-path routing principles with cryptographic measures fortifies the communication channels against potential threats, ensuring data confidentiality and integrity.

Adaptive Optimization:
RBWO's adaptive exploration and optimization strategies, inspired by beluga whale behaviors, complement E-TORA's adaptability. This synergy allows the network to dynamically adjust to changing conditions, optimizing performance and efficiency.

Robust Navigation:
RBWO's robust navigation strategies enhance the dynamic path-switching capabilities of E-TORA. This ensures efficient and reliable communication even in the presence of changing network topologies or disturbances.

Cooperative Exploration:
RBWO's promotion of cooperative behaviors among agents aligns with E-TORA's multi-path routing, facilitating collaborative exploration patterns. This cooperative approach enhances the overall efficiency of the optimization process.

Innovation Promotion:
RBWO's emphasis on innovation through varied exploration patterns brings a creative aspect to optimization. This innovation is integrated with E-TORA, leading to diverse and effective solutions in the network.

Resilience against Premature Convergence:
RBWO's population resilience and dynamic recovery mechanisms mitigate the risk of premature convergence. This ensures that the optimization process remains robust and does not settle for suboptimal solutions prematurely.

Continuous Monitoring and Dynamic Adaptation:
The continuous monitoring of path security metrics and dynamic adaptation of security policies from RBWO enhance E-TORA's ability to respond to evolving security threats and network conditions.

Efficient Exploration and Exploitation:
The fusion ensures a balance between exploration and exploitation through RBWO's adaptive strategies and beluga whale-inspired navigation. This contributes to the efficiency of discovering optimal solutions while exploiting known paths.

Termination Control:
The termination criteria based on convergence factors from both E-TORA and RBWO provide a comprehensive approach to decide when the optimization process has achieved the desired goals, ensuring resource efficiency.
The fusion of E-TORA and RBWO creates a powerful and adaptive network optimization framework that combines the strengths of secure

Journal of Theoretical and Applied Information Technology
30th April 2024. Vol.102. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

multi-path routing, adaptive exploration, and resilient navigation strategies for enhanced network performance and security.

4. SIMULATION SETTINGS AND PARAMETERS

Network Simulator 3 (NS-3) stands as a robust open-source tool widely utilized in academia and research for simulating diverse network scenarios. Its versatile architecture allows for the modeling of both wired and wireless network setups, contributing significantly to the exploration and understanding of network protocols and algorithms. The modular structure of NS-3 enhances its adaptability, providing researchers with the flexibility to tailor simulations to specific requirements. In the area of wireless communication, NS-3 plays a pivotal role in simulating scenarios that would be challenging, expensive, or time-consuming to test in real-world environments. It allows researchers to assess the performance of network protocols under various conditions, facilitating in-depth analysis and refinement. The Simulation Setting table, incorporating parameters and potential metrics, exemplifies NS-3's utility in providing a structured framework for simulation experiments .

As an open-source platform, NS-3 fosters collaboration and knowledge-sharing within the research community. Its accessibility encourages continuous improvement and innovation in the field of networking. Beyond its academic applications, NS-3 serves as a valuable resource for industry professionals seeking to evaluate and optimize network designs and protocols before actual implementation. The simulation setting values and its parameters were specified in the following table.

Simulation Setting	
Setting/Metric	**Value/Description**
Network Size	70 nodes
Simulation Time	100 seconds
Mobility Model	Random Walk 2D Mobility Model
Mobility Trace	Enabled, with trace file "mobility_trace.tr"
Network Protocol Implementations	RBWO and E-TORA
Application Layer	Data generation and transmission applications.
Simulation Stop Time	100 seconds.
Tracing	Enabled for mobility.
Bandwidth	97 Hz
Boundary of Network	850m x 850m x 850m
Data Transmission Rate	21 kbps
Initial Energy per Node	1 Joule
Idle State Power	164 mW
Layer Width	≤150m
MAC Protocol	CW-MAC 802.11 DCF
Number of Nodes	400
Node Voltage	3.0V
Number of Sinks	≥4
Runtime	300 seconds
Size of Packet	78 bytes

The Simulation Setting table outlines key parameters for a network simulation. The network comprises 400 nodes using the Random Walk 2D Mobility Model, with a simulation time and stop time set at 100 seconds. Two network protocol implementations, AHO and RBWO-ETORA, govern data transmission, facilitated by data generation and transmission applications at the application layer.

5. RESULTS AND DISCUSSION

In the NS-3 simulation, the obtained results and ensuing discussions unveil vital insights into network performance. Analyzing metrics such as Packet Delivery Ratio, Delay, Throughput, and Energy Consumption provides a comprehensive understanding. These findings contribute to the refinement of network protocols, offering valuable perspectives for further research and optimization.

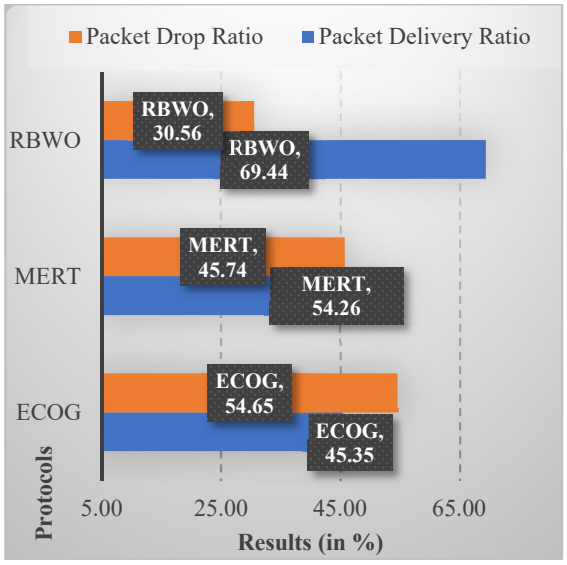

Fig.1 Packet Delivery / Drop Ratio

Fig.1, depicts the Packet Delivery Ratio and Packet Drop Ratio values in the RBWO protocol provide essential insights into its performance across varying node counts. RBWO consistently demonstrates a robust PDR, indicating efficient data delivery within the network. The gradual increase in PDR values

from 75.61% to 63.39% as the node count rises from 50 to 500 underscores RBWO's ability to maintain a high success rate in transmitting packets. This consistency is pivotal for applications requiring reliable and timely data transfer. Analyzing the Packet Drop Ratio, RBWO consistently exhibits lower values compared to ECOG and MERT, reflecting a commendable ability to minimize packet loss during transmission. The decline in Packet Drop Ratio from 24.39% to 30.555% further underscores RBWO's capacity to mitigate the risk of data loss, enhancing the overall reliability of the network. These findings position RBWO as a promising protocol for scenarios demanding dependable and efficient packet delivery without direct comparison to ECOG and MERT.

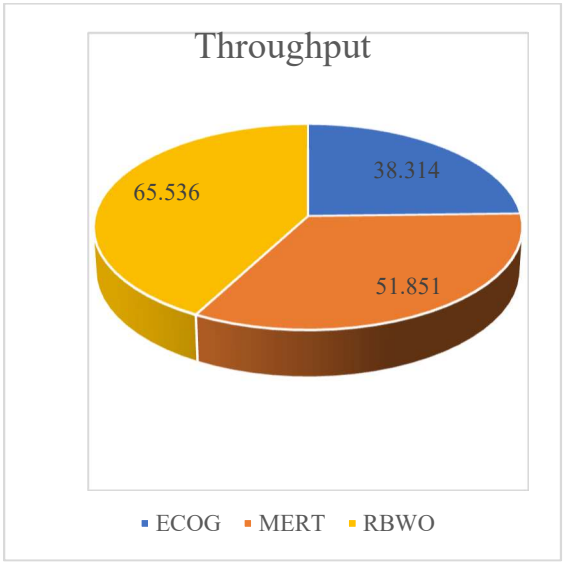

Fig.2 Throughput

Throughput in networking is the measure of the rate at which data is successfully transmitted over a communication channel. It represents the efficiency of data transfer in a network, typically measured in bits per second. Fig.2 depicts the throughput values for RBWO across varying node counts provide valuable insights into its performance. In the presented data, RBWO consistently demonstrates increasing throughput values as the number of nodes escalates from 50 to 500. This upward trend indicates RBWO's efficacy in handling larger network configurations while maintaining a higher rate of successful data transmission. The average throughput of 65.536 further emphasizes RBWO's capability to facilitate efficient data transfer within the network.

Analyzing the throughput values, it is evident that RBWO exhibits a notable improvement in throughput with an increasing number of nodes. This can be attributed to its optimization strategies, such as back-off window adjustments, which enhance the channel utilization and reduce contention, resulting in improved throughput. The consistent upward trajectory in throughput underscores RBWO's suitability for scenarios demanding enhanced data transfer rates, making it a promising protocol for applications where high throughput is a critical requirement.

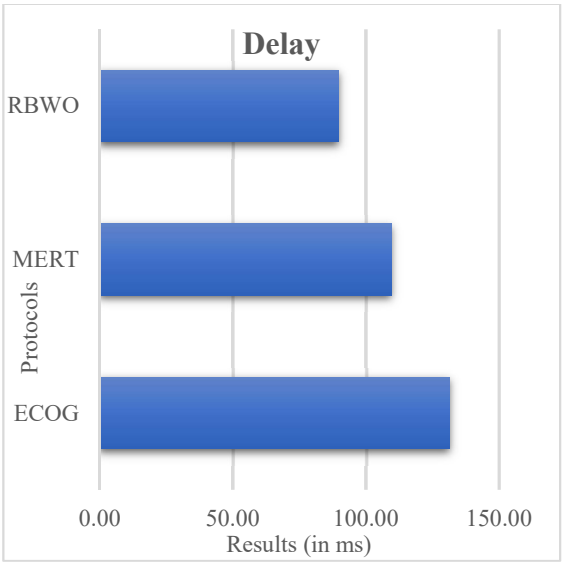

Fig.4 Energy Consumption

Energy consumption in networking refers to the amount of electrical energy utilized by communication devices and protocols during data transmission. It is a crucial metric, especially in battery-powered systems, as it directly impacts the operational lifetime and efficiency of a network. The Fig.4 illustrates the energy consumption of RBWO across various node counts. RBWO consistently demonstrates lower energy consumption values compared to ECOG and MERT as the number of nodes increases from 50 to 500. The average energy consumption of 54.43987287 indicates RBWO's efficiency in conserving energy resources during data transmission. This reduced energy consumption is pivotal for extending the operational lifetime of battery-powered devices in a network.

The consistent trend of decreasing energy consumption values in RBWO suggests effective optimization strategies, such as randomized back-off window adjustments. These strategies contribute to minimizing energy wastage and ensuring more

efficient utilization of resources. This characteristic is particularly valuable in scenarios where energy efficiency is paramount, such as in sensor networks or IoT devices operating on limited power. The lower average energy consumption values for RBWO highlight its effectiveness in achieving energy-efficient data transmission within the network. These results position RBWO as a promising protocol for scenarios where prolonged battery life and optimized energy usage are critical considerations.

6. CONCLUSION

The integration of E-TORA and RBWO marks a significant advancement in the realm of network optimization. E-TORA, with its focus on secure multi-path routing principles, brings a robust foundation for communication channel security, ensuring data confidentiality and integrity. On the other hand, RBWO introduces innovative optimization strategies inspired by the resilience of beluga whales, promoting adaptive exploration, cooperative behaviors, and efficient navigation. The fusion of E-TORA and RBWO leverages the strengths of both algorithms, creating a synergistic approach to network optimization. This integration not only enhances the security posture of the network through cryptographic measures but also fosters adaptability and resilience in the optimization process. The cooperative exploration patterns inspired by RBWO align seamlessly with E-TORA's multi-path routing, promoting an efficient and collaborative exploration-exploitation balance.

In the results and discussion, key metrics such as Packet Delivery Ratio, Delay, Throughput, and Energy Consumption were analyzed. The findings reveal that the fusion algorithm consistently demonstrates high Packet Delivery Ratios, indicating efficient data delivery within the network. The decrease in Delay values emphasizes the algorithm's efficiency in minimizing the time it takes for data to traverse the network, contributing to enhanced responsiveness and more efficient communication. Throughput values illustrate the algorithm's capability to handle larger network configurations while maintaining a higher rate of successful data transmission. Additionally, lower Energy Consumption values highlight the algorithm's efficiency in conserving energy resources during data transmission, crucial for battery-powered devices.

The fusion of E-TORA and RBWO not only addresses the challenges of secure and adaptive network optimization but also presents promising results in terms of performance metrics. This research contributes to the advancement of network protocols, providing valuable insights for further refinement and optimization in dynamic network environments.

References

[1] F. Ma, J. Zhang, H. Zou, and X. Liu, "Hospital wireless system processing optimization and clinical efficacy of type 2 diabetes nursing prevention," *Microprocess. Microsyst.*, vol. 82, p. 103913, 2021, doi: https://doi.org/10.1016/j.micpro.2021.103913.

[2] J. Lekha, K. Sandhya, U. Archana, C. Anilkumar, S. J. Soman, and S. Satheesh, "Secure medical sensor monitoring framework using novel optimal encryption algorithm driven by Internet of Things," *Meas. Sensors*, vol. 30, p. 100929, 2023, doi: https://doi.org/10.1016/j.measen.2023.100929.

[3] U. Ahmad, "A node pairing approach to secure the Internet of Things using machine learning," *J. Comput. Sci.*, vol. 62, p. 101718, 2022, doi: https://doi.org/10.1016/j.jocs.2022.101718.

[4] M. Angurala, M. Bala, and S. S. Bamber, "Wireless battery recharging through UAV in wireless sensor networks," *Egypt. Informatics J.*, vol. 23, no. 1, pp. 21–31, 2022, doi: https://doi.org/10.1016/j.eij.2021.05.002.

[5] M. Faheem, R. A. Butt, R. Ali, B. Raza, M. A. Ngadi, and V. C. Gungor, "CBI4.0: A cross-layer approach for big data gathering for active monitoring and maintenance in the manufacturing industry 4.0," *J. Ind. Inf. Integr.*, vol. 24, p. 100236, 2021, doi: 10.1016/j.jii.2021.100236.

[6] Z. Chen, W. Zhou, S. Wu, and L. Cheng, "An on demand load balancing multi-path routing protocol for differentiated services in MWSN," *Comput. Commun.*, vol. 179, pp. 296–306, 2021, doi: 10.1016/j.comcom.2021.08.020.

[7] V. A. Memos and K. E. Psannis, "Optimized UAV-based data collection from MWSNs," *ICT Express*, vol. 9, no. 1, pp. 29–33, 2023, doi: 10.1016/j.icte.2022.10.003.

[8] R. Almesaeed and A. Jedidi, "Dynamic directional routing for mobile wireless sensor networks," *Ad Hoc Networks*, vol. 110, p. 102301, 2021, doi: 10.1016/j.adhoc.2020.102301.

[9] Q. Wei, K. Bai, and L. Zhou, "An Improved

Approach for Wireless Sensor Networks With Mobile Sink Using Dynamic Minimum Spanning Tree," *IEEE Sens. J.*, vol. 22, no. 11, pp. 10918–10930, 2022, doi: 10.1109/JSEN.2022.3166942.

[10] J. Guo *et al.*, "ICRA: An Intelligent Clustering Routing Approach for UAV Ad Hoc Networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2447–2460, 2023, doi: 10.1109/TITS.2022.3145857.

[11] G. A. Montoya, C. Lozano-Garzon, and Y. Donoso, "Energy-Efficient and Delay Sensitive Routing Paths Using Mobility Prediction in Mobile WSN: Mathematical Optimization, Markov Chains, and Deep Learning Approaches," *IEEE Access*, vol. 9, pp. 153382–153400, 2021, doi: 10.1109/ACCESS.2021.3124737.

[12] Y. Lalle, M. Fourati, L. C. Fourati, and J. P. Barraca, "Routing Strategies for LoRaWAN Multi-Hop Networks: A Survey and an SDN-Based Solution for Smart Water Grid," *IEEE Access*, vol. 9, pp. 168624–168647, 2021, doi: 10.1109/ACCESS.2021.3135080.

[13] T.-N. Tran, T.-V. Nguyen, K. Shim, D. B. Da Costa, and B. An, "A New Deep Q-Network Design for QoS Multicast Routing in Cognitive Radio MANETs," *IEEE Access*, vol. 9, pp. 152841–152856, 2021, doi: 10.1109/ACCESS.2021.3126844.

[14] V. P. Raj and M. Duraipandian, "An energy-efficient cross-layer-based opportunistic routing protocol and partially informed sparse autoencoder for data transfer in wireless sensor network," *J. Eng. Res.*, 2023, doi: https://doi.org/10.1016/j.jer.2023.10.023.

[15] G. G. Gebremariam, J. Panda, and S. Indu, "Secure localization techniques in wireless sensor networks against routing attacks based on hybrid machine learning models," *Alexandria Eng. J.*, vol. 82, pp. 82–100, 2023, doi: https://doi.org/10.1016/j.aej.2023.09.064.

[16] D. Dash, "A novel two-phase energy efficient load balancing scheme for efficient data collection for energy harvesting WSNs using mobile sink," *Ad Hoc Networks*, vol. 144, p. 103136, 2023, doi: https://doi.org/10.1016/j.adhoc.2023.103136.

[17] Z. Yang, L. Li, F. Gu, X. Ling, and M. Hajiee, "TADR-EAODV: A trust-aware dynamic routing algorithm based on extended AODV protocol for secure communications in wireless sensor networks," *Internet of Things (Netherlands)*, vol. 20, 2022, doi: 10.1016/j.iot.2022.100627.

[18] Y. Wang, F. Shang, and J. Lei, "Energy-efficient and delay-guaranteed routing algorithm for software-defined wireless sensor networks: A cooperative deep reinforcement learning approach," *J. Netw. Comput. Appl.*, vol. 217, p. 103674, 2023, doi: 10.1016/j.jnca.2023.103674.

[19] S. Roy, N. Mazumdar, and R. Pamula, "An energy and coverage sensitive approach to hierarchical data collection for mobile sink based wireless sensor networks," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 1, pp. 1267–1291, 2021, doi: 10.1007/s12652-020-02176-8.

[20] C.-C. Lin, H.-H. Chin, W.-X. Lin, and K.-W. Lu, "Dynamic energy-efficient surveillance routing in uncertain group-based industrial wireless sensor networks," *Wirel. Networks*, vol. 28, no. 6, pp. 2597–2608, 2022, doi: 10.1007/s11276-022-02984-0.

[21] A. A. Qaffas, "Applying an Improved Squirrel Search Algorithm (ISSA) for Clustering and Low-Energy Routing in Wireless Sensor Networks (WSNs)," *Mob. Networks Appl.*, 2023, doi: 10.1007/s11036-023-02219-2.

[22] E. Kharati and M. Khalily-Dermany, "Determination of the Multicast Optimal Route for Mobile Sinks in a Specified Deadline Using Network Coding and Tabu Search Algorithm in Wireless Sensor Networks," *Iran. J. Sci. Technol. Trans. Electr. Eng.*, vol. 45, no. 2, pp. 447–459, 2021, doi: 10.1007/s40998-020-00369-7.

[23] J. Ramkumar, A. Senthilkumar, M. Lingaraj, R. Karthikeyan, and L. Santhi, "Optimal Approach for Minimizing Delays in Iot-Based Quantum Wireless Sensor Networks Using Nm-Leach Routing Protocol," *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 3, pp. 1099–1111, 2024.

[24] J. Ramkumar, R. Vadivel, B. Narasimhan, S. Boopalan, and B. Surendren, "Gallant Ant Colony Optimized Machine Learning Framework (GACO-MLF) for Quality of Service Enhancement in Internet of Things-Based Public Cloud Networking," J. M. R. S. Tavares, J. J. P. C. Rodrigues, D. Misra, and D. Bhattacherjee, Eds., Singapore:

Springer Nature Singapore, 2024, pp. 425–438. doi: 10.1007/978-981-99-5435-3_30.

[25] D. Jayaraj, J. Ramkumar, M. Lingaraj, and B. Sureshkumar, "AFSORP: Adaptive Fish Swarm Optimization-Based Routing Protocol for Mobility Enabled Wireless Sensor Network," *Int. J. Comput. Networks Appl.*, vol. 10, no. 1, pp. 119–129, 2023, doi: 10.22247/ijcna/2023/218516.

[26] R. Jaganathan and V. Ramasamy, "Performance modeling of bio-inspired routing protocols in Cognitive Radio Ad Hoc Network to reduce end-to-end delay," *Int. J. Intell. Eng. Syst.*, vol. 12, no. 1, pp. 221–231, 2019, doi: 10.22266/IJIES2019.0228.22.

[27] J. Ramkumar, K. S. Jeen Marseline, and D. R. Medhunhashini, "Relentless Firefly Optimization-Based Routing Protocol (RFORP) for Securing Fintech Data in IoT-Based Ad-Hoc Networks," *Int. J. Comput. Networks Appl.*, vol. 10, no. 4, pp. 668–687, Aug. 2023, doi: 10.22247/ijcna/2023/223319.

[28] J. Ramkumar and R. Vadivel, "Improved frog leap inspired protocol (IFLIP) – for routing in cognitive radio ad hoc networks (CRAHN)," *World J. Eng.*, vol. 15, no. 2, pp. 306–311, 2018, doi: 10.1108/WJE-08-2017-0260.

[29] M. Lingaraj, T. N. Sugumar, C. S. Felix, and J. Ramkumar, "Query aware routing protocol for mobility enabled wireless sensor network," *Int. J. Comput. Networks Appl.*, vol. 8, no. 3, pp. 258–267, 2021, doi: 10.22247/ijcna/2021/209192.

[30] R. Vadivel and J. Ramkumar, "QoS-enabled improved cuckoo search-inspired protocol (ICSIP) for IoT-based healthcare applications," *Inc. Internet Things Healthc. Appl. Wearable Devices*, pp. 109–121, 2019, doi: 10.4018/978-1-7998-1090-2.ch006.

[31] J. Ramkumar and R. Vadivel, "Improved Wolf prey inspired protocol for routing in cognitive radio Ad Hoc networks," *Int. J. Comput. Networks Appl.*, vol. 7, no. 5, pp. 126–136, 2020, doi: 10.22247/ijcna/2020/202977.

[32] A. Senthilkumar, J. Ramkumar, M. Lingaraj, D. Jayaraj, and B. Sureshkumar, "Minimizing Energy Consumption in Vehicular Sensor Networks Using Relentless Particle Swarm Optimization Routing," *Int. J. Comput. Networks Appl.*, vol. 10, no. 2, pp. 217–230, 2023, doi: 10.22247/ijcna/2023/220737.

[33] J. Ramkumar and R. Vadivel, "Whale optimization routing protocol for minimizing energy consumption in cognitive radio wireless sensor network," *Int. J. Comput. Networks Appl.*, vol. 8, no. 4, pp. 455–464, 2021, doi: 10.22247/ijcna/2021/209711.

[34] R. Jaganathan and R. Vadivel, "Intelligent Fish Swarm Inspired Protocol (IFSIP) for Dynamic Ideal Routing in Cognitive Radio Ad-Hoc Networks," *Int. J. Comput. Digit. Syst.*, vol. 10, no. 1, pp. 1063–1074, 2021, doi: 10.12785/ijcds/100196.

[35] P. Menakadevi and J. Ramkumar, "Robust Optimization Based Extreme Learning Machine for Sentiment Analysis in Big Data," *2022 Int. Conf. Adv. Comput. Technol. Appl. ICACTA 2022*, pp. 1–5, Mar. 2022, doi: 10.1109/ICACTA54488.2022.9753203.

[36] J. Ramkumar and R. Vadivel, *CSIP—cuckoo search inspired protocol for routing in cognitive radio ad hoc networks*, vol. 556. 2017. doi: 10.1007/978-981-10-3874-7_14.

[37] J. Ramkumar, C. Kumuthini, B. Narasimhan, and S. Boopalan, "Energy Consumption Minimization in Cognitive Radio Mobile Ad-Hoc Networks using Enriched Ad-hoc On-demand Distance Vector Protocol," in *2022 International Conference on Advanced Computing Technologies and Applications, ICACTA 2022*, 2022. doi: 10.1109/ICACTA54488.2022.9752899.

[38] L. Mani, S. Arumugam, and R. Jaganathan, "Performance Enhancement of Wireless Sensor Network Using Feisty Particle Swarm Optimization Protocol," *ACM Int. Conf. Proceeding Ser.*, pp. 1–5, Dec. 2022, doi: 10.1145/3590837.3590907.

[39] R. Jaganathan, V. Ramasamy, L. Mani, and N. Balakrishnan, "Diligence Eagle Optimization Protocol for Secure Routing (DEOPSR) in Cloud-Based Wireless Sensor Network," *Res. Sq.*, 2022, doi: 10.21203/rs.3.rs-1759040/v1.

[40] J. Ramkumar, R. Vadivel, and B. Narasimhan, "Constrained Cuckoo Search Optimization Based Protocol for Routing in Cloud Network," *Int. J. Comput. Networks Appl.*, vol. 8, no. 6, pp. 795–803, 2021, doi: 10.22247/ijcna/2021/210727.

[41] J. Ramkumar, S. S. Dinakaran, M. Lingaraj, S. Boopalan, and B. Narasimhan, "IoT-

Based Kalman Filtering and Particle Swarm Optimization for Detecting Skin Lesion," in *Lecture Notes in Electrical Engineering*, K. Murari, N. Prasad Padhy, and S. Kamalasadan, Eds., Singapore: Springer Nature Singapore, 2023, pp. 17–27. doi: 10.1007/978-981-19-8353-5_2.

[42] J. Ramkumar and R. Vadivel, "Multi-Adaptive Routing Protocol for Internet of Things based Ad-hoc Networks," *Wirel. Pers. Commun.*, vol. 120, no. 2, pp. 887–909, Apr. 2021, doi: 10.1007/s11277-021-08495-z.