Hindawi
Security and Communication Networks
Volume 2018, Article ID 6328504, 30 pages
https://doi.org/10.1155/2018/6328504

Research Article

An Aggregate Signature Based Trust Routing for Data Gathering in Sensor Networks

Jiawei Tang,[1] Anfeng Liu ,[1] Ming Zhao ,[2] and Tian Wang[3]

[1]*School of Information Science and Engineering, Central South University, Changsha 410083, China*
[2]*School of Software, Central South University, Changsha 410006, China*
[3]*Department of Computer Science and Technology, Huaqiao University, Xiamen, Fujian Province 361021, China*

Correspondence should be addressed to Anfeng Liu; afengliu@mail.csu.edu.cn

Received 27 August 2017; Revised 4 November 2017; Accepted 6 December 2017; Published 23 January 2018

Academic Editor: Ding Wang

An Aggregate Signature based Trust Routing (ASTR) scheme is proposed to guarantee safe data collection in WSNs. In ASTR scheme, firstly, the aggregate signature approach is used to aggregate data and keep data integrity. What is more important, a light aggregate signature based detour routing scheme is proposed to send abstract information which includes the data sending time and ID of data, nodes' ID to the sink over different paths which can verify whether the data reaches the sink safely. In addition, the trust of a path is evaluated according to the success rate of the path. The trust of paths susceptible to frequent attack will be lowered and the path with high trust will be selected for data routing to avoid data gathering through low trust path and thereby increase the success rate of data gathering. Our comprehensive performance analysis has shown that, the ASTR scheme is able to effectively ensure an increase in success rate of data transmission by 23.23%, reduce the data amount loaded by the node by 53.59%, reduce the redundant data by 41.70%.

1. Introduction

With a large number of sensing devices' connection to Internet of Everything (IoE) [1, 2], our life has become favorably convenient. As one of the most important parts for IoT, the wireless sensor networks (WSNs) are increasingly applied in various aspects of industry [3–6], environment monitoring [7], life [8, 9], and medical health monitoring [10]. With the emergence of edge computing [11], the network architecture is also rapidly developing in computing mode from the cloud computing [11, 12] of network center to edge computing [13], where the sensor network plays an important role [14–16]. However, preservation of information and sensitive data is one of the biggest challenges in sensor networks [17–19]. Due to the high costs, the hardware of sensor nodes is relatively simple and is often unattended in hostile environment, hence prone to different kinds of attacks [20–22]. According to the statistics, there are over 30 kinds of attacks against the sensor network [21, 23], including black attack, clone attack, selective forwarding attack, and false data injection attack [20–23]. It is necessary to provide solid as well as evident solutions as countermeasures against threats in sensor networks; otherwise the application of sensor networks will be hindered [20–23]. For example, black hole attack [23] is such an attack behavior which can drop all the data transferred through the black holes, and the selective forwarding attack (SFA) [20, 24] can intelligently drop some packets to damage network which is difficult to detect and defense such attack [23]. The false data injection attack produces much false data to consume the precious energy of the sensor network, which will cause the advanced death of network exception [21, 23]. Therefore, it is a challenge issue to evade the attack scope of malicious node and ensure the node produces safe and effective data which can reach the sink as verified [1, 21].

This paper mainly studies how to defend against the attacks that adversely affect the data transmission and data integrity of WSNs. Generally speaking, this type of attack behaviors has the following characteristics: (1) *Damaging the Data Integrity.* Compromised node (CN) [20, 21] attack

belongs to this type of attack behaviors: the adversary physically compromises a subset of nodes to eavesdrop information. In this way, the compromised node is able to forge false data and launch a false data injection attack or the malicious nodes are able to forge the signature of other nodes and send a large amount of false data or even tamper data. (2) *Damaging the Data Security*. This means the malicious nodes are able to impede the safe routing of data so that the data cannot reach the destination [23]. For example, black hole attack: it can drop all data through the black node; in comparison, selective forwarding attack is more complex and can intelligently drop some selected key data, which not only endangers the network but also makes the attack behavior difficult to detect. Another one is denial of service (DOS) attack [23] which refers to the following: the adversary interferes with the normal operation of the network by actively disrupting, changing, or even paralyzing the functionality of a subset of nodes. The common consequences of this type of attacks are as follows: areas within which the adversary can either passively intercept or actively block information delivery [23].

Some researches have been conducted on defense against malicious nodes. Among them, some were conducted against a single type of attack behavior; for example, the literature [23] proposed the multibranch routing scheme for defense against selective forwarding attack. Some other researches were not conducted on the safety security strategy against a certain attack; for example, the traceback strategy proposed in literature [22] belongs to the defense attack strategies. In traceback strategy, the IDs of pass-through routing nodes are added to the data packet, so when an attack occurs, the attack path will be reconstructed, which helps the system obtain the location of attack source and take measures to clear the malicious node. However, the past researches have a prominent shortcoming: they only considered the circumstance where the data packets produced by the source node are directly routed to the sink, but the wireless sensor network has a typical difference with other networks; that is, its node energy is limited, and hence it is an important research issue to minimize its energy consumption. Data aggregation is an effective method to reduce the energy consumption of WSNs [1] and, due to the correlation between the data packets produced by different nodes of WSNs in time and space, if data packets produced by nodes with approximate locations and time are sent to the sink upon data aggregation, the data amount will be significantly reduced to save the energy consumption of network [1]. In practice, the data aggregation rate may reach over 70%, so the load of data by nodes is only 30%, that is, less than 1/3 of the original after the data aggregation. In some special data aggregation functions, such as the data aggregations to obtain max, min, and avg, only one data packet is output after the data aggregation of n data packets and the data aggregation rate is very high, greatly improving the performance of WSNs [1]. The data aggregation and collection are significant research issues for cloud computing and fog computing which have been studied by many researchers [25, 26].

Although the data aggregation plays an important role in WSNs, it fuses multiple data packets into one, so the issues caused by attack against the fused data packet are more complex than those in the past researches [1]. Specifically speaking, the following challenges will be met: (1) Integrity issue of data packet: malicious nodes may produce much false data, so if no proper measure is taken, it will be difficult to determine whether false data exist after the data aggregation and which are source nodes that produce the false data. (2) The fact that an attack will bring more serious loss to the network: the data packet formed upon data aggregation contains more information, so an attack may cause more serious consequence. (3) Routing safety issue: when some attacks, such as black hole attack and selective forwarding attack, occur, the data packet will be attacked and dropped before reaching the sink, but the sink cannot determine whether the network has been attacked or any data packet has been lost, which is considered the most dangerous circumstance. With so many challenges existing in the data aggregation network and to the best of our knowledge, seldom researches have been made on attack defense of WSNs in data aggregation. In this paper, an aggregate signature based trust routing (ASTR) scheme is proposed to guarantee safe data collection for data aggregation in WSNs. The main contributions of this paper are as follows.

(1) An aggregate signature based trust routing (ASTR) scheme is proposed to guarantee data security and data integrity of data collection in WSNs. In ASTR scheme, each node signatures its data and abstract information (i.e., the data sending time and ID of data, node ID) and sends them to aggregator, and the aggregator compresses those data signatures into a short one packet called data packet and sends it to sink over $\mathscr{M} \mid \mathscr{M} \geq 1$ different paths. In this way, the sink can verify the data aggregate signature is valid if and only if every single signature used in the aggregation is valid. So the data integrity can be guaranteed. On the other hand, the aggregator also compresses those abstract information signatures into another short one packet called abstract packet and sends it over another $\mathscr{N} \mid \mathscr{N} \geq 1$ detour routing to sink, which can verify whether the data has reached the sink safely. We conducted a series of analysis in theory and determined the required values of $\mathscr{M}, \mathscr{N}$ to guarantee the data can reach the sink safely, so the safe routing of data to the sink can be guaranteed. The past researches often considered only one aspect, that is, data security (probability for data to reach the sink safely) or data integrity. However, the ASTR scheme proposed by this paper can ensure both the data integrity and data security, making the research more meaningful.

(2) The ASTR scheme adopts the trust-based routing method to further improve the success rate of routing. Adopting the strategy of sending $\mathscr{M}$ data routings and $\mathscr{N}$ abstract routings (called $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method) is able to improve the success rate of routing but at higher costs in energy consumption. Hence, the paper further proposes a method to select high trust routing based on the trust of paths, which is able to reduce the number of data and abstracts needed to ensure a high success rate of routing or is able to improve the success rate of routing while consuming the same energy and further optimize the system performance.

(3) Through our extensive theoretical analysis and simulation study, we demonstrate that, for TSTR scheme which adopts the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method, both data integrity and data security can be achieved simultaneously. For the network with a packet loss ratio of 10%, the probability for data to reach the sink safely is raised by 23.23%. If a routing path with high trust is selected with the trust-based method, the success rate of routing will increase by at least 3% without additional system cost. If the success rate of routing is guaranteed equal to that of the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method, the data amount loaded by the nodes will be reduced by 53.95%, which demonstrates the outstanding performance of ASTR scheme.

The rest of our paper is organized as follows: In Section 2, a literature review of the research related to this work is presented. Then the system model and problem statement are described in Section 3. In Section 4, we proposed an efficient aggregate signature based trust routing (ASTR) scheme. The performance analysis of the ASTR scheme is provided in Section 5. Finally, Section 6 provides the conclusion.

2. Related Work

Data security and data integrity are the two most important concerns in data gathering of WSNs and also the objects of this research. Therefore, this section mainly discusses some research work related to data security and data integrity. At first, it introduces the existing researches on data security, which mainly focuses on the method and strategy to guarantee the safe routing of data to the sink. The functions based on this strategy can be classified into the following.

(1) Guaranteeing the safe transmission of data to the sink no matter whether an attack exists in the network: the early researches did not detect the existence of attacks (such as black, DOS, and SFA attack) and, instead, adopted a multipath routing method to reliably guarantee the data can reach the sink [21, 23]. This strategy has an advantage of wide applicability in effectively overcoming similar attack behavior that impedes the transmission of data to the sink [21]. However, the strategy cannot detect the existence of an attack or identify malicious nodes [15].

Karlof and Wagner [21] first suggest that multipath routing can be used to counter these types of attacks. Messages routed over k paths whose nodes are completely disjoint are completely protected against such selective forwarding attacks or black hole attacks involving at most k compromised nodes and still offer some probabilistic protection when over k nodes are compromised. In literature [23], a SEDR scheme is proposed to defense black hole attack. In SEDR scheme, a data packet is divided into m shares which are sent to the sink over different nonintersected paths. Even if some shares are dropped by attacker during routing, the sink can restore the whole data packet with only k shares. The attacker cannot get the contents of data packet from less than k shares. The shortcoming of these researches lies in the following: (a) it can neither detect the existence of attacker in the network nor determine the location of malicious node [22]; (b) it consumes much energy, especially when the number of paths is large [21].

(2) Being able to detect the existence of attacker but not identify malicious nodes: this type of strategy has a main point: detect whether attacks exist during the data transmission and once you find the attacks, send an alarm message to inform the system of the existence of attacks. However, the strategy is not or weakly able to identify malicious nodes.

Sun et al. [24] have proposed a multidataflow topologies (MDT) method to countermeasure such attacks. In MDT, the sensor nodes are divided into two-dataflow topologies, and both dataflow topologies can cover the whole network; therefore it is sufficient to control the entire network that the sink only gets one report from either topology. Through two topologies the sink can defend against similar attacks. If a malicious node exists in one topology, the sink can still obtain packets from other topologies. The shortcoming of this method lies in the following: too weak to resist the attacks and both dataflow topologies can be attacked when the number of attackers is large, which will cause a failure of the strategy. In addition, the method will pay high prices in energy consumption and so forth.

(3) Being able to detect the existence of attacker and identify malicious nodes: the type of strategies has a main point: firstly detect whether an attacker exists in the network and if any attacker is detected, use a certain mechanism to identify malicious nodes and expel the malicious nodes from the data routing, eliminating their adverse effects.

Xiao et al. [20] have proposed a checkpoint-based multihop acknowledgement (CHEMAS) scheme for identifying suspect nodes in selective forwarding attack. In CHEMAS scheme, some intermediate nodes over a forwarding path are randomly selected as checkpoint nodes which are responsible for generating acknowledgements for each packet received. Each intermediate node in a forwarding path has the potential to detect abnormal packet loss and identify suspect nodes if it does not receive enough acknowledgements from the downstream checkpoint nodes.

Besides, the attack detecting strategies can be classified into distributed detection strategies and centralized detection strategies based on their execution characteristics. In distributed detection strategy, each source node and participant node of the data routing or neighbor node monitors and detects the existence of attack. Generally the source node summarizes the detection results and reports them to the system (or sink). In centralized detection strategy, a central information processing center (such as sink) is in place and monitoring information from all nodes is reported to and processed by the information processing center. Generally speaking, the distributed strategy has robust performance and the centralized strategy has a shortcoming of single-point failure.

The above classification is only one of the commonly used methods. Actually, different attack detecting strategies usually have multiple characteristics; for example, a distributed strategy may be able to identify malicious nodes; some centralized strategies may have only detection function.

Another group of researches were conducted not against specific attack behaviors but on a widely applicable method to defend against attacks. Among them, traceback is a common

method different from the above ones that adopt active defense against attacks. In the abovementioned methods, the strategies usually aim to design some methods to avoid attacks to nodes and send data packets to the sink successfully. In comparison, the main point of traceback method lies not in how to avoid attack but in designing a method to mark the origin of source nodes and routing paths, which is able to determine the attack source once an attack occurs and then take measures to eliminate the attack source to ensure the network security. The mechanism can be described as monitoring in advance and imposing punishment after an attack event occurs [22]. This type of strategies mainly adopts two methods, namely, mark and logging. The mark method adds the node ID during the routing of data packet to the sink and determines the location of malicious node with a high probability when the network is attacked. The logging method is mainly adopted based on the mark method. In the mark method, as the data packet is routed towards the sink, an increasing number of node IDs are added to the data packet, so the data packet is becoming longer, which increases the energy consumption of sensor nodes. When a node receives the data packet, the logging method checks the length of ID attached to the data packet. When the length exceeds a certain value, the ID information will be saved on the node to reduce the energy consumption of nodes. Therefore, if more information is marked in the traceback method, the sink will own more information when the network is attacked, determine the location of malicious nodes, and remove them more easily. However, when more information is marked and more IDs are attached to the data packet, the nodes will consume more energy. The main point of research on traceback method lies in how to minimize the number of marks and improve the accuracy of determining the location of source node. To achieve this target, the probability mark method is usually adopted. For details about these researches, see literature [22].

The preceding paragraphs mainly discuss the existing researches on data security with the main purpose of effectively routing the data to the sink. Another important work is how to ensure the data integrity, so the next paragraph introduces some researches on this topic.

The main method to ensure data integrity is the digital signature technology which ensures the source of data packet can be verified. This type of methods is widely used in WSNs [1]. The main ideas of these methods are as follows: a source node needs to provide a digital signature to each of the data when sending data packets, so the sink can determine whether the data is sent by the real node, which will prevent the illegal node from sending false data. For details of these researches, see [27], where the authors proposed an effective and safe method to defend against the false data injection attack. However, the digital signature method has shortcomings: the digital signature has a requirement on hardware and increases the length of data packet, which affects the energy consumption, delay, and communication bandwidth. Therefore, how to effectively ensure the data integrity and data security of WSNs remains an important challenge issue.

3. System Model and Problem Statement

3.1. System Model. The network model in this paper is a typical planar periodic data collection wireless sensor network similar to [28–33]. Its system model is as follows.

(1) n homogeneous sensor nodes are randomly deployed in a two-dimensional planar network with a radius of R, a sink is at the center, and the node density is ρ. The node communication radius is r. The data sensed by the network is periodic data collection type. For example, the sensor network is a farm monitoring the information, such as temperature and moisture within an area. Each node produces one data packet in a sensor period and sends it to the sink via multihop relay [34, 35] and the data vector collected in a period is $X = [x_1, x_2, \ldots, x_n]$. The network lifetime is defined as the number of data collection periods when the first node dies [31, 32, 35]. The communication is considered to be perfect, so the lost packets are considered dropped as a result of attack. Although the real communication channel is not perfect, the sender may adopt various methods, such as sen-wait ARQ protocol to ensure the successful transmission of packets to the receiver. The ASTR scheme can also be used in networks with imperfect communication channels under condition that the system parameters have been modified.

(2) Data aggregation model: it adopts data aggregate method for data collection to form a typical data aggregation model similar to [1]. In such data aggregation mode, when the network is collecting data, a set of aggregators are selected and other nodes are called simply nodes. Each simple node belongs to one of the aggregators in which that simple node can send its data to that aggregator directly. The node s_i is said to be the member of aggregator s_j if the node s_i belongs to aggregator s_j. The aggregator node receives the data of all member nodes and aggregates them into one data packet and then multihop routes the data packet to the sink with the shortest routing method. $\mathfrak{D}_i$ represents the original data packet produced by node s_i and $\mathfrak{S}(s_i, s_j)$ represents the aggregation result of data of node s_i and node s_j and $\mathfrak{T}_i$ represents the final aggregation result of the data of all input nodes and node s_i.

When aggregator s_i receives the data packet sent by member node j, it aggregates the data packet $\mathfrak{D}_j$ sent by node j and the current data packet of aggregator s_i (original data packet $\mathfrak{D}_i$ of aggregator s_i or the intermediate aggregation result $\mathfrak{S}_i$ of the data of aggregator s_i and its member nodes) with the following formula:

$$\mathfrak{T}\left(s_i, s_j\right) = \max\left(\mathfrak{S}_i, \mathfrak{D}_j\right) + \left(1 - c_{i,j}\right)\min\left(\mathfrak{S}_i, \mathfrak{D}_j\right), \quad (1)$$

where $c_{i,j}$ is the correlation coefficient between nodes s_i and s_j [4, 15]. A larger $c_{i,j}$ indicates a higher correlation between the data of nodes and a smaller length of data packet formed after the data aggregation.

3.2. Security Model. The attacker is considered to have a strong intelligence [20, 22, 23]. It is actually a malicious node or a node that obtains the legal status through compromising a small portion of sensor nodes and then drops some data packets at a certain probability (if the drop probability is 1,

Table 1: Network parameters.

Symbol	Description	Value
d_0	Threshold distance (m)	87
r_s	Sensing range (m)	15
E_{elec}	Transmitting circuit loss (nJ/bit)	50
e_{fs}	Power amplification for the free space (pJ/bit/m^2)	10
e_{amp}	Power amplification for the multipath fading (pJ/bit/m^4)	0.0013
E_{init}	Initial energy (J)	0.5

then it is black hole attacker; others it is likely to be selective forwarding attacker or DOS attacker), the aim is to try not to expose themselves, and to make the greatest harm to the network. Those attackers can also forge real nodes to launch various attacks, such as false data injection attack. At the same time, the attackers can also collude to launch attacks. In this paper, we assume that the proportion of malicious nodes is small, for example, smaller than ϱ, most nodes in the network are normal nodes [23]. It is obvious that if most nodes in a network are malicious nodes, the network safety cannot be guaranteed.

3.3. Energy Consumption Model. In this paper, we adopt the topical energy consumption model in [31, 33, 35], where the transmission energy consumption, ω_t, follows (2) and energy consumption, ω_r, for receiving follows (3):

$$\begin{aligned} \omega_{t,1}(d) &= lE_{\text{elec}} + l\varepsilon_{\text{fs}}d^2 \quad \text{if } d < d_0, \\ \omega_{t,2}(d) &= lE_{\text{elec}} + l\varepsilon_{\text{amp}}d^4 \quad \text{if } d \geqq d_0, \end{aligned} \tag{2}$$

$$\omega_r = lE_{\text{elec}}, \tag{3}$$

where E_{elec} represents transmitting circuit loss. Both the free space (d^2 power loss) and the multipath fading (d^4 power loss) channel models are used. If the transmission distance is less than the threshold d_0, the power amplifier loss is based on free-space model; if the transmission distance is larger than or equal to the threshold d_0, respectively, the multipath attenuation model is used. ε_{fs} and ε_{amp} are the energy required by power amplification in the two models.

3.4. Problem Statement. The main goal of this paper is to design a data gathering scheme for WSNs that guarantees the data integrity and data security and improves network energy utilization while ensuring long network lifetime. It can be expressed as follows.

(1) Data Integrity. It is assurance to the recipient that the data came from the expected sender and has not been altered in transit [1], although the data is sent to the sink after data aggregation and multihop routed.

(2) Maximizing the Probability of Successively Routing the Data and Abstract to the Sink. The probability of successively routing data packets to the sink can be defined as the ratio between the number of data packets received by the sink and the total number of data packets sent by the network. The maximum data routing success rate can be computed as follows:

$$\max(\mathfrak{P}_D) = \max\left(\frac{D_s}{D_t}\right), \tag{4}$$

where D_t represents the total number of data packets sent in the network and D_s represents the number of data packets successively received by the sink.

Besides, abstract information reaching the sink also has positive effect on the network safety. It can indicate whether the linked data has been sent. If the sink receives the abstract information but fails to receive the linked data packet, it will recognize that the data packet has been attacked and so not received. Therefore, the ASTR scheme will also improve the success rate for abstract information to reach the sink:

$$\max(\mathfrak{P}_A) = \max\left(\frac{A_s}{A_t}\right), \tag{5}$$

where A_t represents the total amount of abstract information sent in the network and A_s represents the amount of abstract information received by the sink.

(3) Maximizing Energy Utilization. Energy utilization is the ratio of the total energy consumed by the network to the total initial energy of the network while the network dies, as shown in

$$\max(\mathscr{R}_u) = \max\left[\frac{\left(\sum_{i=1}^{n} \mathfrak{Q}_i\right)}{\left(\sum_{i=1}^{n} E_{\text{ini}}^i\right)}\right], \tag{6}$$

where i is the ith node in the network, n is the total number of nodes in the network, $\mathfrak{Q}_i$ represents the energy consumption of n_i, and E_{ini}^i represents the initial energy of n_i, which is given in Table 1. The maximization of network energy utilization will improve the effective use of network energy, so that the ratio of energy consumption to the initial energy in the network is largest.

(4) Maximizing Network Lifetime. Network lifetime is defined as the death time of the first node in the network [31, 34, 35]. Considering that the energy consumption of the ith node in the network is $\mathfrak{Q}_i$, its initial energy is E_{ini}^i, and there are n nodes in the network. To maximize the lifetime of the whole network, the network lifetime of the first node to die in the network should be maximized. Therefore, (7) can be obtained:

$$\max(\mathscr{L}) = \max\left[\min_{1 \leq i \leq n}\left(\frac{E_{\text{ini}}^i}{\mathfrak{Q}_i}\right)\right]. \tag{7}$$

In summary, the research objectives are as follows:

$$\max(\mathfrak{P}_D) = \max\left(\frac{D_s}{D_t}\right),$$

$$\max(\mathfrak{P}_A) = \max\left(\frac{A_s}{A_t}\right),$$

$$\max(\mathscr{R}_u) = \max\left[\frac{(\sum_{i=1}^{n} \omega_i)}{(\sum_{i=1}^{n} E_{\text{ini}}^i)}\right], \tag{8}$$

$$\max(\mathscr{L}) = \max\left[\min_{1\le i\le n}\left(\frac{E_{\text{ini}}^i}{\omega_i}\right)\right],$$

s.t. data is integrity.

4. The ASTR Scheme

4.1. The Research Motivation of ASTR Scheme. This section discusses the research motivation of ASTR scheme. Firstly, the energy consumption of the network data collection shows that when the network contains no attackers, the energy consumption is as shown in Figure 1 (M in the figure represents that the same data packets are sent for M times, the same below). Figure 1 shows that all data in the network can successively reach the sink because no attackers exist. Therefore, the energy consumption of nodes close to the sink area is very high. However, if attackers exist, the attack behaviors of attackers will cause a certain proportion of data packets to be attacked during the routing, dropped, and not able reach the sink. In this case, the number of data packets reaching the sink is smaller than that number in network without attackers, so its energy consumption decreases. It should be noted that the decrease of energy consumption is caused by the reduction of data packets reaching the sink as a result of attack instead of improvement in network performance. This is not a good condition because the idealistic effect for guaranteeing data security is to ensure all data packets in the network can reach the sink and the highest energy consumption is the same as the network that is not attacked. One of the methods to improve the reach rate of data packets is to send the same data for M times over different paths, which is able to significantly raise the reach rate of data. As shown in Figure 1, assuming the probability of successively sending data packets under attack $p = 0.90$, the reach rate is only 0.90 for packets routed by one hop; if the times of sending the same data packets $M = 2$, the one-hop reach rate is 99%, apparently higher than 0.90.

This method is actually the multipath routing (MPR) scheme described above. It is direct and effective and so widely researched. However, the shortcomings of the method are rarely researched. Figure 1 shows that when $M = 2$ and $p = 0.90$, the actual number of data reaching the sink has exceeded the number of data when no data packet is lost. However, the effective reach rate of data packets is only 55%. It demonstrates that when this method is used, a large amount of repetitive data packets will reach the sink, but some data packets fail to reach the sink. The repetitive data packets (redundant data packets) reaching the sink are useless

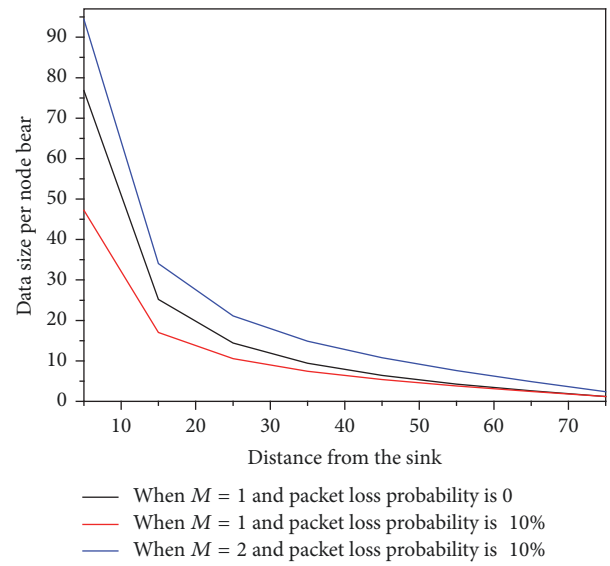

Figure 1: Data loaded by nodes under different data collection strategies.

to improve the reach rate of data packets but increase the energy consumption of the network and shorten the network lifetime.

If we can reduce the redundant data reaching the sink and maintain the success rate for data packets to reach the sink, the network lifetime can be extended and the reach rate of data can be ensured at the same time. Based on the above ideas, this paper proposed the ASTR scheme which sends the light abstract information join data packets to improve the network security and energy efficiency. Instead of simply sending $\mathscr{M}$ data packets, this method sends $\mathscr{N}$ abstracts (containing the information such as node ID, time, and data packet ID) while it sends $\mathscr{M}$ data (relatively small and generally $\mathscr{M} = 1$), which is known as $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method. If the message received by the sink contains no data packets, the sink will notify the node to send $(\mathscr{M} + \mathscr{N})$ messages again until it receives the data packets or the number of sending times reaches the maximum value k. Compared with the past strategies, the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ data collection method with maximum value k can effectively reduce the energy consumption of the network. This is because of the following: if $\mathscr{M} = 1$ in the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method, once the sink receives the data packets, the subsequent sending actions will be stopped, so the sink cannot receive repetitive data packets. As a result, the sink will receive only redundant abstracts but no redundant data packets with this method. Abstracts are lightweight packets with a length of 1/10 or 1/100 of the length of data packets; hence, sending lightweight abstracts instead of heavy data packets is conducive to reducing energy consumption.

Table 2 compares the data amount sent by the node 5 hops from the sink in $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing strategy and multipath routing strategy of sending data the same packets for M times while ensuring the same reach rate of data packets. The first line of Table 2 represents the data in M multipath

Table 2: Comparison of data amount loaded by nodes in different strategies.

	(M, data, sp) (2, 200, 0.83)	(M, data, sp) (3, 300, 0.93)	(M, data, sp) (4, 400, 0.97)	(M, data, sp) (5, 500, 0.99)
$k = 3$	131.82 ($N = 1$)	155.81 ($N = 2$)		
$k = 4$	142.12 ($N = 1$)	171.56 ($N = 2$)		
$k = 5$	146.51 ($N = 1$)	179.08 ($N = 2$)	223.37 ($N = 4$)	
$k = 6$	148.31 ($N = 1$)	182.52 ($N = 2$)	208.03 ($N = 3$)	247.26 ($N = 5$)
$k = 7$	149.02 ($N = 1$)	184.06 ($N = 2$)	210.17 ($N = 3$)	231.35 ($N = 4$)

routing strategy and $(M, data, sp)$ in the first line represent the number of repeatedly sent data packets, data amount sent by nodes, and the proportion of data received by the receiving nodes. For example, (2, 200, and 0.83) represent the following: the number of repeatedly sent data packets: $M = 2$, data amount sent by the sender = 200 bits, and the proportion of data received by the receiver = 0.83. Other lines represent the parameters of the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ strategy, where $\mathscr{M} = 1$ and k represents the times of repetitive sending. Table 2 shows that when M is 2 in MPR scheme and the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing strategy is used, the data amount loaded by nodes is only 70% of the data amount in M multipath routing strategy. As M increases, the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing strategy presents better performance. When $M = 5$, the data amount loaded by nodes is only 50% of that in M multipath routing strategy. The $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing strategy proposed in this paper has prominent advantages.

The second research motivation: in the preceding research motivation, the message is set free after being sent. It is obvious that selecting a trustable routing path for every hop during the routing will effectively improve the success rate of routing and reduce the sending times of data packets. In ASTR scheme, a data packet will not be sent again once received. Therefore, adopting a trustable routing path will improve the success rate of routing, reduce the sending times of data, and thereby effectively enhance the network performance.

The third research motivation is adopting the data aggregation method to reduce the energy consumption. As shown in Figure 2, the network will undertake much less data after adopting the data aggregation and thereby effectively extend the network lifetime.

4.2. The Design of ASTR. This section introduces the detailed design of ASTR scheme. ASTR scheme is as shown in Figure 3. It mainly consists of the following important parts: (1) data aggregate signature; (2) $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method; (3) trust routing method.

(1) Aggregate Signature Stage. In this stage, ID-based aggregate signature technology [1] is adopted in ASTR scheme. ID-based aggregate signature can ensure the data packets of several source nodes are sent to the aggregator and, after aggregating signature and sending the data to the sink through multiple hops, the aggregator can provide assurance to the recipient that the message came from the expected sender and has not been altered in transit [1]. Hence, in ASTR

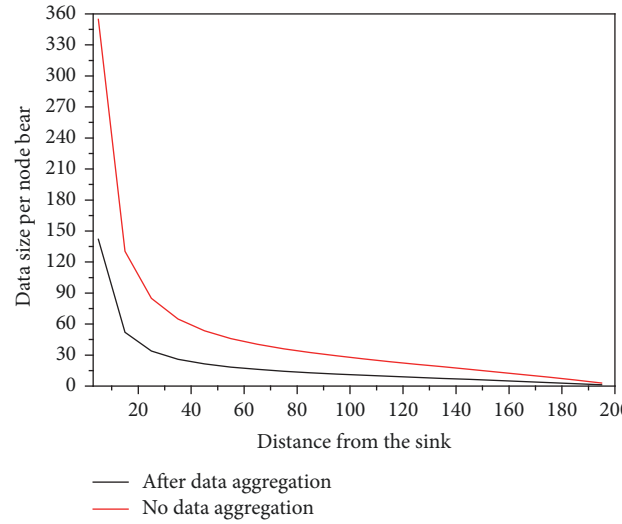

Figure 2: Comparison of data amount in data aggregation and non-data aggregation strategies.

scheme, the data packets are not directly sent to the sink but sent after data aggregation, which effectively reduces the data amount loaded by nodes (see Figure 2). The process of data aggregation is shown in Figure 3. When the nodes s_0, s_1, s_2, s_3, and s_4 intend to send the data packets to the sink, they will select one node among them, such as node s_0 as the aggregator while other nodes become the member nodes of aggregator node s_0 and send data packets to the aggregator node s_0. After receiving the data packets sent by all member nodes, the aggregator node s_0 adopts the aggregate signature method in [1] to aggregate them into one data packet and sends the packet to the sink (if $\mathscr{M} > 1$, the data packet will be sent to the sink in a method similar to multipath routing). Reference [1] has shown that the data aggregation method can be authenticated for each of the data of node. Besides, the selection of aggregator is similar to that of cluster head which can be found in [36]. On the other hand, the ASTR scheme can simultaneously send $\mathscr{N}$ abstracts which are also produced by the aggregator. When receiving all data packets from member nodes, the aggregator will know the IDs of all member nodes, IDs of data packets, and data production time. Therefore, the aggregator can produce the abstracts from the owned information during the data aggregation with the aggregate signature method. The abstract is short in

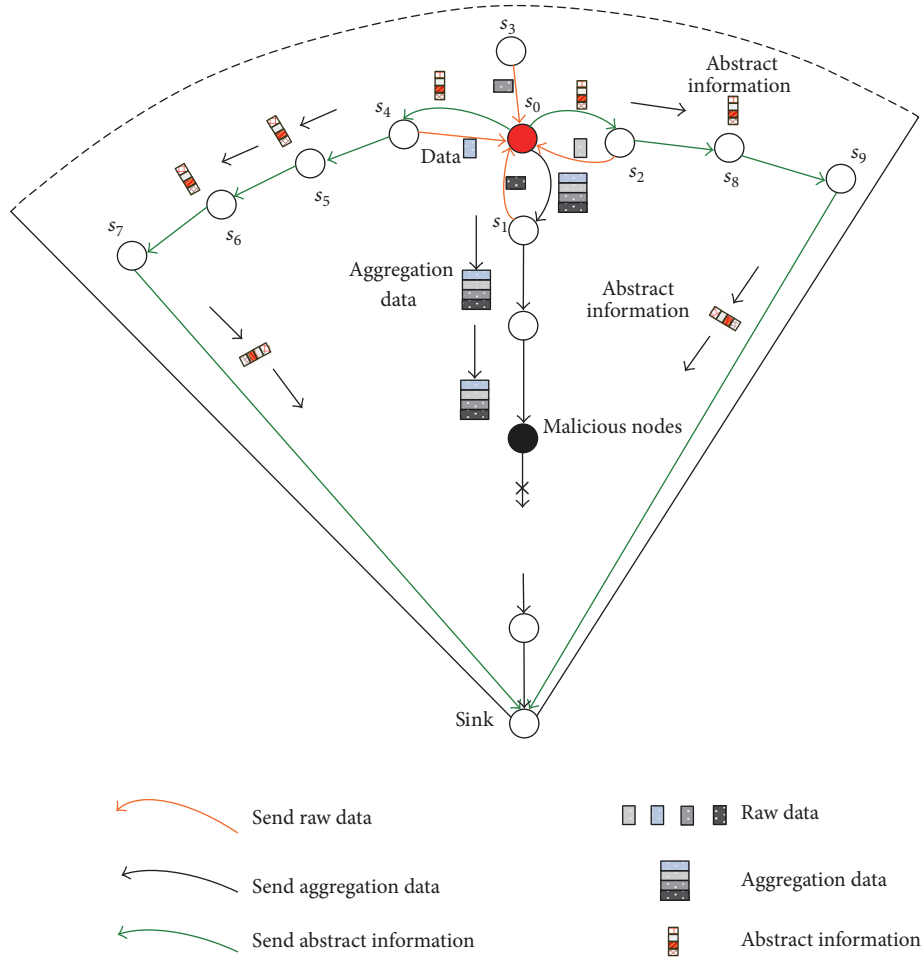

Figure 3: The framework of ASTR scheme.

length and further shortened after the data aggregate, which effectively improves the network performance.

(2) $\mathcal{R}(\mathcal{M}, \mathcal{N})$ Routing Method. After the aggregate signature, both data packet and abstract have been produced, so this section mainly describes how to effectively route the data packet to the sink, that is, $\mathcal{R}(\mathcal{M}, \mathcal{N})$ routing method. The ASTR scheme ensures the probability for data to successively reach the sink is higher than the specified level q. The $\mathcal{R}(\mathcal{M}, \mathcal{N})$ routing method is the method to send $\mathcal{M}$ data packets and $\mathcal{N}$ abstracts at one time and route them to the sink over different routing paths with k times of sending to the maximum. In this paper, the operation of sending $\mathcal{M}$ data packets and $\mathcal{N}$ abstracts at one time is called one sending of $(\mathcal{M} + \mathcal{N})$, written as $\mathbb{S}(\mathcal{M} + \mathcal{N})$. The process of $\mathcal{R}(\mathcal{M}, \mathcal{N})$ routing is as follows:

(a) Sender performs one $\mathbb{S}(\mathcal{M} + \mathcal{N})$ operation; $\mathcal{K} = 1$.

(b) If the sink fails to receive data or abstract, the data packet sending fails and the data sending will end.

(c) Otherwise, if the sink receives the data sent successfully, the sending of this data packet will stop.

(d) If the sink receives only abstract but no data packet, it will notify the sender to perform the $\mathbb{S}(\mathcal{M} + \mathcal{N})$ operation again.

(e) The sender will detect whether the number of resending instances has reached the maximum value k. If $\mathcal{K} < k$, it will perform another $\mathbb{S}(\mathcal{M} + \mathcal{N})$ operation and otherwise gives up the sending of data. k is the maximum number of operations calculated based on the probability for data to successively reach the sink required by the application (see formula (17)).

The process of $\mathbb{S}(\mathcal{M} + \mathcal{N})$ operation is as follows: firstly, the aggregator copies the data packet for $\mathcal{M}$ times during an operation and the $\mathcal{M}$ data packets are routed to the sink over $\mathcal{M}$ paths. At the same time, it copies the abstract for $\mathcal{N}$ times and the $\mathcal{N}$ abstracts are routed to the sink over $\mathcal{N}$ paths. The following describes the routing process of abstract to explain the routing process of data or abstract. The routing process for sending of the ith abstract by aggregator s_0 is taken as an example to describe the $\mathbb{S}(\mathcal{M} + \mathcal{N})$ operation. As shown in Figure 3, aggregator s_0 firstly generates a random number $\mathcal{d}_i$ in $\{1, \mathcal{d}\}$ and $\mathcal{d}_i$ represents the length of the ith

abstract routed horizontally before being routed to the sink with the shortest routing method. In this paper, horizontal routing refers to each time when the node selects a node on the left (right) that is the same hops as itself from the sink as the next relay node for routing. In this way, aggregator s_0 selects its neighbor node s_4 on the left as the relay node and sends the abstract to s_4. s_4 selects its neighbor node s_5 following the same direction. The process proceeds until the abstract is routed to node s_7 and the horizontal routing stops when its routing distance reaches $\mathscr{A}_i$. Starting from node s_7, the node will select the neighbor node closest to the sink until the abstract is routed to the sink. The routing process of other $\mathscr{N}-1$ abstracts is similar to the above. However, the difference lies in that the other $\mathscr{N}-1$ abstracts will select the node that has not been selected by the preceding nodes or a highly trustable node as the relay node. The routing process of data packets is very similar to the routing process of abstract. The value of $\mathscr{M}$ for routing of data packets is usually small; for example, $\mathscr{M}=1$, and its routing process is as shown in Figure 3.

(3) Trust Routing Scheme. In ASTR scheme, the trust-based routing is established for improving the reach rate of data packets. The basis of adopting trust routing is as follows: in ASTR scheme, if the sink fails to receive the data packet but receives the abstract, the sink will notify the sender to send the data again and the sender will recognize that an attacker exists in the routing path of data packet (in this paper, the communication is assumed perfect with no loss of data packet, so the loss of data packet is caused by attack). Therefore, the sender decreases the trust of the routing path which the data just passed through to prevent another data packet from passing the path containing an attacker, which increases the success rate of routing. Actually, the sender does not know the nodes through which the data is routed, so it can only mark neighbor nodes. In other words, the sender will decrease the trust of neighbor nodes of the failing routing path to reduce the probability of another data forwarding through the neighbor node and thereby increase the success rate of routing. $\mathscr{T}_{i,j}$ represents the trust of node s_i on neighbor node s_j. When the sender node s_i successfully sends the data packet through neighbor node s_j, the trust of $\mathscr{T}_{i,j}$ will be lifted by ∂; on the contrary, if the data sending fails through s_j, the trust of $\mathscr{T}_{i,j}$ will be decreased by ∂:

$$\mathscr{T}_{i,j} = \mathscr{T}_{i,j} + \partial \quad \text{if } s_i'\text{s data to sink successfully via } s_j, \quad (9)$$

$$\mathscr{T}_{i,j} = \mathscr{T}_{i,j} + \partial, \quad \text{if } s_i'\text{s data to sink failure via } s_j.$$

After the calculation of trust, the sender node s_i will select the node with high trust from the neighbor nodes for data routing, which can effectively improve the success rate of routing.

The ASTR scheme has been completely introduced, so the following will briefly describe the advantages of ASTR scheme compared with the past schemes. Generally speaking, the ASTR scheme has the following advantages superior to the preceding schemes.

(1) The ASTR scheme adopts the aggregate signature technology so that the source data can be authenticated by sink. In addition, after the data aggregation is adopted, the network load is significantly reduced, which greatly improves the security and extends the network lifetime compared with the past schemes.

(2) The ASTR scheme proposes the $\mathscr{R}(\mathscr{M},\mathscr{N})$ routing method which can more effectively improve the probability for data to reach the sink safely and reduce the data amount, indicating an obvious improvement in both security and network lifetime compared with the past schemes. To ensure the data can successively reach the sink, the past schemes usually adopt the M multipath routing scheme (MPR), which forces the nodes to undertake a large amount of data, but the proportion of effective data reaching the sink is low. The ASTR scheme mainly improves the past schemes in two aspects: (a) reducing the sending of heavy data and increasing abstract to lift the success rate for data to reach the sink and reduce the data amount loaded by nodes and (b) changing the method in past MPR schemes to send M data at one time. It sends only a small number (generally 1) of data and sends again when the first sending fails, which effectively reduces the redundant data sent and achieves a high efficiency.

(3) The ASTR scheme adopts a trust routing mechanism that can further improve the success rate of routing.

The detailed description of ASTR scheme is provided in Algorithm 1.

5. Performance Analysis

5.1. The Calculation and Comparison of Data Amount. This section mainly compares the data amount loaded by nodes in the ASTR scheme proposed in this paper and other schemes and thereby demonstrates that the scheme proposed by this paper can effectively reduce the data amount loaded by nodes. Table 3 shows all symbols used in this paper. Firstly, Theorem 1 provides the data amount loaded by nodes in the network where the data packets are transmitted as in a safe network though the network contains attackers that will cause packets loss.

Theorem 1. *If packets loss ratio is 1-p, the data packet will be sent only once after data aggregation and the distance from the data packet to the sink is l, the data amount loaded by the node with $l = hr + x$ can be calculated as follows:*

$$Q_x = \varepsilon(\tau+\delta+\Gamma)\cdot\left(1+\sum_{i=1}^{z}\frac{l+ir}{l}\cdot p^i\right)\Big| \quad z=\left\lfloor\frac{R-l}{r}\right\rfloor. \quad (10)$$

Proof. As shown in Figure 4, we should consider the data amount loaded by the w_x wide annular area where the node is located. The annular area where the node n_x is located will surely undertake the data transfer of nodes in w_x wide annular areas at $l+r, l+2r, \ldots, l+zr$. If w_x is very small, the nodes in this area can be considered undertaking identical

(1) aggregate signature stage
(1) **For** each node **Do**
(2) running aggregator determining algorithm which is is similar to cluster-head selection algorithm in [36];
(3) **End for**
// now, nodes either belong to aggregators or belong to member nodes
(4) **For** each member node **Do**
(5) send its data as well as node ID, data time to its aggregator
(6) **End for**
(7) **For** each aggregator node s_0 **Do**
(8) s_0 aggregate its member nodes' data into a data packet $\mathfrak{D}_0$ using ID-based aggregate signature technology as [1];
(9) s_0 aggregate its member nodes' abstract into an abstract $\mathfrak{A}_0$ using ID-based aggregate signature technology as [1];
(10) **End for**
(2) $\mathscr{R}(\mathscr{M},\mathscr{N})$ stage
(11) s_0 compute k according to the q using formula (17);
//k is the maximum retransmission times of $\mathbb{S}(\mathscr{M}+\mathscr{N})$
(12) Let $\mathscr{K}=0$;
(13) s_0 select a set of values $\{\mathscr{d}_1,\mathscr{d}_2,\ldots,\mathscr{d}_{\mathscr{N}}\}$ random from $\{1,\mathscr{d}\}$
(14) s_0 select $\mathscr{N}$ neighbor nodes $\{s_0^1,s_0^2,\ldots,s_0^{\mathscr{N}}\}$ with higher trust $\mathscr{T}_{0,?}$ than the rest neighbor nodes;
(15) s_0 send $\mathfrak{A}_0$ to each node in the set $\{s_0^1,s_0^2,\ldots,s_0^{\mathscr{N}}\}$;
// begin abstract routing
(16) **For** each s_0^j in the set $\{s_0^1,s_0^2,\ldots,s_0^{\mathscr{N}}\}$ **Do**
(17) **While** $\mathscr{d}_j>0$ **Do**
(18) select its highest trust of left (right) neighbor node $s_0^{j,\mathrm{nex}}$;
(19) send $\mathfrak{A}_0$ to $s_0^{j,\mathrm{nex}}$;
(20) $\mathscr{d}_j=\mathscr{d}_j-1$;
(21) $s_0^j\leftarrow s_0^{j,\mathrm{nex}}$;
(22) **End While**
(23) **While** s_0^j is not sink **Do**
(24) select its high trust and the nearest to sink neighbor $s_0^{j,\mathrm{nex}}$;
(25) send $\mathfrak{A}_0$ to $s_0^{j,\mathrm{nex}}$;
(26) $s_0^j\leftarrow s_0^{j,\mathrm{nex}}$;
(27) **End While**
(28) **End for** //end abstract route to sink
(29) s_0 select $\mathscr{M}$ neighbor nodes $\{s_0^1,s_0^2,\ldots,s_0^{\mathscr{M}}\}$ with higher trust $\mathscr{T}_{0,?}$ than the rest neighbor nodes;
(30) s_0 send $\mathfrak{D}_0$ to each node in the set $\{s_0^1,s_0^2,\ldots,s_0^{\mathscr{M}}\}$;
// begin data routing
(31) **For** each s_0^j in the set $\{s_0^1,s_0^2,\ldots,s_0^{\mathscr{M}}\}$ **Do**
(32) **While** $\mathscr{d}_j>0$ **Do**
(33) select its highest trust of left (right) neighbor node $s_0^{j,\mathrm{nex}}$;
(34) send $\mathfrak{D}_0$ to $s_0^{j,\mathrm{nex}}$;
(35) $\mathscr{d}_j=\mathscr{d}_j-1$;
(36) $s_0^j\leftarrow s_0^{j,\mathrm{nex}}$;
(37) **End While**
(38) **While** s_0^j is not sink **Do**
(39) select its high trust & the nearest to sink neighbor $s_0^{j,\mathrm{nex}}$;
(40) send $\mathfrak{D}_0$ to $s_0^{j,\mathrm{nex}}$;
(41) $s_0^j\leftarrow s_0^{j,\mathrm{nex}}$;
(42) **End While**
(43) **End for** //end data route to sink
(44) Let $\mathscr{K}=\mathscr{K}+1$;
(45) aggregator s_0 wait for the message from sink;
(46) **If** message from sink indicate that data routing failure **then**
(47) **If**$\mathscr{K}<k$
(48) goto step (13);
(49) **End if**
(50) **End if**

Algorithm 1: Continued.

(3) Trust computing stage
(50) aggregator s_0 get a message from sink
(51) **If** message from sink indicate that data routing failure **then**
(52) $\mathcal{T}_{0,j} = \mathcal{T}_{0,j} - \partial$;
// j is the neighbor s_j of s_0 that data routing through s_j
(53) **Else**
(54) $\mathcal{T}_{0,j} = \mathcal{T}_{0,j} + \partial$;
(55) **End if**

Algorithm 1: The algorithm of aggregate signature based trust routing (ASTR) scheme.

Table 3: Parameters related to the calculation.

Symbol	Description
R	Network radius
n	Total number of nodes in the area
r	Emission radius of nodes
h	Number of hops from a node to the base station
p	Success rate of each hop when each data packet is sent to the sink
n_x	Any node in the network
w_x	Width of the annular area where the node n_x is located
ρ	Distribution density of nodes in the network
ε	Network coefficient

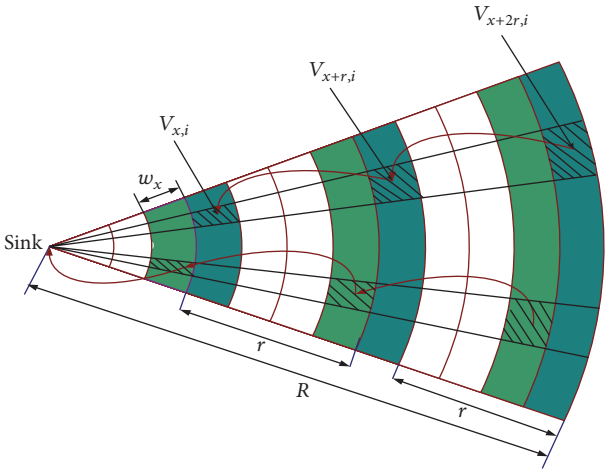

Figure 4: The data forwarding of each node.

data amount. The area of annular area where the node n_x is located is as follows: $S_{n_x} = 2\pi l w_x$

The total number of nodes in the area is

$$N_{n_x} = S_{n_x} \cdot \rho = 2\pi\rho l w_x. \tag{11}$$

It receives and transfers the data in $l + ir \mid i \in \{1, z\}$ area and the area and number of nodes of the annular area V_{l+ir} where $l + ir$ is located are, respectively,

$$\begin{aligned} S_{V_{l+ir}} &= 2\pi (l + ir) w_x, \\ N_{V_{l+ir}} &= 2\pi\rho (l + ir) w_x. \end{aligned} \tag{12}$$

Similarly, the number of nodes of the areas whose data transfer is loaded by V_l area:

$$\begin{aligned} &2\pi\rho l w_x + 2\pi\rho (l + r) w_x + 2\pi\rho (l + 2r) w_x + \cdots + \\ &2\pi\rho (l + zr) w_x \mid \quad z = \left\lfloor \frac{R - l}{r} \right\rfloor. \end{aligned} \tag{13}$$

In a period, each node produces the following: one data packet, one ID packet, and one time axis. The size of data packet is τ, the size of ID packet is δ, and the size of time axis is Γ. Therefore, the data amount produced by a node in a period is $\tau + \delta + \Gamma$. Assume the aggregation is ε, but when the data packet is being transmitted to the base station, the probability for successful transmission of each hop is p and the node in area V_{l+ir} has a distance of i hops from i, so the probability of transmitting the data packet of V_{l+ir} area to area V_l is p^i and the number of data packets of V_{l+ir} area loaded by the node in V_l area is $2\pi\rho(l + ir)w_x \cdot p^i$. The data will time the aggregation coefficient after the data aggregation ε, so the data amount loaded by each node in V_l area can be calculated as follows:

$$\begin{aligned} Q_x &= \varepsilon (\tau + \delta + \Gamma) \\ &\quad \cdot \frac{\left[2\pi\rho l w_x + \sum_{i=1}^{z} 2\pi\rho (l + ir) w_x p^i\right]}{2\pi\rho l w_x} \\ &= \varepsilon (\tau + \delta + \Gamma) \left(1 + \sum_{i=1}^{z} \frac{l + ir}{l} \cdot p^i\right). \end{aligned} \tag{14}$$

□

The ASTR scheme adopts the $\mathcal{R}(\mathcal{M}, \mathcal{N})$ routing method. If the message received by the sink contains no data packet, the sink will notify the node to resend $(\mathcal{M} + \mathcal{N})$ message until the sink has received the data packet or the resending instances have reached the maximum value k. The following calculate the data amount of the $\mathcal{R}(\mathcal{M}, \mathcal{N})$ routing method.

Theorem 2. *Assume the node has a distance of l from the sink, $l = hr + x$, the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method is adopted, and the size of abstract is μ times of the size of data packet. If the number of hops of the abstract during horizontal routing is a random number in $\{1, \mathscr{d}\}$, and the data amount loaded by the node is as*

$$\begin{aligned} S_x &= \mathscr{M}Q_x + \left(Q_x \cdot p^z + \mu\mathscr{N} \sum_{j=1}^{\mathscr{d}/2} p^j \right) \\ &= \mathscr{M} \cdot \varepsilon(\tau + \delta + \Gamma) \cdot \left(1 + \sum_{i=1}^{z} \frac{l + ir}{l} \cdot p^i \right) \\ &\quad + \varepsilon(\tau + \delta + \Gamma) \cdot \left(1 + \sum_{i=1}^{z} \frac{l + ir}{l} \cdot p^i \right) \cdot p^z + \mu\mathscr{N} \\ &\quad \cdot \sum_{j=1}^{\mathscr{d}/2} p^j \Big| \quad z = \left\lfloor \frac{R - l}{r} \right\rfloor. \end{aligned} \tag{15}$$

Proof. Theorem 1 shows that the data amount loaded by each node is Q_x when one data packet is sent, so the data amount loaded by each node is $\mathscr{M}Q_x$ when $\mathscr{M}$ data packets are sent. Different from data packet, the abstract will be horizontally routed by $\mathscr{d}/2$ hops during sending. Therefore, the incremental data amount loaded by each node is $\mathscr{N} \sum_{j=1}^{\mathscr{d}/2} p^j$ and the data amount loaded by each node is $Q_x \cdot p^z + \mu\mathscr{N} \sum_{j=1}^{\mathscr{d}/2} p^j$ when $\mathscr{N}$ abstracts are sent. □

Theorem 2 provides the data amount loaded by the node in the network which is l from the sink when the data aggregation is adopted with a packet loss ratio of 1-p and the $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method is used. It the MPR scheme is adopted, the data amount loaded by a node in the network is as shown in Figures 5–7. Figure 5 shows that, in MPR scheme, the data amount loaded by the node closed to the sink increases more rapidly with the increase of $\mathscr{M}$ because the node closer to the network center has to undertake all data packets routed from periphery of the network. Therefore, the energy of nodes near the sink will be consumed fast and the lifetime of the whole network is short. Figure 6 compares the data amount loaded by the node under different packet loss ratios. Figure 6 shows that when the success rate of successive transmission of each hop p decreases and the node still sends $\mathscr{M}$ data packets, the data amount loaded by each node will be reduced due to packet loss. Figure 6 also shows that if each node sends one less repetitive data packet and the success rate of transmission of each hop decreases by 0.1, the data amount has no change compared with the above. It should be noted that the success rate of transmission of each hop is positively correlated with the reliability of the whole network. Figure 7 compares the data amount in networks of different sizes. It shows that when other variables are all the same, the node in a larger network will undertake more data. Regardless of the network size, the broken line representing the data amount flattens at the same location which is 1 hop from the sink. According to the data, the nodes in a circle with the sink as the center and r as the radius undertake 45.87% of the total data amount of all nodes.

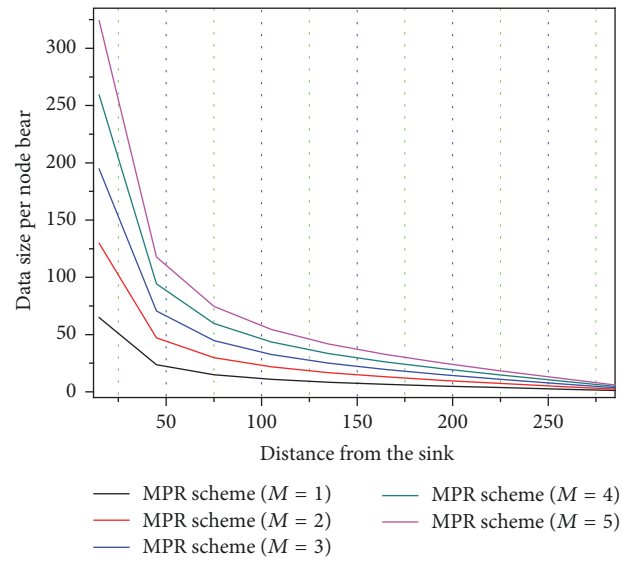

Figure 5: Data amount loaded by a node in MPR scheme when the node sends different numbers of data packets ($p = 0.90$).

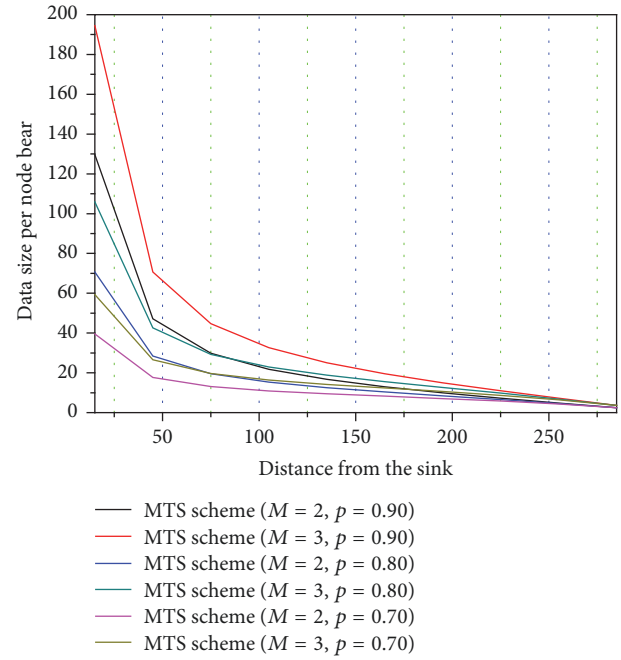

Figure 6: Data amount loaded by a node in MPR scheme with different success rate of transmission of each hop ($R = 300$).

It the ASTR scheme, that is, $\mathscr{R}(1, \mathscr{N})$, routing method is adopted and $\mathscr{M} = 1$ and $\mathscr{N}$ abstracts are sent, and the data amount loaded by a node in the network is as shown in Figure 8. Figure 8 shows that when the more abstracts are sent, the data amount will increase. During the comparison between Figures 8 and 5, the intersection of the broken line and longitudinal axis shows that the data amount is 115.56 when $\mathscr{N} = 5$, smaller than the data amount 129.64 when $\mathscr{M} = 2$, so the increase of number of abstracts has no significant impact on the data amount. Therefore, increasing the number

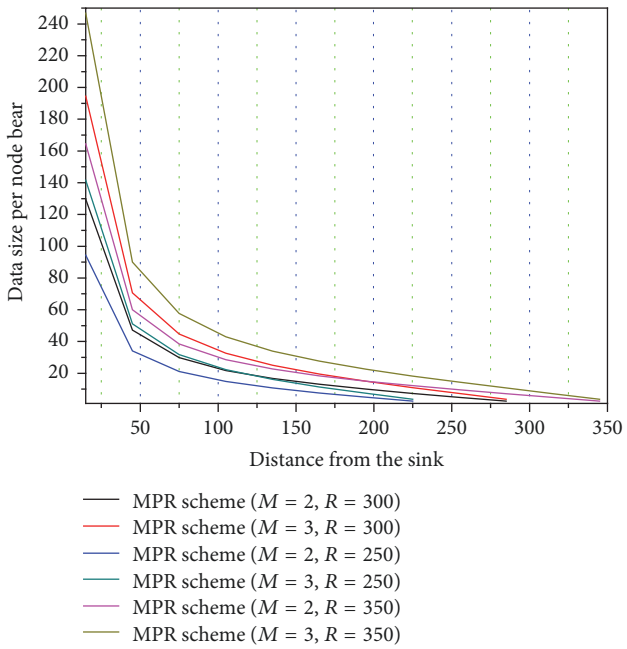

Figure 7: Data amount loaded by a node in MPR scheme for network of different sizes (p = 0.90).

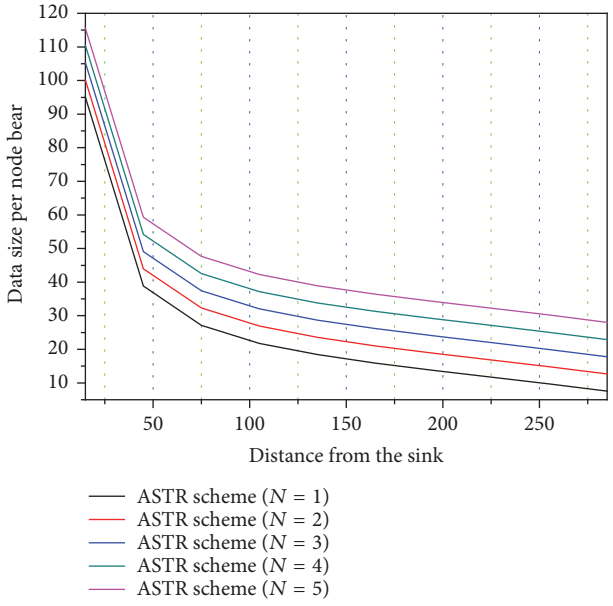

Figure 8: Data amount loaded by a node in ASTR scheme when the node sends different number of abstracts (p = 0.90, $\mathscr{M}$ = 1).

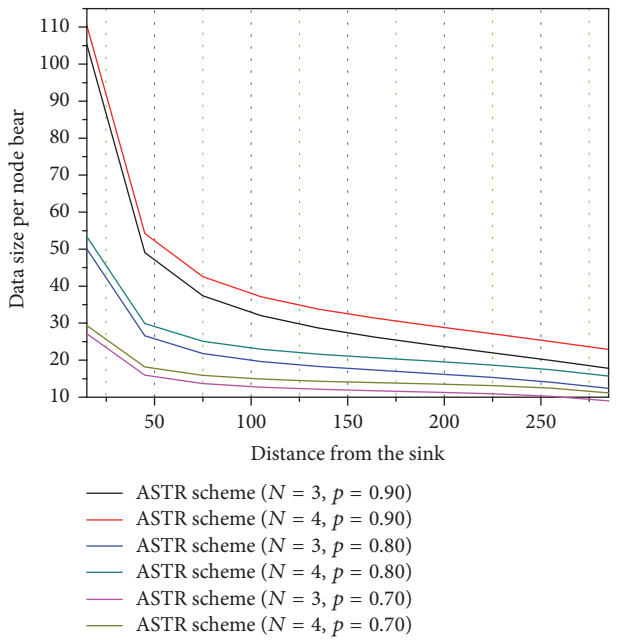

Figure 9: Data amount loaded by a node in ASTR scheme under different success rate of transmission of each hop (R = 300; $\mathscr{M}$ = 1).

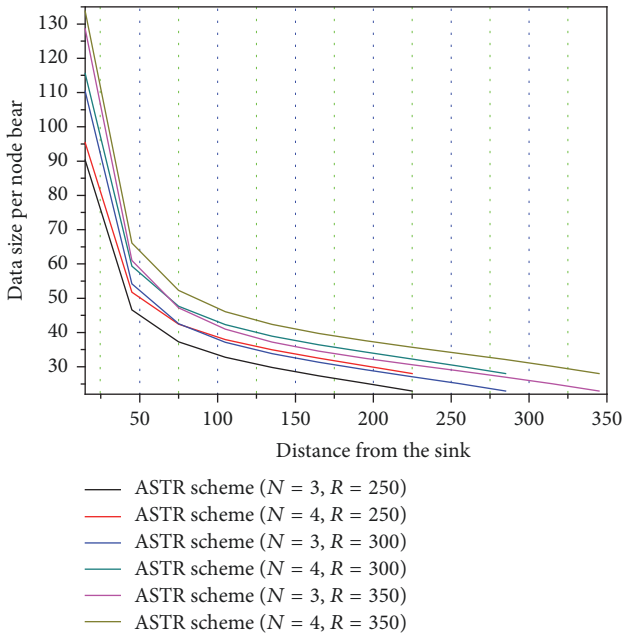

Figure 10: Data amount loaded by a node in ASTR scheme for network of different sizes (p = 0.90; $\mathscr{M}$ = 1).

of abstracts is a better scheme compared with increasing the number of data packets to improve the successful reach rate because the node closer to the sink undertakes less data amount, which will extend the lifetime of network and improve the network performance.

Figure 9 compares the data amount loaded by each node in ASTR scheme under different packet loss ratios. Figure 9 clearly shows that the increase in number of abstracts has no great impact on the data amount, but the increase of p will cause a significant increase of data amount. Figure 10 compares the data amount loaded by each node when $\mathscr{M}$ = 1 and $\mathscr{N}$ abstracts are sent.

A comparison is conducted between the data amount in MPR scheme and ASTR scheme (see Figures 5–7 versus Figures 8–10). In Figures 5–10, the experiment results show

that, in ASTR scheme, the data amount loaded by a node decreases significantly, proving the effectiveness of the ASTR scheme.

In Figures 5–10, the comparison of data amount provides the comparison of data amount of the whole network. Two schemes adopt different parameters and have different data amount reaching the sink, so they are not compared under the same conditions. To effectively compare the performance of the two schemes, we compare the data amount loaded by each node under the same parameters when the data packet is only one hop from the sink.

In ASTR scheme, assume $\mathscr{M} = 1$, $\mathscr{N}$ abstracts are sent, and the success rate of transmitting message for each hop is p. In this case, the probability for at least one message to reach the sink is $\sigma = 1 - (1-p)^{(1+\mathscr{N})}$ in ASTR scheme, where p represents the probability for the data packet to reach the sink. If the data packet is received, no resending is needed and the process will end; if no message is received, the sink will not know the sending of data packet and the process will also end. Otherwise, if the received message contains no data packet, the data packet will be resent and the number of packets sent for each time is also $(1 + \mathscr{N})$. The data packet will be sent for the second time when the first message is received but will contain no data packet and the probability of this case is $(1-p)\sigma$. In the second sending, the reach probability of data packet is still p and the probability of sending and receiving the data packet is $(1-p)\sigma p$. Similarly, the data packet will be sent for the third attempt when the messages are successfully sent in the preceding two attempts but contain no data packet and the probability of this case is $(1-p)^2\sigma^2$. In the third sending, the reach probability of data packet is p and the probability of successively sending the message and receiving the data packet in the third attempt is $(1-p)^2\sigma^2 p$. Therefore, the probability of the kth sending attempt is $(1-p)^{k-1}\sigma^{k-1}p$.

If the upper limit of number of sending attempts is k, the message can only be sent for k times in maximum. If the data packet has not been successfully sent at the kth attempt, it will not send again. If the data packet is successfully sent after less than k attempts, it will not be sent again either. In this case, the total success rate of sending of data packets is as follows:

$$\begin{aligned}\beta_{1+\mathscr{N}} &= p + (1-p)\sigma p + (1-p)^2\sigma^2 p + \cdots \\ &\quad + (1-p)^{k-1}\sigma^{k-1}p = p\sum_{k=1}^{a}\left((1-p)\sigma\right)^{k-1}, \\ \sigma &= 1 - (1-p)^{(1+\mathscr{N})}.\end{aligned} \tag{16}$$

If the application requires a probability of q for data to successively reach the sink, k should abide by the following formula:

$$\beta_{1+\mathscr{N}} = p\sum_{k=1}^{a}\left((1-p)\sigma\right)^{k-1} > q. \tag{17}$$

Theorem 3. *In ASTR scheme, let $\mathscr{M} = 1$, $\mathscr{N}$ abstracts are sent, the success rate of transmission of message of each hop is p, the node is on hop from the sink and the upper limit of the number of sending attempts is k, and the expected number of sending attempts can be calculated as follows:*

$$\begin{aligned}w = &\frac{1}{\left[1-\sigma(1-p)\right] * \sum_{k=1}^{a}\left((1-p)\sigma\right)^{k-1}} \\ &* \left[\frac{1-\sigma^k(1-p)^k}{1-\sigma(1-p)} - k\sigma^k(1-p)^k\right].\end{aligned} \tag{18}$$

Proof. The probability of successively sending the data packet at one time is p and the number of sending attempts is 1; the first sending of data packet fails, the probability of successively sending at the second attempt is $(1-p)\sigma p$, and the number of sending attempts is 2; if the sending of data packet fails for two times, the probability of failure to send the data packet is $(1-p)^2\sigma^2 p$ and the number of sending attempts is 3.

The expected number of sending attempts is

$$\begin{aligned}w &= 1\times\frac{p}{\beta_{1+\mathscr{N}}} + 2\times\frac{(1-p)\sigma p}{\beta_{1+\mathscr{N}}} + 3\times\frac{(1-p)^2\sigma^2 p}{\beta_{1+\mathscr{N}}} \\ &\quad + \cdots + k\times\frac{(1-p)^{k-1}\sigma^{k-1}p}{\beta_{1+\mathscr{N}}} \\ &= \frac{p}{\beta_{1+\mathscr{N}} * \left[1-\sigma(1-p)\right]} \\ &\quad * \left[\frac{1-\sigma^k(1-p)^k}{1-\sigma(1-p)} - k\sigma^k(1-p)^k\right] \\ &= \frac{1}{\left[1-\sigma(1-p)\right] * \sum_{k=1}^{a}\left((1-p)\sigma\right)^{k-1}} \\ &\quad * \left[\frac{1-\sigma^k(1-p)^k}{1-\sigma(1-p)} - k\sigma^k(1-p)^k\right].\end{aligned} \tag{19}$$

□

Theorem 4. *In MPR scheme, M data packets are sent, the success rate of each hop of message transmission is p and the node is one hop from the sink, the actual number of data packets reaching the sink is*

$$G = \sum_{1}^{M} i * C_M^i p^i (1-p)^{M-i} = Mp. \tag{20}$$

Assume the length of the data packet is l_m, and the total data amount is

$$D_m = G * l_m. \tag{21}$$

Proof. Only data packets are sent, so the probability for at least data packet to reach the sink is $1 - (1-p)^M$. In M attempts, the probability for only one data packet to reach the sink is $C_M^1 p * (1-p)^{M-1}$, so the actual number of data packets sent when a data packet reaches the sink is $1C_M^1 p * (1-p)^{M-1}$; the probability for only two data packets to reach

the sink is as follows: $C_M^2 p^2 * (1-p)^{M-2}$, so the actual number of packets sent when two data packets reach the sink is as follows: $2C_M^2 p^2 * (1-p)^{M-2}$. Similarly, the actual number of data packets sent when M data packets reach the sink is as follows: $MC_M^M p^M * (1-p)^{M-M} = Mp^M$. Therefore, when the data packet is sent for M times, the average number of data packets reaching the sink is $G = \sum_1^M i * C_M^i p^i (1-p)^{M-i}$. The length of each data packet is l_m, so the total data amount is $D_m = G * l_m$. □

In MPR scheme, we take the method of routing M times and sending a packet each time. If we change it a little bit, we will get another scheme. In another scheme, we take the method of routing M times but each time c packets are sent. We call this scheme CMPR scheme. Deduction 5 is obtained by Theorem 4.

Deduction 5. In CMPR scheme, each time c packets are sent, cM data packets are sent in total, the success rate of each hop of message transmission is p, and the node is one hop from the sink, the actual number of data packets reaching the sink is

$$F = \sum_1^{cM} i * C_{cM}^i p^i (1-p)^{cM-i} = cMp. \tag{22}$$

Assume the length of the data packet is l_m, and the total data amount is

$$d_m = F * l_m. \tag{23}$$

Proof. Only data packets are sent, so the probability for at least data packet to reach the sink is $1-(1-p)^{cM}$. In M attempts, the probability for only one data packet to reach the sink is $C_{cM}^1 p * (1-p)^{cM-1}$, so the actual number of data packets sent when a data packet reaches the sink is $1C_{cM}^1 p * (1-p)^{cM-1}$; the probability for only two data packets to reach the sink is as follows: $C_{cM}^2 p^2 * (1-p)^{cM-2}$, so the actual number of packets sent when two data packets reach the sink is as follows: $2C_{cM}^2 p^2 * (1-p)^{cM-2}$. Similarly, the actual number of data packets sent when cM data packets reach the sink is as follows: $cMC_{cM}^{cM} p^{cM} * (1-p)^0 = cMp^{cM}$. Therefore, when the data packet is sent for M times, the average number of data packets reaching the sink is $F = \sum_1^{cM} i * C_{cM}^i p^i (1-p)^{cM-i}$. The length of each data packet is l_m, so the total data amount is $d_m = F * l_m$. □

Theorem 6. *In ASTR scheme, let $\mathcal{M} = 1$, $\mathcal{N}$ abstracts are sent, the success rate of each hop of message transmission is p, and the node is one hop from the sink. If the maximum number of resending attempts is km the actual number of data packets reaching the sink is*

$$F = (1 + \mathcal{N}) * w. \tag{24}$$

Assume the length of each data packet is l_m and the length of each abstract is l_n, and the total data amount is

$$D_n = wl_m + w\mathcal{N}l_n. \tag{25}$$

Proof. In ASTR scheme, if 1 data packet and $\mathcal{N}$ abstracts are sent every time, the probability for at least one message to reach the sink is $\sigma = 1-(1-p)^{(1+\mathcal{N})}$. It the data packet needs not to be sent again, the process will end; if no message is received, the sink will not know the sending of data packet and the process will also end; otherwise, if the message is received but contains no data packets, the message will be sent again and a $(1+\mathcal{N})$ will be sent each time.

Therefore, the actual number of messages reaching the sink is $(1+\mathcal{N}) * p$ when the data packet is successively sent at one time and $2(1+\mathcal{N}) * (1-p)\sigma p$ when the data packet is successively sent after two attempts. Similarly, the actual number of messages reaching the sink is $k(1+\mathcal{N}) * (1-p)^{k-1}\sigma^{k-1} p$ when the data packet is successively sent after k attempts and the average number of messages actually reaching the sink is as follows: $F = (1+\mathcal{N}) * w$ (w is the expected sending attempts). Assume the length of each data packet is l_m and the length of each abstract is l_n, and the total data amount is $D_n = wl_m + w\mathcal{N}l_n$. □

Theorem 7. *If the success rate of each hop of message transmission is p and the node is one hop from the sink, the effective data amount received by the sink is*

$$\mathbb{R} = l_m * p. \tag{26}$$

Proof. The effective data amount is defined as the data amount of one data packet reaching the sink. There is a packet loss ratio, so the effective data amount is as follows: $\mathbb{R} = l_m * p$. □

Theorem 4 and Deduction 5 provide the actual data amount reaching the sink in MPR scheme and CMPR scheme. In the CMPR scheme, we let it send two packets at a time now. With different successful rate of transmission of each hop, Figures 11 and 12 can be gotten.

Figures 11 and 12 show that the successful reach rate of MPR scheme is very close to the successful reach rate of CMPR scheme, but the data amount of reaching the sink of MPR scheme is only half of the CMPR scheme. Therefore, the performance of MPR scheme is better under comprehensive consideration.

Figures 11 and 12 show the comparison between CMPR scheme and MPR scheme. The comparison of CMPR scheme and ASTR scheme is shown in Table 4.

As can be obtained from Table 4, the CMPR scheme has improved the success rate of reaching the sink, but the limitation is that too many data packets are sent, so that the sink receives a lot of redundant data. Considering the comprehensive network condition, we do not think that CMPR scheme is a better scheme, so we will only compare MPR scheme and ASTR scheme.

Theorems 4 and 6 provide the actual data amount reaching the sink in MPR scheme and ASTR scheme. Figure 11 compares the data amount and effective data amount reaching the sink in two schemes when the probability for at least one data packet to reach the sink is 0.99, 0.999, 0.9999, and 0.99999.

Figure 13 clearly shows that as the application has higher requirement on the probability for data to reach the sink

Table 4: The comparison between CMPR scheme and ASTR scheme when $c = 2$ and $p = 0.80$.

CMPR scheme		ASTR scheme	
The successful reach rate	Data amount of reaching the sink	The successful reach rate	Data amount of reaching the sink
0.9984	320	0.96	131.5241
0.999936	480	0.992	148.4281
0.999997	640	0.9984	162.1123
1	800	0.99968	174.9072

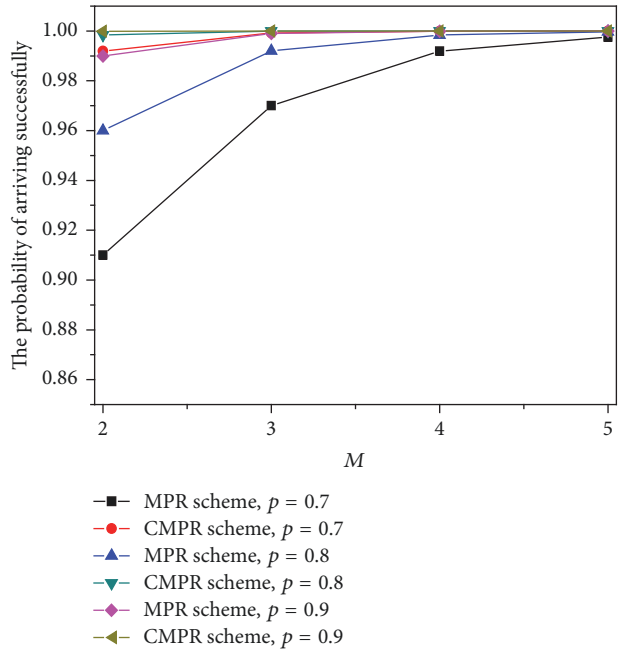

Figure 11: The successful reach rate of MPR scheme and CMPR scheme with different value of p when the number of sending times is 2, 3, 4, and 5, respectively ($c = 2$).

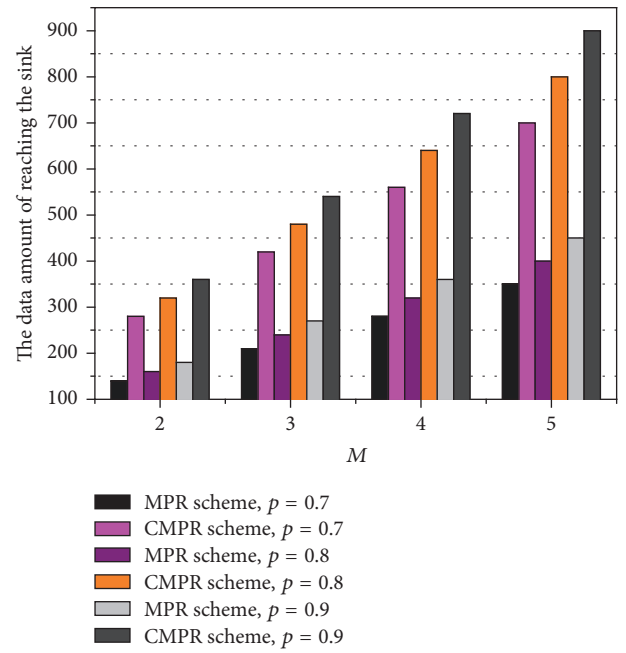

Figure 12: The data amount of reaching the sink successfully of MPR scheme and CMPR scheme with different value of p when the number of sending times is 2, 3, 4, and 5 ($c = 2$).

(increased from $q = 0.99$ to $q = 0.99999$), the data amount rises rapidly in MPR scheme. With the number of data packets sent plus one each time and assuming the length of a data packet is 100, the actual data amount reaching the sink will increase by $100 * p = 90$ each time; however, in ASTR scheme, the rise of data amount is very gentle. The number of abstracts increases by 1 each time and the number of expected sending attempts w changes insignificantly as $\mathcal{N}$ increases. Assuming the length of abstract is 10, the incremental data amount calculated following the formula in Theorem 6 is very small. The effective data amount calculated following Theorem 7 is 99, 99.9, 99.99, and 99.999, respectively.

Theorem 7 proposes the concept of effective data. Currently, we can further propose the concept of redundant data amount which refers to part of the data amount actually reaching the sink and in excess of the effective data amount. Therefore, the redundant data amount reaching the sink in two schemes can be, respectively, calculated according to Figure 13 as follows: deducting the effective data amount from the actual data amount reaching the sink to get Figure 14 and dividing the redundant data amount by actual data amount

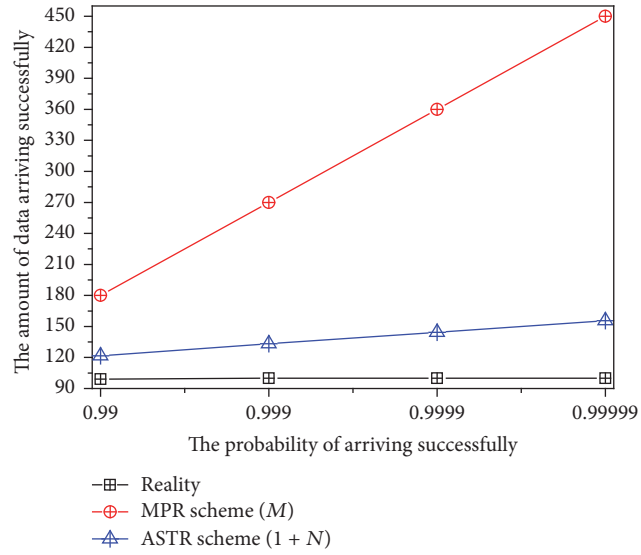

Figure 13: Actual data amount reaching the sink and effective data amount in two schemes when $p = 0.90$.

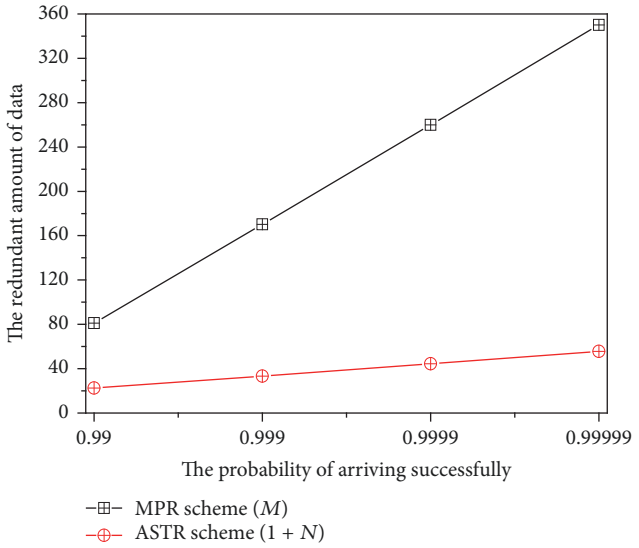

Figure 14: Comparison of redundant data amount between two schemes when $p = 0.90$.

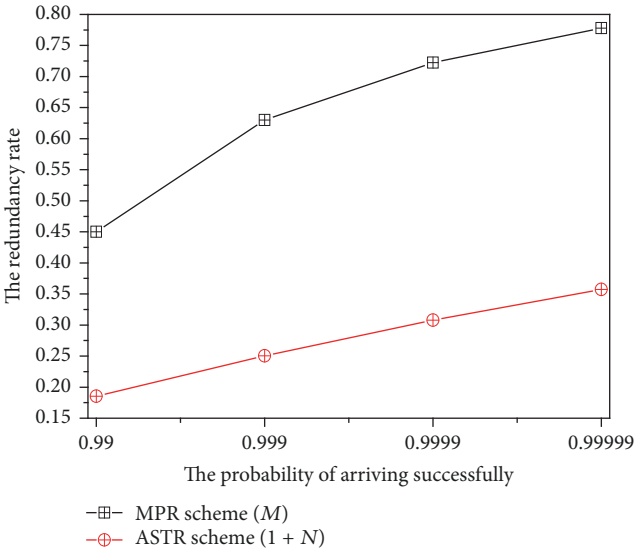

Figure 15: Comparison of data redundancy rate between two schemes when $p = 0.90$.

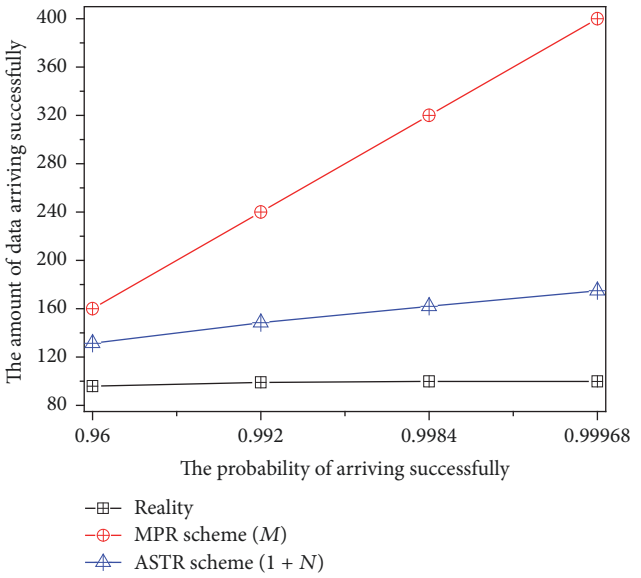

Figure 16: Actual data amount reaching the sink and effective data amount in two schemes when $p = 0.80$.

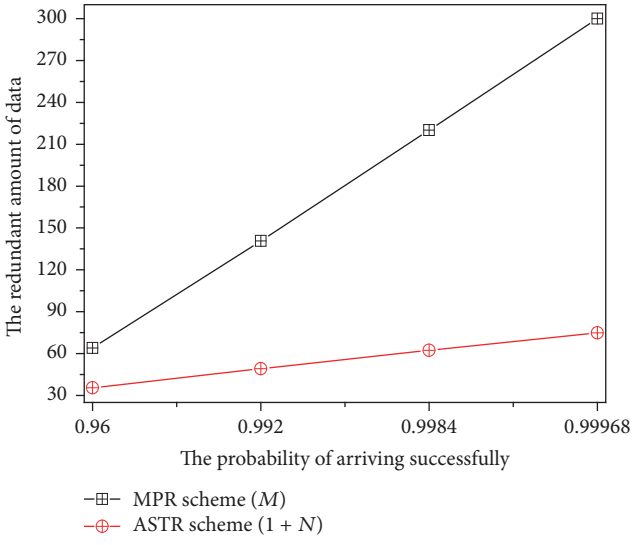

Figure 17: Comparison of redundant data amount between two schemes when $p = 0.80$.

reaching the sink to get Figure 15. Apparently, the redundant data amount and redundancy rate of the ASTR scheme are smaller than those of the MPR scheme because the ASTR scheme does not repetitively send very long data packets. Therefore, the ASTR scheme can both reduce the energy consumption of the network and extend the network lifetime.

Figures 16, 17, and 18 show the experiment results when $p = 0.80$. According to the figures, the change of data amount is similar to the change when $p = 0.90$, which means the ASTR scheme has less redundant data amount and better performance than the MPR scheme regardless of the success rate of each hop of the transmission.

Figures 19 and 20 compare the probability for at least one data packet to reach the sink in MPR scheme and ASTR scheme when the actual data amount reaching the sink is the same. The experiment aims to visibly reflect the following: when consuming the same network resources, the ASTR scheme can achieve a higher probability for the sinker to receive at least one data packet compared with the MPR scheme. Figure 19 shows that when the data amount rises from 140 to 210 in ASTR scheme, the probability for at least one data packet to reach the sink increases rapidly and when the data amount reaches 210, the probability gets close to 1. To determine the causes of this condition more visibly, Figures 21 and 22 explain them clearly.

It is clear that the radian of cambered surface in Figure 21 is small, but the radian of that in Figure 22 is large. The

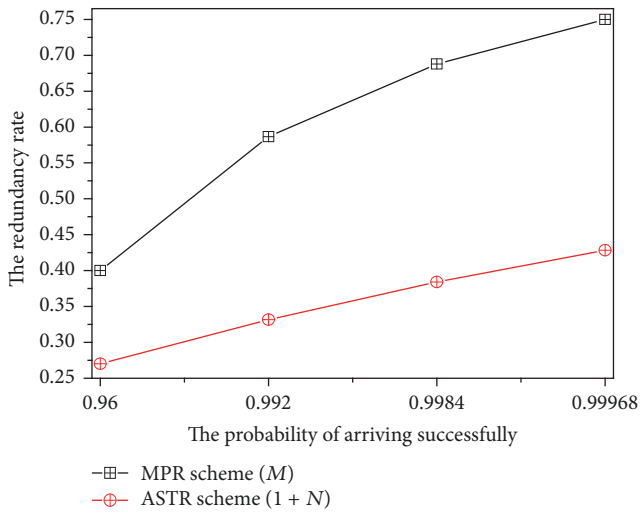

Figure 18: Comparison of data redundancy rate between two schemes when $p = 0.80$.

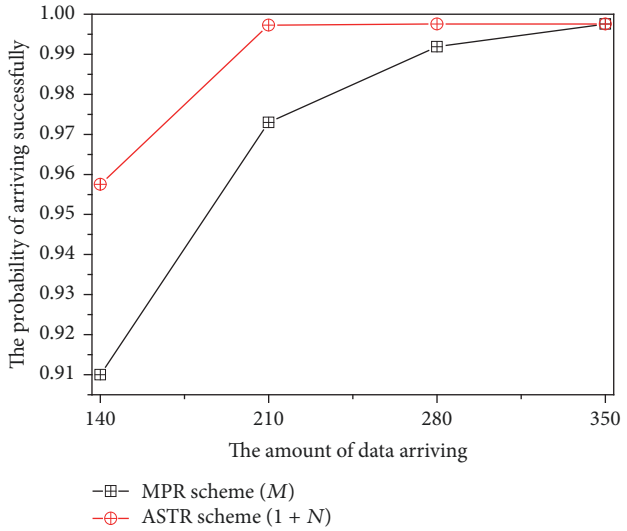

Figure 19: Comparison of success rate of data transmission in two schemes when the actual data amount reaching the sink is the same.

middle cambered surface has a large radian. When observing along the x-axis in two figures, we see the maximum number of sending attempts k ranges from 1 to 7, the data amount increases from 50 to 150, and the probability increases from 0 to 0.90, which demonstrates that, in ASTR scheme, increasing the number of sending attempts has significant effect on improving the probability while the data amount only increases a little. When the data amount increases to around 200, the probability has reached the saturated level. Therefore, we draw the conclusion: adopting the ASTR scheme can quickly achieve the purpose of data integrity with only small increment in data amount.

5.2. The Calculation and Comparison of Success Rate of Data Transmission. In this paper, the security of the network is not

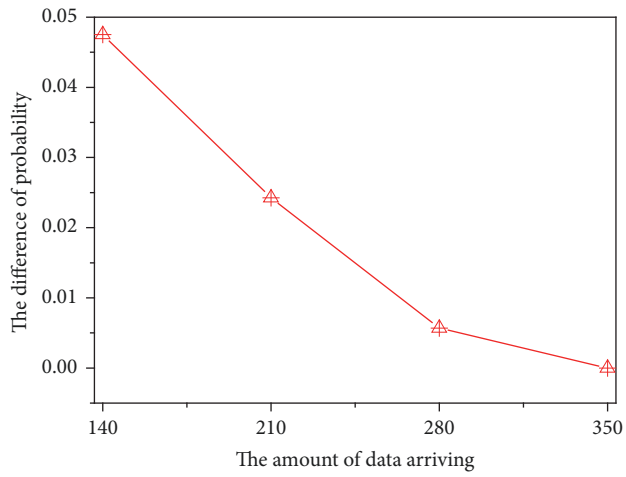

Figure 20: Difference between the success rate of data transmission in ASTR scheme and MPR scheme when the actual data amount reaching the sink is the same.

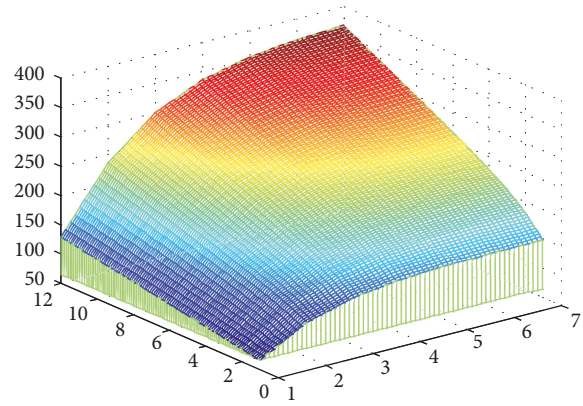

Figure 21: Relationship between k, $\mathcal{N}$ and data amount ($\mathcal{M} = 1$) in ASTR scheme.

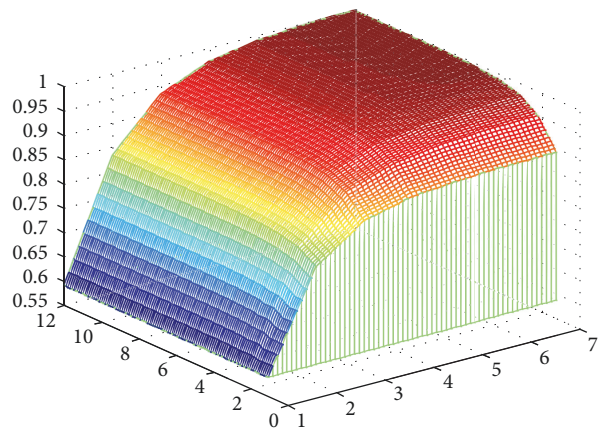

Figure 22: Relationship between $k, \mathcal{N}$ and success rate of data transmission ($\mathcal{M} = 1$) in ASTR scheme.

only related to data packets but also related to abstract. In this section, we analyze the overall probability for the sink to receive the data packet and abstract to verify that the data security can be improved under the ASTR scheme.

Theorem 8 and Deduction 9 show, respectively, the success rate for the data packet and abstract to reach the sink and the overall probability for the sink to receive the data packet and abstract when the success rate of the transmission of each hop is p and $\mathscr{M}$ data packets and $\mathscr{N}$ abstracts are sent at one time.

Theorem 8. *Assume the success rate of each hop is p when each data packet is sent to the base station and the node sending the data packet is h hops from the base station. The number of hops is randomly selected from* $\{1, \mathscr{d}\}$ *for horizontal routing of the abstract, so the expected length of horizontal routing is* $\mathscr{d}/2$ *and its number of hops from the base station is h, the same as the data packet. Therefore, the average number of total hops sent by the abstract to the base station is* $h + \mathscr{d}/2$. *Assume the number of data packets and abstracts is* $\mathscr{M}$, $\mathscr{N}$, *respectively.* $\mathscr{A}_1$ *means the sink fails to receive the data packet,* $\mathscr{A}_2$ *means the sink receives at least one data packet,* $\mathscr{A}_3$ *means the sink fails to receive the abstract, and* $\mathscr{A}_4$ *means the sink receives at least one abstract.* $f_1(\mathscr{M}, h)$ *and* $f_2(\mathscr{M}, h)$, *respectively, represent the probability of case* $\mathscr{A}_1$*and case* $\mathscr{A}_2$ *and* $g_1(\mathscr{N}, h, \mathscr{d})$ *and* $g_2(\mathscr{N}, h, \mathscr{d})$, *respectively, represent the probability of case* $\mathscr{A}_3$ *and case* $\mathscr{A}_4$:

$$
\begin{aligned}
f_1(\mathscr{M}, h) &= \left(1 - p^h\right)^{\mathscr{M}}, \quad \text{case } \mathscr{A}_1,\\
f_2(\mathscr{M}, h) &= 1 - \left(1 - p^h\right)^{\mathscr{M}}, \quad \text{case } \mathscr{A}_2,\\
g_1(\mathscr{N}, h, \mathscr{d}) &= \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}, \quad \text{case } \mathscr{A}_3,\\
g_2(\mathscr{N}, h, \mathscr{d}) &= 1 - \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}, \quad \text{case } \mathscr{A}_4.
\end{aligned}
\tag{27}
$$

Proof. Each data packet needs h hops when routed to the sink, so the probability for each data packet to successively reach the sink is p^h and the probability for each data packet to fail in reaching the sink is $1 - p^h$. Therefore, the probability for all of $\mathscr{M}$ data packets to fail in reaching the sink is $(1 - p^h)^{\mathscr{M}}$ and the probability for the sink to receive at least one data packet is $1 - (1 - p^h)^{\mathscr{M}}$. The distance of horizontal routing of abstract is $\{1, \mathscr{d}\}$, so the expected length of horizontal routing is $\mathscr{d}/2$. The number of hops of the abstract from the base station is h, the same as the data packet, so the total expected length of sending one abstract to the base station is $h + \mathscr{d}/2$. Therefore, the probability of successively routing each abstract to the sink is $p^{h+\mathscr{d}/2}$ and the probability of failing to route each data to the sink is $1 - p^{h+\mathscr{d}/2}$. Similarly, the probability for all of $\mathscr{N}$ data packets to fail in reaching the sink is $(1 - p^{h+\mathscr{d}/2})^{\mathscr{N}}$ and the probability for the sink to receive at least one data packet is $1 - (1 - p^{h+\mathscr{d}/2})^{\mathscr{N}}$. □

The overall probability for the sink to receive the data packet and abstract can be calculated based on the success rate of transmission of data packet and abstract, so Deduction 9 can be obtained from Theorem 8.

Deduction 9. $\mathscr{B}_1$ means the sink receives neither data packet nor abstract, $\mathscr{B}_2$ means the sink receives no data packet but receives at least one abstract, $\mathscr{B}_3$ means the sink receives no abstract but receives at least one data packet, and $\mathscr{B}_4$ means the sink receives at least one data packet and at least one abstract. $\varphi_1(f_1, g_1), \varphi_2(f_1, g_2), \varphi_3(f_2, g_1)$, and $\varphi_4(f_2, g_2)$, respectively, represent the probability of case $\mathscr{B}_1$, case $\mathscr{B}_2$, case $\mathscr{B}_3$, and case $\mathscr{B}_4$:

$$
\begin{aligned}
\varphi_1(f_1, g_1) &= \left(1 - p^h\right)^{\mathscr{M}} * \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}, \quad \text{case } \mathscr{B}_1,\\
\varphi_2(f_1, g_2) &= \left(1 - p^h\right)^{\mathscr{M}} * \left[1 - \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}\right], \quad \text{case } \mathscr{B}_2,\\
\varphi_3(f_2, g_1) &= \left[1 - \left(1 - p^h\right)^{\mathscr{M}}\right] * \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}, \quad \text{case } \mathscr{B}_3,\\
\varphi_4(f_2, g_2) &= \left[1 - \left(1 - p^h\right)^{\mathscr{M}}\right] * \left[1 - \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}\right], \quad \text{case } \mathscr{B}_4.
\end{aligned}
\tag{28}
$$

Proof. According to Theorem 8, whether the base station receives at least one data packet is independent of whether it receives at least one abstract, so the combined probability is the product of two probabilities; that is, $\varphi_{i,j}(f_i, g_j) = f_i(\mathscr{M}, h) * g_j(\mathscr{N}, h, \mathscr{d})$. □

Deduction 9 provides the occurrence probability of the cases in security analysis. The first case is that the sink receives neither data packet nor abstract. If we want to limit the probability of this case to $< u$, Deduction 10 can be obtained based on Deduction 9.

Deduction 10. Assume the success rate of each hop when each packet is sent to the sink, the number of hops for sending one abstract to the base station is $h + \mathscr{d}/2$, the number of data packets is $\mathscr{M}$, and the number of abstracts is $\mathscr{N}$. To limit the probability for the sink to receive neither data packet nor abstract smaller than u, the values of $\mathscr{M}$, $\mathscr{N}$ shall meet the following conditions:

$$
\begin{aligned}
\mathscr{M} &> \log_{1-p^h} \frac{u}{\left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}}, \quad \text{when } \mathscr{N} \text{ is a fixed value,}\\
\mathscr{N} &> \log_{1-p^{h+\mathscr{d}/2}} \frac{u}{\left(1 - p^h\right)^{\mathscr{M}}}, \quad \text{when } \mathscr{M} \text{ is a fixed value.}
\end{aligned}
\tag{29}
$$

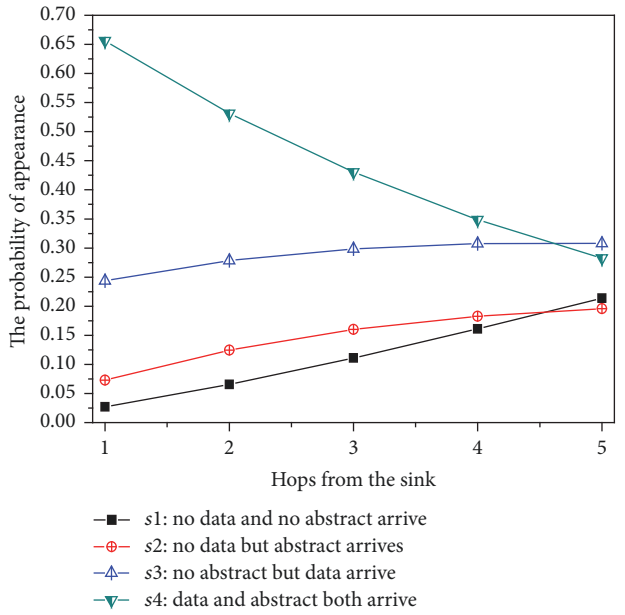

Figure 23: Probability of the four cases for nodes different hops from the sink ($p = 0.90$, $\mathscr{d} = 4$, $\mathscr{M} = 1$, and $\mathscr{N} = 1$).

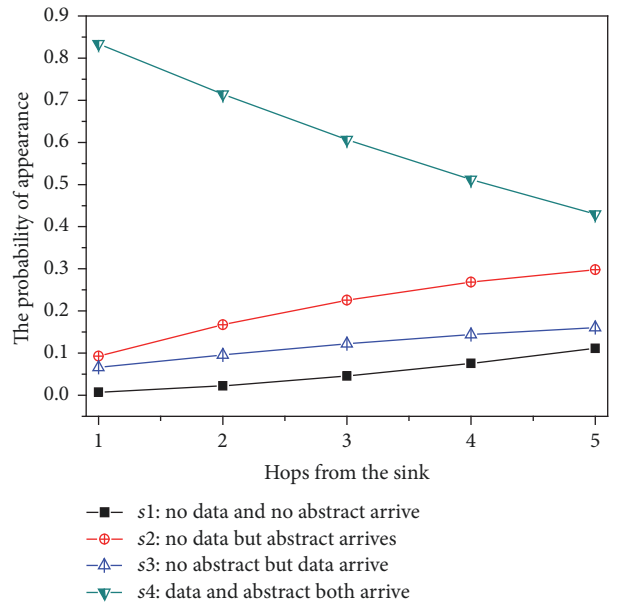

Figure 24: Probability of the four cases for nodes different hops from the sink ($p = 0.90$, $\mathscr{d} = 4$, $\mathscr{M} = 1$, and $\mathscr{N} = 2$).

Proof. According to Deduction 9, the probability for the sink to receive neither data packet nor abstract is

$$\begin{aligned}\varphi_1(f_1, g_1) &= f_1(\mathscr{M}, h) \times g_1(\mathscr{N}, h, \mathscr{d}) \\ &= \left(1 - p^h\right)^{\mathscr{M}} * \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}.\end{aligned} \tag{30}$$

Now assume the value of $\mathscr{N}$ has been determined, $(1 - p^{h+\mathscr{d}/2})^{\mathscr{N}}$ is a constant, and the following inequation is obtained:

$$\begin{aligned}&\varphi_1(f_1, g_1) < u, \\ \text{i.e. } &\left(1 - p^h\right)^{\mathscr{M}} * \left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}} < u,\end{aligned} \tag{31}$$

and take the logarithm of both sides and obtain the following inequation:

$$\mathscr{M} > \log_{1-p^h} \frac{u}{\left(1 - p^{h+\mathscr{d}/2}\right)^{\mathscr{N}}}. \tag{32}$$

Similarly, assume the value of $\mathscr{N}$ has been determined, $(1 - p^h)^{\mathscr{M}}$ is a constant, and the following inequation is obtained:

$$\mathscr{N} > \log_{1-p^{h+\mathscr{d}/2}} \frac{u}{\left(1 - p^h\right)^{\mathscr{M}}}. \tag{33}$$

□

Figures 23–26 provide the probability of the above four cases when the nodes different hops from the sink send the message to the sink with different values of parameters $\mathscr{d}$, $\mathscr{M}$, and $\mathscr{N}$. The comparison between Figures 23 and 24 shows that when $\mathscr{N}$ increases by 1, the probability for the sink to receive no data packet but at least one abstract, represented by the red line, will increase and the probability for the sink to receive at least one data packet but no abstract, represented by the blue line, will decrease. According to Theorem 8, when the number of abstracts is larger, the probability for the sink to receive at least one abstract will increase. In addition, we can see that the probability for the sink to receive both data packet and abstract, represented by the green line, increases significantly, which means increasing the number of abstracts can achieve to the purpose of enhancing the data security. The

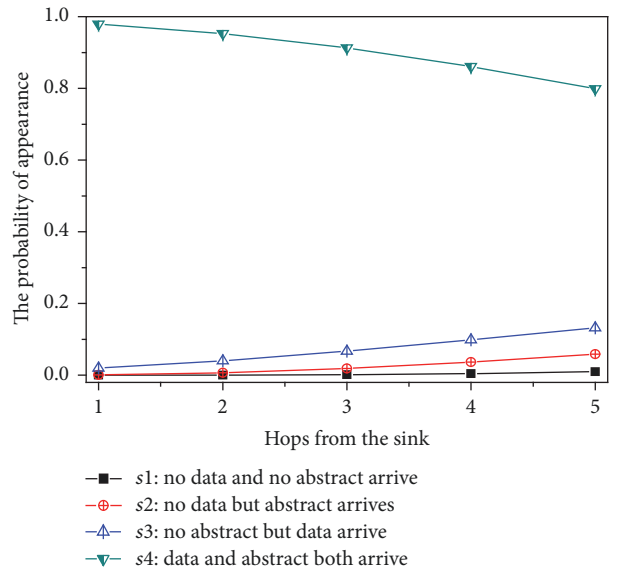

Figure 25: Probability of the four cases for nodes different hops from the sink ($p = 0.90$, $\mathscr{d} = 4$, $\mathscr{M} = 3$, and $\mathscr{N} = 3$).

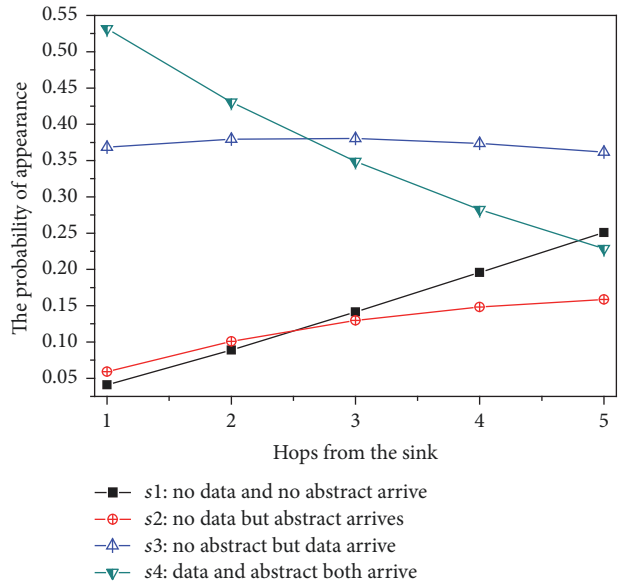

FIGURE 26: Probability of the four cases for nodes different hops from the sink ($p = 0.90$, $\mathscr{d} = 8$, $\mathscr{M} = 1$, and $\mathscr{N} = 1$).

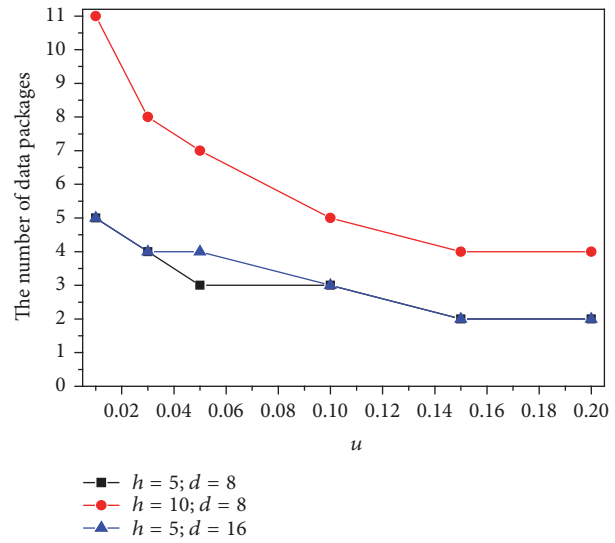

FIGURE 27: Minimum value of $\mathscr{M}$ to meet the conditions when $\mathscr{N}$ is fixed.

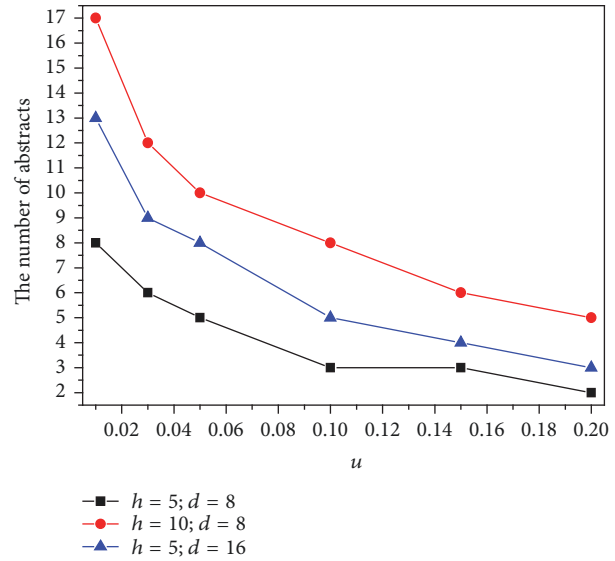

FIGURE 28: Minimum value of $\mathscr{N}$ to meet the conditions when $\mathscr{M}$ is fixed.

comparison between Figures 23 and 25 shows that when the number of data packets and abstracts is increased by the same value, the relative probability of this case remains unchanged and the network achieves a very favorable reliability. Figure 26 shows the case where the number of hops is increased for horizontal routing of the abstract. With the four figures taken together, when $\mathscr{M} : \mathscr{N} = 1 : 1$, the probability for the sink to receive the data packet but no abstract is larger than the probability for the sink to receive the abstract but no data packet; on the contrary, when $\mathscr{M} : \mathscr{N} = 1 : 2$, the probability for the sink to receive the abstract but no data packet is larger than the probability for the sink to receive the data packet but no abstract. Increasing $\mathscr{M}$, $\mathscr{N}$ improves the probability for the sink to receive both data packet and abstract.

Figures 27 and 28 are experiment figures prepared according to Deduction 10. The meaning of Deduction 10 is as follows: ensuring the probability for the sink to receive neither data packet nor abstract is lower than a very small value, fixing any one of $\mathscr{M}$, $\mathscr{N}$ and alternating another variable to meet the above conditions. Figure 27 shows the minimum value obtained to meet the requirement on $\mathscr{M}$ when u is any value within 0.01~0.20 and $\mathscr{N}$ is fixed. Figure 27 shows clearly that when the distance of horizontal routing is 8, the node 10 hops from the sink have to send more data packets than the node 5 hops from the sink to guarantee a reach rate of smaller than u. In addition, when the number of hops from the sink is 5, the node whose distance of horizontal routing is 16 has to send more data packets than the node whose distance of horizontal routing is 8 to guarantee a reach rate of smaller than u. Each hop of the horizontal routing also has a packet loss ratio and more hops will decrease the reach rate of abstract. The value of $\mathscr{N}$ has been fixed, so the overall reach probability of both data packet and abstract can only be improved by increasing the value of $\mathscr{M}$. Figure 28 shows the minimum value obtained to meet the requirement on $\mathscr{N}$ when the value of u is between 0.01 and 0.20 and the value of $\mathscr{M}$ is fixed.

5.3. Performance Comparison of the Whole Network with the Same Reliability. This section analyzes and compares the actual data amount reaching the sink in two schemes when the success rate of data transmission of all nodes in the whole network is q; that is, the whole network has the same reliability.

Theorem 11. *Assuming the number of hops of node n_x from the sink, the MPR scheme is adopted, the reliability of the whole network is q and the success rate of each hop is p, and the*

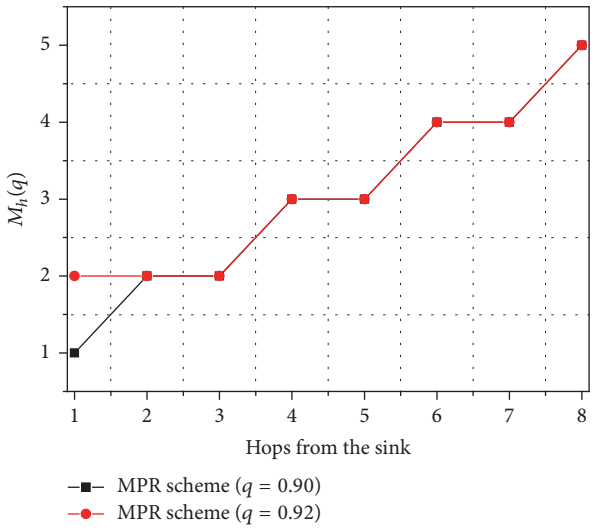

Figure 29: The number of required resending attempts of nodes different hops from the sink to ensure the reliability of the whole network reaches 0.90 and 0.92 in MPR scheme.

number of resending attempts of nodes different hops from the sink are as follows:

$$M_h(q) = \left\lceil \frac{\log(1-q)}{\log(1-p^h)} \right\rceil. \tag{34}$$

Proof. The success rate of each hop is p and the success rate that a message is successively sent to the sink is p^h.

If the message is sent by the node for $M_h(q)$ times, the probability for at least one data packet to reach the sink is

$$1-(1-p)^{M_h(q)}, \tag{35}$$

which is also the reliability of the network q, so the following equation is obtained:

$$q = 1-(1-p)^{M_h(q)}. \tag{36}$$

Take logarithm of both sides of the equation after transposition and obtain the following equation:

$$M_h(q) = \left\lceil \frac{\log(1-q)}{\log(1-p^h)} \right\rceil. \tag{37}$$

□

Figure 29 shows the required resending attempts of nodes different hops from the sink to ensure the reliability of the whole network reaches 0.90 and 0.92, respectively. Figure 29 shows clearly that more resending attempts are required for the node farther from the sink because the node is more hops from the sink and each sink has a packet loss ratio; Figure 30 shows the number of maximum resending attempts k of nodes different hops from the sink to ensure the reliability of the whole network reaches 0.90 and 0.92, respectively, in our ASTR scheme. The scheme uses two variables—namely, k and $\mathcal{N}$—and we fix one of them and treat the other one as a variable. For example, when $\mathcal{N} = 4$, the figure clearly shows that we can determine the upper limit of resending attempts of nodes different hops from the sink.

It should also be noted that, in ASTR scheme, when the reliability is improved, the value of $\mathcal{N}$ should be adjusted accordingly. For example, when the reliability is 0.95, the value of $\mathcal{N}$ should be increased to achieve the probability for the data to reach the sink required by the application q.

Theorem 12. *Assuming the distance from the node n_x to the sink is l, $l = hr + x$, the MPR scheme is adopted and the reliability of the whole network is q, the data amount sent and the data amount received by node n_x—$Q_{h,x}^{1,t}$ and $Q_{h,x}^{1,r}$—are as follows:*

$$\begin{aligned} Q_{h,x}^{1,t} &= M_h^h(q) + M_{h+1}^h(q)\cdot\frac{l+r}{l} + M_{h+2}^h(q) \\ &\quad \cdot\frac{l+2r}{l} + \cdots + M_{h+z}^h(q)\cdot\frac{l+zr}{l}, \\ Q_{h,x}^{1,r} &= Q_{h,x}^{1,t} - M_h^h(q), \\ M_h^h(q) &= \left\lceil \frac{\log(1-q)}{\log(1-p^h)} \right\rceil, \\ M_{h+i}^h(q) &= \left\lceil \frac{\log(1-q)}{\log(1-p^{h+i})} \right\rceil \cdot p^i. \end{aligned} \tag{38}$$

Proof. According to Theorem 11, the number of required resending attempts of nodes different hops from the sink $M_h(q)$ $(h \in \{1, z\})$ is as follows:

$$M_h(q) = \left\lceil \frac{\log(1-q)}{\log(1-p^h)} \right\rceil. \tag{39}$$

The data amount to be transmitted by node n_x can be calculated as follows. The node has a data packet which is h hops from the sink and should be sent to the sink, so the number of data packets that should be resent for one data packet is

$$M_h^h(q) = M_h(q) = \left\lceil \frac{\log(1-q)}{\log(1-p^h)} \right\rceil. \tag{40}$$

The number of required resending attempts of the data at $l + ir$ is as follows:

$$M_{h+i}(q) = \left\lceil \frac{\log(1-q)}{\log(1-p^{h+i})} \right\rceil. \tag{41}$$

The number of data packets to be transmitted when the data has reached node n_x is as follows:

$$M_{h+i}^h(q) = M_{h+i}(q)\cdot p^i. \tag{42}$$

There is $(l + ir)/l$ data at $l + ir$, so the total number of data packets from sources with different distance from node n_x and to be sent by node n_x can be calculated as follows:

$$\begin{aligned} Q_{h,x}^{1,t} &= M_h^h(q) + M_{h+1}^h(q)\cdot\frac{l+r}{l} + M_{h+2}^h(q)\cdot\frac{l+2r}{l} \\ &\quad + \cdots + M_{h+z}^h(q)\cdot\frac{l+zr}{l}. \end{aligned} \tag{43}$$

The received data amount is calculated by deducting the data amount of the node itself from the data sent by the node; that is, $Q_{h,x}^{1,r} = Q_{h,x}^{1,t} - M_h^h(q)$. □

Theorem 13. *Assuming the distance from the node n_x to the sink is l, $l = hr + x$, the ASTR scheme is adopted and the reliability of the whole network is q, the proportional coefficient of the size of abstract and data packets is μ, the data amount sent and the data amount received by node n_x—$S_{h,x}^{1,t}$ and $S_{h,x}^{1,r}$—are as follows:*

$$\begin{aligned}
S_{h,x}^{1,t} &= (1+\mu\mathcal{N})\cdot\left[w_h^h(q) + w_{h+1}^h(q)\cdot\frac{l+r}{l}\right.\\
&\quad \left. + w_{h+2}^h(q)\frac{l+2r}{l} + \cdots + w_{h+z}^h(q)\cdot\frac{l+zr}{l}\right],\\
S_{h,x}^{1,r} &= S_{h,x}^{1,t} - (1+\mu\mathcal{N})\cdot w_h^h(q),\\
w_h^h(q) &= \frac{1}{\left[1-\sigma_h(1-p^h)\right] * \sum_{k=1}^{a}\left((1-p^h)\sigma_h\right)^{k-1}}\\
&\quad * \left[\frac{1-{\sigma_h}^k\left(1-p^h\right)^k}{1-\sigma_h(1-p^h)} - k{\sigma_h}^k\left(1-p^h\right)^k\right],\\
&\qquad \text{where: } \sigma_h = 1-\left(1-p^h\right)^{(1+\mathcal{N})},
\end{aligned} \tag{44}$$

$$\begin{aligned}
&w_{h+i}^h(q)\\
&= \frac{p^i}{\left[1-\sigma_{h+i}(1-p^{h+i})\right] * \sum_{k=1}^{a}\left((1-p^{h+i})\sigma_{h+i}\right)^{k-1}}\\
&\quad * \left[\frac{1-{\sigma_{h+i}}^k\left(1-p^{h+i}\right)^k}{1-\sigma_{h+i}(1-p^{h+i})} - k{\sigma_{h+i}}^k\left(1-p^{h+i}\right)^k\right],\\
&\qquad \text{where: } \sigma_{h+i} = 1-\left(1-p^{h+i}\right)^{(1+\mathcal{N})}.
\end{aligned} \tag{45}$$

Proof. According to Theorem 3, the number of required resending attempts of the node that is h hops from the sink $w_h(q)$ ($h \in \{1, z\}$) is as follows:

$$\begin{aligned}
w_h(q) &= \frac{1}{\left[1-\sigma_h(1-p^h)\right] * \sum_{k=1}^{a}\left((1-p^h)\sigma_h\right)^{k-1}}\\
&\quad * \left[\frac{1-{\sigma_h}^k\left(1-p^h\right)^k}{1-\sigma_h(1-p^h)} - k{\sigma_h}^k\left(1-p^h\right)^k\right],\\
&\qquad \text{where } \sigma_h = 1-\left(1-p^h\right)^{(1+\mathcal{N})}.
\end{aligned} \tag{46}$$

The data amount to be transmitted by node n_x can be calculated as follows. The node has a data packet which is h hops from the sink and should be sent to the sink, so its expected data amount to be resent is as follows:

$$\begin{aligned}
w_h^h(q) &= w_h(q)\\
&= \frac{1}{\left[1-\sigma_h(1-p^h)\right] * \sum_{k=1}^{a}\left((1-p^h)\sigma_h\right)^{k-1}}\\
&\quad * \left[\frac{1-{\sigma_h}^k\left(1-p^h\right)^k}{1-\sigma_h(1-p^h)} - k{\sigma_h}^k\left(1-p^h\right)^k\right],\\
&\qquad \text{where } \sigma_h = 1-\left(1-p^h\right)^{(1+\mathcal{N})}.
\end{aligned} \tag{47}$$

For data packet at $l+ir$, the expected number of resending attempts at the beginning is as follows:

$$\begin{aligned}
&w_{h+i}(q)\\
&= \frac{1}{\left[1-\sigma_{h+i}(1-p^{h+i})\right] * \sum_{k=1}^{a}\left((1-p^{h+i})\sigma_{h+i}\right)^{k-1}}\\
&\quad * \left[\frac{1-{\sigma_{h+i}}^k\left(1-p^{h+i}\right)^k}{1-\sigma_{h+i}(1-p^{h+i})} - k{\sigma_{h+i}}^k\left(1-p^{h+i}\right)^k\right],\\
&\qquad \text{where } \sigma_{h+i} = 1-\left(1-p^{h+i}\right)^{(1+\mathcal{N})}.
\end{aligned} \tag{48}$$

The expected data amount to be transmitted when the data packet reaches node n_x is

$$w_{h+i}^h(q) = w_{h+i}(q)\cdot p^i. \tag{49}$$

There is $(l+ir)/l$ data at $l+ir$, each node sends 1 data packet and $\mathcal{N}$ abstracts each time and the proportional coefficient of the size of abstract and data packet is μ, so $1+\mu\mathcal{N}$ data is sent. The total expected data amount from sources with different distance from node n_x and to be sent by node n_x can be calculated as follows:

$$\begin{aligned}
S_{h,x}^{1,t} &= (1+\mu\mathcal{N})\cdot\left[w_h^h(q) + w_{h+1}^h(q)\cdot\frac{l+r}{l}\right.\\
&\quad \left. + w_{h+2}^h(q)\frac{l+2r}{l} + \cdots + w_{h+z}^h(q)\cdot\frac{l+zr}{l}\right].
\end{aligned} \tag{50}$$

The received data amount is calculated by deducting the data amount of the node itself from the data sent by the node; that is, $S_{h,x}^{1,r} = S_{h,x}^{1,t} - (1+\mu\mathcal{N})\cdot w_h^h(q)$. □

The data amount sent and the data amount received by each node in MPR scheme and ASTR scheme are calculated according to Theorems 7 and 8, respectively. The received data amount has no significant difference with the sent data amount, so we will directly compare the data amount sent by the node.

Figure 31 compares the data amount sent by each node in two schemes when the reliability of the whole network is 0.90 and 0.92. Figure 31 shows that the difference in data amount between the two schemes is more significant for a

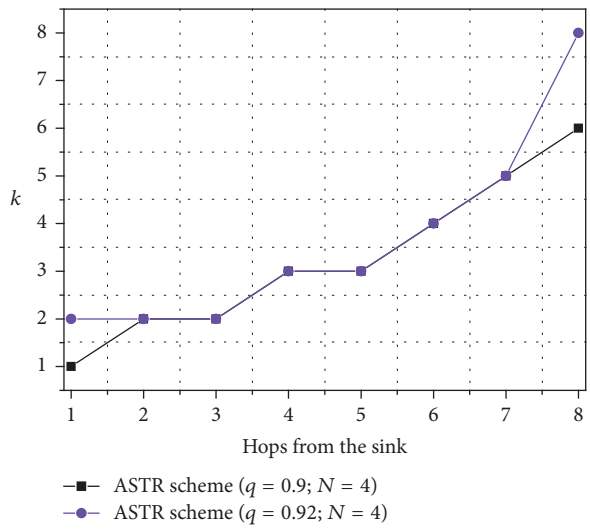

Figure 30: Upper limit of the number of required resending attempts of nodes different hops from the sink to ensure the reliability of the whole network reaches 0.90 and 0.92 in ASTR scheme ($\mathcal{N} = 4$).

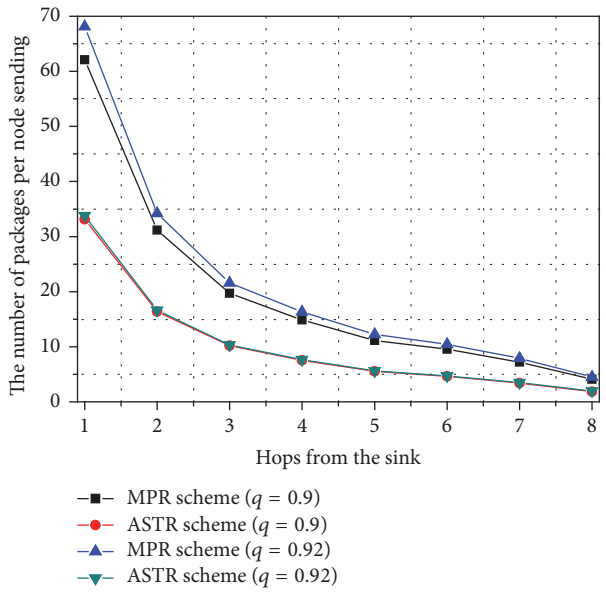

Figure 31: Data amount sent by each node in two different schemes when the reliability of the whole network is 0.90 and 0.92.

node closer to the sink. The data amount in MPR scheme is around twice the data amount in ASTR scheme, so the ASTR scheme can reduce the sent and received data amount while ensuring the network reliability, which both lowers the energy consumption and improves the network performance.

Figure 31 shows that when the network reliability is improved, the data amount loaded by each node will also increase. When the network reliability is improved, the data amount loaded by each node will also increase. The data amount in MPR scheme is around 2.3 times that in ASTR

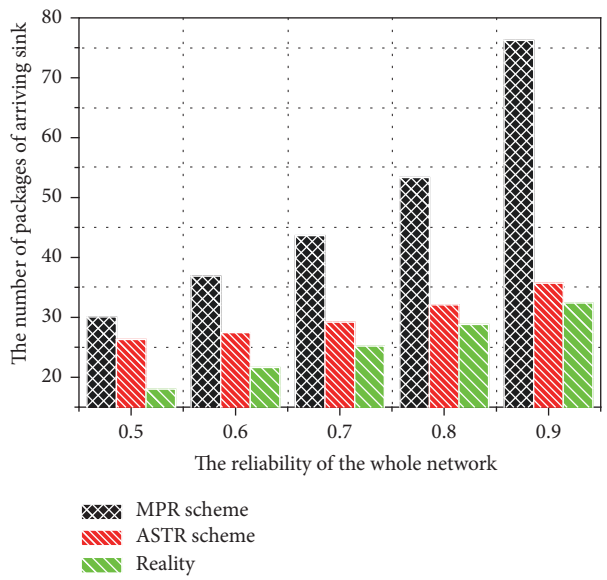

Figure 32: Data amount reaching the sink and effective data amount in two schemes under different network reliability.

scheme. Therefore, the improvement of the network reliability makes the advantage of ASTR scheme in performance more prominent.

Figures 32 and 34 compare the data amount reaching the sink in MPR scheme and ASTR scheme when the whole network has the same reliability. Figure 32 shows that the data amount reaching the sink in MPR scheme is more than that in ASTR scheme and the difference is increasingly significant as the reliability is improved. When $q = 0.90$, the data amount reaching the sink in MPR scheme is 53.95% more than that in ASTR scheme. This is because the node far from the sink has a larger number of resending attempts according to Theorem 12. The number of resending attempts is equivalent to the number of data packets and the data amount of one data packet is larger compared with the abstract, so many resent data packets are redundant. In our ASTR scheme, the data packet will be sent again when not received, so no data packet will be repetitively sent. The redundant data amount is completely produced by abstract which contains only small data amount, so the data amount reaching the sink and redundant data amount are both small. Figure 33 shows the percentage of redundant data packets in two schemes. The result is the same as the conclusion of preceding analysis: the percentage of redundant data is very high in MPR scheme and relatively low in ASTR scheme. Figure 33 also provides the information that as the reliability is improved, the redundancy of MPR scheme gradually increases. When the reliability is 0.90, the redundancy rate of MPR scheme even reaches 50.7%; on the contrary, the redundancy rate in ASTR scheme is gradually decreasing and reaches as low as 9% when the reliability is 0.90. So ASTR scheme can reduce the redundant data amount by 41.70%.

Figure 35 is prepared based on the above related data. It compares the guaranteed network reliability in two schemes when the same data amount reaches the sink. The figure

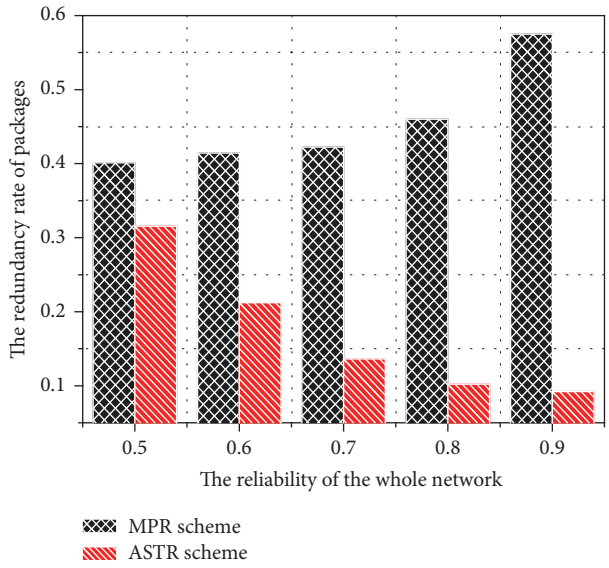

Figure 33: Percentage of redundant data packets in two schemes.

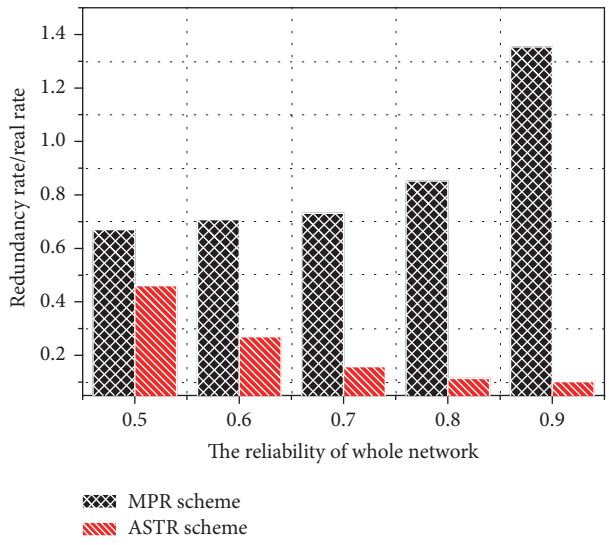

Figure 34: Percentage of redundant/effective data packets in two schemes.

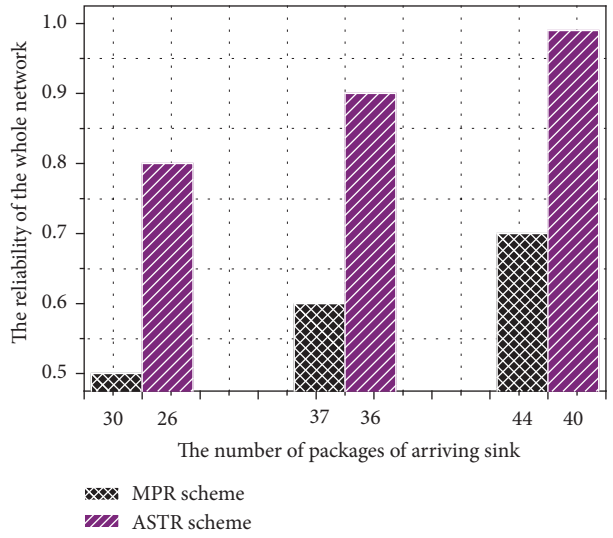

Figure 35: Comparison of the guaranteed network reliability in two schemes when the same data amount reaches the sink ($p = 0.90$).

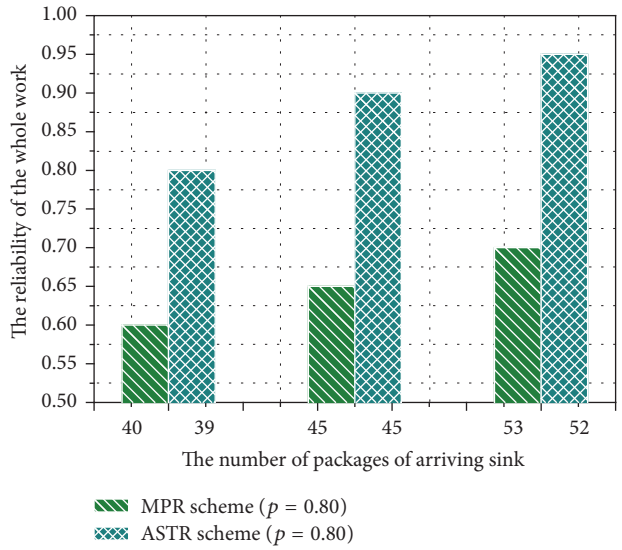

Figure 36: Comparison of the guaranteed network reliability in two schemes when the same data amount reaches the sink ($p = 0.80$).

shows that when the data amount in ASTR scheme is slightly smaller than that in MPR scheme, the reliability is still 30% higher, so sending less data amount in ASTR scheme can ensure a higher reliability.

From this section, all experiment figures are prepared based on the reach rate of each hop $p = 0.90$. To explore the effect of the value of p on this experiment, we get the following three results. Figure 36 is obtained by repeating the same experiment as above when $p = 0.80$ and shows the guaranteed network reliability in two schemes when the same data amount reaches the sink. Compared with Figure 35, the data amount in both MPR scheme and ASTR scheme is improved, but the network performance in ASTR scheme is better because the guaranteed network reliability is about 23.23% higher than that in MPR scheme.

The meaning of Figure 37 lies in comparing the guaranteed network reliability in two schemes when 40 data packets reach the sink with different value of p. The figure shows that when $p = 0.70$, the reliability of ASTR scheme is 5% higher than that of MPR scheme; when $p = 0.80$, the reliability of ASTR scheme increases by 20%; when $p = 0.90$, the reliability of ASTR scheme increases by 29%. It is concluded that when p is larger, the guaranteed network reliability of the ASTR scheme is much higher than that of the MPR scheme, which presents the advantages of ASTR scheme more prominently.

Figure 38 shows the following: generally speaking, when the value of p is larger, the increase of data amount is smaller in two schemes. This is because when the value of p is small in MPR scheme, the packet loss ratio of each hop of the node far

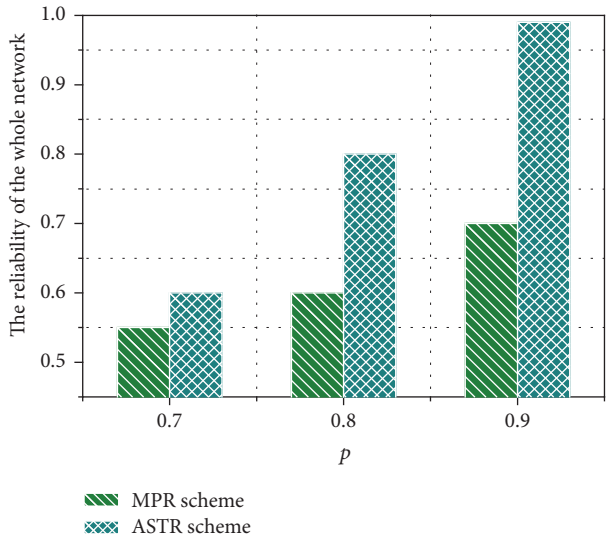

Figure 37: Comparison of guaranteed network reliability in two schemes when 40 data packets reach the sink with different value of p.

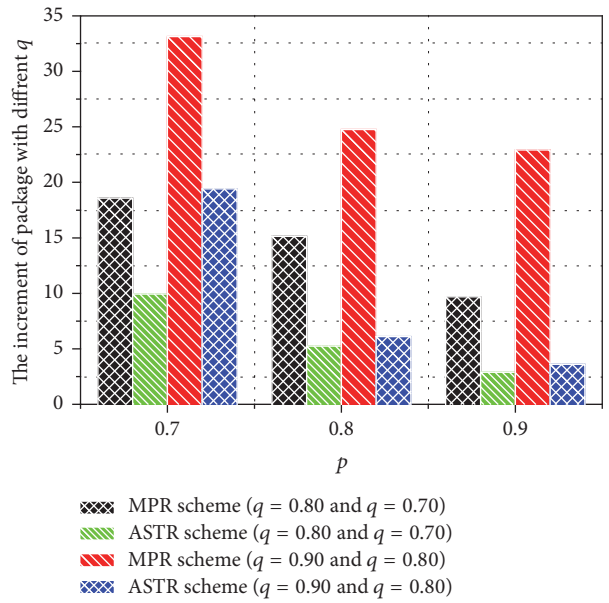

Figure 38: Comparison of increased data amount when q increases by 0.10 each time with different success rate of transmission of each hop.

from the sink will be accumulated and multiple data packets should be sent to ensure the reliability of the whole network is equal to q; for example, when the reliability of the whole network is 0.70: if $p = 0.70$, the node that is 8 hops from the sink should send 21 data packets; if $p = 0.80$, the node should send 7 data packets; if $p = 0.90$, the node should send only 5 data packets. When the number of data packets decreases from 21 to 7 and from 7 to 5, the increase of data amount will fall, so a larger p will cause a smaller increase of data amount in MPR scheme. Similarly, in ASTR scheme, the packet loss ratio of node far from the sink will be accumulated, so the number of abstracts sent and the upper limit of sending attempts should be greatly improved to ensure the reliability of the whole network is q. Next we will analyze the cases when $q = 0.80$ and $q = 0.70$. When q increases from 0.70 to 0.80, the comparison of increased data amount in MPR scheme and ASTR scheme clearly shows that the increased data amount in ASTR scheme is much smaller than that in MPR scheme. This is because when the same value of p, for example, 0.70, is used and the network reliability increases from 0.70 to 0.80, the number of data packets sent by each node in MPR scheme will increase greatly; the upper limit of the number of sending attempts will also increase greatly in ASTR scheme, but the calculated number of expected sending attempts w has no apparent change and the increased data amount of abstract is very small compared with the data amount of data packet. Similarly, when $q = 0.90$ and $q = 0.80$, the increase of data amount in ASTR scheme is smaller than that in MPR scheme.

The meaning of Figure 38 lies in the following: when $p = 0.90$, the required increase of data amount in MPR scheme is 7.5 times that in ASTR scheme to increase the network reliability from 0.80 to 0.90.

5.4. Performance Comparison in Trust Routing. This section analyzes the calculation of successful reach rate and compared the performance in two schemes when the trust routing is used; that is, the trust is improved.

Theorem 14. *Assuming the distance from the node to the sink is l, $l = hr + x$, in MPR scheme and ASTR scheme and the node is h hops from the sink, the success rate is p before the trust routing is adopted and, after the trust routing is used, the increased trust is ∂ and the probability for each message to reach the sink is $\mathbb{P}_h$, the successful reach rates of data packets in MPR scheme and ASTR scheme $\beta^b_{\mathcal{M},h}$ and $\beta^b_{(1+\mathcal{N}),h}$ are, respectively,*

$$\begin{aligned}
\beta^b_{\mathcal{M},h} &= 1 - \left(1 - \mathbb{P}_h\right)^{\mathcal{M}}, \\
\beta^b_{(1+\mathcal{N}),h} &= \mathbb{P}_h \sum_{k=1}^{a} \left(\left(1 - \mathbb{P}_h\right)\sigma^b_h\right)^{k-1}, \\
\mathbb{P}_h &= \left(p + \partial\right)^h, \\
\sigma^b_h &= 1 - \left(1 - \mathbb{P}_h\right)^{(1+\mathcal{N})}.
\end{aligned} \tag{51}$$

Proof. In MPR scheme, if a node is h hops from the sink, the probability for each message to reach the sink is $\mathfrak{P}_h = p^h$, so the original success rate of routing is $\beta^a_{\mathcal{M},h} = 1 - (1 - \mathfrak{P}_h)^{\mathcal{M}}$; after the trust routing is used, the probability for each message to reach the sink is $\mathbb{P}_h = (p+\partial)^h$, so the success rate of routing becomes $\beta^a_{\mathcal{M},h} = 1 - (1 - \mathbb{P}_h)^{\mathcal{M}}$.

In ASTR scheme, if a node is h hops from the sink, the probability for each message to reach the sink is $\mathfrak{P}_h = p^h$, so the original success rate of routing is

$$\begin{aligned}
\beta^a_{(1+\mathcal{N}),h} = \mathfrak{P}_h + \left(1 - \mathfrak{P}_h\right)\sigma^a_h\mathfrak{P}_h + \left(1 - \mathfrak{P}_h\right)^2\sigma^{a2}_h\mathfrak{P}_h \\
+ \cdots + \left(1 - \mathfrak{P}_h\right)^{k-1}\sigma^{ak-1}_h\mathfrak{P}_h
\end{aligned}$$

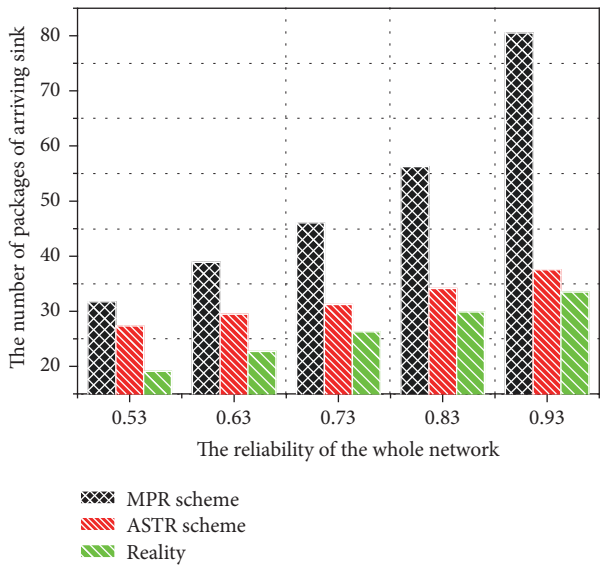

Figure 39: Data amount reaching the sink and effective data amount in two schemes with different network reliability after the trust is improved ($p = 0.91$).

$$= \mathfrak{P}_h \sum_{k=1}^{a} \left(\left(1 - \mathfrak{P}_h\right) \sigma_h^a \right)^{k-1},$$

$$\text{where: } \sigma_h^a = 1 - \left(1 - \mathfrak{P}_h\right)^{(1+\mathcal{N})}. \tag{52}$$

After the trust routing is used, the probability for each message to reach the sink is $\mathbb{P}_h = (p + \partial)^h$, so the success rate of routing becomes

$$\begin{aligned} \beta_{(1+\mathcal{N}),h}^b &= \mathbb{P}_h + \left(1 - \mathbb{P}_h\right) \sigma_h^b \mathbb{P}_h + \left(1 - \mathbb{P}_h\right)^2 \sigma_h^{b^2} \mathbb{P}_h \\ &\quad + \cdots + \left(1 - \mathbb{P}_h\right)^{k-1} \sigma_h^{b^{k-1}} \mathbb{P}_h \\ &= \mathbb{P}_h \sum_{k=1}^{a} \left(\left(1 - \mathbb{P}_h\right) \sigma_h^b \right)^{k-1}, \end{aligned} \tag{53}$$

$$\text{where: } \sigma_h^b = 1 - \left(1 - \mathbb{P}_h\right)^{(1+\mathcal{N})}.$$

□

Figures 39, 40, 41, and 42 are obtained in the same network as set in Figures 32–35 in the last section. After the trust routing is adopted, if p increases from 0.90 to 0.91, the trust routing method will improve the network reliability. The horizontal coordinate in Figure 39 shows that the reliability increases by 3%, which means the trust routing can improve the reliability of the whole network and thereby enhance the data security.

The meaning of Figure 43 lies in the following: when the increase of trust ∂ is 0.01, 0.02, or 0.03, we use the ASTR scheme to ensure the same data amount reaches the sink and compare the guaranteed network reliability in three cases. Figure 43 shows that a larger ∂ will result in a higher reliability.

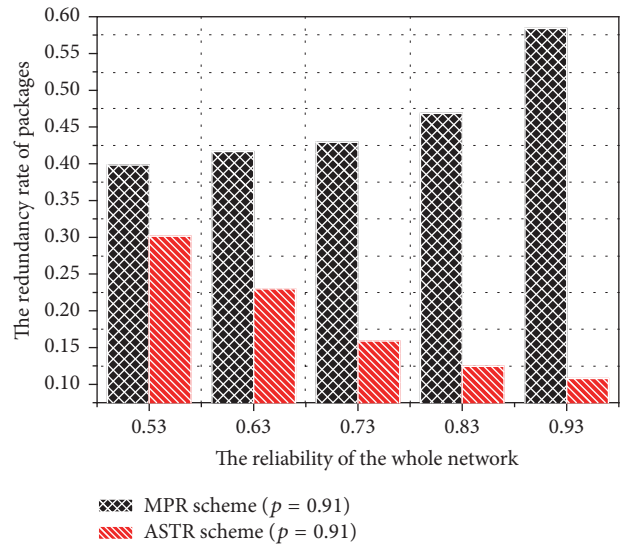

Figure 40: Redundancy rate in two schemes after the trust is improved.

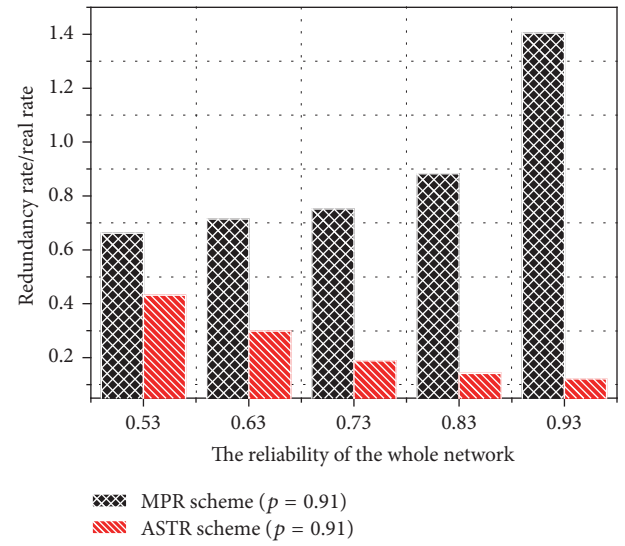

Figure 41: Redundancy rate/effective rate in two schemes after the trust is improved.

Figure 44 compares the improvement of network reliability when p changes to 0.91, 0.92, and 0.93 after the trust routing is adopted and when p is 0.90 before the trust routing is adopted in ASTR scheme if the same data amount reaches the sink. The figure clearly shows that, after the trust routing is adopted, the network reliability is improved and a larger increase of trust ∂ will improve the reliability more significantly. We will select the path with a higher reliability when sending data packets to improve the probability for the data packets to reach the sink and reduce the packet loss ratio. As the network operates, the packet loss ratio will be increasingly low.

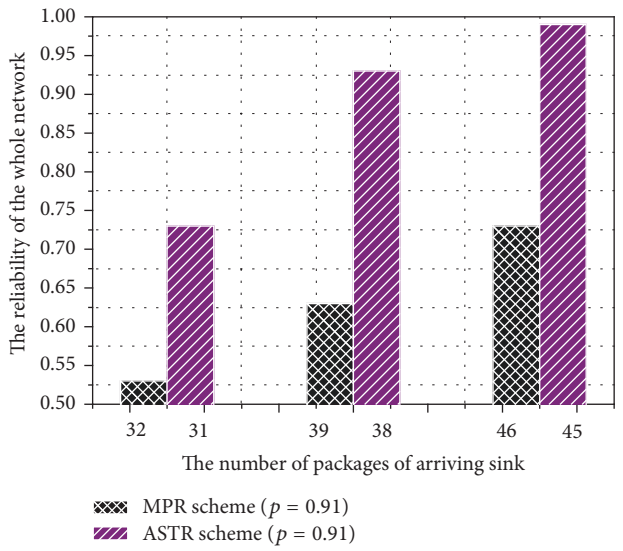

Figure 42: Guaranteed network reliability in two schemes when the same data amount reaches the sink after the trust is improved (p = 0.91).

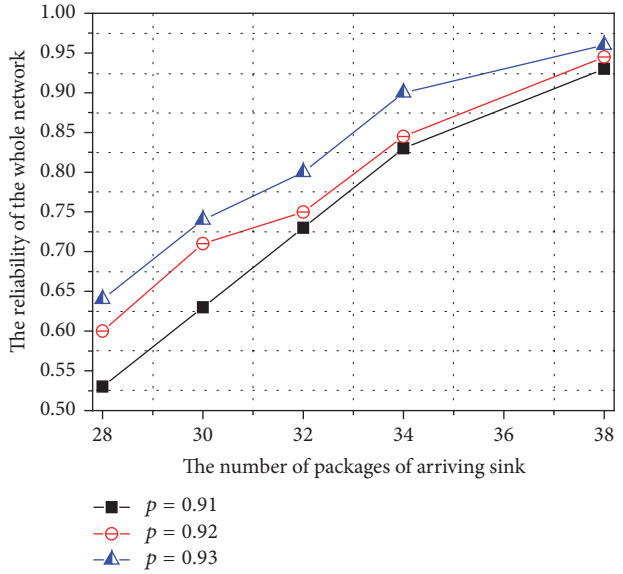

Figure 43: Guaranteed network reliability in ASTR scheme with different trust when the same data amount reaches the sink.

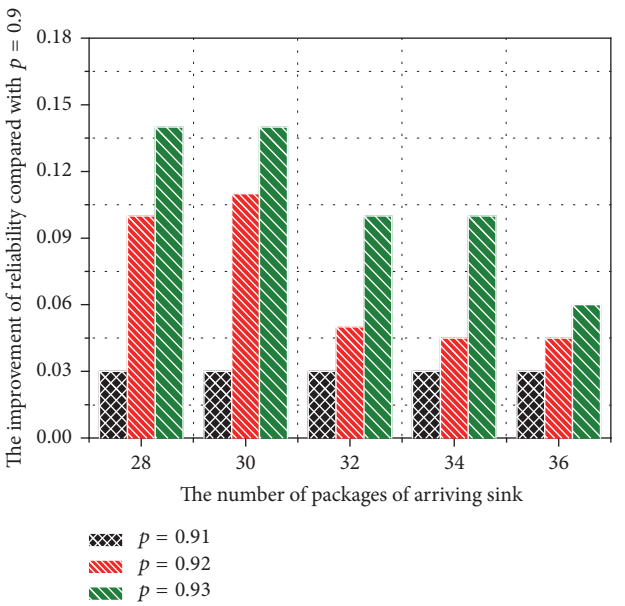

Figure 44: Improvement of network reliability when the same data amount reaches the sink and the value of p increases from 0.90 to 0.91, 0.92, and 0.93 in ASTR scheme.

6. Conclusions

Internet of Everything (IoE) [1–3] leverages the ubiquity of smart sensor-equipped devices such as sensor based devices, smartphones, and vehicle sensor devices to collect information at low cost and provide a new paradigm for solving the complex data sensing based applications from the significant demands of critical infrastructure such as surveillance systems, remote patient care systems in healthcare, intelligent traffic management, and automated vehicles in transportation environmental and weather monitoring systems. Despite its great potential in our lives, sensor based IoE also exposes users to new security threats, which can impact human users' health and safety. Data authentication, as an important defense, can be used to prevent unauthorized attack in wireless sensor networks. In this paper, an aggregate signature based trust routing (ASTR) scheme is proposed to guarantee safe data collection in WSNs. Firstly, the aggregate signature approach is used to aggregate data and keep data integrity. Then, a $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method is proposed to improve the probability for the data to safely reach the sink and reduce the redundant data transmission in order to extend the network lifetime. The $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing method overcomes the shortcomings of sending too much data and low data security in the past multipath routing. In $\mathscr{R}(\mathscr{M}, \mathscr{N})$ routing, some lightweight abstracts are used to replace the heavy data, which is able to effectively reduce the network load and improve the routing security. Finally, the ASTR scheme adopts a trust routing method to further improve the security of routing. The results of our strict theoretical analysis show that the ASTR scheme can effectively increase the safe reach rate of data routing by 23.23%, reduce the data amount on the node by 53.95%, and reduce the redundant data amount by 41.70%.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61772554, 61379110, 61572526, and 61572528) and the National Basic Research Program of China (973 Program) (2014CB046305).

References

[1] L. Shen, J. Ma, X. Liu, F. Wei, and M. Miao, "A Secure and Efficient ID-Based Aggregate Signature Scheme for Wireless Sensor Networks," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 546–554, 2017.

[2] A. Liu, Q. Zhang, Z. Li, Y. Choi, J. Li, and N. Komuro, "A green and reliable communication modeling for industrial internet of things," *Computers Electrical Engineering*, vol. 58, pp. 364–381, 2017.

[3] Q. Jiang, S. Zeadally, J. Ma, and D. He, "Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks," *IEEE Access*, vol. 5, pp. 3376–3392, 2017.

[4] Q. Jiang, J. Ma, F. Wei, Y. Tian, J. Shen, and Y. Yang, "An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 76, pp. 37–48, 2016.

[5] S. He, D. Shin, J. Zhang, J. Chen, and P. Lin, "An Exchange Market Approach to Mobile Crowdsensing: Pricing, Task Allocation, and Walrasian Equilibrium," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 4, pp. 921–934, 2017.

[6] Z. Chen, M. Ma, X. Liu, A. Liu, and M. Zhao, "Reliability Improved Cooperative Communication over Wireless Sensor Networks," *Symmetry*, vol. 9, no. 10, p. 209, 2017.

[7] X. Duan, C. Zhao, S. He, P. Cheng, and J. Zhang, "Distributed Algorithms to Compute Walrasian Equilibrium in Mobile Crowdsensing," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4048–4057, 2017.

[8] D. Zeng, L. Gu, L. Lian, S. Guo, H. Yao, and J. Hu, "On cost-efficient sensor placement for contaminant detection in water distribution systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2177–2185, 2016.

[9] C. Hu, M. Li, D. Zeng, and S. Guo, "A survey on sensor placement for contamination detection in water distribution systems," *Wireless Networks*, 2016.

[10] X. Chen, Y. Xu, and A. Liu, "Cross Layer Design for Optimizing Transmission Reliability, Energy Efficiency, and Lifetime in Body Sensor Networks," *Sensors*, vol. 17, no. 4, p. 900, 2017.

[11] J. Wang, A. Liu, T. Yan, and Z. Zeng, "A resource allocation model based on double-sided combinational auctions for transparent computing," *Peer-to-Peer Networking and Applications*, pp. 1–18, 2017.

[12] H. Li, D. Liu, Y. Dai, T. Luan, and S. Yu, "Personalized Search over Encrypted Data with Efficient and Secure Updates in Mobile Clouds," *IEEE Transactions on Emerging Topics in Computing*, 2016.

[13] Z. Su, Q. Xu, M. Fei, and M. Dong, "Game theoretic resource allocation in media cloud with mobile social users," *IEEE Transactions on Multimedia*, vol. 18, no. 8, pp. 1650–1660, 2016.

[14] A. Liu, Z. Chen, and N. N. Xiong, "An Adaptive Virtual Relaying Set Scheme for Loss-and-Delay Sensitive WSNs," *Information Sciences*, vol. 424, pp. 118–136, 2018.

[15] G. Yang, S. He, and Z. Shi, "Leveraging Crowdsourcing for Efficient Malicious Users Detection in Large-Scale Social Networks," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 330–339, 2017.

[16] T. Wang, Y. Li, G. Wang, J. Cao, M. Z. Bhuiyan, and W. Jia, "Sustainable and Efficient Data Collection from WSNs to Cloud," *IEEE Transactions on Sustainable Computing*, 2017.

[17] Q. Jiang, J. Ma, X. Lu, and Y. Tian, "An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 8, no. 6, pp. 1070–1081, 2015.

[18] D. Wang, N. Wang, P. Wang, and S. Qing, "Preserving privacy for free: Efficient and provably secure two-factor authentication scheme with user anonymity," *Information Sciences*, vol. 321, Article ID 11496, pp. 162–178, 2015.

[19] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 3, pp. 312–325, 2016.

[20] B. Xiao, B. Yu, and C. Gao, "CHEMAS: Identify suspect nodes in selective forwarding attacks," *Journal of Parallel and Distributed Computing*, vol. 67, no. 11, pp. 1218–1230, 2007.

[21] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 293–315, 2003.

[22] M. Zhou, M. Zhao, A. Liu, M. Ma, T. Wang, and C. Huang, "Fast and Efficient Data Forwarding Scheme for Tracking Mobile Targets in Sensor Networks," *Symmetry*, vol. 9, no. 11, p. 269, 2017.

[23] A. Liu, Z. Zheng, C. Zhang, Z. Chen, and X. Shen, "Secure and energy-efficient disjoint multipath routing for WSNs," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3255–3265, 2012.

[24] H.-M. Sun, C.-M. Chen, and Y.-C. Hsiao, "An efficient countermeasure to the selective forwarding attack in wireless sensor networks," in *Proceedings of the IEEE Region 10 Conference, TENCON 2007*, 4, 1 pages, November 2007.

[25] J. Xu, X. Liu, M. Ma, A. Liu, T. Wang, and C. Huang, "Intelligent Aggregation Based on Content Routing Scheme for Cloud Computing," *Symmetry*, vol. 9, no. 10, p. 221, 2017.

[26] T. Li, Y. Liu, L. Gao, and A. Liu, "A cooperative-based model for smart-sensing tasks in fog computing," *IEEE Access*, vol. 5, pp. 21296–21311, 2017.

[27] R. X. Lu, X. D. Lin, H. J. Zhu, X. H. Liang, and X. Shen, "BECAN: a bandwidth-efficient cooperative authentication scheme for filtering injected false data in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 32–43, 2012.

[28] J. Gui, L. Hui, and N. Xiong, "A Game-Based Localized Multi-Objective Topology Control Scheme in Heterogeneous Wireless Networks," *IEEE Access*, vol. 5, pp. 2396–2416, 2017.

[29] J. Gui and J. Deng, "A Topology Control Approach Reducing Construction Cost for Lossy Wireless Sensor Networks," *Wireless Personal Communications*, vol. 95, no. 3, pp. 2173–2202, 2017.

[30] H. Xin and X. Liu, "Energy-Balanced Transmission With Accurate Distances for Strip-Based Wireless Sensor Networks," *IEEE Access*, vol. 5, pp. 16193–16204, 2017.

[31] X. Liu, Y. Liu, H. Song, and A. Liu, "Big Data Orchestration as a Service Network," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 94–101, 2017.

[32] Y. Xu, A. Liu, and C. Changqin, "Delay-aware program codes dissemination scheme in internet of everything, mobile information systems," *Mobile Information Systems*, vol. 2016, Article ID 2436074, 18 pages, 2016.

[33] D. Zeng, P. Li, S. Guo, T. Miyazaki, J. Hu, and Y. Xiang, "Energy Minimization in Multi-Task Software-Defined Sensor Networks," *IEEE Transactions on Computers*, vol. 64, no. 11, pp. 3128–3139, 2015.

[34] J. Gui and K. Zhou, "Flexible adjustments between energy and capacity for topology control in heterogeneous wireless multi-hop networks," *Journal of Network and Systems Management*, vol. 24, no. 4, pp. 789–812, 2016.

[35] A. Liu, X. Liu, Z. Tang, L. T. Yang, and Z. Shao, "Preserving Smart Sink-Location Privacy with Delay Guaranteed Routing Scheme for WSNs," *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 3, pp. 1–25, 2017.

[36] X. Liu, "A novel transmission range adjustment strategy for energy hole avoiding in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 67, pp. 43–52, 2016.