symmetry

MDPI

Article

Energy Efficient Fuzzy Adaptive Verification Node Selection-Based Path Determination in Wireless Sensor Networks

Muhammad Akram [1] and Tae Ho Cho [2,*]

[1] College of Information and Communication Engineering, Sungkyunkwan University, Suwon 16419, Korea; akram.khan@skku.edu
[2] College of Software, Sungkyunkwan University, Suwon 16419, Korea
* Correspondence: thcho@skku.edu; Tel.: +82-31-290-7221

Received: 7 August 2017; Accepted: 30 September 2017; Published: 10 October 2017

Abstract: Wireless sensor networks are supplied with limited energy resources and are usually installed in unattended and unfriendly environments. These networks are also highly exposed to security attacks aimed at draining the energy of the network to render it unresponsive. Adversaries launch counterfeit report injection attacks and false vote injection attacks through compromised sensor nodes. Several filtering solutions have been suggested for detecting and filtering false reports during the multi-hop forwarding process. However, almost all such schemes presuppose a conventional underlying protocol for data routing that do not consider the attack status or energy dissipation on the route. Each design provides approximately the equivalent resilience in terms of protection against compromised node. However, the energy consumption characteristics of each design differ. We propose a fuzzy adaptive path selection to save energy and avoid the emergence of favored paths. Fresh authentication keys are generated periodically, and these are shared with the filtering nodes to restrict compromised intermediate filtering nodes from the verification process. The scheme helps delay the emergence of hotspot problems near the base station and exhibits improved energy conserving behavior in wireless sensor networks. The proposed scheme provides an extended network lifetime and better false data filtering capacity.

Keywords: fuzzy; adaptive; energy; en-route filtering; wireless sensor networks

1. Introduction

Wireless sensor networks (WSNs) represent a sustainable solution for continuous observation of arbitrary events in the physical sphere and consist of resource-constrained nodes installed in a range of hostile environments [1–4]. Sensor networks are predicted to intermingle with the physical world and also support various novel applications [5]. In most applications, sensor nodes are deployed in exposed environments, and therefore are susceptible to physical attacks, potentially compromising the node's cryptographic keys [6]. Attackers can insert false sensing reports through compromised nodes, which both cause false alarms at the base station (BS) and deplete the limited energy resources of the individual nodes in the network [7]. False Report Injection Attack (FDIA) and False Vote Injection Attack (FVIA) are two serious attacks among several others that can occur in WSNs [7,8]. Sensor observations are temporally and spatially correlated in WSNs [9]. Spatial and temporal correlations in the observations of closely located sensor nodes are exploited to reduce the propagation of redundant data and multiple nodes authenticate the data to be forwarded by a single designated node or cluster head. Cluster-based organization helps reduce the communication overhead and data redundancy through data fusion and data collection [10]. However, a multitude of nodes in a cluster can add bogus Message Authentication Codes (MACs) to a genuine event data generated

at the cluster. The cluster head forwards the same report, on a low-cost path, to the base station (BS). The verification nodes on the path attempt to verify the forwarded report by authenticating the MACs attached to it. The report is dropped as soon as the detected false MACs number hits a pre-set threshold value. Attaching bogus MACs to legitimate reports results in denial of service (DoS). Such reports are repeatedly generated and forwarded to the BS by the event reporting cluster head until the BS acknowledges its receipt. The regeneration, retransmission, and repeated verification of such reports causes drain of the limited energy resources in the sensor network. In FRIA, several nodes within a cluster conspire to fabricate a report about a non-existent event in the surrounding environment and attach MACs to it. The reception and forwarding and en-route verification of these fabricated reports drains a significant amount of the limited energy resources at the intermediate verification nodes [11]. For the sake of clarity, a cluster head that generates the report is referred to as an event-cluster head or as an e-CH, whereas all the other cluster heads are simply referred to as cluster heads or as CHs hereafter.

In the past, several filtering schemes have been proposed in an effort to counter the two attacks, viz. FVIA and FRIA [7,12–17]. Either static [17] or dynamic [18] key management schemes are used to generate keys in en-route filtering schemes. As a defense against FRIA, various solutions, such as location based resilient security (LBRS) [19], the statistical en-route filtering scheme (SEF) [5], the key inheritance-based filtering scheme (KIF) [20], the interleaved hop-by-hop authentication scheme (IHA) [16], and the dynamic en-route filtering scheme (DEF) [12] have been proposed. These security techniques prevent forwarding of reports of non-existent events in the cluster and filter false reports during the forwarding process. However, these methods also inadvertently make it easier for adversaries to launch FVIA, and all the reports with a single false vote/MAC attached to them are dropped en-route if SEF, DEF, IHA, KIF, or LBRS are used. As a countermeasure against FVIA, different security solutions such as the multi-path en-route filtering scheme (MEF) [21], probabilistic voting-based filtering scheme (PVFS) [7], and false negative resilient SEF (FNRSEF) [22] have been proposed. In PVFS, reports with false votes less than the threshold are forwarded to the BS, whereas multiple copies of the reports are transmitted through multiple routes in the MEF and FNRSEF.

However, the previously mentioned schemes are not effective in terms of saving energy when attacks do not occur [23], and due to the extra computational and communication overhead. The associated energy and communication costs are seldom discussed by researchers who propose secure WSN protocols.

Existing routing protocols can be categorized into three groups: the one-hop model, the multi-hop planar model and the cluster-based hierarchical model [7]. The one-hop model is impractical for large scale WSNs, and it does not accommodate filtering schemes. In the multi-hop planar model, due to no division of sensors in the network, compromised nodes in arbitrary locations can conspire with each other to launch a FRIA attack because there are no divisions between sensors in the network [5]. Cluster-based data routing has proven to be effective in minimizing energy consumption, managing network topology and aggregating data in WSNs [24].

In WSNs, multi-hop cluster head communication is more energy efficient, and the CHs collaborate with each other to forward their data to the BS [25]. Thus, data forwarding on a multi-hop path through intermediate CHs is a more realistic solution [25]. Cluster head communication facilitates energy efficient and safe routing of data in WSNs. A cluster based trust-aware routing protocol allows cluster member nodes to forward reports through the trusted CHs towards the BS [26]. The trust-aware routing protocol allows for re-election of a new CH to maintain safe routing in the network. Multi-hop cluster head-based communication is scalable and provides energy efficiency in WSNs [27,28]. Therefore, cluster-based model becomes a natural option due to its suitability as a filtering mechanism. Cluster head communication saves energy in multi-hop communication based WSNs [7,26].

In [11], we proposed the Fuzzy Adaptive Selection of Intermediate verification Nodes (referred to as FASIN hereafter), which adaptively selects verification nodes based on the attack situation, the energy levels and the distance of the nodes. FASIN surpasses the PVFS in terms of energy saving and

extended network life in the sensor network, and it provides equivalent security against FRIA and FVIA at the same time.

We propose a fuzzy rule-based route selection coupled FASIN that uses dynamic authentication key dissemination technique to achieve earlier detection of fabricated reports during the filtering process and enhance the energy efficiency of the filtering-based WSNs.

The proposed method considers the following three important factors in the sensor network for selecting the fittest routing path:

1. The number of verification nodes on the path within h hops from the e-CH where $h = L$ (the number of nodes in a cluster).
2. The hop-count distance ratio of the routing path being considered and the longest path between the e-CH and the BS.
3. The remaining energy status of the intermediate nodes on the path.

By considering the above three input parameters, fuzzy inferencing carried out at the e-CH helps choose the most appropriate routing path for the current situation. FASIN is used in conjunction with the path selection to improve the performance of the less fit routes in an effort to increase their chance of being selected and avoid the emergence of a favored path anomaly.

The remainder of this paper is organized as follows. In Section 2, we present an overview of the PVFS and FASIN algorithms and provide a rationale for the proposed scheme. The proposed method is presented and explained in Section 3. Simulation results are presented and discussed in Section 4, followed by a study of related work in Section 5. Finally, Section 6 presents the conclusion.

2. Background and Rationale

Most filtering schemes have been able to counter one particular kind of attack while leaving space for other attacks to occur in the network. FRIA and FIVA cause serious damage to the sensor network and result in rapid depletion of energy resources and denial of service, respectively. They further cause a decrease in the network lifetime and increased information loss when a certain number of sensor nodes die due loss of energy. PVFS and FASIN are designed to counter both types of attacks. In the following sections, we briefly discuss PVFS and FASIN techniques for countering FRIA and FVIA attacks to provide a motivation for our suggested scheme.

2.1. PVFS

PVFS makes use of a probabilistic key assignment routine for report authentication. In PVFS, the sensor network is organized into clusters, each comprised of L nodes. A set of finite keys is assigned to every cluster, CHs are elected, and only their one-hop neighbors are the member nodes of the same cluster. PVFS assumes that the creation of clusters, dissemination of keys, and path discovery are performed immediately after the sensor deployment. During the phase of path discovery, every cluster head acquires the IDs of all the intermediate upstream CHs on the paths between the cluster and the base station (BS), their distance to the BS in hop counts, and its own distance to the BS. The source CH probabilistically selects intermediate CHs as verification nodes on each of the c available paths, where the value of c may be different for different clusters. Each member node in the event cluster individually share their report generation keys with the selected verification nodes. During generation key sharing, a session is established between the member node and the verification node using a pairwise session key establishment protocol.

2.2. FASIN

FASIN [7] uses the voting method to verify real reports, as in PVFS. However, FASIN fundamentally differs from PVFS in the selection of intermediate verification nodes. Initially, in the path discovery process, each cluster head acquires the IDs of the upstream CHs, their distances to the e-CH and its own distance to the BS, as well as information about the energy status of the

CHs [7,11,29]. Initially the e-CH randomly chooses intermediate verification CHs on all c different routing paths available to the e-CH. Each routing path contains L verification nodes, which is equal to the cluster size. After the selection of a verification node, the e-CH notifies each node in its cluster to securely exchange their generation keys with the corresponding verification nodes using the session key generated through pairwise key establishment protocols. FASIN makes use of the fuzzy inferencing engine implemented at the e-CH for the selection of the verification nodes during the operation of the network. The remaining energy level and the distance of the verification nodes, and the current attack situation in the network are considered during the verification nodes selection. FASIN improves the proximity of the verification nodes in response to an increase in attack rate, increases the network energy efficacy and improves the network operational life time.

FASIN helps choose verification nodes dynamically and can be used to improve the performance and fitness of the otherwise less fit paths because they are either long or the verification nodes on those paths are relatively farther apart than those on the fittest path.

Therefore, we propose using FASIN with a fuzzy adaptive path selection method for data forwarding to improve the selection chances of the paths that are otherwise least likely to be selected for data forwarding unless a node or route failure occurs.

2.3. Rationale

Like in most en-route filtering scheme, PVFS and FASIN use minimum cost forwarding, i.e., the shortest path for forwarding the event reports. However during the path discovery phase, the e-CH discovers c (the value of c may be different for different clusters) different paths to the sink to overcome a node failure. The shortest path is always used for data forwarding until a node failure occurs on the path. Due to this uneven distribution of workload in the intermediate nodes, the energy dissipation behavior is uneven across the network. Therefore, nodes with extra workloads (i.e., receiving, verifying and forwarding the data) die out sooner due to rapid energy loss. Therefore, the energy of the CHs on the shorter paths deplete rapidly compared to that of the nodes/CHs on other paths [30] even though the number of verification nodes on every path is same (and therefore so is the filtering strength). However, some paths have more verifications nodes closer to the e-CH than other paths. The path that has the lowest average distance of the intermediate verification nodes from the e-CH has more verification nodes closer to the e-CH. It will be more convenient to filter false reports earlier if the attack rate is higher.

Equation (1) calculates the average number of verification keys stored on a filtering CH_i.

$$N_{ver-keys} = c \cdot (d_{max} - d_i), \quad (1)$$

d_{max} is the distance between the farthest CH and the BS and d_i is the distance between the CH_i and the BS [7]. CHs that are closer to the BS hold more keys than the CHs that are farther away, and thus they are expected to perform more verifications for several different clusters that generate reports. Moreover, if filtering nodes are nearer to the BS, false reports may travel farther before they are filtered. Since nodes around the BS perform data-relay activities more often on behalf of several clusters, their energy drains more rapidly than other nodes in the network. Therefore, a hotspot problem occurs around the BS. Selection of verification nodes away from the BS and closer to the report generating CHs helps to filter false reports earlier and avoid the hotspot problem from happening around the BS.

Moreover, authentication keys are created and shared only once during the initialization phase of network deployment. In FASIN, the same shared keys are exchanged between the newly chosen and previously active verification nodes. In PVFS, the filtering nodes are chosen probabilistically and remain fixed during the entire lifetime of the network. However, one serious problem with the static sharing of authentication keys and verification node selection is that the filtering nodes may get compromised, and the adversary can obtain all the verification keys on the compromised intermediate

node. Therefore, it is highly desirable that both the verification nodes be chosen dynamically, and the authentication keys be created and shared afresh with the intermediate verification nodes.

In most routing protocols, only the energy is considered during the selection of routing paths. However, in the presence of a security attack, the selection of such paths may result in a rapid depletion of energy resources of nodes than the paths that have not been chosen so far [31].

PVFS is static and does not adapt to the frequency of the attacks, data generation rate, and the energy on the nodes participating in the data forwarding and verification. Furthermore, PVFS does not use back check keys and drop report acknowledgment [11].

In FASIN, the filtering nodes are initially chosen randomly. However, as time passes, the nearness of the verification nodes to the location of the e-CH is gradually improved adaptively in response to the current attack situation and the average node energy on the path. The dynamic selection of verification nodes in FASIN only takes place on the same path used for data forwarding, i.e., the chosen shortest path. Thus, the energy of nodes on the same path may still be depleted compared to the energy of the nodes on the unchosen paths.

Therefore, both early filtering of the fabricated reports and balancing the workload on the chosen filtering nodes and along the routing path are of utmost importance in improving the energy efficiency and increasing the lifetime of WSNs.

In a WSN, the radio unit on the sensor consumes the most energy. In single hop communication based WSNs, CHs located far from the BS have a larger energy burden due to the long-haul communication link. Therefore, multi-hop cluster head communication is more energy efficient, and the CHs collaborate with each other to forward their data to the BS [25]. Thus, data forwarding on a multi-hop path through intermediate CHs is more realistic in this case [25]. The cluster head selection can be periodically rotated with in a cluster to balance the energy consumption. MR-LEACH [32] is a multi-hop routing strategy for energy conservation in WSNs in which an intermediate CH serves as a relaying node. In MR-LEACH, the intermediate nodes are only the CHs forming a tree rooted at the BS, and member nodes serve as leaves. The context aware architecture for probabilistic voting-filtering scheme in [29] also presupposes a multi-hop cluster head-based data forwarding technique due to its inherent energy conserving feature. LEACH provides extended network lifetimes [33]. Similarly, in our proposed scheme, the role of the CH can be rotated with in the cluster to achieve energy balancing as proposed in [25,26,32,34]. However, a discussion of such CH rotation techniques is beyond the scope of this paper.

3. Dynamic Key Sharing and Fuzzy Adaptive Route Selection

In this section, we discuss the problem definition, goals, assumptions, the model for the threats considered, and present a thorough explanation of the proposed scheme.

3.1. Problem Definition

3.1.1. Hot Spot Problem

In multi-hop communication, nodes near the BS are burdened with relaying a large amount of data to the BS, which creates a hotspot problem. Rapid depletion of energy at nodes around the BS eventually disconnects the rest of the network from the BS.

3.1.2. False Data Injection and False MAC Injection

Injection of false votes by the adversaries is intended to deplete the energy of the intermediate relay nodes. Similarly, legitimate data endorsed by compromised nodes are corrupted by attaching bogus MACs and are dropped en-route during en-route verification. This prevents true event information from reaching the BS.

3.1.3. Network Portioning Due to Constant Use of Minimum Cost Forwarding Paths

The potential issue associated with the multi-hop based minimum cost and multipath data forwarding schemes is that the minimum cost path is constantly used until it fails. This scheme does not provide the best network lifetime. Constant use of a minimum cost path results in depletion of energy at nodes on the same path and causes a network partition.

3.2. Goals

We propose a distributed fuzzy heuristic based method that minimizes the effect of the previously mentioned problems in WSNs. Sensor nodes in the network can carry out fuzzy computations, and a fuzzy logic based system can be hard-coded on a node to reduce the code size.

3.2.1. Load Balancing

Fuzzy adaptive selection of data forwarding path potentially balances the load between data forwarding paths. Load balancing spreads energy utilization across nodes in the network, potentially resulting in longer network lifetime [35].

3.2.2. Fuzzy-Based Selection of Data Forwarding Path

Our proposed method chooses the suitable path among the several available paths in response to input factors such as the number of verification nodes within h hops on every path, the residual energy status of the routing paths, and the distance between the e-CH and the BS via each path.

Our proposed scheme uses FASIN as a submodule only for the adaptive selection of the intermediate verification nodes on the selected path.

3.2.3. Dynamic Authentication Key Sharing with the Verification Nodes

Whenever FASIN selects intermediate verification nodes, the member nodes in the event cluster generate authentication keys and share them with the chosen verification nodes afresh. This minimizes the chances of stale keys being captured by the adversary to create False MACs or forge false reports and restricts the previous verification nodes from duplicating the authentication keys.

3.2.4. Identification of Unhealthy Nodes Near the BS

Our proposed scheme determines the qualification of the potential filtering path and identifies those unhealthy nodes on the path whose energy is below a threshold value given the distance between the e-CH and the BS.

3.3. System Model and Assumptions

We study a large-scale wireless sensor network whose nodes are densely deployed and organized in a cluster-based topology. After the nodes are deployed, they do not change their initial locations and remain stationary during the operation of the network. The reason for choosing a cluster-based hierarchical model is that it is most appropriate for multi-hop communication and assists the common en-route filtering paradigm [7]. When the density of sensor nodes is high enough, cluster merge methods can be used to include no less than L nodes in the cluster [7,29]. There are several techniques available to make clusters, merge clusters, choose CHs and produce cluster IDs [26]. We assume that the sensing range of sensor nodes is larger than their transmission range, and every CH uses a longer transmission range than that of the member nodes in the cluster [7,8]. All nodes are assigned node IDs prior to network deployment. A simple cluster formation scheme is assumed; that is all nodes within one hop distance join to form a cluster, and the nodes that have the smallest IDs in the clusters are elected to be CHs [7]. For the sack of simplicity, it is assumed that sensor nodes cannot be compromised for a very small interval of time during which the formation of clusters, key distribution, and discovery of paths are performed [7,11]. We further assume that the network is exposed to only

FRIA and FVIA [7,11]. Since the nodes are spatially correlated, clusters located adjacent to each other contend with each other whenever an event happens, and the wining cluster prepares the report. Each member node checks the report for consistency with its observations and appends a MAC to it. After having received all the votes, e-CH randomly picks a preset number of votes, including its own vote, and affixes them to the report and transmits the report to the BS. Compromised intermediate verification nodes can treat false votes on a report as true votes, and they can similarly invalidate true votes. Compromised nodes can be exploited to launch several other types of attacks; however, a discussion of all other possible attacks, their impact, and countermeasures is beyond the scope of this paper.

3.4. Threat Model

Adversaries can compromise sensor nodes and exploit them to inject false reports and false votes in the network. Adversaries can also injection false reports through an outside entity. The frequency of the attacks increases gradually with an increasing number of nodes being compromised by the adversaries. Adversaries can insert a large number of fabricated reports and attach false votes to the legitimate reports to intensify the effect of the attacks. Compromised nodes belonging to the same cluster can communicate with each other to generate fabricated reports, whereas compromised nodes belonging to separate clusters cannot conspire with one another.

3.5. Dynamic Authentication Key Sharing

In PVFS, LBRS, SEF, FASIN, KIF, DEF, FASIN and IHA, the verification keys are disseminated to the intermediate verification nodes only once during the initialization phase and are assumed to never expire. The disadvantage of such a key sharing scheme is that it is static, and an adversary can compromise the intermediate verification nodes and gather the authentication keys stored on those nodes. To counter this problem and make the proposed scheme safer and capable of adaptively responding to the increasing attack, we propose that the nodes in the event reporting cluster dynamically update and share their authentication keys with the recently chosen intermediate verification nodes.

We assume that every member node is preloaded with a key from the global key pool which serves as a seed-key for the one-way hash chain of generation keys. Let's say that a node n_i gets $key_m^{n_i}$ during the key assignment stage. This key is used as a seed key to create a one-way hash chain of size, m: $key_{m-1}^{n_i}, key_{m-2}^{n_i}, \ldots, key_1^{n_i}, key_0^{n_i}$, where $key_j^{n_i} = f^{m-j}(key_m^{n_i})$, using a pseudo random one-way function f. The keys will be used in reverse order of their creation. That is $key_0^{n_i}$ will be the first authentication key to share with the intermediate verification node. Either all the keys can be created at once and stored at the node, or only seed key $key_m^{n_i}$ can be stored and the other keys can be computed on demand. In practice, the hybrid method helps decrease the storage overhead due to the small re-computation effort [36]. The storage efficient mechanism in [36] only requires $\log(m)$ storage and $\log(m)$ computation to access a key in a one-way hash chain of size m.

Node n_i's seed key is $key_m^{n_i}$.

The one-way hash chain created by node n_i is $key_{m-1}^{n_i}, key_{m-2}^{n_i}, \ldots, key_1^{n_i}, key_0^{n_i}$.

A one-way hash chain is created recursively: $key_{m-1}^{n_i} = f(key_m^{n_i})$.

Therefore, node n_i creates any $key_j^{n_i}$ in a one-way hash chain using f:$key_j^{n_i} = f^{m-j}(key_m^{n_i})$.

By definition: $f^j(k) = f^{j-1}(f(k))$.

The size of the one-way hash chain is m.

Node n_i shares its keys with the verification nodes in this order: $key_0^{n_i}$, $key_1^{n_i}, key_1^{n_i}, \ldots, key_{m-1}^{n_i}, key_m^{n_i}$.

We assume that the length of each key chain is such that they last for the entire duration of the communication [37]. The size of the one-way hash chain can be calculated by estimating the network mission lifetime and the battery life of the sensor nodes under normal operation of the network [38]. Given the battery life of each sensor node b_i, $i = 1, \ldots, n$, and a network mission lifetime of B for

normal operation of the network, we can calculate the size of the one-way hash chain m_i for sensor node n_i, or a single size m for all the hash chains.

$$m_i = \frac{b_i}{l}\,; i = 1, \ldots,\ n.$$

$$m = \frac{B}{l}\,.$$

Here, $B = \frac{1}{n}\sum_{i=1}^{n} b_i$ and l is the cycle length, which depends on the frequency of fuzzy inferencing operations for the selection of intermediate verification nodes.

Implementation of a one-way hash chain provides backward secrecy. That is, even if a $key_j^{n_i}$ is shared by a node n_i with one of the intermediate verification nodes during session/cycle j, and is revealed to an adversary, it does not compromise future secret authentication keys. As $key_{m-1}^{n_i} = f(key_m^{n_i})$, it is computationally impossible for the adversary to calculate the previous forward key. The period of key dissemination is determined when the e-CH performs the verification node selection. Every verification node CH_i shares their separate symmetric key, k_{CH_i-BS} , with the BS station to produce their verification signatures. A session key can be created using pairwise key establishment protocols, as in [39], to securely transmit the encrypted generation key, i.e., $E_{n_i}\left(key_j^{n_i}\right)$, to the chosen verification node. Alternatively, the e-CH may instruct all member nodes to calculate their new generation keys and share with the verification nodes chosen for the current round until e-CH performs another verification node selection procedure when the current round expires.

Two association nodes, i.e., a member node in the source cluster and an intermediate CH, need to share an authentication key over multiple hops based on one of the id-based schemes [39–43]. All these schemes have comparable computational overhead and communication cost [16].

The average number of verification keys stored by an intermediate CH is less than or equal to $c{\cdot}(d_{max} - d_i)$. where, d_{max} is the distance between the BS and the farthest CH. there are $4{\cdot}2h$ number of CHs that are h hops away from the BS for a uniform distribution of CHs. Initially, a CH that is $h + j$ hops away from the BS selects an intermediate CH_i that is h hops away with a probability: $\frac{d_i}{d_i+j}$. The average number of CHs that are $h + j$ hops away from the BS select one CH that is h hops away from the BS as a verification node on the path is: $\frac{8\cdot(d_i+j)}{8\cdot d_i}$. Every CH selects c different paths and the average number of verification keys a CH that is h hops away from the BS stores for the CH that is $h + j$ hops away from the BS is 1. The total number of verification keys is less than or equal to $\sum_{j=1}^{d_{max}-d_i} c$. Secure cryptographic one-way hash functions can be used with techniques such as salting, key stretching, chaining, etc., to make it difficult for an attacker to compute or find the one-way hash chain key in any rainbow table [44]. However, a discussion of these techniques is beyond the scope of this paper.

The compromise of one-way hash function bears a resemblance to the compromise of a node in the sensor network that uses the same function to generate future authentication keys. If an attacker gets hold of the one-way hash function by compromising a node, the hacker can generate another key using the same function. Our proposed scheme uses the cluster-based organization which limits the degree to which compromised nodes could misuse their keys, as it becomes useless for nodes belonging to different clusters to conspire. Every report must be validated by s distinct votes using keys generated by the nodes belonging to a single cluster. The adversary must gather s distinct keys generated by s different one-way hash functions in a single cluster to successfully create a false report that eludes intermediate verification nodes. It also makes the identification and exclusion of compromised nodes easier. After a false vote is detected, the node which generated the same vote and the cluster it belongs to can be found and then isolated from the network.

3.6. Fuzzy Adaptive Selection of Data Forwarding Routes

PVFS has functional power and the flexibility to deal with both FRIA and FVIA attacks, making it a good choice for our proposed scheme to further improve energy efficiency, load balancing, and

increased network life. There are always c different paths available to the e-CH; however, it uses only the shortest path for forwarding the reports [7,11]. For each data generating cluster, every path will have exactly the same number of verification nodes (L). Therefore, the abilities of each route to detect false reports are equal in terms of the number of verification keys. However, due to the probabilistic selection of the verification nodes, their distances relative to the e-CH are different. That is some nodes are closer to the e-CH, while others are farther away. It is very likely that a false report may travel farther on the shortest path than on a longer path since we know that:

$$\begin{gathered} if\ p(n_i, d_{lo}) = \frac{d_i}{d_{lo}},\ p(n_i, d_k) = \frac{d_i}{d_k},\ p(n_i, d_{sh}) = \frac{d_i}{d_{sh}}, \\ then\ p(n_i, d_{lo}) < p(n_i, d_k) < p(n_i, d_{sh})\ for\ d_{lo} > d_k > d_{sh}, \end{gathered} \tag{2}$$

Here, d_i represents the distance between the BS and the ith CH on any data routing path, and d_{lo}, d_k, and d_{sh} represent the distance between e-CH and the BS via the longest, k-hop long, and shortest path, respectively. Equation (2) show the probability that a node that is i hops away from the BS and located on a specific path will be chosen as a verification node. It shows that, though the distance from the BS may be same on different paths, however the probability of a node being chosen as a verification node is different from that of other nodes. Therefore, nodes farther from the BS on longer paths are more likely to become verification nodes than nodes on shorter paths. Choosing verification nodes near the BS implies that they hold more authentication keys, and false data are highly likely to travel near the BS before they are filtered by the verification nodes. False data traveling near the BS result in a higher energy cost on nodes around the BS. Therefore, the energy-hole problem will occur sooner than in other configurations.

Figure 1 shows how a cluster head chooses between the different available upward paths. The number of verification keys within, say $h = 8$ hops, is greater on the upper path than on the lower path. Moreover, we can say that the energy status of the intermediate nodes on the upper path is better than that of the verification nodes on the lower path. Therefore, it is ideal for the e-CH to choose the upper path, although its hop length is greater than the lower path. The rationale for considering verification nodes only within L hops (where $h = L$) is that all the paths have exactly the same number of verification nodes. Therefore, the filtering power of each path is essentially the same. A few verification nodes might not be closer to the e-CH, but may be closer to the BS station. It makes sense to only consider the number of verification nodes within L hops rather than considering all the nodes since we use the threshold values (such as $s \leq L$) to establish the number of MACs necessary to endorse a report, and check if the number of recorded true votes T_t or false votes T_f have reached threshold values.

The topology creates the routes using Dijkstra's shortest path algorithm [45,46] for the cluster based model [7,12,47–49]. The selection of a minimum cost multi-constrained path is NP-complete [23]. The CH also identifies the 2nd, 3rd, ... , cth shortest paths during the route discovery phase to make use of these in case a node fails on the shortest path [1,18]. Each CH can discover c different paths, where c is a variable and its value may be different for every cluster.

Routing paths are created by flooding a control message broadcasted by the BS. Most routing protocols use this technique for establishing paths [50,51]. This control message contains the sender ID, hop-count from the BS and the energy information. When the control message is received by a node, it stores the information in the control message. The node forwards the control message after updating it by inserting its own information into the control message. This technique is particularly useful because the control message is propagated with an increasing size farther from the BS, preventing nodes around the BS from relaying heavier control messages. The cluster represented by e-CH may receive more than one instance of the control message traversing different paths from the BS to the cluster represented by e-CH. After receiving the control message, the e-CH computes the fitness of the inward paths through fuzzy inferencing.

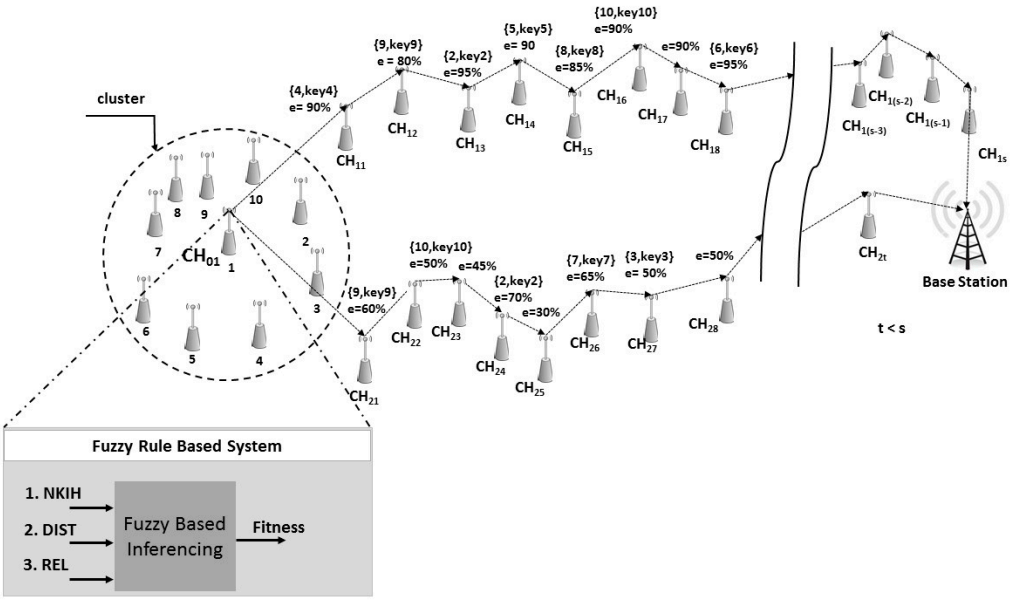

Figure 1. Selection of data forwarding route.

Authors in [35] propose techniques to proactively setup and maintain alternate paths in addition to the shortest path at the expense of some energy which minimize the likelihood of invoking data flooding to discover alternate paths. Low-rate data is sent along multipaths for maintaining multipaths without the need for network-wide data flooding. The *k*-disjoint (here *k* has same value as *c*) multipath scheme favors the cluster based en-route filtering framework as it ensure to avoid duplication of verification keys on the intermediate node and comparatively incurs lower maintenance overhead at low densities. In our scheme, since only CHs participate in data forwarding, therefore the density of the multihop inter-cluster communicating nodes is significantly lower than the actual sensor node-density in the network. Braided paths incur lesser maintenance overhead than disjoint paths. However, the use of braided paths does not favor en-route filtering scheme with distinct verification keys on the intermediate nodes on a single path. In the braid, an alternate path routes around a single shortest path node. Alternate paths are not independent, and a combination of failures on the shortest path breaks all substitute paths. They are also more prone to pattern failure [35]. Moreover, the repetition of verification keys on different verification nodes increases, and those nodes that possess distinct verification keys are likely to be left out in braided multipaths resulting in a weak en-route filtering capability.

We consider that the number of votes to attach to a report is $s = 5$, and the false votes needed to drop a report is $T_f = 3$. We assume the number of votes required to mark a report safe is $T_t = 4$. The value of T_t can be set to a lower value to save energy. However, a smaller value of T_t implies that an attacker can compromise fewer nodes in a cluster and attach their MACs to the false report. The false report may travel all the way to the BS if T_f is much smaller and the number of votes attached by the compromised nodes is $\geq T_t$. As soon as the value of T_t reaches the threshold, there is no need to further verify the same report even if the remaining votes are false, because they are not sufficient to drop a report. The value of s is chosen such that a minimum of half the cluster member nodes should endorse the report. Choosing a value of $s > 5$ is useless because either of the threshold values T_f or T_t would have been reached before the next verification node verifies the report. If the value of s is chosen to be smaller, the adversaries can easily create counterfeit reports in a few compromised nodes and attach false votes to them. $T_f = 1$ implies that only the FRIA attack is being considered and addressed. If we choose $T_f = 2$, then the attacker can compromise few nodes in the cluster and attach false votes to real reports that are dropped en-route. With a larger cluster size, the condition for picking at least

half the nodes to cast votes may be relaxed, because choosing more nodes to vote implies that more verification nodes are involved in the verification process.

3.6.1. En-Route Filtering

When an intermediate CH receive a report, it will inspect whether all the MACs belong to the same cluster by comparing C_{id} (cluster ID) with $floor\left(\frac{i}{L}\right)$ for each (i, MAC_i) pair in the report. The intermediate node will verify the report if it holds the corresponding key at index i in its memory.

Verification CHs that possess a verification key corresponding to a report use two binary verification sequences to record the verification result, i.e., $vote_v$ and $vote_t$. $vote_v$ records the number of verified votes and $vote_t$ records the number of verified true votes. When the value of $vote_t$ reaches the threshold, the flag Acceptr is set to 1. The report with the flag bit Acceptr set to 1 is not verified again on the path. This saves the remaining verification nodes from investing energy in computing MACs for the verification.

In the worst case, a false report travels all the way to the BS whereas in the best case, the false report is immediately dropped at a node T_f hops away from the e-CH. For a true report, the best case is when it arrives at the BS station with the minimum verifications performed on it. In the worst case, a true report gets dropped just before the BS receives it, with the maximum verification operations performed on it. Equations (3) and (4) give the total amount of energy invested in forwarding the false and the true reports, respectively:

$$E_{false\_report} = \begin{cases} (n+1)(E_{transmit} + E_{receive}) + s{\cdot}E_{verify} \; worst\; case \\ T_f\left(E_{transmit} + E_{receive} + E_{verify}\right) \; best\; case \end{cases}, \tag{3}$$

$$E_{true\_report} = \begin{cases} n{\cdot}\left(E_{transmit} + E_{receive}\right) + s{\cdot}\; E_{verify} \; worst\; case \\ (n+1)\left(E_{transmit} + E_{receive}\right) + T_t{\cdot}E_{verify}\right) \; best\; case \end{cases}, \tag{4}$$

where, n = number of intermediate nodes on the path, $E_{transmit}$, $E_{receive}$, E_{verify} are energies invested in transmitting, receiving and verifying a report, respectively, T_t and T_f are true and false vote threshold values respectively, and s is the number of votes attached to a report.

3.6.2. Input and Output Variables

Our method makes use of fuzzy inferencing carried out at the e-CH and estimates the qualification of the data forwarding paths. Whenever the e-CH chooses to select one among the available paths, it considers the following factors as input variables into the fuzzy module:

1. $NKIH$ = The number of verification keys within h hops from the event cluster where $h = L$;
2. $DIST$ = The hop-count distance ratio between the routing path being considered and the longest path between the e-CH and the BS;
3. REL = The average of the residual energy levels of the nodes on the data forwarding path [31].

$DIST$ is calculated as:

$$DIST = \frac{d_n}{d_L} \times 100\ (\%), \tag{5}$$

where, d_n = the length of the path being considered for selection, d_L = the longest path length between the e-CH and the BS.

$NKIH$ is calculated using:

$$NKIL = n/L, \tag{6}$$

where, n = the number of keys within h hops on the path being considered, L = the cluster size (which is equal to the number of total authentication keys for the same event-reporting cluster).

The authentication keys are shared by the cluster member nodes with the intermediate CHs after they are selected by e-CH. Therefore, the e-CH knows their IDs and distances as it does in [7,11,29].

If the distance between the e-CH and BS is greater than or equal to L the e-CH initially probabilistically chooses the number of verification nodes, which is equal to the size of cluster i.e., L verification nodes [7,11,29], for a given path between the BS and the e-CH. For a distance less than L hops, the e-CH may select an intermediate node to possess more than one authentication key for its cluster. The BS maintains the global key pool and performs the final verification of the reports. Therefore, for a cluster within L hops of the BS, all the verification/authentication keys happen to be within L hops, including those keys on the BS. For a cluster with a distance greater than L hops from the BS, the e-CH chooses L verification nodes on the path probabilistically as it does in [7,11,29].

REL is calculated as in Equation (7):

$$REL = \sum_{i=1}^{n} \frac{E_{residual\,i}}{n} \times 100\ (\%), \tag{7}$$

where, $E_{residual\,i}$ = remaining energy of the intermediate CH_i on the data forwarding path, L = total number of the verification CHs on the path.

The path lengths are known to the CH from the route discovery phase [7,11].

To calculate the energy consumptions of the nodes, the free space model is adopted as presented in [52]:

$$E_{residual} = E_{initial} - E_c(t), \tag{8}$$

$$E_c(t) = E_t + E_r, \tag{9}$$

$$E_t = L \times E_{elec} + L \times \varepsilon_{free-space} \times d^2, \tag{10}$$

$$E_r = L \times E_{elec}, \tag{11}$$

$E_{residual}$ is the residual energy of the node at time t, $E_{initial}$ denotes the initial energy, E_t is the total transmission energy, and E_r represents the energy consumed for data reception. E_{elec} represents the energy required to run the radio electronics of the node. ε_{free_space} is the energy consumption parameter in free-space, L is the data size, and d is the distance between the transmitter and receiver.

The BS acknowledges the receipt of the report and informs the e-CH of the receipt of the report. The e-CH knows the number of the reports generated within the cluster and therefore can deduce the attack ratio as given by [11]:

$$Attack\ ratio = \frac{Reports_{generated} - Reports_{acknowledged}}{Reports_{generated}} \times 100(\%) \tag{12}$$

The output of the fuzzy inference systems is the inferred fitness of the filtering path that is being considered for the selection.

Rule-based fuzzy schemes are a handy instrument for imprecise reasoning when there is some ambiguity in the reasoning process and fuzziness in the information [53,54]. The inference engine uses fuzzy membership functions and fuzzy rules to make a decision. Each node can execute these fuzzy computations, and so the fuzzy inferencing modules can be hardcoded onto small nodes, which can reduce the size of the code [31].

The output variable, i.e., fitness, is defuzzified by using the centroid method or CoA (Center of Area) to produce a crisp value. The weighted average of the fuzzy set A is calculated by the formula given in the following equation.

$$\mathrm{CoA} = \sum_{i=1}^{n} \mu_{\mathrm{A}}(\mathrm{x}_i) \times \mathrm{x}_i / \sum_{i=1}^{n} \mu_{\mathrm{A}}(\mathrm{x}_i), \tag{13}$$

3.6.3. Fuzzy Membership Function

The input and output fuzzy membership functions of the fuzzy inference system are depicted in Figure 2. These membership functions have been carefully chosen based on the simulation

results. The fuzzy set membership functions of every input variable are adjusted to obtain desirable output values.

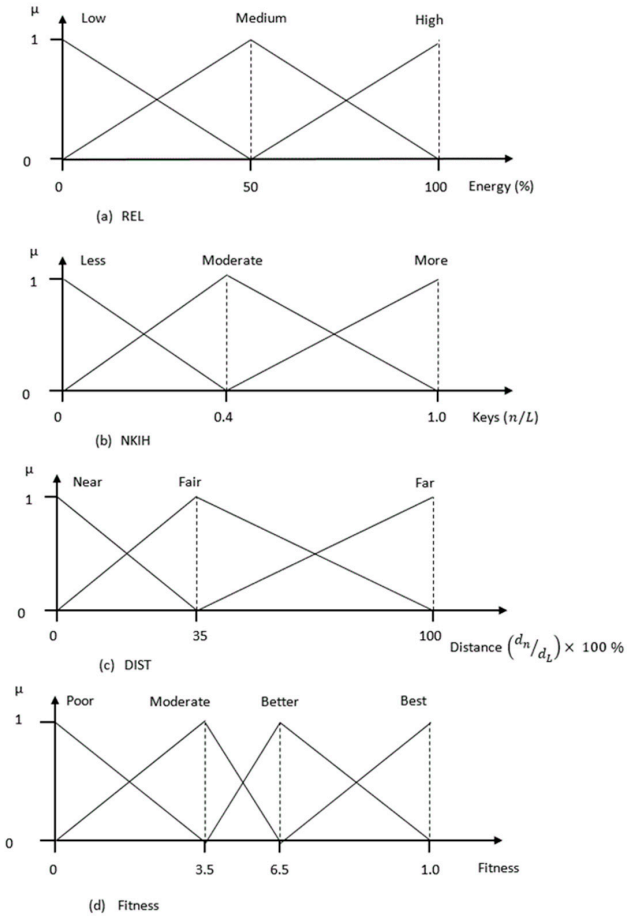

Figure 2. Fuzzy membership functions. (**a**) REL; (**b**) NKIH; (**c**) DIST; (**d**) Fitness.

Each of the 3 input variables have 3 labels; therefore, the rule base of the fuzzy inference system contains 27 ($3 \times 3 \times 3$) rules. Table 1 shows some of the rules.

Table 1. Fuzzy if-then rules.

Rule No.	IF			THEN
	NKIH	**DIST**	**REVN**	**Fitness**
01	Less	Near	Low	Poor
03	Less	Near	High	Moderate
13	Moderate	Fair	Low	Poor
14	Moderate	Fair	Medium	Better
15	Moderate	Fair	High	Best
20	More	Near	Medium	Best
21	More	Near	High	Best
23	More	Fair	Medium	Better
24	More	Fair	High	Best

3.6.4. Fitness Evaluation and Path Selection

An evaluation function computes the qualification of each path. The path with the highest qualification value is chosen as a data forwarding path. The evaluation function $Q(P_i)$ of a given path P_i is given by the following equation:

$$Q(P_i) = F(P_i) - \omega \cdot N(P_i), \quad (14)$$

where, $F(P_i)$ = the fitness value of path P_i calculated through fuzzy inferencing, $N(P_i)$ = the number of all those CHs on path P_i for which: $\frac{CH's\ remaing\ energy}{distance\ between\ e-CH\ and\ BS} \leq Threshold$, and ω = weighting factor.

$F(P_i)$ denotes "fitness", the output of fuzzy inferencing carried out by the e-CH to calculate the fitness of path P_i whose fuzzy membership functions are depicted in Figure 2d. This fitness value is calculated by considering three input parameters for the path P_i: (i) the remaining energy of the CHs on the path P_i, (ii) the number of verification keys within L hops of e-CH on path P_i, and (iii) the distance between the e-CH and the BS via path P_i.

The value of ω ranges between 0 and 1 with both being inclusive. If $\omega = 1$, then e-CH chooses a path which may have a lower fitness value than the highest fitness value but has fewer unhealthy nodes. If $\omega = 0$, then the e-CH selects a path that has the highest fitness value produced by the fuzzy inferencing regardless of the number of unhealthy nodes on the path.

Algorithm 1 presents the routine for route selection. If an intermediate node's energy level drops below a certain value given the distance between the e-CH and the BS, the e-CH marks it and considers the number of such nodes while computing the qualification of the path for routing as in Equation (14). This helps to prevent the hotspot problem from occurring early, and such nodes may be used by report generating clusters that are relatively closer to the BS than the cluster currently represented by e-CH. If there are multiple routes with the same highest qualification value $Q(P)$, then the path with the lowest $N(P)$ value among them is chosen for routing.

Algorithm 1. *PathSelect*(c paths).

1. $P_A = \{P_i \mid P_i$ is a path between the e-CH and the BS and $i = 1, \ldots, c\}$;
2. P_c: current routing path
3. $F(P_i)$: P_i's fitness value calculated through fuzzy inferencing
4. $d_{o,i}$: distance between the e-CH and the BS on path P_i
5. $REL_{j,i}$: residual energy of node n_j on path P_i
6. $N(P_i)$: number of nodes on path P_i with $REL_{j,i}/d_{o,i} \leq Threshold$
7. $Q(P_i)$: P_i's qualification value
8. Δt: time interval
9. **Initialize**;
10. $\omega = 0.5$;
11. Find c paths, choose L verification nodes on each path
12. P_c: = shortest path;
13. **While** (true)
14. use P_c for time duration Δt;
15. **For** each $P_i \in P$ do
16. Calculate $F(P_i)$;
17. Calculate $Q(P_i) = F(P_i) - \omega \cdot N(P_i)$
18. **end for**
19. Choose P_c:= P_i with $Q(P_i) = \max(Q(P_1), \ldots, Q(P_c))$;
20. **if** $Q(P_i) = Q(P_k) = \max(Q(P_1), \ldots, Q(P_c))$ for $i \neq k$ **Then**
21. Choose P_c with highest Q(P) and least N(P) values;
22. **end if**
23. Call *selectVN*(P_c);
24. **end while**

Algorithm 1 makes a call to *selectVN*(P_i) for the selection of intermediate verification nodes on the selected path P_i [11].

In Algorithm 2, the fuzzy inferencing for FASIN produces two outputs [11]:

1. Substitution: the number of current filtering nodes to be substituted with the newly chosen filtering nodes.
2. Exclusion: determination of whether or not to exclude the upstream CHs from being considered as potential filtering nodes in the future whenever their remaining energy is less than a threshold value η.

Algorithm 2. *SelectVN*(P_i).

```
1.  N: set of intermediate nodes likely to be chosen as verification nodes
2.  REL_i: remaining energy of node CH_i
3.  Dist_i: distance of node CH_i from the e-CH
4.  S ⊆ N: set of verification nodes to be substituted
5.  output: output of the fuzzy inferencing for verification node selection
6.  μ: constant
7.  if output = (exclusion ∧ substitution) then
8.     for each CH_i ∈ N do
9.        if REL_i/Dist_i < μ then
10.          N:= N – CH_i;
11.       end if
12.    end for
13. end if
14. if output = (substitution ∨ exclusion) then
15.    for each CH_j ∈ S do
16.       find a CH_i ∈ N such that REL_j/Dist_j < REL_i/Dist_i; and
17.       Choose CH_i as a verification node;
18.    end for
19. end if
```

Apparently, for a particular source cluster, a report that is delivered to the BS via a longer path consumes more energy than if it were delivered on the shortest path. However, the overall energy efficiency of our proposed scheme is due to the fact that, in the absence of adaptive path selection scheme, the shortest path always has heavy traffic. Therefore, nodes on the shortest path are depleted sooner and the shortest path may become unavailable to any source cluster in the future. Load balancing spreads energy utilization across nodes in the network, potentially resulting in longer network lifetime [35].

Since the proximity of the verification nodes improves with the passage of time, it is expected that all the verification nodes on a path P_i will eventually be accommodated within $h = L$ hops. Therefore, the early detection and filtering probability of false reports will improve resulting in energy savings. The probability of a CH_i detecting a false vote is given by:

$$p_{CH_i} = \begin{cases} \frac{s-1}{L} \cdot \frac{d_o - d_i}{L}, \; d_o > d_i \geq (d_o - L) \text{ when all the verification nodes are within L hops} \\ \frac{(s-1)}{L} \cdot \frac{d_i}{d_o}, \text{ when all the verification nodes are probabilistically chosen} \\ \frac{s-1}{L} \cdot \frac{1}{d_o}, \text{ when all the verification nodes are randomly chosen} \end{cases}$$

Here, d_o is the number of hops between the e-CH and the BS and d_i is the number of hops between the CH_i and the BS.

Since the verification nodes are initially chosen probabilistically in our proposed scheme, and their proximity to the e-CH improves with time, the probability of CH_i detecting a false vote is given by $\frac{s-1}{L} \cdot \frac{d_0 - d_i}{L} \geq p_{CH_i} \geq \frac{(s-1)}{L} \cdot \frac{d_i}{d_o}$.

Our proposed scheme uses the Mamdani model for fuzzy inferencing [55]. The time and space complexities of the proposed fuzzy inferencing technique are given by $O\left(n_{rules} \times n_{input\_dim}\right)$ where

n_{rules} is the number of fuzzy if-then rules and n_{input_dim} is the number of input dimensions (fuzzy membership functions) [56].

4. Simulation Results

Simulation experiments have been performed to confirm the efficiency of the proposed method in a custom simulator developed in Microsoft Visual C++ 2012. Table 2 lists the network configurations and parameters. The network is static and contains 4000 nodes in a two dimensional environment delimited in a 1000 × 1000 m^2 area. The number of keys per cluster is $L = 10$, which is equal to the number of nodes in a cluster. We assume $T_f = 3$, $T_t = 4$ and $s = 5$. The number of nodes being compromised increases with time, and consequently the attack intensity also increases. We also assume that intermediate CHs can be compromised and the authentication keys stored on them can be exploited to invalidate a true vote or authenticate a false vote as true during the verification process.

The BS is positioned at the top-left edge of the network and contains information about the sensor node IDs, their locations, and keys. An adversary can only launch FVIA and FRIA attacks using compromised nodes. Due to the cluster based organization of sensor nodes, compromised nodes belonging to different clusters cannot conspire.

Initially, all nodes are fully charged and we assume that no attacks occur in the network organization and initialization phase. CHs are elected, and each CH discovers more than one paths to the BS. The transmission range of the CHs is greater than that of the ordinary nodes. We use Ye et al.'s model of energy consumption [5] in which it consumes 16.25/12.5 µJ per byte to transmit/receive, and 15 µJ to generate a vote. Each report is 36 bytes and each MAC is 4 bytes in size. It takes a verification node 75 µJ of energy to verify a report [48,57].

Table 2. Simulation configuration.

Parameters	Value
Number of nodes	4000
Network size	1000 × 1000 m^2
Base station location	0 × 0 m
Number of clusters	400
Nodes in a cluster (cluster size)	10
Votes required to endorse a report (s)	5
False votes threshold to drop a report (T_f)	3
Votes required to accept a report as true report (T_t)	4
Size of report	36 bytes
Size of MAC	4 bytes
Energy consumed to:	
Generate a MAC	15 µJ
Verify a report	75 µJ
Transmit/receive a byte	16.25/12.5 µJ

As illustrated in Figure 3, our suggested method's capability to detect false votes early during the process of verification is better than that of the PVFS. The value of h (the number of hops from the e-CH) significantly affects the false vote detection probability of the path. Choosing $h \geq L$ still produces better performance. If h is increased such that $h \geq d_i$ for every $i = 1, 2, \ldots, c$ (d_i = hop count distance of ith path), then the performance of the proposed scheme gets slightly closer to and compares well with that of FASIN in the early detection of false votes. However, the closer the value of h is to L, the better the performance is in terms of detecting false votes earlier.

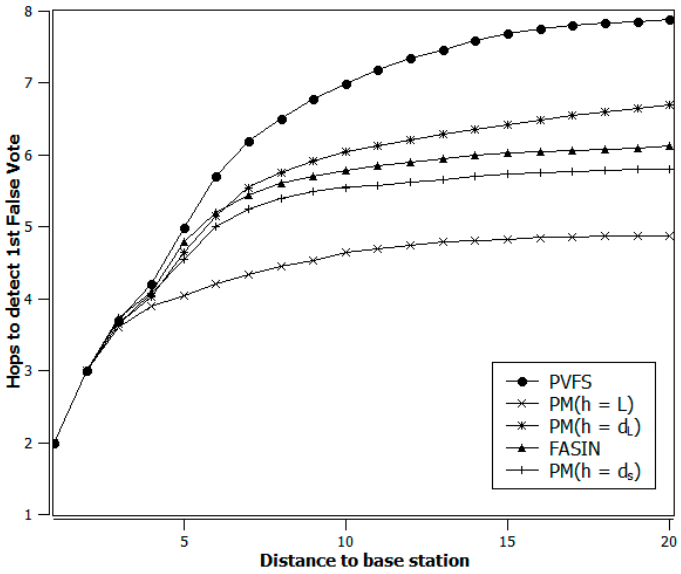

Figure 3. Filtering performance analysis.

Figure 4 provides a comparative investigation of the performance of the PVFS, FASIN, and fuzzy-based path selection with and without FASIN in terms of the percentage of packet drops at each hop. It is obvious from the figure that the performance of the scheme in Figure 4d is far better than the performance of the schemes in Figure 4a,b. There are some similarities in the false-packet filtering behavior depicted in Figure 4c,d. However, close observation reveals that the scheme in Figure 4d achieves superior performance compared to that in Figure 4c by filtering more false reports as early as possible. This improvement in early filtering performance is due to the fact that:

- The data forwarding path is being selected dynamically, and
- The intermediate verification nodes' average distance from the e-CH improves over time based on security threat and energy status.

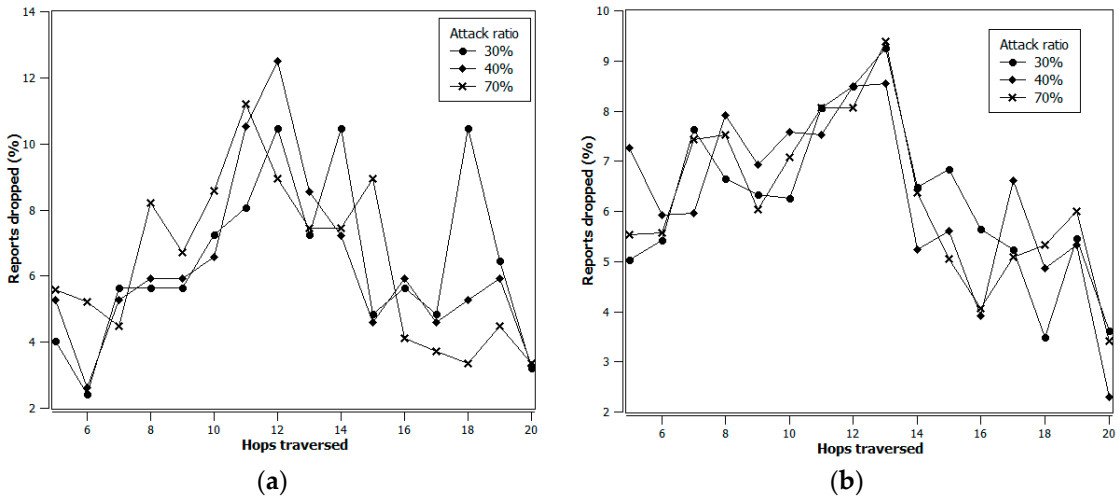

Figure 4. *Cont.*

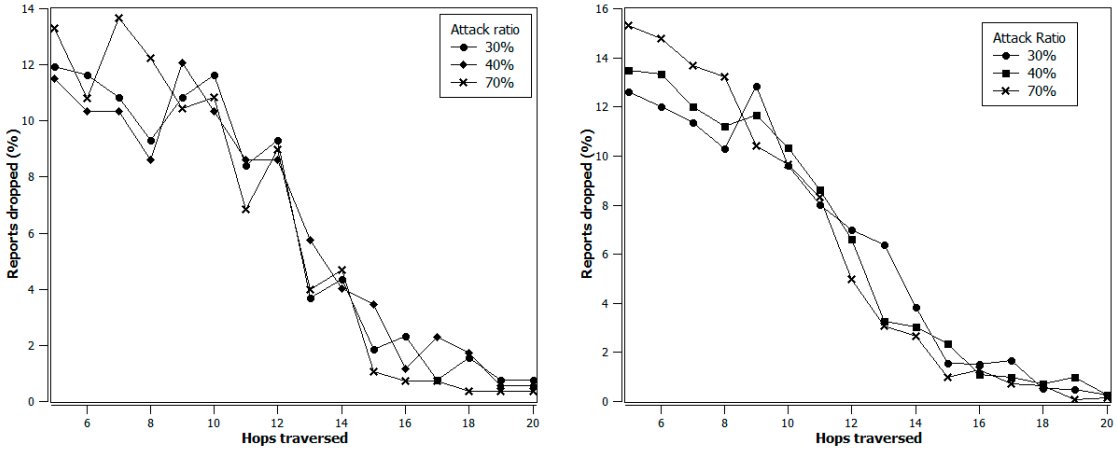

Figure 4. Fabricated packets dropped en-route. (**a**) Probabilistic voting-based filtering scheme (PVFS); (**b**) Fuzzy adaptive path selection without FASIN; (**c**) Fuzzy adaptive selection of verification nods (FASIN); (**d**) Proposed method.

Figure 5 shows that, as long as the value of T_f is same for all three schemes, i.e., PVFS, FASIN, and our proposed method, the early filtering of fabricated reports is better in our proposed scheme as the distance between the e-CH and the BS increases. It is extremely desirable to relieve the nodes closer to the BS of the task of verification because they contain more verification keys and must perform verification more often. All those reports for which T_t reaches its threshold value are marked safe and are not verified any further, which eases the task of the verification nodes located farther from the e-CH.

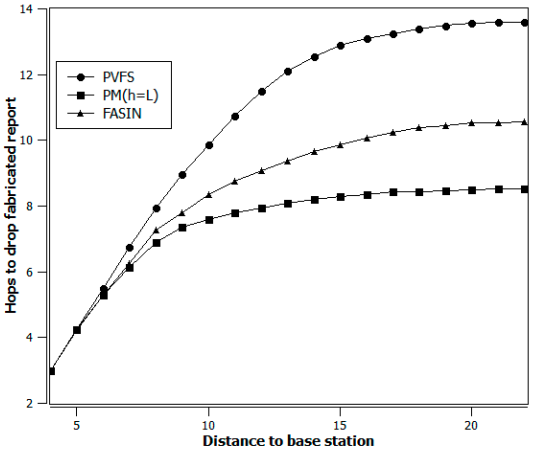

Figure 5. Fabricated report filtering performance.

Figure 6 shows the average energy consumed in forwarding a false report before it is filtered by an intermediate verification node. Initially, when there is no false data insertion in the network and the sensor nodes are fully charged, the proposed scheme chooses the shortest path for data delivery like in PVFS and FASIN. However, the false data rate increases, and the energy of the intermediate nodes decreases with the passage of time. In the proposed scheme, the e-CH makes fuzzy inferencing about the path selection and chooses the best path given the current insertion rate of false data and the status of the energy of the intermediate paths. Whereas, FASIN and FVPS switch to the next shortest path for data delivery only after a node failure occurs on the first shortest path. Whenever FASIN switches to the available next shortest path, the verification nodes on that path are sparsely located due to initial

probabilistic selection of the verification nodes. Therefore, false reports travel more hops than in the proposed method and consume more energy until FASIN improves the proximity of the verification node to the e-CH. In PVFS, verification nodes are probabilistically chosen only once and they remain fixed. The bend in the curve for PVFS in Figure 6 is due to the fact that verification nodes that are near the source cluster have a higher probability of being chosen as verification nodes on a longer path than on the shortest path as provided by Equation (2).

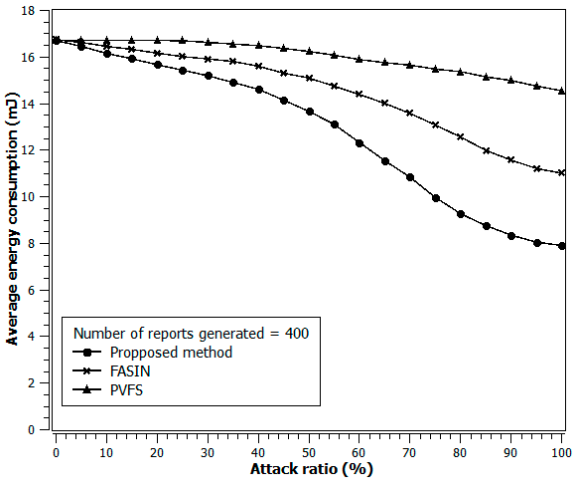

Figure 6. Average energy consumption per false report.

Intermediate nodes may ultimately die earlier than other nodes because they are constantly involved in data forwarding as extra energy is expended during data receiving, verifying and forwarding. Consequently, loss of information inevitably occurs due to the fact that information generated by event reporting clusters cannot reach the BS because one or more of the intermediate nodes on the data routing paths have been depleted and have gone dead.

Figure 7a,b show information delivery and loss analysis of PVFS, FASIN, and the proposed method. The proposed method outperforms PVFS as well as FASIN with the improved information delivery and reduced information loss. It is very obvious from Figure 7a,b that our proposed method helps to prevent energy depletion in the intermediate nodes, hence the network operational time is increased.

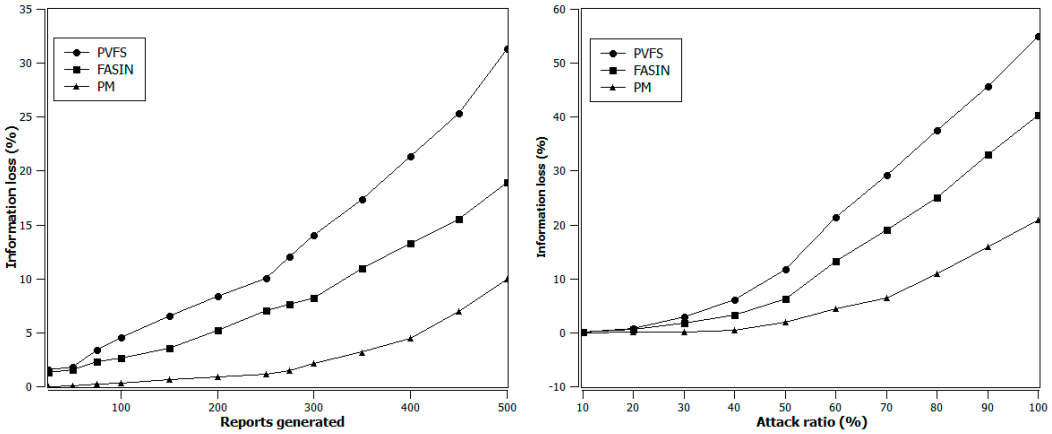

Figure 7. Information delivery and loss analysis. (**a**) Attack ratio = 60%; (**b**) Number of reports generated = 400.

5. Related Work

Techniques in computational intelligence (like heuristic methods) may be used to normalize the multifaceted, nonlinear dynamic behavior of a WSN. Such a heuristic method implements a "rule-of-thumb" function for a proposed scheme, and, consequently, it lacks structure and simplicity. Our proposed scheme is based on fuzzy set theory. Unlike most artificial intelligence techniques, fuzzy set theory calculates decisions in time commensurate with the needs of the WSN. This method is chiefly applied to control WSN behaviors that are difficult to model. Many of the queries regarding this method arise chiefly from its non-numerical nature, i.e., the stability of the whole scheme, completeness of the rules, etc. But, mathematical and analytical approaches have their own restrictions, whereas this study shows that heuristic programs may practically and effortlessly be used in WSNs to achieve desirable behavior in the WSN.

Recently, computational intelligence techniques have been used to resolve issues relating to security, data aggregation and filtering, network deployment, location awareness optimization, and data routing in WSNs. Use of fuzzy logic is appropriate for achieving near optimization in data routing in cluster-based WSNs. Fuzzy logic is the ultimate choice for the optimization of filtering node and data-forwarding route selection due to the simplicity of the fuzzy membership functions [11,48].

PVFS is a probabilistic data filtering scheme that has the objective of countering FRIA and FVIA attacks and achieving energy savings. PVFS lacks the ability to protect against compromised filtering nodes and does not utilize acknowledgements of the dropped reports [7]. In LEACH, cluster heads are chosen probabilistically; however, LEACH does not give importance to the distribution density of the sensor nodes and their energy status during their selection [58]. In the dynamic en-route filtering scheme (DEF) [12], filtering nodes are selected randomly within a predetermined hop count distance from the event detecting node before the routing paths are setup. Therefore, in DEF, the filtering power is mainly defined by the selection of the routing paths. It is very likely that some of the paths will not filter a single false report during data forwarding. Moreover, DEF does not counter the FVIA attack. FPMS in [31] was recommended to improve the detection of false data for the routing paths in the DEF based network. However, the performance is still determined by the number of filtering nodes on a path along with other elements, like the distance and energy of the nodes on the path. Since the number of filtering nodes on every path is not equal, it is very likely that favored paths may emerge and a path consisting of more filtering nodes is used excessively. Another anomaly is that a shorter path with a higher energy level may still not be able to filter out false reports since it has fewer filtering nodes. In SEF [5], verification nodes on the routing path authenticate the MACs of the report with a probability. However, SEF performs well when no more than a few nodes are compromised and the limited filtering capability of SEF is mainly decided by the detection probability of each node. The detection power of SEF is significantly influenced by the choice of the upstream routes. Several efforts have been made in an attempt to enhance the filtering and energy efficiency of SEF. Sun et al. [59] suggested a path selection method that aims at refining the detection capability of SEF by considering the number of filtering keys on the upstream paths and their distances. However, in the presence of heavy traffic, favorite paths for data routing may emerge with heavy traffic on them, and the selection of paths leads to uneven energy consumption. The path renewal method in [60] proposes that an intermediate node on a routing path can renew its upstream path by considering its energy status and the degree of communication traffic it receives. However, such a path renewal proposition is based on the hypothesis that any super-node having a greater number of sub-nodes consumes more energy than measuring the actual amount of data it receives and forwards. Moreover, the message overhead involved in the flooding of energy messages, eviction messages, and fare messages is greater.

Multi path routing was proposed to defend against selective forwarding attacks. But multipath routing entails increased communication overhead by the increased number of data forwarding paths. Lee et al. presented a fuzzy-based reliable date delivery method in which the number of paths is controlled by the fuzzy consideration of the network energy level and the number of malicious nodes. However, this method supposes that the number of malicious nodes are known beforehand, to select the number of paths for data delivery, and it relies on the message flooding technique to inform the

source nodes of the new paths formed after periodic inspection by the base station [61]. In CCEF [15], information between the BS and the HS is sent bidirectionally on the same path, rendering the CCEF scheme impractical for dynamic topologies. A false report can always make it to the BS because only session keys experience en-route verification in the presence of the malicious nodes. Furthermore, CCEF uses a public key algorithm for the commutative ciphering, which is unsuitable for sensor networks because of the limited energy resources and restricted computational capabilities of the sensor nodes [17]. STEF proposed in [14] counters FRIA and path-based DoS attacks while leaving out FVIA attacks, which causes dropping of true data en-route to the BS. In IHA [16], the BS can detect and filter false reports till the number of compromised nodes is below a certain value. Hybrid energy efficient distributed clustering (HEED) in [62] was proposed to periodically select CHs considering their energies; however, this approach causes substantial overhead in the network resulting in a decreased network lifetime and needless energy depletion [48]. In BCDCP [34], every CH has to disseminate specific information, including information about the location and energy, to the BS in every round, which requires a vast amount of energy due to the heavy communication in the network. TICK proposed in [13] is a dynamic key and en-routing filtering scheme that does not require transmission of the explicit keying messages needed to avoid old keys. Rather, it uses the node's local time values as a one-time dynamic encryption key. However, the scheme relies heavily on the intermediate nodes' capability to accurately compute the keys to verify the report, and the size of the tick window used depends on the data dissemination rate. The larger the size of the tick window, the more time it will take to compute the correct key. A distant forwarding node may classify a healthy incoming report as false. SOBAS [63] proposed an improvement in TICK by introducing a selective re-encryption mode. SOBAS requires intermediate nodes to accurately calculate the key used by the node that generated the report. However, to correctly compute the key, SOBAS assumes that the distance between the two nodes is small enough to find the correct key in the time window. SPINS [64] requires a time synchronization practice, packet storage time, and delay in key disclosure at each node, which result in data transmission delays [48].

Several energy aware centralized routing protocols have been proposed wherein the BS keeps track of the energy dissipation rate in sensor nodes and periodically selects cluster heads and routing paths [33,34,65,66]. In all these schemes, the BS updates the energy status of the nodes in the network based on the amount of data generated and received. Requiring the sensor nodes in the network to share their actual energy status in a multi-hop communication model accelerates the hotspot problem near the BS. The energy invested in computations is negligible compared to that invested during transmission and reception of data.

In some proposed schemes, intermediate data forwarding nodes can aggregate the inward packets from different clusters together with its own packet. The correlation degree of data coming from different clusters is very low and makes data aggregation at intermediate relay nodes an impractical option [25].

Authors in [67] investigated the problem of constructing virtual backbones to increase the network life and proposed a distributed algorithm to find a backbone in the dual-radio network whenever a new backbone is needed. Similarly, a virtual backbone construction heuristic for maximizing the network lifetime in dual-radio based WSNs was proposed in [68]. In both the previously mentioned studies, sensor nodes are supposed to be equipped with two radio interfaces: short range and long range radios. We investigate the energy efficiency problem in WSNs from an entirely different perspective wherein the security against the previously mentioned attacks, the filtering strength of the available intermediate routes (for example: the number of verification keys on the path and the average distance of verification nodes from the data generating cluster) are considered while developing a heuristic based solution for path selection that provides the maximum security against the FVIA and FRIA attacks at the same time. In contrast, the solutions proposed in [67,68] are scheduling strategies that consider the energy power of the nodes while constructing a backbone. One of the important pieces of information required in choosing the energy efficient path for data dissemination is the number of upstream nodes holding the corresponding data authentication keys and their distances from the data generating cluster.

Authors of [69] studied and investigated the scheduling of virtual data aggregation trees to try to maximize the network life. An en-route filtering scheme requires that the data being verified by the intermediate verification nodes be generated by the nodes whose corresponding authentication keys are shared with the intermediate verification nodes. In en-route filtering schemes, only data that is temporally and spatially correlated can be aggregated making authenticity verification at the intermediate verification nodes easy and convenient [29]. Moreover, input information related to the verification keys at the intermediate verification nodes is important in determining a suitable path for information delivery, and those keys need to be distinct to avoid repeated authentications. A virtual data aggregation tree may include more nodes possessing the same verification key, which makes them verify the same report several times against a single authentication key.

6. Conclusions

In WSNs, sensor nodes suffer from limited energy and computational restrictions. Numerous data routing and dynamic or static data filtering schemes have been proposed that try to conserve energy in WSNs. Data routing protocols aimed at saving energy are always proposed without considering security measures. Moreover, data filtering schemes are proposed without regard to energy efficient data routing, wherein most of the underlying routing protocols are a low-cost-first routing protocol.

In multi-hop communication, nodes near the BS are burdened with relaying heavy data to the BS, which creates a hotspot problem. Rapid depletion of energy at nodes around the BS eventually disconnect the rest of the network from the BS. Injection of false votes by adversaries have objective of depleting the energy of the intermediate nodes that relay the data generated by the source cluster. Similarly, legitimate data endorsed by compromised nodes with bogus MACs get dropped en-route during the en-route filtering that blocks true information from reaching the BS. Constant use of a minimum cost path results in depletion of energy at nodes on the same path and causes network partitioning. To tackle the unbalanced energy consumption problem in WSN from a data routing and filtering scheme perspective, we proposed a fuzzy rule-based data route and intermediate filtering node selection scheme. Our scheme allows cluster member nodes to generate authentication keys afresh and share these with the intermediate verification nodes. The proposed method demonstrates effective and actual efficiency in the WSNs, and extends the network lifetime while improving filtering capacity due to dynamic authentication key sharing. The data forwarding is carried out along the suitable path chosen in response to input factors such as energy status, the number of verification keys within a certain number of hops on the path, and the distance between the cluster and the base station. Dynamic intermediate verification node selection helps to improve the fitness of the route by improving the proximity of verification nodes to the event cluster. Compromised intermediate verification nodes no longer possess network-life-long authentication keys for report verification, which results in improved detection and filtering of false reports.

Acknowledgments: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. NRF-2015R1D1A1A01059484).

Author Contributions: Muhammad Akram and Tae Ho Cho conceived and designed the study. Muhammad Akram performed the simulations, analysis, and wrote the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Han, X.; Cao, X.; Lloyd, E.L.; Shen, C. Fault-Tolerant Relay Node Placement in Heterogeneous Wireless Sensor Networks. *IEEE Trans. Mob. Comput.* **2010**, *9*, 643–656.
2. Xu, J.; Zhou, X.; Han, J.; Li, F.; Zhou, F. Data Authentication Model Based on Reed-Solomon Error-Correcting Codes in Wireless Sensor Networks. *IETE Tech. Rev.* **2013**, *30*, 191–199. [CrossRef]
3. Ding, C.; Yang, L.; Wu, M. Localization-Free Detection of Replica Node Attacks in Wireless Sensor Networks using Similarity Estimation with Group Deployment Knowledge. *Sensors* **2017**, *17*, 160. [CrossRef] [PubMed]

4. Ye, F.; Chen, J.; Li, Y. Improvement of DS Evidence Theory for Multi-Sensor Conflicting Information. *Symmetry* **2017**, *9*, 69.
5. Ye, F.; Luo, H.; Lu, S.; Zhang, L. Statistical En-Route Filtering of Injected False Data in Sensor Networks. *IEEE J. Sel. Areas Commun.* **2005**, *23*, 839–850.
6. Lee, H.Y.; Cho, T.H. Fuzzy Adaptive Selection of Filtering Schemes for Energy Saving in Sensor Networks. *IEICE Trans. Commun.* **2007**, *90*, 3346–3353. [CrossRef]
7. Li, F.; Srinivasan, A.; Wu, J. PVFS: A Probabilistic Voting-Based Filtering Scheme in Wireless Sensor Networks. *Int. J. Secur. Netw.* **2008**, *3*, 173–182. [CrossRef]
8. Hu, Y.; Perrig, A.; Johnson, D.B. Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks. In Proceedings of the INFOCOM 2003, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, CA, USA, 30 March–3 April 2003; pp. 1976–1986.
9. Misra, S.; Das, S.; Obaidat, M. Context-Aware Quality of Service in Wireless Sensor Networks. *IEEE Commun. Mag.* **2014**, *52*, 16–23. [CrossRef]
10. Wang, Z.; Zeng, P.; Zhou, M.; Li, D.; Wang, J. Cluster-Based Maximum Consensus Time Synchronization for Industrial Wireless Sensor Networks. *Sensors* **2017**, *17*, 141. [CrossRef] [PubMed]
11. Akram, M.; Cho, T.H. Energy Efficient Fuzzy Adaptive Selection of Verification Nodes in Wireless Sensor Networks. *Ad Hoc Netw.* **2016**, *47*, 16–25. [CrossRef]
12. Yu, Z.; Guan, Y. A Dynamic En-Route Scheme for Filtering False Data Injection in Wireless Sensor Networks. In Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM 2006), Barcelona, Spain, 23–29 April 2006; pp. 294–295.
13. Uluagac, A.S.; Beyah, R.A.; Copeland, J.A. TIme-Based Dynamic Keying and En-Route Filtering (TICK) for Wireless Sensor Networks. In Proceedings of the 2010 IEEE in Global Telecommunications Conference (GLOBECOM 2010), Miami, FL, USA, 6–10 December 2010; pp. 1–6.
14. Kraub, C.; Schneider, M.; Bayarou, K.; Eckert, C. STEF: A Secure Ticket-Based En-Route Filtering Scheme for Wireless Sensor Networks. In Proceedings of the Second International Conference on Availability, Reliability and Security, Washington, DC, USA, 10–13 April 2007; pp. 310–317.
15. Yang, H.; Lu, S. Commutative Cipher Based En-Route Filtering in Wireless Sensor Networks. In Proceedings of the IEEE 60th Vehicular Technology Conference, VTC2004-Fall, Los Angeles, CA, USA, 26–29 September 2004; pp. 1223–1227.
16. Zhu, S.; Setia, S.; Jajodia, S.; Ning, P. An Interleaved Hop-by-Hop Authentication Scheme for Filtering of Injected False Data in Sensor Networks. In Proceedings of the 2004 IEEE Symposium on Security and Privacy, Berkeley, CA, USA, 12 May 2004; pp. 259–271.
17. Eschenauer, L.; Gligor, V.D. A Key-Management Scheme for Distributed Sensor Networks. In Proceedings of the 9th ACM Conference on Computer and Communications Security, Washington, DC, USA, 18–22 November 2002; pp. 41–47.
18. Eltoweissy, M.; Moharrum, M.; Mukkamala, R. Dynamic Key Management in Sensor Networks. *IEEE Commun. Mag.* **2006**, *44*, 122–130. [CrossRef]
19. Yang, H.; Ye, F.; Yuan, Y.; Lu, S.; Arbaugh, W. Toward Resilient Security in Wireless Sensor Networks. In Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing, Urbana-Champaign, IL, USA, 25–27 May 2005; pp. 34–45.
20. Lee, H.Y.; Cho, T.H. Key Inheritance-Based False Data Filtering Scheme in Wireless Sensor Networks. In Proceedings of the International Conference on Distributed Computing and Internet Technology, Bhubaneswar, India, 20–23 December 2006; pp. 116–127.
21. Kim, M.S.; Cho, T.H. A Multipath En-Route Filtering Method for Dropping Reports in Sensor Networks. *IEICE Trans. Inf. Syst.* **2007**, *90*, 2108–2109. [CrossRef]
22. Lee, H.; Cho, T. False Negative-Resilient Report Generation for the Statistical Filtering in Sensor Networks. In Proceedings of the International Conference on Network and Mobile Computing (NMC'06), Negeri Sembilan, Malaysia, 28–29 August 2006; p. 27.
23. Lee, H.Y.; Cho, T.H. A Scheme for Adaptively Countering Application Layer Security Attacks in Wireless Sensor Networks. *IEICE Trans. Commun.* **2010**, *93*, 1881–1889. [CrossRef]
24. Wei, C.; Yang, J.; Gao, Y.; Zhang, Z. Cluster-Based Routing Protocols in Wireless Sensor Networks: A Survey. In Proceedings of the 2011 International Conference On Computer Science and Network Technology (ICCSNT), Harbin, China, 24–26 December 2011; pp. 1659–1663.

25. Chen, G.; Li, C.; Ye, M.; Wu, J. An Unequal Cluster-Based Routing Protocol in Wireless Sensor Networks. *Wirel. Netw.* **2009**, *15*, 193–207. [CrossRef]
26. Safa, H.; Artail, H.; Tabet, D. A Cluster-Based Trust-Aware Routing Protocol for Mobile Ad Hoc Networks. *Wirel. Netw.* **2010**, *16*, 969–984. [CrossRef]
27. Singh, S.K.; Kumar, P.; Singh, J.P. A Survey on Successors of LEACH Protocol. *IEEE Access* **2017**, *5*, 4298–4328. [CrossRef]
28. Ray, A.; De, D. Energy Efficient Clustering Protocol Based on K-Means (EECPK-Means)-Midpoint Algorithm for Enhanced Network Lifetime in Wireless Sensor Network. *IET Wirel. Sens. Syst.* **2016**, *6*, 181–191. [CrossRef]
29. Nam, S.M.; Cho, T.H. Context-Aware Architecture for Probabilistic Voting-Based Filtering Scheme in Sensor Networks. *IEEE Trans. Mob. Comput.* **2017**, *16*, 2751–2763. [CrossRef]
30. Zhang, L.; Yin, N.; Fu, X.; Lin, Q.; Wang, R. A Multi-Attribute Pheromone Ant Secure Routing Algorithm Based on Reputation Value for Sensor Networks. *Sensors* **2017**, *17*, 541. [CrossRef] [PubMed]
31. Lee, H.Y.; Cho, T.H. Fuzzy-Based Path Selection Method for Improving the Detection of False Reports in Sensor Networks. *IEICE Trans. Inf. Syst.* **2009**, *92*, 1574–1576. [CrossRef]
32. Farooq, M.O.; Dogar, A.B.; Shah, G.A. MR-LEACH: Multi-Hop Routing with Low Energy Adaptive Clustering Hierarchy. In Proceedings of the 2010 Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM), Venice, Italy, 18–25 July 2010; pp. 262–268.
33. Heinzelman, W.B.; Chandrakasan, A.P.; Balakrishnan, H. An Application-Specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Trans. Wirel. Commun.* **2002**, *1*, 660–670. [CrossRef]
34. Muruganathan, S.D.; Ma, D.C.; Bhasin, R.I.; Fapojuwo, A.O. A Centralized Energy-Efficient Routing Protocol for Wireless Sensor Networks. *IEEE Commun. Mag.* **2005**, *43*, S8–S13. [CrossRef]
35. Ganesan, D.; Govindan, R.; Shenker, S.; Estrin, D. Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* **2001**, *5*, 11–25. [CrossRef]
36. Coppersmith, D.; Jakobsson, M. Almost Optimal Hash Sequence Traversal. In Proceedings of the International Conference on Financial Cryptography, Bermuda, UK, 11–14 March 2002; pp. 102–119.
37. Perrig, A.; Canetti, R.; Song, D.; Tygar, J.D. Efficient and Secure Source Authentication for Multicast. In Proceedings of the Network and Distributed System Security Symposium, NDSS, San Diego, CA, USA, 7–9 February 2001; pp. 35–46.
38. Liu, C.; Cao, G. Spatial-Temporal Coverage Optimization in Wireless Sensor Networks. *IEEE Trans. Mob. Comput.* **2011**, *10*, 465–478. [CrossRef]
39. Du, W.; Deng, J.; Han, Y.S.; Varshney, P.K.; Katz, J.; Khalili, A. A Pairwise Key Predistribution Scheme for Wireless Sensor Networks. *ACM Trans. Inf. Syst. Secur.* **2005**, *8*, 228–258. [CrossRef]
40. Blom, R. An Optimal Class of Symmetric Key Generation Systems. In Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques, Paris, France, 9–11 April 1984; pp. 335–338.
41. Blundo, C.; De Santis, A.; Herzberg, A.; Kutten, S.; Vaccaro, U.; Yung, M. Perfectly-Secure Key Distribution for Dynamic Conferences. In Proceedings of the Annual International Cryptology Conference, Santa Barbara, CA, USA, 16–20 August 1992; pp. 471–486.
42. Liu, D.; Ning, P. Location-Based Pairwise Key Establishments for Static Sensor Networks. In Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks, Washington, DC, USA, 27–30 October 2003; pp. 72–82.
43. Liu, D.; Ning, P.; Li, R. Establishing Pairwise Keys in Distributed Sensor Networks. *ACM Trans. Inf. Syst. Secur.* **2005**, *8*, 41–77. [CrossRef]
44. Ah Kioon, M.C.; Wang, Z.S.; Deb Das, S. Security Analysis of Md5 Algorithm in Password Storage. *Appl. Mech. Mater.* **2013**, *347–350*, 2706–2711. [CrossRef]
45. Lu, R.; Lin, X.; Zhu, H.; Liang, X.; Shen, X. BECAN: A Bandwidth-Efficient Cooperative Authentication Scheme for Filtering Injected False Data in Wireless Sensor Networks. *IEEE Trans. Parallel Distrib. Syst.* **2012**, *23*, 32–43.
46. Johnson, D.B. A Note on Dijkstra's Shortest Path Algorithm. *J. ACM* **1973**, *20*, 385–388. [CrossRef]
47. Yang, X.; Lin, J.; Yu, W.; Moulema, P.; Fu, X.; Zhao, W. A Novel En-Route Filtering Scheme Against False Data Injection Attacks in Cyber-Physical Networked Systems. *IEEE Trans. Comput.* **2015**, *64*, 4–18. [CrossRef]
48. Nam, S.M.; Cho, T.H. A Fuzzy Rule-Based Path Configuration Method for LEAP in Sensor Networks. *Ad Hoc Netw.* **2015**, *31*, 63–79. [CrossRef]
49. Al-Riyami, A.; Zhang, N.; Keane, J. An Adaptive Early Node Compromise Detection Scheme for Hierarchical WSNs. *IEEE Access* **2016**, *4*, 4183–4206. [CrossRef]

50. Ye, F.; Chen, A.; Lu, S.; Zhang, L. A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks. In Proceedings of the Tenth International Conference on Computer Communications and Networks, Scottsdale, AZ, USA, 15–17 October 2001; pp. 304–309.
51. Al-Karaki, J.N.; Kamal, A.E. Routing Techniques in Wireless Sensor Networks: A Survey. *IEEE Wirel. Commun.* **2004**, *11*, 6–28. [CrossRef]
52. Lee, G.; Kong, J.; Lee, M.; Byeon, O. A Cluster-Based Energy-Efficient Routing Protocol without Location Information for Sensor Networks. *J. Inf. Process. Syst.* **2005**, *1*, 49–54. [CrossRef]
53. Serrano, N.; Seraji, H. Landing Site Selection using Fuzzy Rule-Based Reasoning. In Proceedings of the 2007 IEEE International Conference On Robotics and Automation, Roma, Italy, 10–14 April 2007; pp. 4899–4904.
54. Jelušič, P.; Žlender, B. Discrete Optimization with Fuzzy Constraints. *Symmetry* **2017**, *9*, 87. [CrossRef]
55. Lee, K.H. *First Course on Fuzzy Theory and Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006.
56. Balázs, K.; Kóczy, L.T.; Botzheim, J. Comparison of Fuzzy Rule-Based Learning and Inference Systems. In Proceedings of the 9th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics, CINTI, Budapest, Hungary, 5–8 November 2008; pp. 61–75.
57. Moon, S.Y.; Cho, T.H. Key Index-Based Routing for Filtering False Event Reports in Wireless Sensor Networks. *IEICE Trans. Commun.* **2012**, *95*, 2807–2814. [CrossRef]
58. Heinzelman, W.R.; Chandrakasan, A.; Balakrishnan, H. Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 7 January 2000; Volume 2, p. 10.
59. Sun, C.I.; Lee, H.Y.; Cho, T.H. A Path Selection Method for Improving the Detection Power of Statistical Filtering in Sensor Networks. *J. Inf. Sci. Eng.* **2009**, *25*, 1163–1175.
60. Kim, J.M.; Han, Y.S.; Lee, H.Y.; Cho, T.H. Path Renewal Method in Filtering Based Wireless Sensor Networks. *Sensors* **2011**, *11*, 1396–1404. [CrossRef] [PubMed]
61. Lee, H.; Cho, T. Fuzzy-Based Reliable Data Delivery for Countering Selective Forwarding in Sensor Networks. In Proceedings of the 4th international conference on Ubiquitous Intelligence and Computing, Hong Kong, China, 11–13 July 2007; pp. 535–544.
62. Younis, O.; Fahmy, S. HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks. *IEEE Trans. Mob. Comput.* **2004**, *3*, 366–379. [CrossRef]
63. Uluagac, A.S.; Beyah, R.A.; Copeland, J.A. Secure Source-Based Loose Synchronization (SOBAS) for Wireless Sensor Networks. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *24*, 803–813. [CrossRef]
64. Perrig, A.; Szewczyk, R.; Tygar, J.D.; Wen, V.; Culler, D.E. SPINS: Security Protocols for Sensor Networks. *Wirel. Netw.* **2002**, *8*, 521–534. [CrossRef]
65. Nasser, N.; Chen, Y. SEEM: Secure and Energy-Efficient Multipath Routing Protocol for Wireless Sensor Networks. *Comput. Commun.* **2007**, *30*, 2401–2412. [CrossRef]
66. Chen, Y.; Nasser, N. Energy-Balancing Multipath Routing Protocol for Wireless Sensor Networks. In Proceedings of the 3rd International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, Waterloo, ON, Canada, 7–9 August 2006; p. 21.
67. Liu, B.; Pham, V.; Nguyen, N. An Efficient Algorithm of Constructing Virtual Backbone Scheduling for Maximizing the Lifetime of Dual-Radio Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 475159. [CrossRef]
68. Liu, B.; Pham, V.; Nguyen, N. A Virtual Backbone Construction Heuristic for Maximizing the Lifetime of Dual-Radio Wireless Sensor Networks. In Proceedings of the 2015 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), Adelaide, Australia, 23–25 September 2015; pp. 64–67.
69. Nguyen, N.; Liu, B.; Pham, V.; Luo, Y. On Maximizing the Lifetime for Data Aggregation in Wireless Sensor Networks using Virtual Data Aggregation Trees. *Comput. Netw.* **2016**, *105*, 99–110. [CrossRef]