Postprint of a Privacy-Preserving Data Fusion Algorithm Based on Distributed Compressed Sensing and Hash Functions

Authors: Kou Lan, Liu Ning, Huang Hongcheng, Zhang Yan

Date: 2018-11-29T00:00:00+00:00

Abstract

To address security issues such as privacy leakage, incomplete information, and data tampering during data fusion and transmission in crowdsensing networks, a data fusion privacy protection algorithm based on distributed compressed sensing and hash functions is proposed. First, the distributed compressed sensing method is employed to perform sparse observation on the sensing data, thereby eliminating redundant data; second, a one-way hash function is utilized to compute the hash value of the sensing data observations, which is then padded together with unrestricted dummy data into the sensing data observations to achieve the goal of concealing the real sensing data; finally, after extracting the dummy data at the aggregation node, the hash value of the sensing data is recomputed and the data integrity is verified. Simulation results demonstrate that the algorithm simultaneously ensures the protection of data confidentiality and integrity while significantly reducing communication overhead, exhibiting strong applicability and scalability in practical applications.

Full Text

Preamble

Vol. 37 No. 1
Application Research of Computers
ChinaXiv Cooperative Journal

Data Aggregation Privacy Protection Algorithm Based on Distributed Compressive Sensing and Hash Function

Kou Lan, Liu Ning, Huang Hongcheng†, Zhang Yan
(School of Communication & Information Engineering, Chongqing University of Posts & Telecommunications, Chongqing 400065, China)

Abstract: To address security issues such as privacy leakage, incomplete information, and data tampering during data aggregation and transmission in crowd sensing networks, this paper proposes a data aggregation privacy protection algorithm based on distributed compressive sensing and hash functions. First, the distributed compressive sensing method is employed to sparsely observe the sensed data and remove redundant information. Second, a one-way hash function is used to compute the hash value of the observed data, which is then padded into the observation values along with unconstrained camouflage data to conceal the true sensor data. Finally, after extracting the camouflage data at the sink node, the hash value of the sensed data is recomputed to verify data integrity. Simulation results demonstrate that the algorithm simultaneously ensures data confidentiality and integrity protection while significantly reducing communication overhead, exhibiting strong applicability and scalability in practical applications.

Keywords: privacy protection; distributed compressed sensing; one-way hash function; crowd sensing networks; data aggregation

0 Introduction

Crowd sensing utilizes ordinary users' mobile devices (such as smartphones, smartwatches, and tablets) as basic sensing units to collaboratively distribute sensing tasks and collect data through intentional and unintentional cooperation via the Internet, thereby accomplishing complex and large-scale social sensing tasks [1]. Compared with traditional sensor networks that require deploying numerous dedicated sensor nodes for specific tasks, crowd sensing networks can leverage users' portable mobile terminals and existing communication links, enabling low-cost implementation of large-scale and fine-grained sensing tasks. Consequently, crowd sensing networks have emerged as a novel means for monitoring traffic, earthquakes, and health information [2].

In hotspot areas (such as shopping malls and office districts), sensing nodes are densely distributed, and the data collected by each node inevitably exhibits strong spatiotemporal correlation with other nodes. Transmitting raw sensed data directly to aggregation nodes would significantly increase resource consumption in the sensing network due to unnecessary transmission of redundant data [3]. Moreover, people are typically concerned only with effective information from sensed data rather than massive raw datasets. Therefore, compressing and fusing raw sensed data before transmission can not only substantially reduce resource consumption during data transmission but also accurately extract features from vast amounts of raw data, improving the efficiency of data collection for sensing platforms.

However, the unreliability of wireless transmission links and the untrustworthiness of fusion nodes make sensed data vulnerable to tampering, eavesdropping, replay attacks, or injection of false data, posing serious threats to data security

[4]. Thus, investigating privacy protection in crowd sensing networks, particularly confidentiality and integrity protection for sensed data during the data fusion phase, holds significant research importance.

1 Related Work

Numerous studies have investigated privacy protection during data aggregation and transmission in crowd sensing networks. Literature [5] proposed a lightweight directional data fusion scheme that generates network topology based on distance to balance cluster head energy consumption. By forming a complex structure with nodes' private factors and raw data and employing additive homomorphic encryption, the scheme achieves data fusion without requiring decryption, ensuring privacy protection performance while reducing computational overhead. Literature [6] introduced a trust-based data aggregation protocol that calculates, monitors, and evaluates node trust values through behavioral observation, promptly detecting and excluding compromised nodes, thereby effectively reducing node energy consumption and improving data transmission reliability. Literature [7] shared encrypted data content using simple techniques without revealing actual data and keys to other nodes, enabling the base station to identify untrusted nodes within fusion node clusters and retransmit data only from intermediate nodes with abnormal sensed data. Since intermediate nodes do not require hop-by-hop encryption and decryption operations, computational overhead is reduced, and because intermediate fusion nodes cannot access real sensed data content, the scheme effectively defends against internal attacks.

Literature [8] proposed a cluster-based privacy-preserving data aggregation method where, according to the aggregation node' s instructions, sensing nodes pad both constrained and unconstrained camouflage data into corresponding positions of sensed data observation values, enhancing data privacy protection performance. Literature [9] presented a data fusion privacy protection system for distributed smart meter data based on Fourier perturbation and wavelet perturbation algorithms, utilizing exponential ElGamal encryption mechanisms to achieve secure communication between users and aggregation nodes. The distributed differential privacy mechanism generates distributed noise based on Gaussian principles, ensuring differential privacy for each user compared with traditional privacy protection systems, though the scheme lacks applicability to other types of sensed data.

To protect sensed data privacy while fusing data with low communication overhead, literature [10] proposed a distributed compressed sensing-based privacy-preserving data aggregation mechanism (DCSPDA) that compresses and observes raw sensed data using distributed compressive sensing, achieving privacy protection with small communication overhead but without considering data integrity verification. Literature [11] introduced positive-negative pairs to confuse real sensed data and determined the number of positive-negative pairs each sensor needed to generate through confusion factors. During data fusion, the scheme employed positive-negative neutralization strategies and time-slot allo-

cation mechanisms to reduce communication overhead and collision rates but still lacked integrity protection for sensed data.

Protocols based on secure multi-party computation enable nodes to exchange seeds and perform joint calculations, effectively protecting data privacy. However, computational and communication overhead increases with the number of involved nodes [12]. Literature [13] proposed a privacy protection scheme for vehicular networks based on secure multi-party computation, combining linear equation theory, anonymous authentication concepts, and oblivious transfer protocols with traditional public-key cryptography algorithms to substantially reduce computational complexity. Literature [14] improved cluster-based data fusion privacy protection algorithms based on secure multi-party computation concepts, where cluster heads select cooperative nodes to complete privacy data aggregation, significantly reducing computational and communication overhead.

To simultaneously ensure confidentiality and integrity protection for sensed data, literature [15] proposed an integrity-protecting private data aggregation (iPDA) algorithm that achieves privacy protection during data fusion through data slicing and assembly, constructing two aggregation trees and using one to supervise whether fusion results are complete. However, the algorithm is only effective against certain attacks; for instance, it cannot detect attacks where adversaries simultaneously tamper with both trees' fusion results. Literature [16] proposed an integrity-checking private data aggregation (ICKPDA) algorithm that protects sensed data privacy by padding keys into sensed data and slicing the padded data, then verifies integrity of reassembled data at aggregation nodes using key correlations. This algorithm balances confidentiality and integrity to some extent but generates substantial communication overhead.

Existing privacy protection methods primarily focus on data confidentiality while neglecting integrity, and schemes that address both aspects incur high communication overhead. To solve this problem, this paper proposes a data aggregation privacy-preserving algorithm based on distributed compressive sensing and hash functions (DAP-DCSHF). The algorithm employs distributed compressive sensing to sparsely observe raw sensed data, reducing network communication overhead. It uses a one-way hash function to compute hash values of sparse observations and pads these hash values along with unconstrained camouflage data generated according to privacy protection requirements into zero-value positions of observation values, enhancing confidentiality protection. The padding position information for hash values and unconstrained camouflage data is encrypted and sent separately to the aggregation node. Finally, after removing camouflage data from received privacy datasets, the aggregation node recomputes hash values and verifies data integrity by comparing the two hash values.

2 Data Aggregation Privacy Protection Algorithm Based on Distributed Compressive Sensing and Hash Function

This paper proposes the DAP-DCSHF algorithm, which improves upon the DCSPDA algorithm by using a one-way hash function to compute hash values of sparse observations. The hash values serve as constrained camouflage data padded into zero-value positions of observation values. At the aggregation node, after removing camouflage data from received privacy datasets, hash values are recomputed, thereby enhancing confidentiality protection, verifying integrity, and reducing communication overhead during data fusion transmission.

2.1 Distributed Compressive Observation Method for Perceptual Data

Distributed compressed sensing (DCS) is a data fusion method that efficiently compresses and observes sensed data while exploiting spatiotemporal correlations among data for reasonable sparse representation. In distributed source coding, whether encoding is performed independently or jointly at the encoding end, joint decoding at the decoding end after transmission yields equivalent information [17]. During the distributed compressive observation phase, data collected by sensing nodes is first observed through distributed compressive sensing.

A joint sparse model (JSM) capable of describing and processing sensed data is established. At fusion nodes, raw multimedia sensed data collected from multiple sensing nodes within communication range undergoes distributed compressive sensing observation:

where J represents the number of sensing nodes within the communication coverage of a fusion node; θ_j is the sparse coefficient of node j' s sensed data; Ψ is the $N \times N$ public sparse basis; Z_C is the public sparse component of sensed data x_j; and Z_j are the independent sparse components of sensed data x_j.

This paper organizes sensing nodes within a certain communication range into a cluster, each containing a cluster head node that serves as the fusion node responsible for intra-cluster data fusion. Cluster heads closer to the aggregation node can act as relay nodes to further complete inter-cluster fusion and transmission. Depending on their distance from the aggregation node, fused data can reach the aggregation node through single-hop or multi-hop transmission.

Practical applications have repeatedly demonstrated that distributed compressive sensing of sensed data produces privacy protection effects similar to direct encryption [18]. The process of sensed data observation and camouflage data insertion is illustrated in Figure 1 [Figure 1: see original paper].

A sparse binary random observation matrix is used to observe sensed data:

where y_j is the M-dimensional distributed compressive sensing observation vector of raw multimedia sensed data; Φ is an $M \times N$ sparse binary random obser-

vation matrix; x_j is the raw sensed data of node j; Ψ is the $N \times N$ public sparse basis; Θ_C is the public sparse coefficient of node j; and Θ_j is the independent sparse coefficient.

The observation process essentially projects the sparse coefficient vector using M row vectors of the $M \times N$ sparse binary random observation matrix Φ, preserving information required for signal reconstruction. The position set of raw multimedia sensed data observation values is denoted by TPS.

2.2 Confidentiality Protection for Perceptual Data

A one-way hash function, also known as a one-way hash function, converts input information strings of arbitrary length into fixed-length output strings from which the input information cannot be easily reverse-engineered. This paper employs the MD5 one-way hash function, which converts input information strings of arbitrary length into 32-bit output strings. The hash value of distributed compressive sensing observation data is computed using the MD5 one-way hash function:

where H_j is the hash value of raw multimedia sensed data observation values, and y_j is the input raw multimedia sensed data observation value.

The distributed compressive sensing observation values of raw multimedia sensed data consist of three parts: public sparse components, independent sparse components, and zero-value components. The hash value is defined as constrained camouflage data and is padded into zero-value positions of observation values, with the padding positions recorded as set $RCPS$. This paper defines a random function that generates arbitrary values within a specified range as unconstrained camouflage data, with the position set for padding unconstrained camouflage data into zero-value positions of observation values denoted by $GCPS$.

When the number of zero values in observation values is less than 32, constrained camouflage data fills all zero-value positions first, with remaining constrained camouflage data placed directly at the end of observation values. When the number of zero values is greater than or equal to 32, all constrained camouflage data can be padded into zero-value positions, and additional unconstrained camouflage data can be inserted according to the strength of confidentiality protection requirements. Generally, since sensed data contains numerous zero values after compressive sensing, both constrained and unconstrained camouflage data can be simultaneously padded into zero-value positions of observation values to enhance confidentiality protection.

The position information for padding constrained camouflage data into zero-value positions of observation values is described as the set $RCPS$ of camouflage data padding positions, which is transmitted independently and encrypted to the aggregation node. Simultaneously, $GCPS$ and the original sensed data observation value position set TPS jointly constitute the sensed privacy data packet $NTPS$:

where equations (2) and (3) respectively represent the public sparse component Z_C and independent sparse component Z_j of raw sensed data x_j.

2.3 Fusion Transmission of Perceptual Data

Generally, sensed data fusion calculations can select comprehensive operations such as addition, multiplication, averaging, maximum, or minimum according to application requirements. Since addition operations have fewer constraints in privacy protection algorithms and other operations can be converted into additive fusion forms, this paper adopts addition operations during sensed data fusion transmission. The addition operation is as follows:

where Sum_j represents the value after node j inserts camouflage data into the observation values of raw multimedia sensed data.

2.4 Integrity Verification of Perceptual Fusion Data

The process of verifying sensed data integrity using a one-way hash function is illustrated in Figure 2 [Figure 2: see original paper].

a) Fusion nodes perform compressive observation on sensed data and compute hash values H_I of observation values using a one-way hash function.

b) After receiving the privacy dataset, aggregation nodes remove camouflage data based on received position information, compute hash values H_O of observation values, and compare the two hash values. If $H_I = H_O$, the sensed data is secure during transmission, passes integrity verification, and proceeds to sensed data reconstruction; otherwise, the data was attacked during transmission, and the destination node discards it as it does not represent authentic sensed data values.

c) At relay nodes, inter-cluster fusion is further completed to form privacy datasets. Depending on fusion nodes' distance from the aggregation node, fused data can reach the aggregation node through single-hop or multi-hop transmission.

d) Aggregation nodes compute hash values H_O of sensed data from privacy datasets after removing camouflage data, comparing whether the two hash values are equal. If $H_I = H_O$, the data is secure during transmission, passes integrity verification, and original data is jointly reconstructed; otherwise, the data was altered during transmission, and the destination node discards it as it does not represent authentic values.

2.5 Reconstruction of Perceptual Fusion Data

Original sensed data is recovered by reconstructing sensed fusion data that passes integrity verification. The distributed compressive sensing information

operator is denoted as $A_{DCS} = \Phi\Psi$. Literature [19] indicates that if A_{DCS} satisfies the condition that the measurement matrix Φ and sparse basis matrix Ψ are incoherent, then the original signal can be precisely reconstructed by solving the optimal l_0-norm problem:

where $\hat{\theta}_j$ is the estimated sparse coefficient obtained by solving the optimization method for node j' s sensed data received by the aggregation node; θ_j is the sparse coefficient of node j's sensed data; and y_j is the data after the aggregation node removes camouflage data from distributed compressive observation values of sensed data and passes integrity verification.

The final recovered original sensed data $\hat{x}_j$ is:

Based on the above theory, the data aggregation privacy protection algorithm based on distributed compressive sensing and hash functions is presented to simultaneously ensure confidentiality and integrity protection for sensed data during data fusion transmission in crowd sensing networks while reducing network communication volume.

Algorithm: Data Aggregation Privacy-Preserving Algorithm Based on Distributed Compressive Sensing and Hash Function (DAP-DCSHF)

a) Employ distributed compressive sensing to sparsely observe sensed data collected by intra-cluster sensing nodes.

b) Compute hash values H_I of observation values using a one-way hash function, and generate unconstrained camouflage data through random functions according to different privacy protection requirements. Pad hash values and unconstrained camouflage data into zero-value positions of observation data to form privacy data packets. Encrypt and send the camouflage data padding position information separately to the aggregation node.

c) Perform additive fusion at relay nodes to form privacy datasets. Depending on fusion nodes' distance from the aggregation node, fused data can reach the aggregation node through single-hop or multi-hop transmission.

d) After removing camouflage data, recompute hash values H_O of observation values from privacy datasets at the aggregation node. Compare whether the two hash values are equal. If $H_I = H_O$, the data is secure during transmission, passes integrity verification, and original data is reconstructed through joint reconstruction; otherwise, the data was altered and is discarded by the destination node.

3 Performance Evaluation

3.1 Experimental Environment

The proposed algorithm is implemented in MATLAB, with simulation datasets using standard Foreman, Hall Monitor, and News video sequences. Each video sequence contains 300 frames, with each frame having a resolution of 256×256.

To evaluate the algorithm' s performance, DAP-DCSHF is compared under identical conditions with the integrity-checking private data aggregation (ICKPDA) algorithm, the distributed compressive sensing-based privacy-preserving data aggregation (DCSPDA) algorithm, and the integrity-protecting private data aggregation (iPDA) algorithm. The evaluation assesses the proposed algorithm' s effectiveness from three perspectives: data confidentiality protection level, data integrity protection level, and communication consumption. Parameter settings for the DAP-DCSHF algorithm in simulations are shown in Table 1 .

Table 1 Algorithm Parameter Settings

Parameter	Value
Network area (m^2)	4500×3400
Node cache (M)	2
Node communication method	Bluetooth
Hash value length	32
Video sequences	Foreman, Hall Monitor, News
Image frame pixels	256×256
Link cracking probability	0.1
Key frame sampling rate	0.5
Non-key frame sampling rate	0.3

3.2 Data Confidentiality Protection Performance Analysis

To objectively analyze the confidentiality protection effectiveness of DAP-DCSHF during sensed data fusion transmission, we assume that position information for padded hash values and unconstrained camouflage data is transmitted securely and confidentially to the aggregation node. Let $||NTPS||$ denote the length of the real sensed data set in DAP-DCSHF, and $||NSS||$ denote the length of the complement of the node secret set composed of real sensed data sparse components and constrained camouflage data. Assuming each link has a cracking probability q, the probability of sensed data leakage on that link is denoted by P_{leak}. The leakage probability is defined as the probability that an attacker can successfully recover real sensed data from intercepted privacy-protected data when a given link is cracked.

An attacker must obtain data from each sensing node in a cluster transmitted to the cluster head node to acquire the fused data within that cluster. Assuming

the number of sensing nodes in a cluster is n, the intra-cluster fused data privacy leakage probability $P_{DAP-DCSHF}^{Prob}$ can be expressed as:

In the DCSPDA algorithm, to obtain intra-cluster fused data, an attacker must first acquire privacy data packets sent from each sensing node to the cluster head node to possibly recover the fused data within the cluster. Therefore, the intra-cluster fused data privacy leakage probability calculation for DCSPDA is similar to DAP-DCSHF.

In the iPDA algorithm, each sensing node first divides its multimedia data into L slices, retains one slice, and distributes the remaining $L-1$ data slices to neighbor nodes while receiving $L-1$ data slices from neighbor nodes. To crack sensed data, an attacker must not only obtain the $L-1$ data slices sent by the node but also crack links of $L-1$ data slices from neighbor nodes. Assuming the number of intra-cluster sensing nodes is n, the data leakage probability P_{iPDA}^{Prob} can be expressed as:

where $P_{Rec}(J=k)$ is the probability that a sensing node receives k data slices. If n nodes receive k slices, then:

where N is the total number of slices. Since iPDA requires constructing two fusion trees, $2L = 3N = n$.

In the ICKPDA algorithm, an attacker must crack both private seeds of a node and its degree links to obtain privacy data. Therefore, the privacy data leakage probability P_{ICKPDA}^{Prob} can be expressed as:

where n is the cluster size and $P_{deg}(k)$ is the probability of a node with degree k.

To intuitively demonstrate the confidentiality protection performance of different algorithms during sensed data transmission, this paper simulates the consequences of data leakage for different algorithms under conditions of cluster size $n=3$, link leakage probability $q=0.1$, and identical original image frames. The results are shown in Figure 3 [Figure 3: see original paper]. When using DAP-DCSHF and DCSPDA algorithms, leaked sensed data cannot completely reconstruct the original sensed image frames. However, when using ICKPDA and iPDA algorithms, leaked sensed data may reveal partial visual features of the original sensed images. The image frames reconstructed by the proposed DAP-DCSHF algorithm are comparatively blurrier than other algorithms, indicating that less original data is leaked and confidentiality protection is more effective.

Figure 4 [Figure 4: see original paper] shows the privacy data leakage probabilities of the four privacy-preserving data fusion algorithms (DAP-DCSHF, DCSPDA, ICKPDA, iPDA) under cluster sizes of 3, 5, and 8. To reduce simulation result errors from random factors, simulation values in this algorithm are averaged over 10 runs.

As shown in Figure 4, the proposed algorithm and DCSPDA algorithm ex-

hibit significantly better performance in protecting sensed data confidentiality compared with the other two algorithms. Since iPDA and ICKPDA achieve confidentiality protection through data slicing and reassembly, their effectiveness depends on the number of slices. Increasing slice count to enhance confidentiality protection simultaneously increases communication overhead. Both DAP-DCSHF and DCSPDA algorithms sparsely observe raw sensed data based on distributed compressive sensing, whose measurement process is equivalent to privacy data encryption. Both algorithms further enhance confidentiality by padding camouflage data into zero-value positions of sparse observation values. Additionally, DAP-DCSHF' s confidentiality protection performance is slightly superior to DCSPDA because DCSPDA' s constrained camouflage data has a smaller value range than observation values, which cannot effectively conceal real data, whereas DAP-DCSHF' s constrained camouflage data consists of hash values whose value range is not constrained by observation values.

3.3 Data Integrity Protection Performance Analysis

The DCSPDA algorithm enhances confidentiality protection during data fusion and reduces communication overhead but does not propose solutions for threats such as replay, forgery, and data tampering attacks. This paper analyzes the algorithm' s data integrity protection performance by evaluating its resistance to replay, forgery, and tampering attacks.

3.3.1 Replay Data Attack When fused data suffers replay attacks: In iPDA, the established red and blue fusion trees are disjoint. One tree' s fusion value changes while the other remains unchanged. Since the two trees'fusion results differ, iPDA can easily detect replay attacks. In ICKPDA, correlation calculations on the final binary data yield unchanged restoration results, so ICKPDA cannot detect replay attacks. In DAP-DCSHF, replay attacks do not alter sensed data observation values, making them difficult to distinguish from original data.

3.3.2 Forged Data Packet Attack When fused data suffers forged data packet attacks: In iPDA, if an attacker forges data packets in only one tree, the two trees' fusion results differ, making detection easy. However, if both trees are forged, since their fusion results remain equal, iPDA cannot detect such attacks.

In ICKPDA, the binary data fusion values obtained by the aggregation node increase correspondingly, and after integrity detection value calculation, the two values remain equal. Therefore, ICKPDA cannot resist forged data packet attacks. In DAP-DCSHF, after removing camouflage data from received privacy data packets, the hash value computed from observation values received by the aggregation node does not equal the original data observation values' hash value, failing integrity verification and being discarded by the aggregation node. Thus, DAP-DCSHF effectively resists forged data packet attacks.

3.3.3 Data Tampering Attack When fused data suffers data tampering attacks: In iPDA, similar to forged packet attacks, if only one tree is tampered with, detection is easy. However, if both trees are tampered with, iPDA cannot detect it.

In ICKPDA, tampering any element of the binary data directly destroys the correlation between binary data elements, enabling detection at the aggregation node. Therefore, ICKPDA effectively resists data tampering attacks. In DAP-DCSHF, tampering changes the hash value of observation values, enabling effective detection of data tampering attacks.

Table 2 presents the performance analysis of DAP-DCSHF, DCSPDA, ICKPDA, and iPDA algorithms in resisting replay, forgery, and data tampering attacks.

Table 2 Performance Analysis of Defense Against Attack

Algorithm	Replay Data Attack	Forged Data Packet Attack	Data Tampering Attack
DAP-DCSHF	○	△	△
DCSPDA	○	×	×
ICKPDA	×	×	△
iPDA	△	○	○

△ *indicates strong attack resistance,* ○ *indicates weak attack resistance,* × *indicates no attack resistance*

3.4 Communication Energy Consumption in Perceptual Data Fusion Transmission

Literature [20] indicates that communication energy consumption of sensing nodes far exceeds computational energy consumption. Therefore, this section analyzes communication energy consumption during sensed data fusion transmission.

First, we analyze the number of data packets each node must send in the four algorithms:

In DCSPDA, fusion nodes must send position information of camouflage data padded into observation value zero-value portions once to aggregation nodes, and transmit privacy-protected data packets to upper-layer relay nodes, ultimately reaching aggregation nodes. Therefore, DCSPDA's communication overhead is $O(3n)$. Since DCSPDA is also a distributed compressive sensing-based privacy-preserving data aggregation algorithm, its communication overhead is roughly equivalent.

In iPDA, to protect sensed data confidentiality through slicing and reassembly and to protect data integrity, two disjoint red and blue fusion trees must be

constructed. Each node must send $L-1$ slices (where L is the number of slices) to neighbor nodes in both fusion trees and send one data packet for tree construction, finally transmitting the reassembled data packet. Therefore, iPDA' s communication overhead is $O((2L-1)n)$. Since $L \geq 2$, the communication overhead is at least $O(5n)$.

In ICKPDA' s data fusion phase, nodes first divide data into l slices, store one slice, encrypt and send the remaining $l-1$ slices to different neighbor nodes. Neighbor nodes decrypt received slices and fuse them with their own data. Simultaneously, each node receives data slices from other nodes and fuses them with another of its own slices. After all nodes complete cyclic fusion, fused data is uploaded to upper-layer nodes. Since slice count l must be greater than or equal to 2, the communication overhead is at least $O(4n)$.

For fair comparison, the same amount of sensed data is transmitted to aggregation nodes, and total data communication volume (in packets) is compared across the four algorithms. Simulation results are shown in Figure 5 [Figure 5: see original paper].

As shown in Figure 5, iPDA and ICKPDA algorithms exhibit significantly higher communication overhead compared with DAP-DCSHF and DCSPDA, with the gap widening as cluster node count increases. Because iPDA requires data slicing and reassembly for confidentiality and constructs two disjoint red and blue fusion trees for integrity verification, performing data fusion on both trees doubles redundant data growth. Although ICKPDA also uses data slicing and reassembly for confidentiality like iPDA, its communication overhead is smaller since each data item only requires one fusion operation.

DAP-DCSHF and DCSPDA algorithms exploit spatiotemporal correlations among sensed data, employing distributed compressive sensing to remove redundant data and capture important sparse components, dramatically reducing data communication overhead. This advantage becomes increasingly evident as intra-cluster node count grows. Since DAP-DCSHF pads constrained camouflage data as 32-bit fixed-length hash values into observation values, while DCSPDA dynamically adjusts constrained camouflage data length according to confidentiality protection requirements, DAP-DCSHF' s communication volume is slightly higher than DCSPDA when cluster node count is small. However, as cluster node count increases, the communication volumes of both algorithms converge.

4 Conclusion

To address severe privacy leakage of sensed data during data fusion transmission in crowd sensing networks, where existing algorithms focus primarily on confidentiality while neglecting integrity, and where schemes addressing both aspects incur high communication overhead, this paper proposes the DAP-DCSHF algorithm. The algorithm employs distributed compressive sensing to sparsely observe sensed data, removing redundancy, and uses a one-way hash function

to compute hash values of sparse observations. These hash values serve as constrained camouflage data, combined with unconstrained camouflage data generated by random functions according to confidentiality requirements, and are padded into zero-value positions of observation values to enhance confidentiality protection. After additive fusion forms privacy datasets transmitted to aggregation nodes, camouflage data is removed and hash values are recomputed to verify integrity. Observation values passing integrity verification are reconstructed to recover original sensed data.

Compared with ICKPDA, DCSPDA, and iPDA algorithms, the proposed algorithm demonstrates superior confidentiality protection performance. In terms of integrity protection against forged and tampered data attacks, it outperforms the other three algorithms. Against replay attacks, its protection performance is slightly inferior to iPDA. Regarding communication energy consumption, the proposed algorithm significantly outperforms ICKPDA and iPDA. When data volume is small, its communication overhead is slightly higher than DCSPDA, but as cluster node count increases, both algorithms' communication volumes converge.

The algorithm simultaneously ensures data confidentiality and integrity protection while substantially reducing network communication overhead, exhibiting strong applicability and scalability in practical applications. However, limitations exist: the algorithm currently applies only to single-scenario sensed data fusion privacy protection problems. In practical applications, smart devices integrate increasingly diverse sensor types, and addressing privacy protection for sensed data fusion in such complex scenarios represents future work.

References

[1] He Hong, Xiang Chaocan, Xiao Shucheng, et al. Survey on crowd-sensing networks [J]. Journal of Jilin University: Information Science Edition, 2016, 34 (3): 374-383.

[2] Sun W, Liu J. Congestion-aware communication paradigm for sustainable dense mobile crowd-sensing [J]. IEEE Communications Magazine, 2017, 55 (3): 62-67.

[3] Boubiche S, Boubiche D E, Bilami A, et al. Big data challenges and data aggregation strategies in wireless sensor networks [J]. IEEE Access, 2018, 2018 (6): 20558-20571.

[4] Zeng Juru, Chen Hong, Peng Hui, et al. Privacy preservation in mobile participatory sensing [J]. Chinese Journal of Computers, 2016, 39 (3): 595-614.

[5] Zhao Xiaomin, Zhu Jiabin, Liang Xueli, et al. Lightweight and integrity-protecting oriented data aggregation scheme for wireless sensor networks [J]. IET Information Security, 2017, 11 (2): 82-88.

[6] Ma Teng, Liu Yun, Zhang Zhenjiang. An energy-efficient reliable trust-based

data aggregation protocol for wireless sensor networks [J]. International Journal of Control and Automation, 2015, 8 (1): 249-253.

[7] Akila V, Sheela T. Preserving data and key privacy in Data Aggregation for Wireless Sensor Networks [C]// Proc of the 2nd International Conference on Computing and Communications Technologies. Piscataway, NJ: IEEE Press, 2017: 282-287.

[8] Wu Dapeng, Yang Boran, Wang Ruyan. A scalable privacy-preserving big data aggregation scheme for wireless sensor networks [C]// Proc of Military Communications Conference. Piscataway, NJ: IEEE Press, 2008: 1-7.

[9] Lyu L J, Law Y W, Jin J, et al. Privacy-Preserving Aggregation of Smart Metering via Transformation and Encryption [C]// Proc of IEEE Trustcom//BigDataSE//ICESS. Piscataway, NJ: IEEE Press, 2017: 472-479.

[10] Wu Dapeng, Yang Boran, Wang Honggang, et al. Privacy-preserving multimedia big data aggregation in large-scale wireless sensor networks [J]. ACM Trans on Multimedia Computing Communications and Applications, 2016, 12 (4): 1-19.

[11] Zhang Jun, Zhu Jianghao, Jia Zongpu, et al. A secret confusion based energy-saving and privacy-preserving data aggregation algorithm [J]. Chinese Journal of Electronics, 2017, 26 (4): 740-746.

[12] Vinodha D, Mary Anita E A. Secure data aggregation techniques for wireless sensor networks: a review [J]. Archives of Computational Methods in Engineering, 2018, 2018 (8): 1-21.

[13] Song Cheng, Zhang Mingyue, Peng Weiping, et al. Privacy protection mechanism based on secure multi-party computation in VANET [J]. Journal of Beijing University of Posts & Telecommunications, 2017, 40 (3): 67-71.

[14] Man Dapeng, Wang Chenye, Yang Wu, et al. Energy-efficient cluster-based privacy data aggregation for wireless sensor networks [J]. Journal of Tsinghua University: Science and Technology, 2017, 2017 (2): 213-219.

[15] He Wenbo, Nguyen H, Liu Xue, et al. iPDA: An integrity-protecting private data aggregation method [J]. Digital Communications and Networks, 2016, 2 (3): 122-129.

[16] Zhou Qiang. Research on secure data aggregation Technology of wireless sensor networks [D]. Nanjing: Nanjing University of Posts and Telecommunications, 2014.

[17] Wang Wen, Zhu Jinkang, Zhang Sihai, et al. Tradeoff between efficiency and delay of distributed source coding for uplink transmissions in machine type communications [C]// Proc of the 9th International Conference on Wireless Communications and Signal Processing. Piscataway, NJ: IEEE Press, 2017: 1-6.

[18] Qian Jianhua, Zhang Xueying. Compressive data gathering based on even clustering for wireless sensor networks [J]. Journal of Computer Applications, 2018, 38 (6): 1691-1697.

[19] Hampton J, Doostan A. Basis adaptive sample efficient polynomial chaos (BASE-PC) [J/OL]. Journal of Computational Physics, 2018, 2018 (371). (2017-07-02) [2018-07-20]. http://doi.org/10.1016/j.jcp.2018.03.035.

[20] Arbi I B, Derbel F, Strakosch F. Forecasting methods to reduce energy consumption in WSN [C]// Proc of the 12th IEEE International Instrumentation and Measurement Technology Conference. Piscataway, NJ: IEEE Press, 2017: 1-6.

Note: Figure translations are in progress. See original paper for figures.