Deep Learning-Based Network Data Security Analysis Research

Kai Huang*

E-mail: 18123969109@163.com

School of Economics and Management, Beijing Jiaotong University，Haidian,100044

Abstract. With the continuous advancement of computer, microelectronics and wireless communication technologies, low-power sensor nodes that integrate various functions such as information acquisition, storage, processing and wireless communication in a tiny volume are rapidly developing. A wireless sensor network consists of a large number of inexpensive sensor nodes deployed in a monitoring area, forming a multi-hop network by means of self-organisation. The aim is to sense, collect and process information about the sensed objects in the coverage area and send it to the observer. Sensor networks have greatly changed the way humans interact with the outside world and improved their ability to understand the world, and can be widely used in national defence and military construction, industrial and agricultural production, environmental monitoring, medical care and other fields. In terms of data management and usage, it is generally accepted that the data collected by the nodes are transmitted directly to the base station or sink for processing and maintenance. However, it was found that this centralised approach to data management is bandwidth intensive, the aggregation point is prone to constitute a network bottleneck due to a single point of failure or attack, and lacks the practical deployment capability to accommodate the growth of sensor networks and some new applications.

Keywords: Network data; Security analysis; Distributed data storage; Data Management

1 Introduction

In the last decade, with the advancement of sensor technology, microelectronics, modern networks and wireless communications, the generation and development of Wireless Sensor Network (WSN) has been promoted and facilitated. As WSN is a hot research direction involving multiple disciplines and highly integrated knowledge, it can be applied in defence and military construction, industrial and agricultural production control, commercial operation services, medical monitoring, intelligent transportation and urban management, natural disaster prevention and emergency rescue, environmental health monitoring, remote regulation and control of harsh environments, etc [1]. It has attracted widespread attention from industry and academia, and is considered one of the key information technologies to be developed in the 21st century [2].

ICBBEM 2023, May 19-21, Hangzhou, People's Republic of China

DOI 10.4108/eai.19-5-2023.2334400

2 Adaptive data security storage research

The integrity verification scheme proposed in this paper is based on dual-grained linear codes. Some basics of dual-grained linear codes are briefly introduced [3].

The internal linear code of the data, or internal code for short, is calculated using algebraic functions inside the data. The form is as follows:

$$\Omega_a(x) = \sum_{i=1}^{k} a^{i-1} x_i$$

Here each data x has k symbols, denoted as x_i. The length of the symbol is p. a is an element of the original domain in $GF(2^p)$. In addition $k \leq 2^p - 1$.

Data interaction linear codes, or interaction codes for short, are computations between data using algebraic operations of the following form:

$$P(y_1, y_2, \ldots, y_n = \sum_{i=1}^{n} \beta^{i-1} y_i$$

where each data $y_i (i \in [1, n])$ has k symbols of symbol length p and β^i is a different element chosen at random from the finite field $GF(2^p)$ satisfying $i \in [1, n], n \leq 2^p - 1$.

An important property of dual-grained linear codes is that the interaction code of an internal code is equal to the internal code of an interaction code. That is:

$$\sum_{j=1}^{n} \beta^{j-1} \left(\sum_{i=1}^{k} a^{i-1} x_{ij} \right) = \sum_{i=1}^{k} a^{i=1} \left(\sum_{j=1}^{k} \beta^{i-1} x_{ij} \right)$$

The internal code is calculated by the data holder and the interaction code is calculated by the data distributor and then distributed to the authenticator [4]. For ease of description, the internal codes are referred to later as digest and the interaction codes as parity [5].

3 Analysis

In this paper, simulation experiments are conducted to verify the effectiveness of the scheme [6]. Figure 1 shows the results of the comparison of the detection rates of the different schemes for the case where the number of captured nodes x is incremented from 100 to 500.

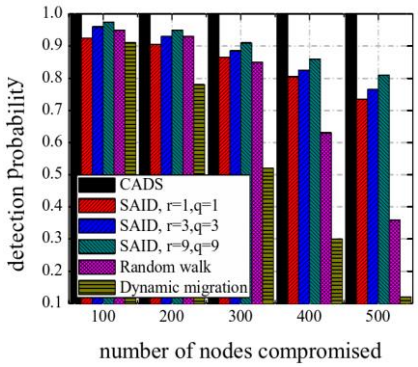
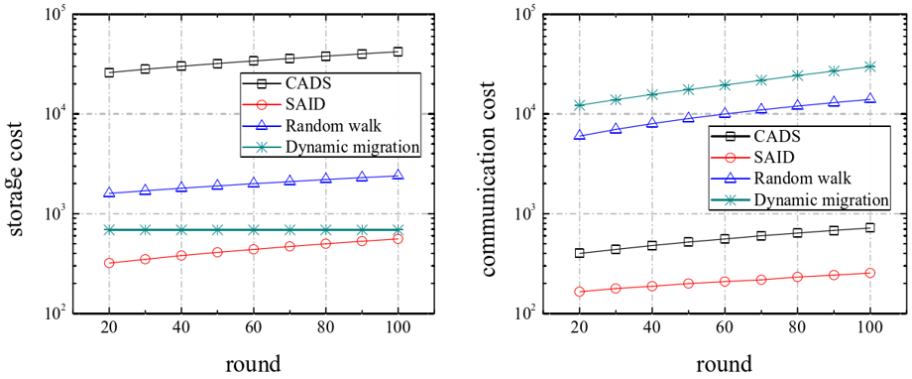
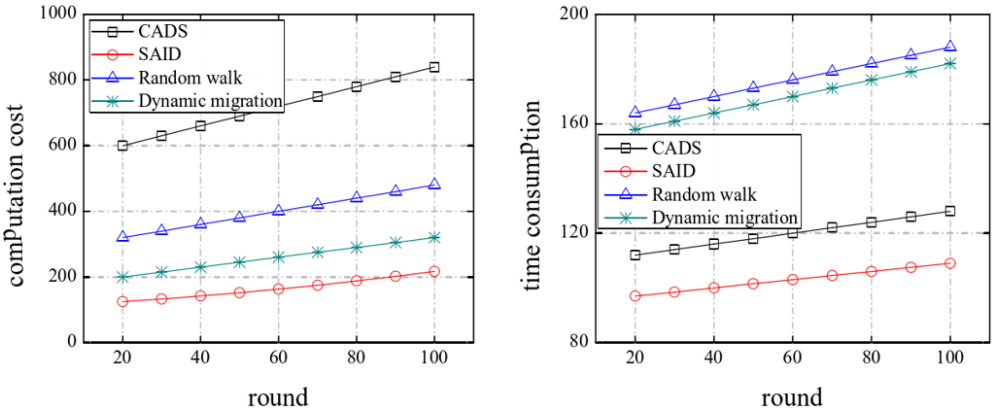


Fig. 1. Comparing the detection rates of various schemes

Figure 2 compares the storage, computational, communication and time overheads of each of the four schemes. As can be seen from the graphs, the SAID option has better indicators than the other three options. This is due to the fact that SAID selects neighbouring nodes for storage and can continuously verify the integrity of the data, saving computational and communication overhead [7]. The CADS scheme, on the other hand, is based on discrete logarithms and requires the introduction of complex calculations. Random walk uses a network-wide broadcast, which has the highest communication cost, and the Dynamic migration solution constantly adjusts the location of the data, which consumes a significant communication overhead. This also reaffirms the validity and feasibility of the programme.



(a) Storage consumption (b) Communication consumption



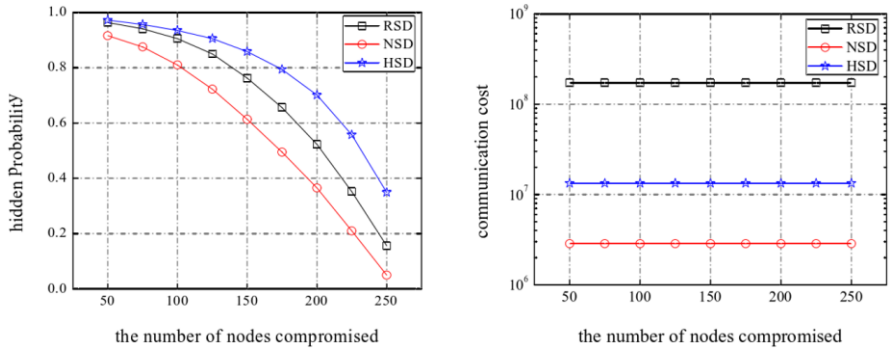
(a) Calculated overhead (b) Time overheads

Fig. 2. Comparing the overheads of the various options

4 Data Retrieval Process

4.1 Experimental results and performance analysis

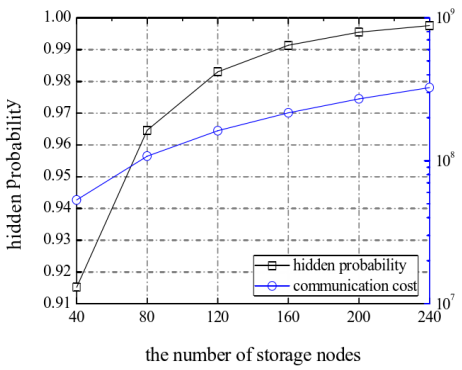
Figure 3(a) compares the data hiding probability with the number of captured nodes. As can be seen from the figure, the larger N' is, the lower the data hiding probability is, and vice versa. This is because as the number of captured nodes in the cell increases, the probability of data being leaked increases [8].



(a) Hidden probabilities (b) Communication overheads

Fig. 3. Symbol distribution versus the number of captured nodes N'

Figure 3(b) illustrates that the communication overhead of the three symbol distribution schemes is independent of the number of captured nodes N'. As can be seen from the previous analysis, the factors that affect the communication overhead are the number of nodes and the density of nodes.



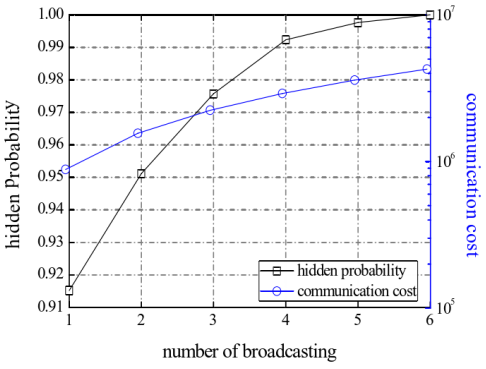
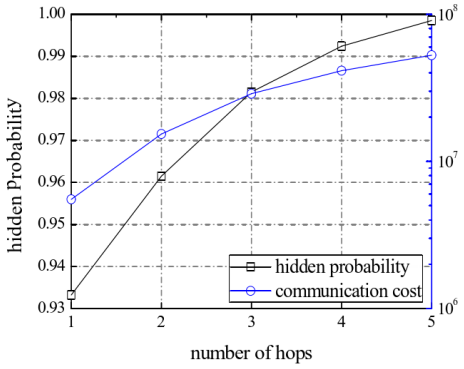


Fig. 4. Effect of n on the RSD scheme. **Fig. 5.** Impact of x on the NSD programme.

Figure 4 shows the impact of n on the RSD scheme. As can be seen from the graph, the larger the n, i.e. the greater the number of storage nodes, the higher the data hiding probability and communication overhead. This is because we choose the (n, k) encoding scheme for the stretching factor to maintain a certain relationship, as n increases, so does k, requiring the retrieval of more symbols, making the data hiding probability higher. From this diagram we can further conclude that by changing the value of (n, k) a better flexibility of data leakage can be guaranteed.

Figure 5 shows the impact of x on the NSD scheme. It can be seen the data hiding probability and communication overhead increase as x increases. Similar to the RSD scheme, as x increases, more storage nodes are selected, and thus the data hiding probability and communication overhead increases.



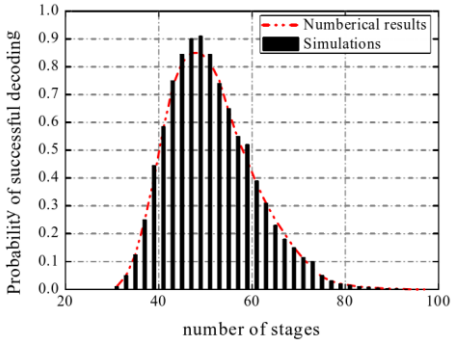


Fig. 6. Impact of h on the HSD programme.**Fig. 7.** Decoding success rate versus number of iterations

Figure 6 shows the effect of h on the HSD scheme. In this we can see the same pattern as in Figures 4 and 5. In general, the larger h is, the higher the probability that the HSD scheme will be able to find enough storage nodes, and for a certain stretch factor, a larger n equals a larger k. The greater the number of symbols that the attacker needs to capture, the greater the number of symbols that the attacker needs to capture. The higher the probability of data hiding, the higher the corresponding communication overhead [9].

The following addresses the efficiency of decoding under the condition that the data is modified after some nodes have been captured. Figure 7 shows the distribution of nodes accessed when the algorithm terminates. The bar chart gives the average value after 1000 simulations and the curve indicates the analytical results. In the experiment $n = 127, k = 31$ and $q = 0.2$ were chosen so that 100 simulation runs gave adequate statistics [10].

In Figure 8, set $k = 31$ constant, n from 127 to 205 and q from 0 to 0.9. Give the probability P_{suc} of successful decoding for different n versus the probability of error q. It can be seen that P_{suc} decreases as q increases. Clearly the more erroneous data there is, the lower the probability of successful decoding of the data. In addition when k is constant the larger n is the higher P_{suc} is. indicates that the larger n the more redundant symbols, the more accessible symbols can be found when decoding fails.

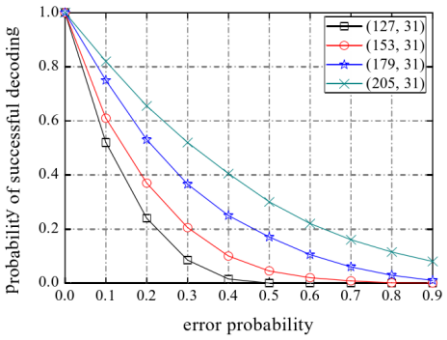


Fig. 8. Decoding success rate versus error rate.

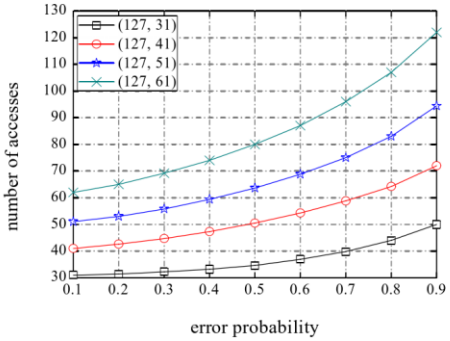


Fig. 9. Number of visits versus error rate

In Figure 9, we set $n = 127$ constant, k from 31 to 61, and q from 0 to 0.9. gives the average number of visits as a function of the probability of error q. The graph shows that the greater the error probability the greater the number of visits required.

5 Conclusion

This paper studies the problem of secure storage of distributed WSN data and first proposes an integrity-verified data storage scheme. Before the data is collected, the nodes holding the data slices can verify their integrity and exclude the upload of incorrect data, saving communication overhead and processing time for the user. On this basis, an adaptive data storage solution has been designed that allows for adaptive storage of data according to the characteristics of the network and security requirements. An efficient data retrieval algorithm is also designed, with optimal communication and computation. Excellent retrieval efficiency in the event of node failure, random errors and data contamination, ensuring data availability and reliability. The former is suitable for environments with dense network deployments, a high number of neighbouring nodes and a relatively low level of security requirements. The latter is suitable for environments with a high level of user capability and a high level of network security. Both solutions have their advantages and both can be applied to resource-constrained sensor networks.

References

[1] Wang Fang. A deep learning-based method for identifying network transmission data anomalies [J]. Modern Electronics Technology,2023,46(06):62-66. doi:10.16652/j.issn.1004-373x.2023.06.012.

[2] Pang Jiale, Zhang Yan. Simulation of repetitive network data annotation under reverse gradient deep learning[J]. Computer Simulation,2022,39(10):467-470+485.

[3] Liu Yumeng. Research on data security situational awareness of wireless communication networks based on deep learning[J]. Information Recorded Materials,2022,23(08):182-185.DOI:10.16009/j.cnki.cn13-1295/tq.2022.08.018.

[4] Yuan Yiying. Deep learning-based model construction for web data analysis [J]. Information and Computer (Theory Edition), 2022,34(12):44-46.

[5] Jia, C. H. Research on malicious load detection of network data stream based on deep learning[D]. Northern Polytechnic University, 2022. doi:10.26926/d.cnki.gbfgu.2022.000070.

[6] Shen Yibo. Deep learning-based mining of key features of communication network data[J]. Journal of Longyan College,2022,40(02):14-19+128.DOI:10.16813/j.cnki.cn35-1286/g4.2022.02.003.

[7] Xiong Lei, Peng Jiqiong, Li Ming, Deng Lundan. Deep learning-based algorithms for personalized mining of grassroots network data[J]. Computer Simulation,2022,39(01):318-321+332.

[8] Pi S. Research on data compression algorithm for wireless sensor networks based on deep learning[J]. Science and Technology Economic Market,2021(10):37-39+42.

[9] Mei Ke, Zeng Changchang. Deep learning algorithm for naval wireless network security analysis[J]. Ship Science and Technology,2021,43(14):169-171.

[10] Ji Chong, Liu Yan. Integrated web big data mining based on semi-supervised deep learning method[J]. Computer Simulation,2021,38(07):313-316.