Application And Performance Evaluation Of Artificial Immune System-Negative Selection Algorithm For Anomalies Detection In Distributed Sensor Networks

Agbu, Alexander Uchenna[1]
Department / Office of National Space Research and Development Agency, (NASRDA)
Abuja, Nigeria
agbualexuche@gmail.com

Ofonime Dominic Okon[2]
Department Of Electrical/Electronic And Computer Engineering,
University of Uyo, Akwa Ibom State Nigeria

Bliss Utibeabasi Stephen[3]
Department Of Electrical/Electronic and Computer Engineering,
University of Uyo, Akwa Ibom State Nigeria

***Abstract*— In this paper, application and performance evaluation of artificial immune system-negative selection algorithm for anomalies detection in distributed sensor networks is presented. The negative selection algorithm (NSA) is used to address the challenge of injection of false data into the distributed sensor network by an attacker when a sensor node or the key management system in a network is compromised. Particularly, the NSA serves as malicious behavior detection strategy to identify the misbehaving nodes in the network, Then, revocation procedures are engaged to revoke the misbehaving nodes and their keys from the network immediately after detecting the faulty nodes or compromise. The performance of the NSA scheme is evaluated in terms of false positives, true positives, false negatives, and true negatives. In all, the results from the experimental setups show that the NSA performs better than the CSA in terms of both detection rate and false positive rate.**

Keywords*— *Anomalies Detection, Homogeneous Wireless Sensor Networks, Artificial Immune System, Distributed Sensor Networks, Clustering, Negative Selection Algorithm

1. Introduction

Over the years, there has been tremendous increase in wireless sensor networks (WSNs) applications [1,2,3,4,5,6,7,8]. Mostly, the sensor nodes in the WSNs are usually installed physically in insecure areas where they are susceptible to compromise [9,10,11,12,13,14,15]. Although, some forms of secure key establishments and management mechanism, such as pairwise keys approach can be adopted as a solution, however, when a sensor node is captured, it is presumed that all information and stored key materials will be exposed to the attacker. In the pairwise keys management strategy, the pairwise keys are stored by the potential neighbors of each sensor node [16,17,18,19]. After an attacker launches attack on one of its neighbor nodes, the attacker will be able to decrypt the information coming from other neighbor nodes directly. However, other links which are not involved directly in this communication will still be secure. Hence, the resiliency of the approach is high due to its deterministic nature.

However, the challenge is the injection of false data into the network by an attacker [20,21,22,23,24]. In this case, an efficient malicious behavior detection strategy is required to identify the misbehaving nodes and revoke them and their keys from the network. In the shared and homogeneous Wireless Sensor Networks (HWSNs), the resource constraint nature of sensor nodes limits the computation, memory, and communication resources which can be deployed for revocation [25,26,27,28]. Accordingly, in this paper, an efficient misbehaving detection scheme based on Artificial Immune System (AIS) for distributed sensor networks is presented [29,30]. In addition, evaluation of the performance of the artificial immune system-negative selection algorithm is presented along with comparison of the performance of the algorithm with other anomalies detection methods for distributed sensor networks

2. Methodology

2.1 Irregularity Detection in Wireless Sensor Network

This paper proposes bio-inspired solution using Negative Selection Algorithm (NSA) of the Artificial Immune System (AIS) for anomalies detection in WSNs. For this reason, an enhanced NSA is implemented and a detector set that holds anomalous packets only will be defined.

The NSA has been useful for detecting anomalies in different ways. However, in this work, NSA is used with some modifications. The system learning is performed for a large dataset and a detector set is generated. Once this step is completed, an injection feature in the detector set is proposed. With the aid of this feature, the detector set can be updated at any stage. This injection procedure is known as vaccination.

The proposed scheme has learning and testing phases which are shown in Figure 1 and Figure 2, respectively. The basic NSA that is capable of doing a single classification is first implemented and the anomalies from the dataset are detected. At this point, there exist two classes, namely, self-set and non-self. Subsequently, the remaining processing is performed on detected non-self and three different anomalies are classified; which are, sensor network packets delayed, packets dropped, and wormholes are detected.

In Figure 1, self-strings are paired with randomly generated strings using character-by-character pairing. Those strings that get matched during the pairing are rejected while those that do not get matched are moved to the detector set. Vaccination can be used to update detector set anytime, and this makes it more efficient. Vaccination enables a user to input any non-self-pattern directly into the detector set, as part of the strategy which makes the function of the detector set more effective.

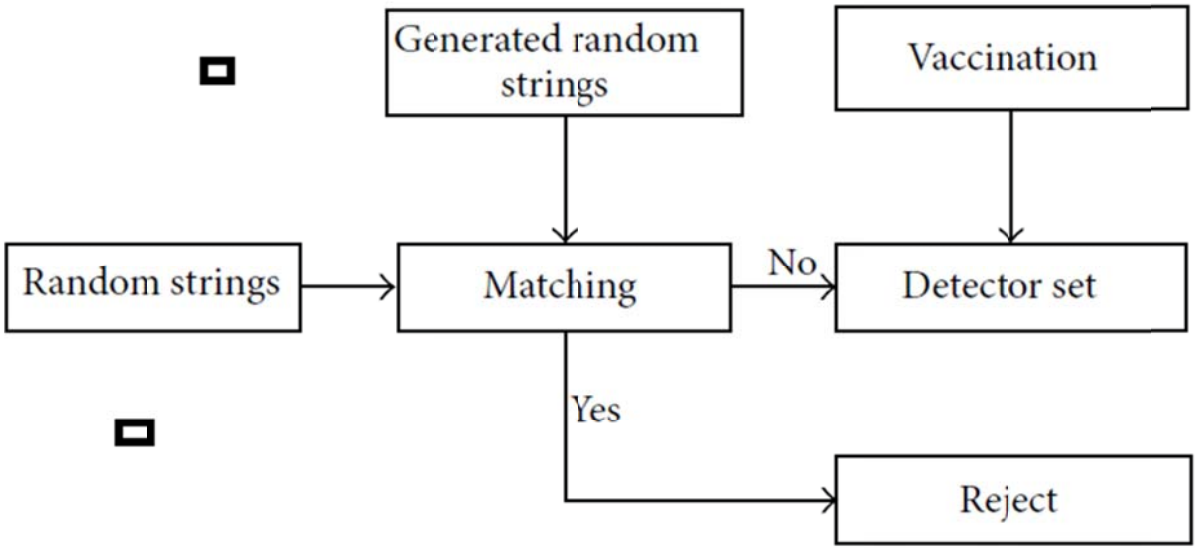

Figure 1: Proposed NSA learning for anomaly detection

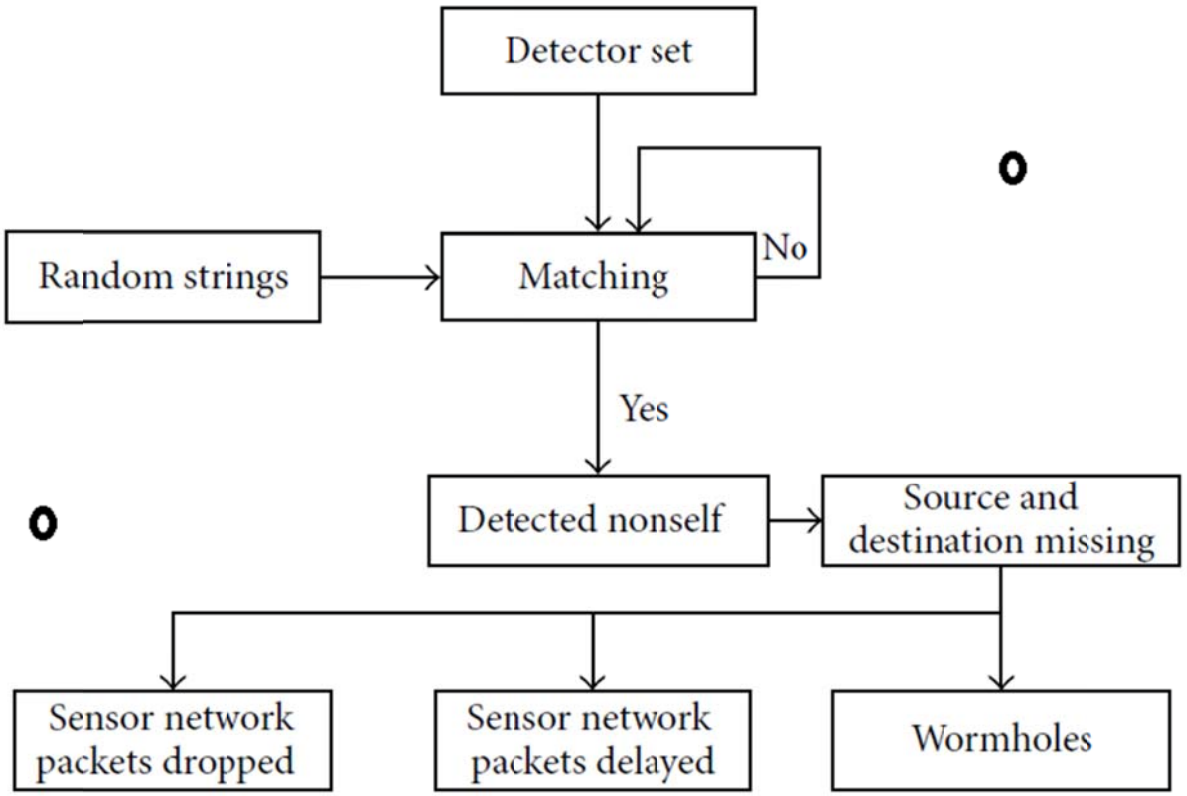

Figure 2 Proposed NSA for testing anomaly detection

In Figure 2, randomly generated strings are paired with the detector set using character-by-character pairing. Those strings that get matched during the pairing are declared as non-self. Source and destination pairing are carried out on non-self and they are further classified as sensor network packets delayed, packets dropped, and wormholes.

2.2 Assumptions and Protocols

In a sensor network, there exists $N = (n(t), e(t))$, where $n(t)$ and $e(t)$ denote the set of nodes and edges, respectively, at any time t. Two nodes, say node A and node B will be able to communicate if and only if they are within the radio transmission range of each other. Then, the route between two nodes in an adhoc network is setup using any routing protocol. In this paper, an Adhoc On-demand Distance Vector (AODV) routing protocol is adopted. Connection is established by this protocol only when it is necessary to route to destination. In this scenario, Route Request (RREQ) is forwarded to all nodes within the network.

Destination or intermediate node replies with a Route Reply (RREP) control packet. This RREP is routed through the same path towards the source as that of RREQ. If eventually, while moving towards the source, the next node ceases to reply, a Request Error (RERR) packet is forwarded to the connection initiator. In ad hoc communication, each node keeps its own routing table, which contains information about the destination node, all registered routes, and hop count for a given destination. Since the transmission is ad hoc, wireless scheme should be synchronized and this is performed on the basis of medium contention. In IEEE 802.11 MAC protocol, carrier sensing is performed by RTS-CTS-DATA-ACK handshake. This handshake can be disabled for situations where packet size is the same or smaller than RTS threshold. The default value for RTS threshold is given as 2347 bytes. This threshold can be modified by a data traffic pattern. The highest data transformation rate for IEEE 802.11b and IEEE 802.11g is 11 and 54Mbit/s, respectively.

2.3 Adding New Nodes

One of the desired features in a scalable key management scheme of WSNs is the ability to add new sensors to the network. It is pertinent for the newly joined sensors to establish secret key with the existing nodes. However, it is important to verify that the prospective new node to be added is not an attacker node. The proposed strategy is robust for adding legitimate sensor to the network. Once this sensor L_x is added, it determines its neighbors using node discovery, and then sends join request to the cluster head (CH), for which it has the strongest RSSI values as given below

$$L_x \rightarrow CH: id_{L_x}, nonce_{L_x}, List, MAC_{K_M,L_x}(id_{L_x} \,||\, nonce_{L_x} ||\, List) \quad (1)$$

Where, id_{L_x} denotes the identity of the legitimate sensor, $nonce_{L_x}$ denotes the random number string generated by a legitimate sensor, MAC_{K_M,L_x} denotes the message authentication code calculated using the master key on the sensor message. The node L_x gets authenticated by the CH by verifying the MAC. If authentication is successful, CH

determines the distributed key for each $L_x's$ neighbors and unicasts the distributed key message to L_x and its neighbors.

2.4 Cluster Key Setup

Cluster key is utilized by both cluster members (CM) and CH to securely broadcast messages within cluster. Once a shared pairwise key between cluster members is established, CH generates cluster key K_C, which is sent to each cluster member. K_C is encrypted with the corresponding shared key between the cluster member and CH. For instance, CH can send to L_u (cluster member) the following message:

$$CH \rightarrow L_u: E_{K_{CH},L_u}(KC) \quad (2)$$

Where K_{CH} denotes the shared key between the legitimate sensor and the cluster head

2.5 Key Revocation

Revocation procedures are engaged immediately after detecting faulty nodes or compromise. The duty of the base station is to monitor sensor behavior and detect a sensor compromise or failure. If a node is compromised, the base station sends this information to the corresponding CH. The CH then broadcasts to its member the revocation message which is made up of the list of key ids to be revoked, where the message is signed with KC. The *Revocation* message is formulated as follows:

$$list\left(id_{gk_1}, id_{gk_2}, \dots, id_{gk_r}\right), MAC_{KC}(list) \quad (3)$$

When any legitimate sensor receives a revocation message, it verifies the MAC to check the integrity of the message and to find those key ids it the key ring, and remove the keys (if found). Some links may disappear after key revocation and the affected nodes must reconfigure those links by restarting the distributed key discovery phase.

2.6 Performance Evaluation of the Anomaly Detection

Among the numerous performance measures, the most popular ones for analyzing the performance of NSA and other AIS algorithms are false positives, true positives, false negatives, and true negatives. These outlined measures are defined below:

i. False positives (FPs) are described when self-patterns are mistakenly identified as non-self-patterns
ii. True positives (TPs) are described when self-pattern are rightly identified as self-pattern
iii. True negatives (TNs) are described when non-self-patterns are rightly identified as non-self-pattern
iv. False negatives (FNs) are described when non-self-patterns are identified as self-pattern

Detection rate (DR), false positive rate (FPR), and accuracy can be calculated by these measures. The computation blueprint is as shown in Equation 4 to Equation 6.

$$DR = \frac{TP}{TP+FN} \quad (4)$$

$$FPR = \frac{FP}{FP+TN} \quad (5)$$

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (6)$$

3. Results and discussion

3.1 The results of experiment1 test for the anomaly detection

Three set of experiments were conducted to test for the anomaly detection. In the first experiment, NSA for small dataset which have normal packets only was implemented. A total anomaly of 10 was inserted at runtime and was detected. Simulations were executed in MATLAB 2019 RA and it took about 8 – 10 seconds to execute. Figure 3 presents the screenshot for the NSA simulation with random anomalies. The average results computed for the proposed simulation is presented in Table 1 in comparison with the popular Agent based intrusion (ABI) and Immune-inspired detection and recovery (IDR) schemes. The data presented in Table 1 is depicted graphically in Figure 4.

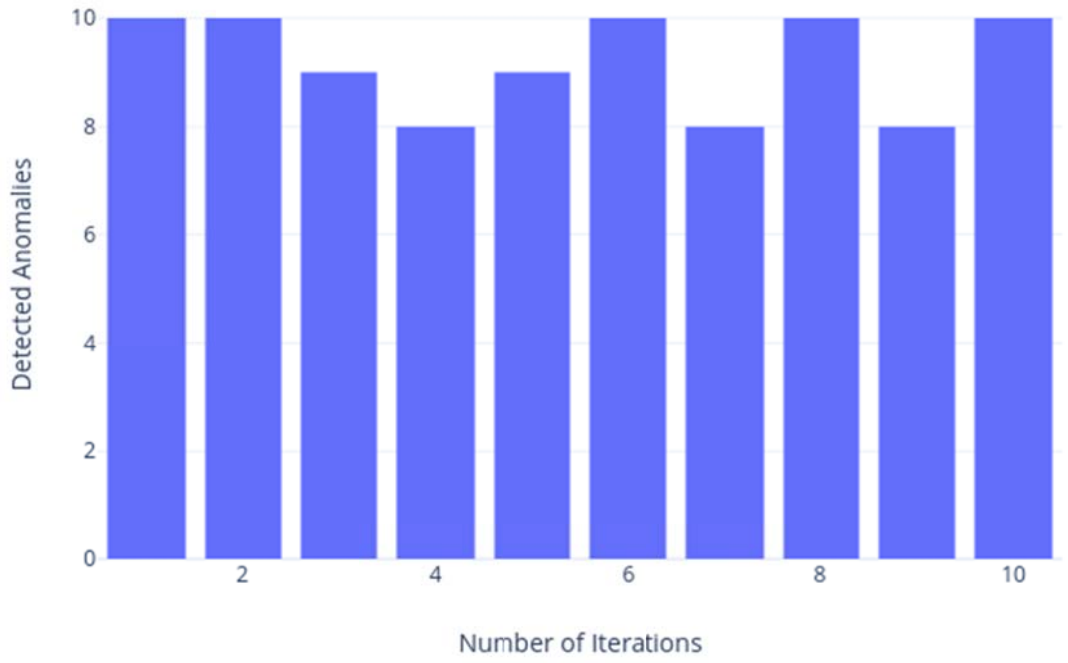

Figure 3: NSA simulation with random anomalies

Table 1: Results of anomalies test iteration.

Number of iterations	**Anomalies detected (Proposed Scheme)**	**Anomalies detected (ABI) scheme**	**Anomalies detected (IDR)**
Iteration 1	10	8	6
Iteration 2	10	7	7
Iteration 3	9	4	6
Iteration 4	8	7	6
Iteration 5	9	7	7
Iteration 6	10	8	7
Iteration 7	8	7	5
Iteration 8	10	6	6
Iteration 9	8	7	7
Iteration 10	10	8	7

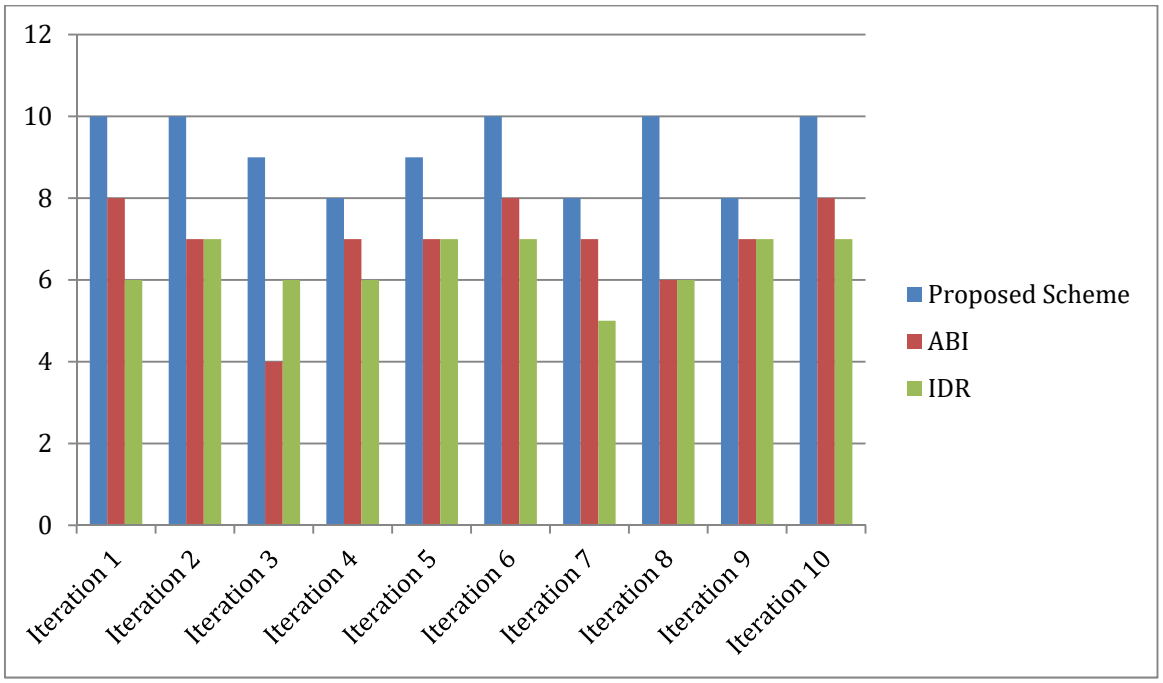

Figure 4: Result of anomalies test iterations

3.2 The results of experiment 2 test for the anomaly detection

In the second experiment, the network sensor dataset provided by [31] were used. The enhanced NSA was implemented, thus self and non-self-network packets were identified. First and foremost, the incoming network strings are compared with self-strings. Those strings that get matched are rejected while others are moved to the detector set. Next, arbitrary strings are compared with the detector set and those that get matched are identified as non-self. Figure 5 depicts the wormholes, packet delayed, and packet dropped found and the average results computed for this simulation is presented in Table 2.

Table 2: Average results for string matching

SN.	Normal packets	Packets delayed	Packets dropped	Wormholes
1	89	20	31	18
2	87	22	38	19
3	84	25	22	20
4	88	22	22	19
5	89	20	30	18

Note that all values presented inn Table 4.3 are in 10^3.

The results presented in Table 4.3 are compared with the original dataset and the values for TP, FN, FP, and TN are computed. The detection rate for this experiment is observed to be 97.3%, while the $FPR = \pm 2.6\%$. DR represents the intermediate result which comprises of the FP and TN. However, the accuracy of the scheme is observed to be 89.1%.

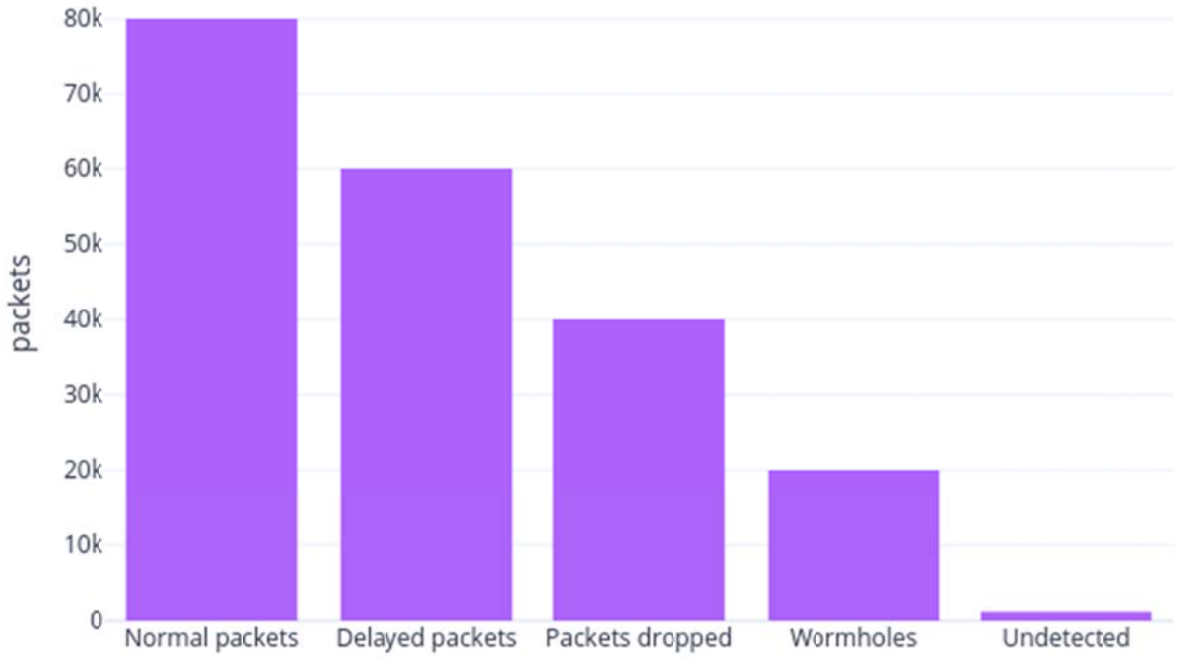

Figure 5: Anomalies detected

3.3 The results of experiment 3 test for the anomaly detection

In experiment 3, clonal selection algorithm (CSA) is implemented for comparison with the proposed NSA. The CSA approach models the production of antigens, which are then bound to specific antigens. A key lock mechanism can be deployed in certain cases for binding processes. It is assumed that those antibodies, which recognize the antigen, are selected for comparison. After comparison, a detector set is generated. The working principles of CSA are presented as follows:

i. Create initial population of antibodies
ii. Execute clonal choice for high affinity comparison (assume 76% threshold)
iii. Generate detector set for antibodies that match the threshold
iv. Introduce arbitrarily generated antibodies to the system
v. Use clonal selection generated detector to identify self and non-self.

The aim of experiment 3 is to compare the performance of NSA with CSA on different data subsets and the results of both FP and anomaly detection is compared as presented in Table 3.

Table 3: Comparison of NSA and CSA

		NSA		CSA	
Datasets	**Total packets**	**Anomalous packets**	**False positives**	**Anomalous packets**	**False positives**
Dataset Part 1	5619	3549	±1.80%	3453	±2.20%
Dataset Part 2	4275	2710	±1.50%	2693	±2.80%
Dataset Part 3	1212	643	±2.10%	657	±1.20%
Dataset Part 4	9435	5821	±2.50%	5863	±3.10%
Dataset Part 5	1263	866	±1.90%	852	±1.50%
Dataset Part 6	3008	1089	±2.20%	1002	±1.77%
Dataset Part 7	4540	1653	±1.20%	1640	±2.32%
Dataset Part 8	1429	463	±2.60%	496	±2.20%
Dataset Part 9	821	283	±1.23%	246	±1.67%
Dataset Part 10	4763	1389	±2.60%	345	±2.10%

The performance of both algorithms on particular datasets is derived from the number of anomalies detected and false positive ratio. From Table 3, it is shown that for dataset parts 1, 2, 4, 5, 7, 9, and 10, NSA yields better result and for datasets 3, 6, and 8, CSA performs better.

This comparison is also performed for the entire dataset. In the first scenario, only some files of the same dataset are

engaged for comparison. In the second case, results of both the algorithms for the entire dataset are produced, as presented in Figure 4.10. Normal and anomalous sensor network packets, undetected packets, and FPR for both the algorithms are presented. The results of the experiments illustrate that the detection rate of NSA is 97.3% and the FPR is ±2.6%, while, for CSA, the detection rate is 88% and false positive rate is 3.4%. This obviously entails that NSA performs better that CSA in terms of both detection rate and false positive rate.

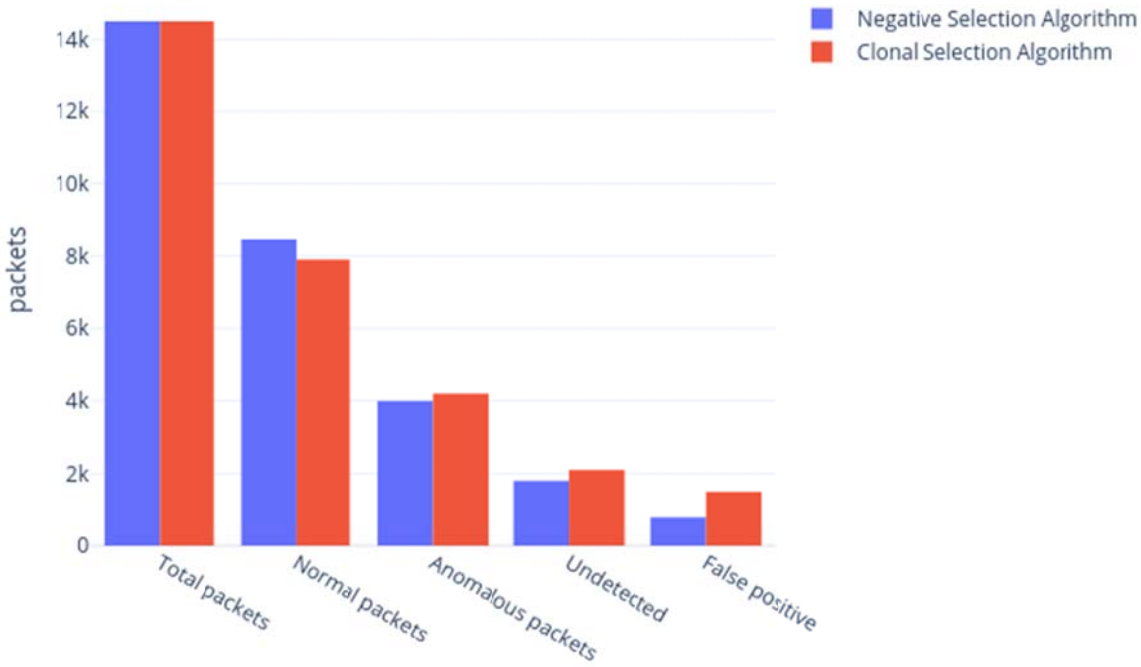

Figure 6: Comparison between NSA and CSA for complete dataset

4. Conclusion

A bio-inspired anomalies detection in Wireless Sensor Network (WSN) solution using Negative Selection Algorithm (NSA) of the Artificial Immune System (AIS) is presented. This is used to address the challenge of injection of false data into the network by an attacker when a sensor node or the key management system in a WSN is compromised by an attacker.

Notably, the NSA presented in this paper serves as malicious behavior detection strategy to identify the misbehaving nodes in the WSN. Revocation procedures are engaged immediately after detecting faulty nodes or compromise. The revocation procedure is used to revoke the misbehaving nodes and their keys from the network. The performance of the NSA scheme is evaluated in terms of false positives, true positives, false negatives, and true negatives. Equally, the performance of the NSA scheme is compared with that of clonal selection algorithm (CSA). In all, the NSA performs better than the CSA in terms of both detection rate and false positive rate.

References

1. Chaudhary, D. D., Nayse, S. P., & Waghmare, L. M. (2011). Application of wireless sensor networks for greenhouse parameter control in precision agriculture. *International Journal of Wireless & Mobile Networks (IJWMN)*, *3*(1), 140-149.
2. Cavalcanti, D., Das, S., Wang, J., & Challapali, K. (2008, August). Cognitive radio based wireless sensor networks. In *2008 Proceedings of 17th International Conference on Computer Communications and Networks* (pp. 1-6). IEEE.
3. Al Ameen, M., Liu, J., & Kwak, K. (2012). Security and privacy issues in wireless sensor networks for healthcare applications. *Journal of medical systems*, *36*(1), 93-101.
4. Singh, H., & Singh, D. (2016, December). Taxonomy of routing protocols in wireless sensor networks: A survey. In *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)* (pp. 822-830). IEEE.
5. Tubaishat, M., Zhuang, P., Qi, Q., & Shang, Y. (2009). Wireless sensor networks in intelligent transportation systems. *Wireless communications and mobile computing*, *9*(3), 287-302.
6. reza Akhondi, M., Talevski, A., Carlsen, S., & Petersen, S. (2010, April). Applications of wireless sensor networks in the oil, gas and resources industries. In *2010 24th IEEE International Conference on Advanced Information Networking and Applications* (pp. 941-948). IEEE.
7. Khedo, K. K., Perseedoss, R., & Mungur, A. (2010). A wireless sensor network air pollution monitoring system. *arXiv preprint arXiv:1005.1737*.

8. Suo, H., Wan, J., Huang, L., & Zou, C. (2012, March). Issues and challenges of wireless sensor networks localization in emerging applications. In *2012 International Conference on Computer Science and Electronics Engineering* (Vol. 3, pp. 447-451). IEEE.
9. Boubiche, D. E., Athmani, S., Boubiche, S., & Toral-Cruz, H. (2021). Cybersecurity issues in wireless sensor networks: current challenges and solutions. *Wireless Personal Communications*, *117*(1), 177-213.
10. Messai, M. L. (2014). Classification of attacks in wireless sensor networks. *arXiv preprint arXiv:1406.4516*.
11. Behrens, H. W., & Candan, K. S. (2018, October). Adversarially-Resistant On-Demand Topic Channels for Wireless Sensor Networks. In *2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS)* (pp. 83-92). IEEE.
12. Othman, S. B., Bahattab, A. A., Trad, A., & Youssef, H. (2015). Confidentiality and integrity for data aggregation in WSN using homomorphic encryption. *Wireless Personal Communications*, *80*(2), 867-889.
13. Mcginthy, J. M., & Michaels, A. J. (2019). Secure industrial Internet of Things critical infrastructure node design. *IEEE Internet of Things Journal*, *6*(5), 8021-8037.
14. Kuchipudi, R., Qyser, A. A. M., & Balaram, V. S. (2016, March). An efficient hybrid dynamic key distribution in Wireless Sensor Networks with reduced memory overhead. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* (pp. 3027-3030). IEEE.
15. Bhandari, M. (2014). Various Security Threats in Wireless Sensor Networks and their Defense Schemes. *Global Journal of Computers & Technology*, *1*(1), 28-33.
16. Chelli, K. (2015, July). Security issues in wireless sensor networks: Attacks and countermeasures. In *Proceedings of the world congress on engineering* (Vol. 1, No. 20, pp. 876-3423).
17. Cheikhrouhou, O. (2016). Secure group communication in wireless sensor networks: a survey. *Journal of Network and Computer Applications*, *61*, 115-132.
18. Abdallah, W., & Boudriga, N. (2016, August). A location-aware authentication and key management scheme for wireless sensor networks. In *2016 22nd Asia-Pacific Conference on Communications (APCC)* (pp. 488-495). IEEE.
19. Gautam, A. K., & Kumar, R. (2021). A comprehensive study on key management, authentication and trust management techniques in wireless sensor networks. *SN Applied Sciences*, *3*(1), 1-27.
20. Li, S., Yılmaz, Y., & Wang, X. (2014). Quickest detection of false data injection attack in wide-area smart grids. *IEEE Transactions on Smart Grid*, *6*(6), 2725-2735.
21. Yu, Z. H., & Chin, W. L. (2015). Blind false data injection attack using PCA approximation method in smart grid. *IEEE Transactions on Smart Grid*, *6*(3), 1219-1226.
22. Liu, X., & Li, Z. (2017). False data attack models, impact analyses and defense strategies in the electricity grid. *The Electricity Journal*, *30*(4), 35-42.
23. Zhang, J., Chu, Z., Sankar, L., & Kosut, O. (2018). Can attackers with limited information exploit historical data to mount successful false data injection attacks on power systems?. *IEEE Transactions on Power Systems*, *33*(5), 4775-4786.
24. Ayad, A., Farag, H. E., Youssef, A., & El-Saadany, E. F. (2018, February). Detection of false data injection attacks in smart grids using recurrent neural networks. In *2018 IEEE power & energy society innovative smart grid technologies conference (ISGT)* (pp. 1-5). IEEE.
25. Sharma, D., Ojha, A., & Bhondekar, A. P. (2019). Heterogeneity consideration in wireless sensor networks routing algorithms: a review. *The journal of supercomputing*, *75*(5), 2341-2394.
26. Mehta, K., & Pal, R. (2017). Energy efficient routing protocols for wireless sensor networks: A survey. *International Journal of Computer Applications*, *165*(3), 41-46.
27. Gonçalves, D. D. O., & Costa, D. G. (2015). A survey of image security in wireless sensor networks. *Journal of Imaging*, *1*(1), 4-30.
28. Mallick, C., & Satpathy, S. (2018). Challenges and design goals of wireless sensor networks: A state-of-the-art review. *International Journal of Computer Applications*, *179*(28), 42-47.
29. Kumar, E. S., Kusuma, S. M., & Kumar, B. V. (2014, March). A random key distribution based artificial immune system for security in clustered wireless sensor networks. In *2014 IEEE Students' Conference on Electrical, Electronics and Computer Science* (pp. 1-7). IEEE.
30. Vargheese, V. S., Jayasudha, G., & Dhivya, V. Secured Adaptive Routing in Mobile Ad Hoc Networks with Artificial Immune System.
31. Dataset for Immune Inspired Misbehavior Detection, 201_,https://www.sim.uni- hannover.de/de/Data-sets/datasets.html.
32. Rizwan, R., Khan, F. A., Abbas, H., & Chauhdary, S. H. (2015). Anomaly detection in wireless sensor networks using immune-based bioinspired mechanism. *International journal of distributed sensor networks*.