Journal of
Sensor and Actuator Networks

MDPI

Article

Resistance to Cybersecurity Attacks in a Novel Network for Autonomous Vehicles

Callum Brocklehurst and Milena Radenkovic *

School of Computer Science, The University of Nottingham, Nottingham NG8 1BB, UK; psycb5@nottingham.ac.uk
* Correspondence: milena.radenkovic@nottingham.ac.uk

Abstract: The increased interest in autonomous vehicles has led to the development of novel networking protocols in VANETs In such a widespread safety-critical application, security is paramount to the implementation of the networks. We view new autonomous vehicle edge networks as opportunistic networks that bridge the gap between fully distributed vehicular networks based on short-range vehicle-to-vehicle communication and cellular-based infrastructure for centralized solutions. Experiments are conducted using opportunistic networking protocols to provide data to autonomous trams and buses in a smart city. Attacking vehicles enter the city aiming to disrupt the network to cause harm to the general public. In the experiments the number of vehicles and the attack length is altered to investigate the impact on the network and vehicles. Considering different measures of success as well as computation expense, measurements are taken from all nodes in the network across different lengths of attack. The data gathered from each node allow exploration into how different attacks impact metrics including the delivery probability of a message, the time taken to deliver and the computation expense to each node. The novel multidimensional analysis including geospatial elements provides evidence that the state-of-the-art MaxProp algorithm outperforms the benchmark as well as other, more complex routing protocols in most of the categories. Upon the introduction of attacking nodes however, PRoPHET provides the most reliable delivery probability when under attack. Two different attack methods (black and grey holes) are used to disrupt the flow of messages throughout the network and the more basic protocols show that they are less consistent. In some metrics, the PRoPHET algorithm performs better when under attack due to the benefit of reduced network traffic.

Keywords: VANETS; opportunistic networks; security

Citation: Brocklehurst, C.; Radenkovic, M. Resistance to Cybersecurity Attacks in a Novel Network for Autonomous Vehicles. *J. Sens. Actuator Netw.* **2022**, *11*, 35. https://doi.org/10.3390/jsan11030035

Academic Editor: Lei Shu

Received: 7 June 2022
Accepted: 11 July 2022
Published: 13 July 2022

1. Introduction

Vehicular edge computing is the integration of emerging mobile edge computing with traditional vehicular networks, and it aims to move communication, computing, and caching resources to close proximity of vehicular users to support V2V, V2I and V2X communications. Autonomous Vehicular edges use a wide range of sensors spanning traditional (such as position, acceleration, temperature) and also state-of-the-arts camera LIDER radar to improve their reliability and provide a range of new services and applications for users (such as virtual reality and smart control). In order to explore and identify the disruption caused to messages by attacking nodes in an opportunistic network, multiple experiments are run. A smart city is used to run the experiments in a controlled environment, changing the style, quantity, and length of attack to provide a multi-dimensional analysis into the protocol that provides best resistance to the attacks. In the city the public transport systems (trams and buses) are autonomous and rely on the information sent through the opportunistic network to function. Fast emerging new sensor devices and 5G+ sensor networks are becoming increasingly pervasive, and we see an explosion of new autonomous vehicle edge applications in which vehicles can sense, store process, communicate, predict

and anticipate any information about themselves, other vehicles, and the environments. The messages include information about traffic, roadside infrastructure, and the location of vehicles around the trams and buses. Without information, the autonomous vehicles would not know where the surrounding cars and pedestrians are so might collide with them. Even if collisions were not caused, traffic hold-ups are a legitimate cause for concern when high risk targets are vulnerable.

In order to test the networking protocols, modified algorithms were produced that allowed cars either to be fully functional, or malfunction as black and grey hole nodes. When the attack starts, cars will either act as black or grey hole nodes depending on which the attackers select. The attackers aim to disrupt the flow of the network and cause harm to those within the city. As the trams and buses are using the data to navigate and manoeuvre around the city, if the data do not reach them they will either not reach their destination or act in a dangerous manner as they do not have the data to avoid doing so. In order to study the nodes throughout the attacks, each of them collects data about each of the messages they receive. The data consist of how long the messages have been stored at the current node, the time at which they were originally sent and the time they were delivered. With the large amount of data gathered, multi-dimensional exploration allowed an insight into how different attacks impact metrics including the sensors' delivery probability of a message, the sensors' time taken to deliver and the computation expense to each node. The analysis of the attacks shows how different algorithms are affected by black and grey hole attacks. The state-of-the-art Maxprop algorithm is compared to a benchmark protocol (Epidemic) as well as other more complex methods.

Throughout the paper, novel networks are referred to. These are new, emerging types of networks which combine autonomous vehicles (as mobile cyber physical systems) with rich onboard sensors, cameras that are able to share and process complex real time multimedia data to help their decisions. Figure 1 [1] shows a method of how vehicles may communicate in a novel network using a hop-based approach. This method allows all vehicles to send/receive messages that are insecure. Security is important when working in safety-critical systems (such as with heavy machinery) and, as such, requires further investigation. The experiments in this paper consider the security provided by different delay-tolerant networking protocols and how they could improve the performance of the network.

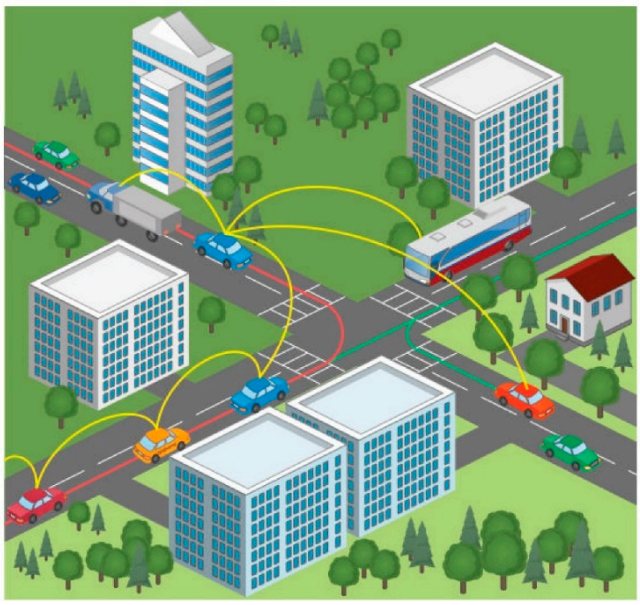

Figure 1. Illustration of how vehicles can communicate with each other in a hop-by-hop manner within a VANET [1].

Presented in the paper are the results of many experiments run for the smart city scenario. These include:

- Multiple novel networking protocols
- Multiple time frames (to remove the impact of simulation time on results)
- Varied attack methods
- Varied attack strengths (proportion of attacking nodes to ordinary)
- Multiple averaged performance metrics (averaged over multiple runs of experiments in order to improve reliability and repeatability)
- Geospatial visualization and analysis

2. Related Work

There are many different applications for networks and in some instances the standardised topologies are not optimised for the purpose. When working with vehicles, an important aspect is the ability for them to be mobile and remain connected to the network. This is where novel networks and their protocols can provide an advantageous solution to a specific problem.

2.1. Next Generation Networks

The nodes within a mobile ad hoc network (MANET) can be many different mobile devices, all connected via Bluetooth and/or a WiFi to form a dynamic network. The recent release of widespread 5G networking capabilities has led to more research on the topic as they are able to support a much larger flow of traffic [2]. A key feature is mobility and allowing the nodes to move freely. Due to the high expense most of the research and experimentation into MANETs is only achievable within computer simulations [3]. MANETs allow communications directly between two nodes rather than having a router between them. They have routing protocols embedded within them in order to allow nodes to communicate in a predefined, consistent manner. Now that mobile phones are becoming more capable (such as incorporating 5G [4]), the networks are becoming much more feasible as the nodes have more capability to perform processes traditionally managed by a fixed infrastructure.

The networks have the potential to save lives as well as generally improve vehicle and road safety [5]. If the networks were widely adopted, the convenience and comfort of passengers and drivers might also be improved. In vehicular ad hoc networks (VANETs), mobility is especially important as not only do most vehicles travel long distances, but they also move at relatively high speeds [6]. VANETs can include multiple types of vehicle and can adapt quickly to real world movements. They also allow vehicles to communicate with road-side equipment: a fixed infrastructure alongside of the road so that the drivers of vehicles can be alerted to information and services. This could include safety and traffic information as well as any services that the driver may need at that given point in time [7].

Flying ad aoc networks (FANETs) are another common form of complex next-generation network. They consist of a number of UAVs that communicate directly with each other, allowing the data to remain more secure than when communicating using infrastructure such as a group base or satellite [8]. Having a network of UAVs able to work together to achieve a high-level goal would allow military operations to be completed quicker and more reliably as it introduces some fault tolerance over a single drone [9]. The use of drones and FANETs can also be combined with other networks such as a VANET in order to create a complex network of networks. This would allow a drone to communicate with a vehicle on the ground using a hop-based approach [10]. This could be used in a smart city scenario where, for example, drones are used to pass data directly to police officers on the ground.

Delay-tolerant networks (DTNs) are useful where reliable communication is not available for message transmission, due to a sparse and intermittent connection. End-to-end connectivity is not guaranteed in a DTN and therefore alternate routing protocols such as the store-and-forward approach increase the probability that a message will be delivered once a connection has been reinstated [11]. They have been developed in order to overcome

technical issues that have been identified in standard networking topologies. When operating in either mobile or extremely sparse environments DTNs used as constant connections are a rare occurrence [12].

Opportunistic networks are viewed as a subclass of DTNs where communication is intermittent. This means that an end-to-end path between destination and source may never exist. Nodes are only connected temporarily, and the topology is highly changeable [13]. One-hop communication is very important in an opportunistic network. This is a wireless and short-range message exchange between nodes with no intermediate node on the communication path, meaning that there is no message-routing. The nodes are able to send and receive arbitrary data to and from any node that they are in direct communication range with [14].

2.2. Security in Opportunistic Networks

Where technology is used, security is always a key consideration. When safety is dependent upon the technology, security is of the utmost importance. Given that in the use of VANETs the network will have an impact upon many vehicles (each made from tonnes of metal) there is an opportunity for a vast amount of damage to be caused to other individuals and bystanders. Due to the open-access nature of a VANET, they are vulnerable to multiple forms of attack that must be defended against [15].

A proven low-cost hardware (Raspberry Pi) method for dealing with faulty nodes is to use a multi-layered approach that allows all devices in a complex network to be self-organised and provide self-adaptive routing algorithms [16]. This could be used to improve resiliency to cybersecurity attacks as an attacking node is equivalent to a node with a deliberate fault.

One method of increasing the security of a MANET is using a reputation-based mechanism [17]. In such a network, every node collects reputation information from neighbouring nodes and gathers indirect reputations from other nodes. These get updated with each interaction so that a given node's reputation is as up-to-date as possible [18].

In order to provide resilience to Black Hole Attacks another proposed method is isolating the nodes from the network entirely. This can be done by holding a master routing table with trusted routes stored [19]. One problem with this is that once it is isolated it would never receive a message again, even if the malfunctioning node began to function again.

Blockchain technologies have recently become popular methods to decentralise and provide smart contracts for greater security in novel networks [20]. A number of key elements make up the technologies: Decentralisation, Transparency, Open Source, Autonomy Immutability and Anonymity [21]. All these factors combined provide a network without fixed infrastructure with a method of remaining secure [22]. It has been proved that such technologies can be used within opportunistic networks [23]. In the proposed method however, two nodes meeting causes their chains to be merged creating a theoretically infinitely long chain.

2.3. Opportunistic Networking Architecture

Using a fully distributed peer-to-peer (P2P) software-defined network (SDN) allows vehicles to communicate in a vehicle-to-vehicle (V2V) manner. The vehicles could also communicate directly with infrastructure (V2I) or any other device with networking capabilities (V2X). Using P2P SDN architecture allows for better path selection if the software allows the communication of routing information [24]. The PRoPHET and MaxProp algorithms take advantage of this as the protocols send reputation information about other nodes within the network in form of probability vectors.

Using vehicles as edges in a network provides interesting challenges as not only do the edges require mobility but they travel at high speeds and can vary in direction at short notice. When considering the architecture required to support the edges, it must adapt to the needs of the vehicles. Using vehicular micro-clouds, following a hierarchical

cloud structure provides edges with the required mobility and also provides a method of communication between nodes [25]. In this architecture, clusters of vehicles can be identified geographically to allow more data to be transferred about the surrounding environment [26]. To design appropriate architecture for vehicular SDNs, fog computing is an important consideration. Fog computing is a decentralised infrastructure allowing data, storage, and compute instances to be located between the data source and the cloud [27]. In this instance the data and storage are shared between the vehicular edges that make up the VANET. Using fog-computing principles in the design of a network model provides methods to handle difficult problems such as orchestration and services as there are pre-existing solutions. These can be applied to a fully distributed P2P SDN architecture and have proved their use in cases such as a traffic management system and accident rescue [28].

3. Delay Tolerant Network Protocols

Routing protocols in DTNs can be separated into two broad categories: replication-based and forwarding-based. In a replication-based protocol, the main principle is that a message is copied multiple times. Replication of a message allows it to be distributed quickly throughout a network as well as decreasing latency and increasing the chance of a successful delivery. However, having too many packets in a network and using replication-based protocols can mean the network gets congested easily and slows down transmission of all messages. In a forwarding-based protocol, the message only exists as a single copy. When it is passed between nodes, once the receiver has got the message, the sender deletes it. This keeps network traffic down to a minimum but vastly increases latency and the chance of a successful delivery is significantly reduced.

In the experiments four different protocols were used:

- Epidemic [29]—A basic benchmark protocol requiring a vast amount of resources to produce reasonable results (Algorithm 1)
- Spray and Wait [30]—A more complex algorithm building upon Epidemic's success and reducing the need for resources (Algorithm 2)
- PRoPHET [31]—A complex algorithm relying on probabilities to route messages as directly as possible and minimising duplications (Algorithm 3)
- MaxProp [32]—An intricate, state-of-the-art protocol relying on probabilities and hop-counts to provide the highest delivery probability using the least resources [33] (Algorithm 4)

3.1. Black and Grey Hole Epidemic

As a flooding-based protocol, the Epidemic routing protocol is constantly replicating and transmitting a message to any node that does not yet have a copy of the message [29]. Due to the persistence of the protocol, it has a relatively high delivery probability, but it requires high overheads. As the protocol is relatively effective as a simple algorithm, it is often used as a benchmark to allow comparisons of other protocols. The protocol was modified to allow attacking vehicles to act as both black and grey holes. The adaptations allowed the functional nodes to communicate with attacking nodes without being aware of their state (attacking or not).

3.2. Black and Grey Hole Spray and Wait

The Spray and Wait routing protocol is a relatively simple yet efficient routing scheme that consists of two phases [30]. Spray and Wait is a combination of ideas from two other routing protocols: Epidemic and Direct Transmission Epidemic is very effective at spreading a message quickly though a network to the destination, however it uses an immense amount of computational resources; direct transmission uses very few resources but takes a long time to deliver the message. Spray and Wait is an effective amalgamation as the spray phase allows the speed of epidemic routing and the wait phase continues until the message is delivered to the required destination node. The biggest issue with the protocol is how to determine the number of copies of the message which should be

spread in the initial stage. If the number is too small, the message may not be delivered, but if the figure is too large it could become identical to the Epidemic Routing Protocol with resulting high computational expense and use of resources. There are currently two common adaptations of the spray and wait protocol, the only difference between them being how the message is distributed during the first phase.

Algorithm 1. Black and Grey Hole Epidemic Router

```
start
while True do
    if Contacted Node then
        if Black Hole then
            Drop Messages
        else if Grey Hole then
            if Dropping Messages then
                Drop Messages
            else
                if Next node does not have messages then
                    Forward copy of messages
            end
        else
            if Next node does not have messages then
                Forward copy of messages
end
```

Every message that originates at a source node within the network is copied a set number of times. Once two nodes meet, they will transfer half of the available copies of each relevant message. This continues until a node only holds one copy of the message and then goes into the wait phase. The reason this adaptation was made is passing messages in this manner allows far a faster spread and therefore lower latency within the network.

Once a message has reached the limit of copies, if the destination node for the message was not found during the previous phase, each of the nodes that has a replication of the message then uses direct transmission in order to deliver it to the required node. The wait phase vastly reduces the computational expenses, however it does run the risk of increasing latency depending upon how well the copy limit has been calculated for the network. The basic algorithm was altered to allow attacking vehicles to act as both black and grey holes. The modifications allowed the functional nodes to send messages to attacking nodes without being aware of the fact the recipients were malfunctioning.

Algorithm 2. Black and Grey Hole Binary Spray and Wait Router

```
start
while True do
    if Contacted Node then
        if Black Hole then
            Drop Messages
        else if Grey Hole then
            if Dropping Messages then
                Drop Messages
            else
                if Next node does not have messages then
                    if Copies available or Next node is destination then
                        Forward messages
            end
        else
            if Next node does not have messages then
                if Copies available or Next node is destination then
                    Forward messages
end
```

3.3. Black and Grey Hole PRoPHET

The more complex PRoPHET (Probability Routing Protocol using History of Encounters and Transitivity) uses probabilities [31]. It calculates the delivery predictability based on contact history between nodes, where a higher value represents a greater chance of future contact with the destination node. This means that a message is only replicated if the delivery predictability of the new node is larger than that of the transmitting node [34]. This keeps overheads low, allowing it to be implemented where there are limited resources [35]. The delivery predictability value allows a projection as to how likely a node is to be in contact with the destination node. This means that a node stores a value for each node it encounters, and then increases the predictability upon future contacts. Only if the receiving node is more likely to deliver a message does the transmission node send the message.

There are more advanced protocols based upon PRoPHET such as PRoPHET+ [36]. The latter takes into consideration other data about the nodes such as the buffer size, location, and status to support the delivery predictability. The further considerations allow the nodes to decide which is most likely to deliver the message if their predictability value is the same. For example if the nodes were both equally likely to deliver the message, but one was much closer than the other, it would receive the message as it would be deemed most likely to deliver the message.

The vanilla PRoPHET protocol was amended to allow communication with grey and black hole nodes. As well as this, another algorithm was generated to allow vehicles to perform as black and grey hole nodes. Unlike the former algorithms, when nodes meet they exchange information about contact history and the probability of delivery. Due to this, the attacking nodes still need to perform some function rather than just removing data. The attacking nodes can falsely publish their probability of delivering a message to increase the likelihood of them receiving (and therefore removing) it.

Algorithm 3. Black and Grey Hole PRoPHET Router

```
start
while True do
    if Contacted Node then
        ExchangeSummaryVector()
        UpdateDeliveryProbabilities()
        if Black Hole then
            Drop Messages
        else if Grey Hole then
            if Dropping Messages then
                Drop Messages
            else
                if Next node does not have messages then
                    if Next node is destination then
                        Forward messages
                    if Next node has greater probability of delivery then
                        Forward Messages
            end
        else
            if Next node does not have messages then
                if Next node is destination then
                    Forward messages
                if Next node has greater probability of delivery then
                    Forward Messages
end
```

3.4. Black and Grey Hole MaxProp

The MaxProp routing protocol is a state-of-the-art algorithm and provides an effective way to route DTN messages [11]. Whereas Spray and Wait replicates a message a fixed

number of times at the source node, the MaxProp protocol replication occurs each time nodes meet and the conditions for copying a message from one node to another are met. These conditions are based on prioritising packet movements through the network based on their hop count. This includes both the schedule of those packets to be transmitted and those to be dropped [32]. In a MaxProp network, all nodes carry all messages until the next transfer occurs. A node will continue to forward a message to multiple others until either the message times out, it is notified of delivery, or is deleted due to a full buffer. The key to the routing algorithm is the nodes knowing which messages should be transmitted and dropped first (Figure 2).

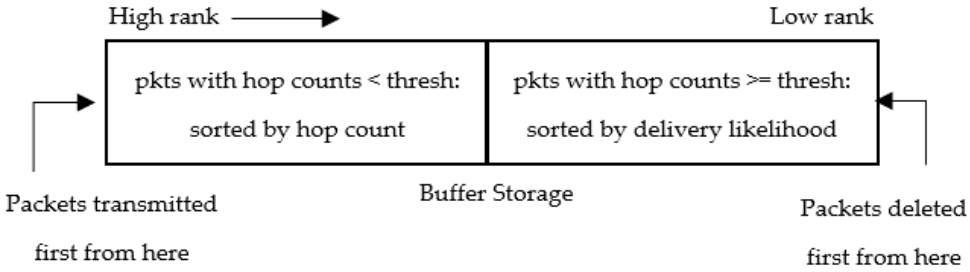

Figure 2. Diagram of how the MaxProp protocol works [32].

To work out the chance of being on the transitive path, each node has a vector with a number of elements equal to one less than the total number of nodes within the network. Initially each node is equally likely to meet any other node next, once a node meets another (node x) the x^{th} element of its vector is incremented by one. Then the entire vector is normalised so that the magnitude of the whole vector is equal to one. Upon initial contact between two nodes, they exchange their vectors, in an optimal solution, every node has an up-to-date, accurate element within the vector containing the probabilities linked to every other node [37]. Then the vectors can be used during a depth-first search in order to find the shortest path to the destination node.

Acknowledgements can be used in the protocol as once a node has successfully received a message, they can be injected into the network [38]. Once it has been flooded, all nodes carrying the original message can free up space by deleting it. Messages with low hop-counts are given priority to be transmitted as this helps get new messages spread quickly and efficiently. The message also contains a list of nodes that it has already visited so that it does not revisit a node.

A node communicating using the standard Maxprop algorithm will only communicate with others using the same protocol. The black and grey hole vehicles use a protocol based upon Maxprop that is modified so that either all or half of the messages can be dropped based upon the parameters set by the attackers. The black and grey hole Maxprop algorithm communicates with standard nodes by impersonating the standard protocol that they are using. The attacking nodes can falsely increase their published chance of delivery so that they receive (and attack) more messages.

3.5. Protocol Comparison

The key differences between the protocols are highlighted in Table 1. If a protocol is replication-aware it indicates that there is a fixed number of copies (replications) within the network and it will not exceed this by making any new copies. Reputation awareness is based upon the data a node retains about other vehicles. In this instance it is concerning how likely a node is to deliver the message to the destination node. If its reputation is better (more likely to deliver the message) then the node will provide the message to it. Social awareness allows nodes to communicate a social status between them, passing data about other nodes to the wider network. This can be done as a part of updating the vector values and changes the likelihood of a vehicle receiving the message. Resource awareness

protocols consider the resources available to the node and will manage the messages it is carrying based upon the capacity of each vehicle.

Algorithm 4. Black and Grey Hole MaxProp Router

```
start
while True do
    if Contacted Node then
        ExchangeSummaryVector()
        UpdateDeliveryProbabilities()
        SortPackets()
        if Black Hole then
            Drop Messages
        else if Grey Hole then
            if Dropping Messages then
                Drop Messages
            else
                if Next node does not have messages then
                    if Next node is destination then
                        Forward messages
                    if Next node has greater probability of delivery then
                        Forward Messages with lowest hop count
            end
        else
            if Next node does not have messages then
                if Next node is destination then
                    Forward messages
                if Next node has greater probability of delivery then
                    Forward Messages with lowest hop count
end
```

Table 1. Table to compare attributes of the protocols.

Attribute	Epidemic	Binary Spray and Wait	PRoPHET	MaxProp
Replication Awareness	No	Yes	Yes	Yes
Reputation Awareness	No	No	Yes	Yes
Social Awareness	No	No	Yes	Yes
Resource Awareness	No	No	Yes	Yes

4. Network Model

An opportunistic network was made from many nodes communicating with the afore-mentioned protocols (unique protocols for each set of experiments). Each of the nodes represents a car that communicates using the protocol to other cars within a close vicinity. These cars could be functioning or could be compromised (black/grey holes). Those that are compromised will not perform as designed but will drop either all of half of the messages instead of forwarding them along. The diagram below (Figure 3) shows how cars communicate with each other and how the messages are spread throughout the network. It illustrates how the location of the attacking node impacts the effect it has on the network and how having more densely packed areas provides more unrestricted paths for message transfer.

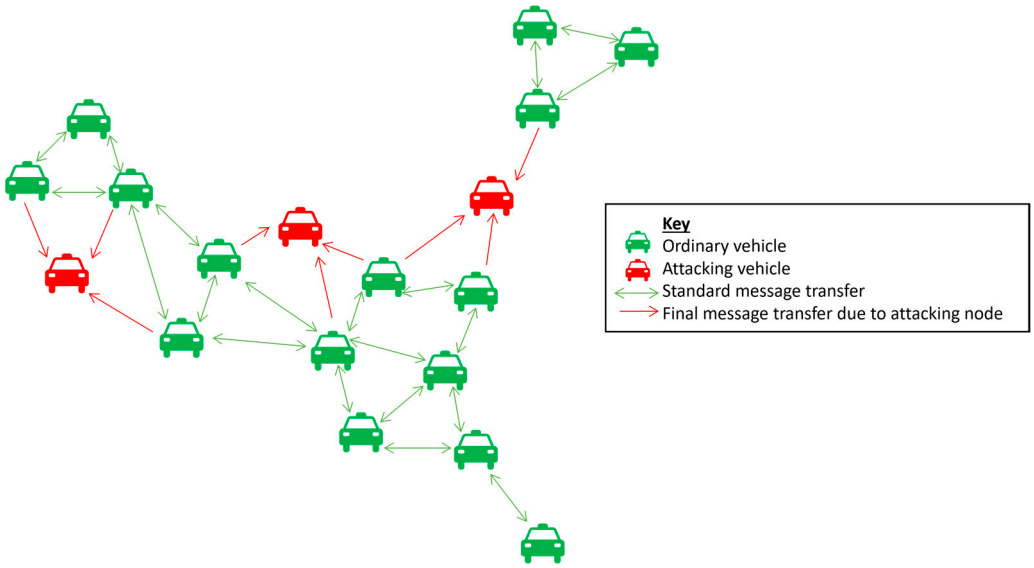

Figure 3. Diagram to show vehicular nodes communicating.

5. Autonomous Vehicle Opportunistic Network

In order to evaluate the effect of attacking vehicles, each node was monitored for the messages it was sending and receiving. The environment was kept constant with the only changes being the substitution of compromised cars for regular cars. The experiments were developed around Helsinki City Centre as a smart city and included varying modes of transport including trams, buses, cars and pedestrians. Each of the individuals was connected using an Opportunistic Network operating over Bluetooth.

In the city centre the trams and buses are autonomous and rely on the messages they receive from other vehicles to get the passengers to their required destination. The messages contain important data about the current traffic in the city, the status of controlled junctions the vehicles must drive through, and the position of other individuals in the vicinity of the autonomous vehicle. The messages originate from other individuals in the network as well as roadside infrastructure such as traffic lights. The map (Figure 4) shows the public transport system within the city that the attackers will be attempting to disrupt.

The aim of the attacking group is to prevent data arriving at the autonomous vehicles by disrupting messages in the VANET, therefore reducing the messages delivered throughout the network. This is to cause commotion within the city and make people's journeys more dangerous. All of the attacks are distributed rather than being focused on a particular target (this could be a specific vehicle or location) this is to cause the most widespread disruption to the network rather than specific localised area.

With no data reaching the autonomous vehicles they will stop (to prevent any dangerous actions being taken), however this could be in the middle of a junction or road causing a hazard to other road users and potentially causing a roadblock. This could be used by the attackers to prevent people arriving at or leaving the city. If only some data are received by the autonomous vehicles, they would function but less reliably than if all messages were received. This in turn could cause unnecessary traffic to certain areas, as well as dangerous manoeuvrers being made. For example, if a message about an individual's proximity to an autonomous bus gets dropped by a black hole vehicle, the bus may attempt to manoeuvre into the individuals' space, causing them harm. This is of more use to attackers trying to endanger the general public due to the resulting unpredictable behaviour of the autonomous vehicles.

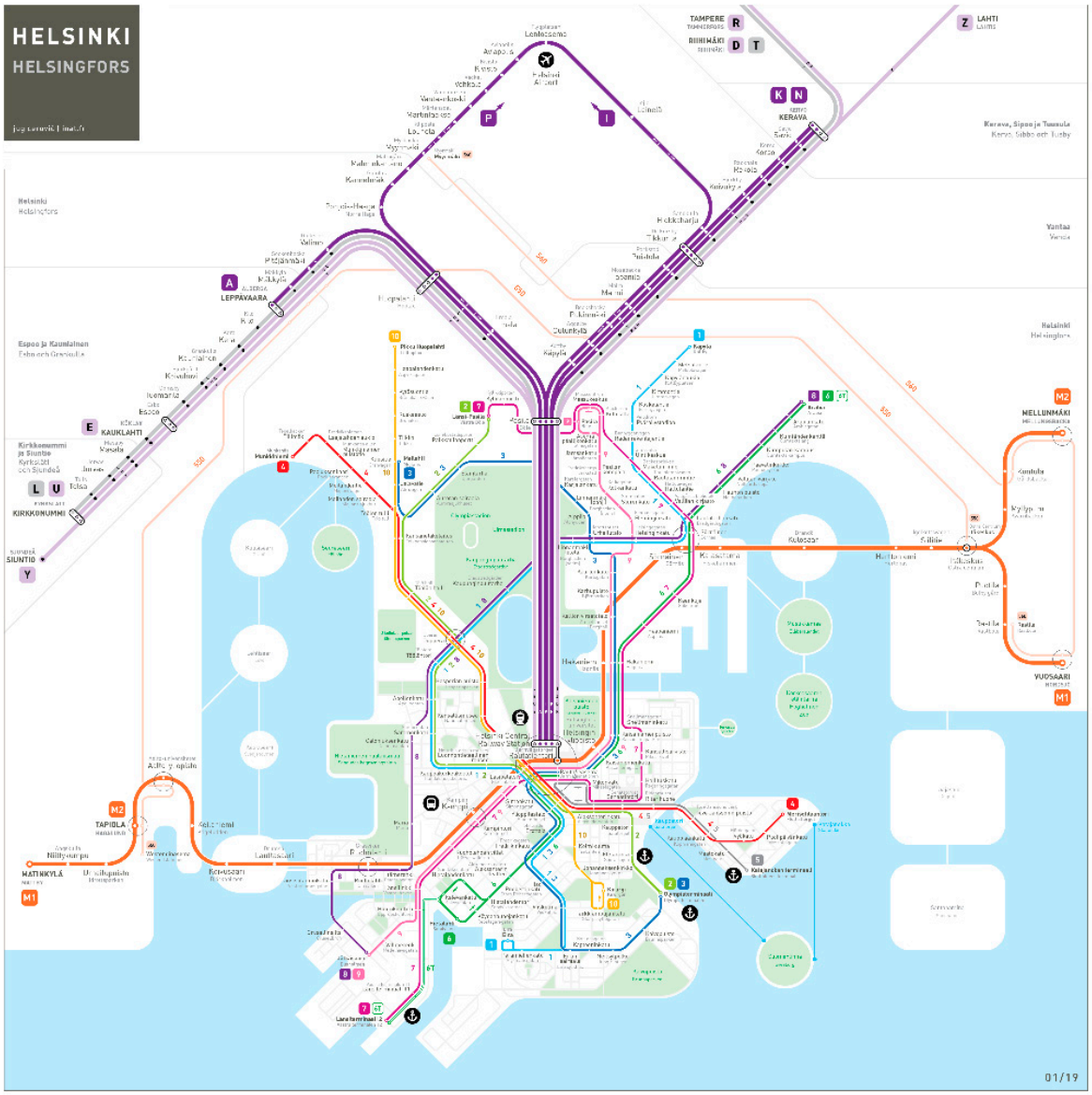

Figure 4. Helsinki Public Transport map.

If a serious traffic jam within the city were to occur as a fault of the network, there could be major consequences. Not only would the blame be put on the technology and damage the reputation for the networks (and technological advance) as a whole, but also to the immediate areas affected. Traffic congestion and road safety (especially crash severity) are shown to have a correlation [39] and thus with congestion in an area, the public are less safe. Studies have also shown that uncontrolled traffic problems (such as the unpredicted ones in this instance) can lead to more fatalities on roads and, as such, should be avoided by any means possible [40].

6. Distributed Attacks

There many different ways of attacking a VANET such as DDos attacks [41] and black or grey hole attacks [42]. In the experiments, attacking nodes are cars that have been tampered with by the attacking group. The cars can function as either black or grey holes [43]. The two different attacks are frequently used to try and gather information as well as disrupt the flow of messages through the network at the same time. Black holes are nodes that do not forward any information and just receive it. Grey holes forward only some of the messages they receive. This scenario will allow exploration into how each of

the selected protocols performs if someone is attempting to disrupt the network using the two attacks. Due to the fact that DTNs could be used in a variety of different situations, it is likely that at some point they would be subject to attempted disruption by an individual or a group of individuals. In this instance the attacking group is attempting to cause harm to individuals within the city by tampering with the autonomous vehicle network. The attacks are coordinated by a group in the hope that it will have maximum impact on the network. When developing any new technology it is important to consider how people may try and abuse it, so research into how attacks may affect performance of a network can help future developers chose how to either prevent, or overcome them. The distributed attacks are formed as the group can work together to achieve the disruption. It is the most likely form of attack on such a dynamic network as any individual attempting to attack is unlikely to interrupt the traffic significantly.

6.1. Distributed Black-Hole Attacks

Black-hole attacks occur when one of the compromised cars is activated and therefore acts as a black hole within the network. The aim of each of these cars is to drop (not forward) each of the messages it receives so that it never reaches the desired destination. This form of attack is thought to be one of the most common approaches to damage the performance and reliability of networks [44], so it is important to compare the performance between protocols to see if one is better at handling it than others. The attacking group is using black-hole attacks to try and prevent the autonomous trams and buses from functioning correctly. The hope is that they will cause harm to other road users and cause traffic chaos across the city.

6.2. Distributed Grey-Hole Attacks

Instead of dropping all messages, grey-hole attacks take a stealthier approach and only remove half of the messages they receive (forwarding the remainder). This makes them harder to detect within a network and allows them to be active for longer. Random Grey Holes do not have any selection criteria for which messages they forward and which they drop, they will do it randomly so that about 50% of messages get forwarded. Further research into selection criteria could make a Grey Hole attack more effective. For example if attackers were searching for a specific type of message they could drop those but forward others so that they were less likely to be discovered. If the attackers were looking to improve their attack vector with more complex methods, they could decide to take all of the messages from individuals but leave those from roadside infrastructure. This would mean that the trams and buses would operate through junctions as if there was no-one around, making them a danger to other road users in the vicinity. This would mean that it would not be safe for any individuals on the streets, either in the autonomous trams and buses or those in their own vehicles.

7. Implementation

7.1. Simulators

There are a variety of different tools that allow the simulation of DTNs and other types of network. Each have their pros and cons, and some are more complex than others to utilise to their full capacity.

7.1.1. Opportunistic Networking Environment Simulator

The opportunistic networking environment (ONE) is a specifically developed simulation tool designed for opportunistic networks [45]. It is designed to allow a direct comparison between routing protocols based on how they would react to real world scenarios. On the initial start-up of the ONE simulator a default simulation is loaded, which makes it easy to understand what the simulator is capable of straight away. It includes a map of Helsinki in order to run its simulations. A user can provide their own maps, however they have to be in multiple layers in a Well-Known Text (.wkt) file type. Due to

the variety of features ONE offers, it was used to perform the experiments making the evaluation metrics easier to collect.

7.1.2. Additions to the ONE Simulator

Although the ONE simulator is shipped with the vanilla protocols, the black- and grey-hole implementations needed to be added. The protocols are written in Java and, as such, could be programmed in a manner to suit the situation. To perform the experiments new routing files were created for each protocol for both the grey and the black hole adaptations. In order for the vanilla implementations to communicate with the attacking nodes, they also needed to be altered (as ordinarily they can only communicate with nodes of the same file type).

7.2. Experiment Parameters

The configuration files were altered and allowed different experiments to have different parameters (Table 2).

Table 2. Table to show the experiment parameters.

Parameter	Varied (Yes/No)	Value(s)
Time (seconds)	Yes	10,800; 18,900; 27,000; 35,100; 43,200
Transmission	No	Bluetooth
Number of Ordinary Cars	Yes	50, 45, 40, 35, 30, 25, 20, 15, 10, 5, 0
Number of Public Transport Vehicles	No	6
Number of Attacking Cars	Yes	0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Protocol	Yes	Binary Spray and Wait, PRoPHET, Maxprop, Epidemic

8. Geospatial Analysis

During the experiments the location of all vehicles was tracked so the density of attacking nodes could be visualised (Figure 5). This shows that although distributed attacks provided the opportunity for wider spread disruption, the less structured approach meant that attacking nodes could cluster together, rendering them less effective. Testing them in a dynamic environment did, however, provide the opportunity for the densities of nodes to change providing a necessary overall view of a complex strategy of both running and attacking autonomous vehicles.

The visualisations can show the entire experiment (animated), a single position in time or a summary of the entire monitoring time. An effective way to ensure the attacks were covering the entirety of the city was to show where the attacking cars had got to within the given time frame (Figure 6). The coverage shows that the attacking nodes reached most of the areas and so the disruption caused should have been relatively high. As anticipated, in the more outlying areas the density of vehicles was less than in the city centre, however this would be true for regular and attacking vehicles so the proportions should not differ greatly.

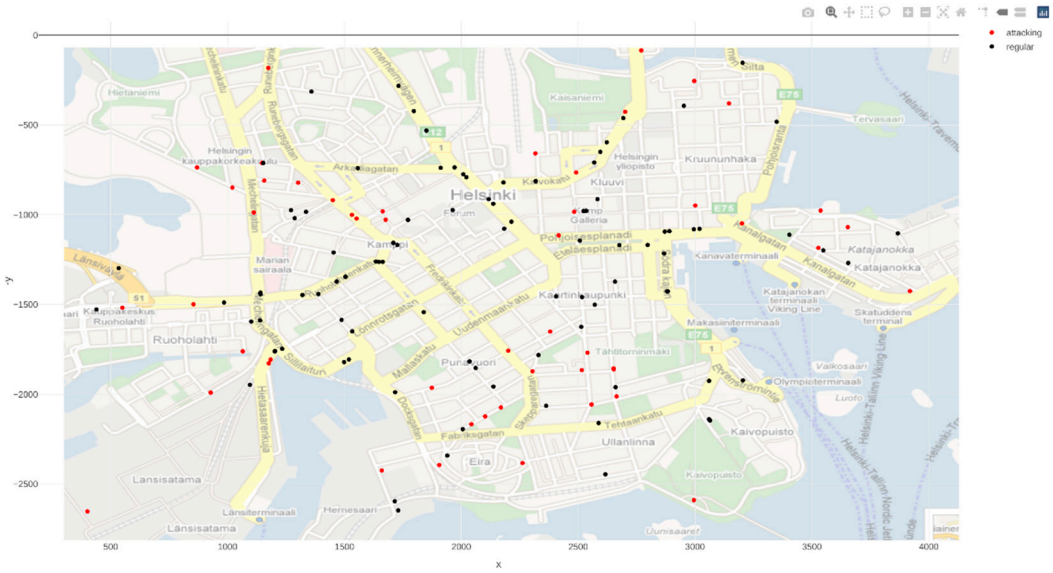

Figure 5. Geospatial Plot of vehicles at a single point in time.

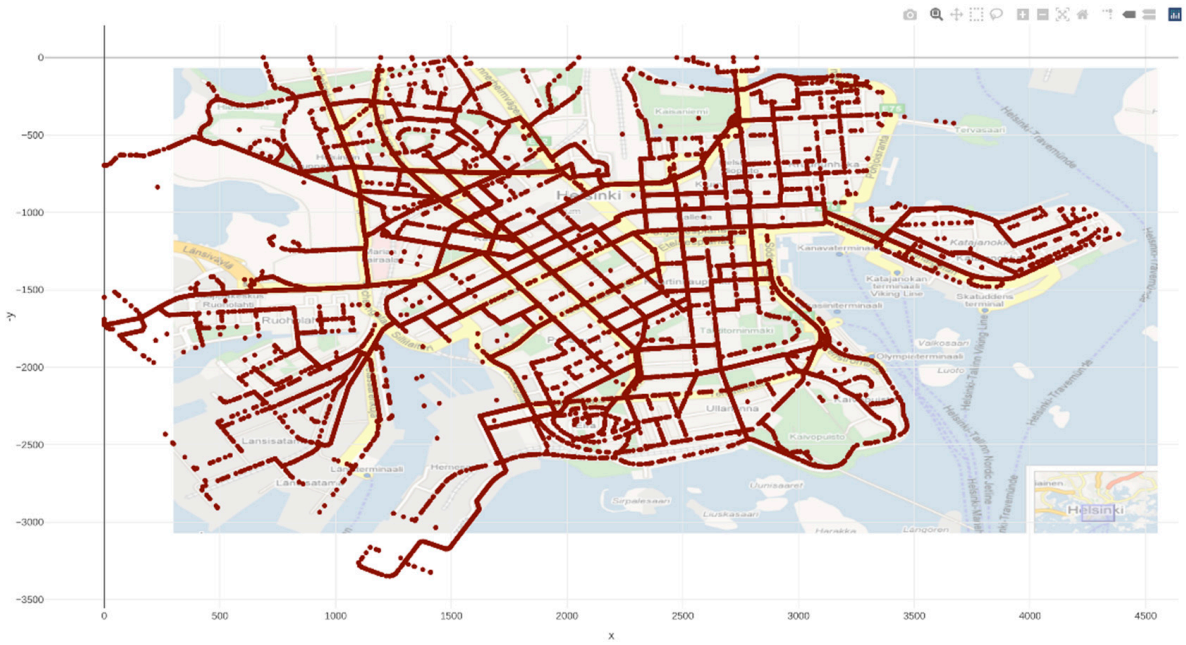

Figure 6. The coverage of the attacking vehicles throughout the attack time.

9. Multidimensional Evaluation

The evaluation metrics were gathered using the reports that the ONE simulator created. After having collected results for 440 experiments (four routing protocols, five time frames, two attack types and eleven attack proportions), python scripts were written to collate the data from individual text files to a usable CSV format. From the CSV files, the averages across all time frames could be calculated in an attempt to present the most reliable data points on the graphs generated using R.

9.1. Average Delivery Probability Throuhout the Network

The average delivery probability (Figure 7) provides the chance that a single message generated from anywhere within the network would be delivered. The delivery probability was produced in a report in the ONE simulator. It considered all of the messages created throughout the experiment and which were delivered. From here it calculated the percent-

age chance that a randomly selected message would be delivered. There are many factors such as sender and receiver location that could effect this, which is why the average is used. The higher the average value, the more likely messages are to have been delivered and thus a higher value indicated a more successful algorithm. From the graph it can be seen that all the protocols benefited from grey holes being introduced, most likely due to the reduced amount of traffic. Black holes have a similar effect up to 20% when performance is then reduced.

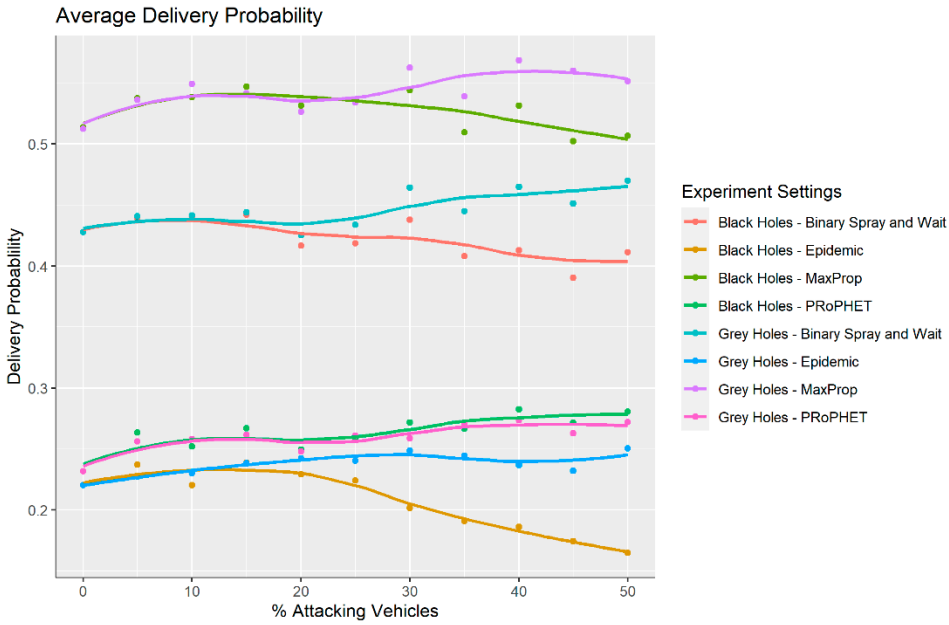

Figure 7. The average delivery probability across multiple experiment settings.

9.1.1. Black-Hole Attack

Looking first at the data points with zero black holes, it can be seen that the MaxProp protocol offered the highest chance of a message being successfully delivered. The graph shows the average taken across all five different attack times. This is not a surprise as the MaxProp Protocol was a more complex algorithm that was developed specifically to improve upon more basic protocols such as the Binary Spray and Wait algorithm and the Epidemic protocol. The surprise was that the PRoPHET protocol was only marginally more likely to deliver the message than the Epidemic protocol.

On introducing attempted disruption, MaxProp and Spray and Wait reacted to an increasing amount of black hole nodes in a similar way. This could be seen by the similar gradient of the linear trend lines. The gradient of the MaxProp trend lines was, however, slightly less negative and this was seen in future sections as well, which showed that MaxProp was more resilient to black-hole attacks. This also meant that it was a valid assumption that the MaxProp algorithm would never perform worse than the Binary Spray and Wait regardless of any black hole nodes added, as long as the current data were representative of those data points that could be estimated using extrapolation techniques. This went to show that when the main priority was a message reaching its desired destination, regardless of any black hole nodes, MaxProp was the algorithm to choose over Binary Spray and Wait.

PRoPHET and Epidemic reacted differently to the introduction of black holes. Epidemic demonstrated a significant reduction in delivery probability as black holes were introduced and PRoPHET showed an increasing proportion of messages delivered. This could not continue indefinitely (as at 100% black holes no messages would be delivered) but it did show that the protocol was more resilient to black holes. This however, did not explain why the probability was increasing. It was most likely due to the reduced traffic in the network. Due to the fact that PRoPHET records all contacts, it is highly likely that

in areas densely populated with nodes the protocol was less effective due to the constant updating of delivery predictabilities.

9.1.2. Grey-Hole Attack

The graph shows that across all four protocols, grey hole nodes had far less of an effect on the network versus black holes. This would be expected given that grey holes only dropped half of the messages rather than all of them. Interestingly, the results of the two types of attack were very similar up to around 25%. From here the black holes decreased the delivery probability significantly whereas for grey holes, the graph indicates that the messages were more likely to be delivered. This was unexpected as the more grey holes, the more the network should be disrupted (more messages dropped). One reason may be that as more grey hole nodes were introduced, there was potential for fewer messages to be generated (a grey hole could just drop its own message instantly).

Maxprop and Spray and Wait performed slightly better with grey holes rather than black which was as expected. PRoPHET appeared to be the other way around, for black holes the highest delivery probability was around 0.27 and for grey it was around 0.25. This was an unexpected result, possibly due to the algorithm trying to reduce the number of copies made. If the grey holes managed to drop the messages before they were replicated, there was a much smaller chance of them being delivered.

Although Maxprop lent itself to being the algorithm of choice to maximise delivery probability, the trend lines indicated that PRoPHET was more consistent. This showed that the complex PRoPHET algorithm was more resilient to both attacks but provided a much lower delivery probability.

9.2. Average Latency for Delivered Messages

Latency is used to measure the amount of time taken for a message to be delivered. The graph (Figure 8) shows the average amount of time, considering all delivered messages, between being sent and received. A lower latency is preferable as it means that messages are delivered faster.

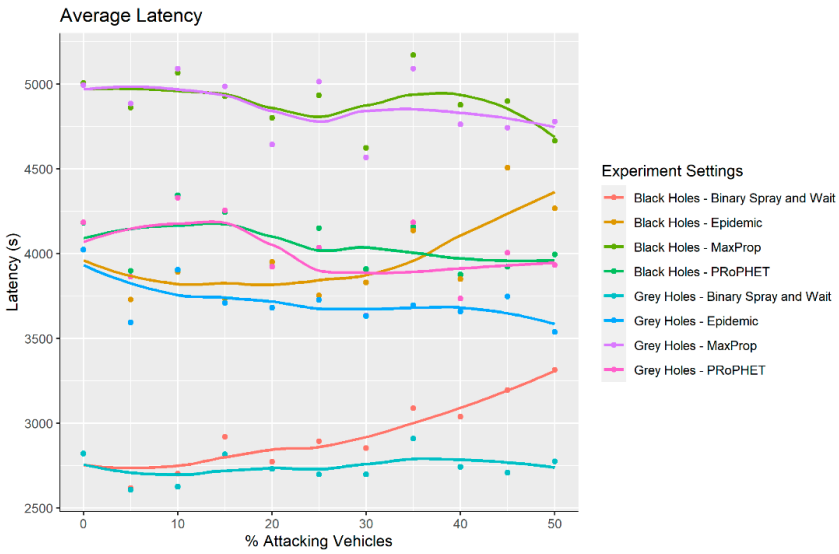

Figure 8. The average latency across multiple experiment settings.

9.2.1. Black Hole Attack

For the range of results collected, MaxProp consistently took longer to deliver the messages successfully. There was a large difference in speed between the all the algorithms, however it was important to recognise that even though Spray and Wait delivered the messages faster, far fewer messages were received at all as demonstrated in the graphs showing the delivery probabilities.

The gradient of the lines suggested that should more and more black holes be introduced, eventually Epidemic, MaxProp and PRoPHET would perform better than Spray and Wait. It was far less affected by black holes. If plotted as a linear trend line, the gradients allowed us to see that Spray and Wait was approximately four times as vulnerable to black holes in terms of latency than Maxprop. Based on the trend lines and by solving the equation: 12.5x + 2671.8 = 3.1x + 4933.9, it can be shown that the crossing point of the algorithms was at the x coordinate of 240.6. This was not possible to extrapolate due to the fact that as the proportion of black holes reaches 100%, no data would be sent through the network so no messages would be delivered.

The difference in delivery speed was not entirely surprising as the spray phase of the basic algorithms meant that messages were initially spread very quickly throughout a network even if not very efficiently. As MaxProp and PRoPHET took a more calculated approach, they took longer. Current results showed that Maxprop offered a more reliable delivery probability and so was more widely considered as the better performing algorithm.

9.2.2. Grey-Hole Attack

For the PRoPHET and Maxprop algorithms, the data were very spread out which suggested that there was not much of a correlation between the proportion of grey-hole nodes and the average latency in the scenarios. This is an interesting observation as black holes slightly reduced the latency and so it would be reasonable to expect that grey holes would do so at half of the rate.

For the Binary Spray and Wait protocol, it seems that the latency was approximately flat. This again is an interesting discovery as for black holes it was increasing at an exponential rate. This showed that the protocol was far better at dealing with grey holes rather than black holes.

The graph shows that the Spray and Wait algorithm consistently provided a lower latency even when dealing with grey hole attacks. The spray phase meant that messages got around the network very quickly rather than the more calculated algorithms that took longer to deliver messages. It was also the only protocol that outperformed the Epidemic protocol, which was often used as a benchmark.

9.3. Average Time a Single Message Remains in a Unique Nodes Buffer

The buffer time is the number of seconds that a message is stored in a single node's buffer. These graphs considered two different averages (mean and median) from all of the nodes for all messages. It waimportant to minimise the buffer time as when a node is carrying a message, it is taking up resources that could be used for other messages or tasks. For both black and grey holes, the average time spent in a buffer was much larger for Spray and Wait than any of the other protocols. This was most likely due to the nature of the algorithm. Because a fixed number of messages were created at the start, they were then stored in a buffer until they were passed to another node. The other protocols created copies when necessary rather than having to store them for sustained periods of time.

9.3.1. Black-Hole Attack

The graphs (Figure 9) show two different averages for the buffer times of the nodes, both the mean and the median. The buffer time is the amount of time that a message is in a node's buffer. For Spray and Wait it was consistently far larger than that of the Epidemic, MaxProp and PRoPHET algorithms. This reflected the fact that during the wait phase a message can spend a very long time in the buffer until direct contact with the destination node was achieved, whereas the others were more efficient in achieving final delivery and then deleting copies of the message.

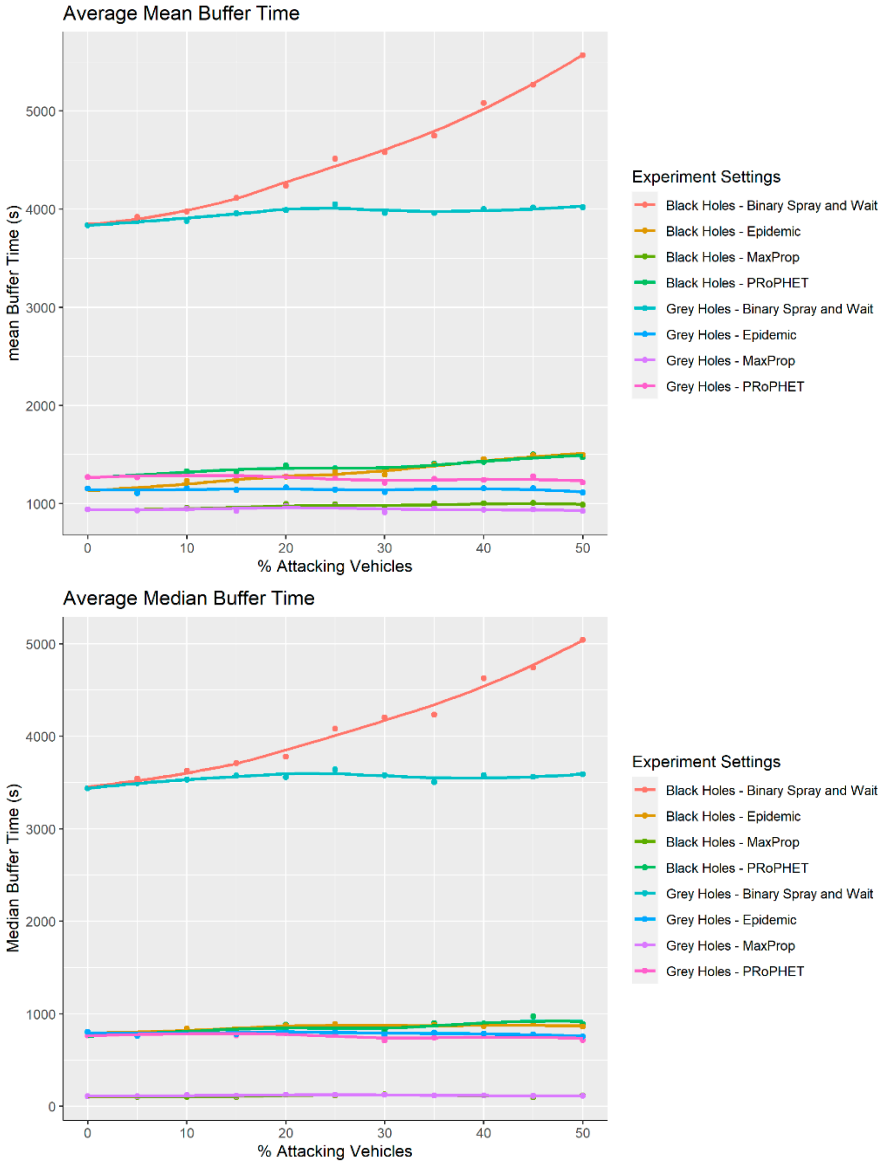

Figure 9. Two averages for the buffer times throughout the experiment.

The quadratic trend lines for the spray and wait algorithm meant that the rate at which the buffer time increases gets larger with an increasing number of black holes.

Both of the trend lines for the MaxProp protocol showed that the buffer time was unaffected by black-hole nodes. This was due to the nature of the algorithm dropping messages once the buffer is full. This was a more controlled way to manage network traffic as it meant that nodes were not carrying excessive amounts of unnecessary data.

Interestingly, the Epidemic and PRoPHET protocols were very similar and also far less effected by black holes than Binary Spray and Wait. It did however show a slight increase in buffer time with an increase in the proportion of black holes. This suggested that although it was definitely more resilient than the Binary Spray and Wait algorithm, Maxprop still outperformed them both.

Both types of averages were included because, as can be seen in the graph, throughout the experiment the mean was larger than the median without fail. This showed that the data were positively skewed. The skewness showed that the majority of the buffer times were smaller than indicated in the mean, it was just a few unusually large results that increased the mean.

9.3.2. Grey-Hole Attack

Again the buffer time graphs show that Spray and Wait provided the worst performance of the algorithms. It also shows that for all protocols, the average time spent in a buffer was far less when under grey-hole attacks than when dealing with black holes. This was presumably due to the fact that grey holes only dropped half of the messages and, as such, messages were forwarded from other nodes faster, spending less time in their buffer.

Similarly to the latency graph, for black holes the Spray and Wait graph was exponential. Having swapped black holes for grey, the graph increased linearly which suggested that the attack had far less impact on the network. It however was still the only protocol that the buffer time appeared to be affected by attacks.

Interestingly, the Maxprop algorithm was the only one to consistently outperform the benchmark Epidemic protocol. This showed that the buffer time was not the primary concern when the other algorithms were being developed.

10. Discussion

Running the scenario at different total times also allowed some insights into how overall time impacts certain metrics. From the data gathered during this experiment, it can be seen that the longer the attack time, the higher the chance of delivery, the higher the latency, and for Spray and Wait the lower the buffer time. The investigation of the effect of an increasing overall attack time on the buffer time of individual nodes looks like an interesting area for further research because, if using the mean, an increase in time indicated an increase in the buffer time. However, if using the median, in the data from this experiment, it was the other way around, as the total attack time increases, the median buffer time decreased.

An interesting observation that can be made is that the delivery probability for the MaxProp and PRoPHET algorithm always increased between 10–15 black holes and then Maxprop significantly dropped again between 25–30. When comparing this to the latency graphs, it can also be seen that the latency was a direct opposite; it reduced between 10–15 and then rose again at 25–30. This could be because the black holes were clearing the network of some of the traffic. If the black holes were discarding messages that had been delivered but were still in the network, it was beneficial since as the traffic cleared, the messages traveled faster and were therefore more likely to be delivered. This affected the more methodical protocols, as for Spray and Wait the messages were replicated so many times (during the spray phase) that there was always an excess of messages travelling through the network.

The difference between the effect of the two types of attacking nodes (grey and black holes) was mostly seen in the delivery probability. Black-hole nodes confidently disrupted the network and reduced the delivery probability for the Maxprop and Spray and Wait algorithms. From the results collected it appeared that for all of the protocols, the introduction of grey-hole nodes increased the probability of a message being delivered. This is probably most likely due to the fact that as messages were dropped, traffic in the network was cleared. Given that it was only dropping half of the messages, it is also likely that as messages are being replicated a copy of the message was still traversing the network and would be delivered by a different node. This means that grey holes could be used by attackers to gather data from the messages without being as detectable.

The most surprising difference between the two type of attacking nodes was in the buffer time graphs. It was here that for the Spray and Wait algorithm, the trend line changed from quadratic to linear. This was not the expectation as, given the nodes perform the same function, just at half the rate, it would be a valid assumption that the rate would be different by a factor of two rather than a factor of x.

To enhance the research further, the attacking nodes could introduce some vector alterations in order to trick the other nodes into sending more data. By increasing their own vector, other nodes think that they are the best route to the destination node. This is a

more realistic and complex method of trying to disrupt a network, as that way the black hole nodes receive and drop more data.

11. Conclusions

Before the introduction of any attacking nodes, the biggest difference in the results was between the Maxprop and Spray and Wait protocols. The biggest trade-off was between speed and probability of delivery. In a scenario where speed was more important than delivery and there was an unlimited number of resources, Spray and Wait would be better suited. This would most likely be in a scenario where if the data delivery was slow then it would be useless, such as information about a traffic jam on a motorway. In that scenario, if a node received the data too slowly, then the likelihood was that they would already be a part of the jam so it would be a pointless message. In scenarios where delivery probability was more important than speed of receipt, MaxProp would be the preferred protocol. The PRoPHET algorithm never outperformed the two protocols, however it could provide a happy medium between the two. This meant that it could be used in situations whether neither of the others were suitable. For example, if a protocol is needed that was faster than Maxprop but not as resource-heavy as Spray and Wait, it would be ideal, provided the use case allows for a lower delivery probability than either of the others.

All of the protocols were affected by the introduction of black-hole nodes into the network. Spray and Wait is more vulnerable to the attacks as delivery probability reduces at a slightly faster rate than MaxProp. PRoPHET appears to deliver more messages with an increased number of black holes suggesting that it is more resistant to the attacks. This is probably due to the methodical way the protocol replicates messages, so reducing the traffic in the network is beneficial. The metric that was most altered during the experiment was the buffer time of the Spray and Wait algorithm. This was one of a few trend lines that were not linear. The quadratic curve meant that if even more black holes were introduced into the network then the rate of increase would continue to get increasingly larger, until eventually it would become unmanageable. This contrasts sharply with Maxprop and PRoPHET where the buffer time was unchanged by the presence of black holes, meaning the protocols were more resilient to black-hole attacks from both a delivery probability and (in particular) a latency perspective. As PRoPHET demonstrated the highest resilience to attacks, in an application where it was highly likely to undergo attack, it should be the protocol of choice. As in this experiment, the network is safety-critical; it is imperative that the network performs predictably. In this instance, the PRoPHET protocol should be used with messages being sent multiple times in order to increase the delivery probability. This would make the network more likely to deliver messages when performing normally and under attack.

12. Future Work

There is potential for the methods and results used to be expanded upon in future work. In order to investigate how secure each node is (meaning how easy it is to corrupt a vehicle) different hardware and methods of communication could be used. One previously theorised, tested and accepted method would be using Raspberry Pis to send messages over Wi-Fi [46]. This would be cheap to implement and could provide insight into how the capability of different hardware and methods of communication affect the ability to deal with cybersecurity attacks.

Another interesting area for further research would be combining the results of the experiments with some other proposed security measures. Using obfuscation algorithms is not uncommon in MANETS in order to provide nodes with anonymity [47]. A novel approach to increase security and allow nodes to remain anonymous is to use a reputation-aware network combined with obfuscation algorithms [48]. If this were to be combined with a protocol already providing some resistance to cybersecurity attacks (e.g. Maxprop) it might provide a very powerful, secure network for sending messages.

The results collected show that for all algorithms the introduction of grey holes increases the delivery probability. This could be due to the reduced traffic within the network. Further research could be carried out to confirm the reasoning and introduce the changes into the algorithms. If it is the reduced traffic then maybe all algorithms should contain methods of removing redundant copies.

Dependent upon the application of the given network, there may be other considerations that affect the likelihood of attacks occurring. For example, if users had to 'opt-in' to allow their mobiles to participate, then the entire city would not be involved (as many would decline). This would make attacks easier as the overall population of nodes would be fewer, meaning the proportion of attacking nodes would be easier to modify. A survey could be conducted to see the proportion of users that might be interested in participating and therefore the realistic proportions of attacking nodes possible.

Another area for further research which is out of the scope for this paper is congestion awareness. For experiments requiring networks which are congestion-aware protocols are available that include metrics for awareness. Both congestion awareness without replication (Cafe) [49] and congestion awareness with adaptive replication (CafREP) [50] provide many real-life solutions to problems. By using urban sensing, monitoring and processing, rich data from humans and vehicles can be applied to different business solutions such as advertising, and required services such as rubbish disposal and on-street parking.

Author Contributions: Conceptualization, C.B. and M.R.; methodology, C.B. and M.R.; software, C.B.; validation, C.B.; formal analysis, C.B. and M.R.; investigation, C.B. and M.R.; resources, C.B. and M.R.; data curation, C.B. and M.R.; writing—original draft preparation, C.B. and M.R.; writing—review and editing, C.B. and M.R.; visualization, C.B. and M.R.; supervision, M.R.; project administration, M.R.; funding acquisition, M.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data sets, configuration setting and code a available from the Nottingham Worktribe repository: https://nottingham-research.worktribe.com/record.jx?recordid=8950898 accessed on 6 June 2022.

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

References

1. Tripp-Barba, C.; Zaldívar-Colado, A.; Urquiza-Aguiar, L.; Aguilar-Calderón, J.A. Survey on Routing Protocols for Vehicular Ad Hoc Networks Based on Multimetrics. *Electronics* **2019**, *8*, 1177. [CrossRef]
2. Coll-Perales, B.; Pescosolido, L.; Gozalvez, J.; Passarella, A.; Conti, M. Next generation opportunistic networking in beyond 5G networks. *Ad Hoc Netw.* **2021**, *113*, 102392. [CrossRef]
3. Hogie, L.; Bouvry, P.; Guinand, F. An overview of manets simulation. *Electron. Notes Theor. Comput. Sci.* **2006**, *150*, 81–101. [CrossRef]
4. Hu, Y.C.; Patel, M.; Sabella, D.; Sprecher, N.; Young, V. *Mobile Edge Computing—A Key Technology Towards 5G*; ETSI White Paper; 2015; Volume 11, pp. 1–16. Available online: https://www.etsi.org/images/files/etsiwhitepapers/etsi_wp11_mec_a_key_technology_towards_5g.pdf (accessed on 6 June 2022).
5. Humayun, M.; Almufareh, M.F.; Jhanjhi, N.Z. Autonomous Traffic System for Emergency Vehicles. *Electronics* **2022**, *11*, 510. [CrossRef]
6. Zeadally, S.; Hunt, R.; Chen, Y.S.; Irwin, A.; Hassan, A. Vehicular ad hoc networks (VANETS): Status, results, and challenges. *Telecommun. Syst.* **2012**, *50*, 217–241. [CrossRef]
7. Kathiriya, H.; Kathiriya, N.; Bavarva, A. Review on V2R Communication in VANET. In Proceedings of the International Conference on Innovations in Automation and Mechatronics Engineering, Vallabh Vidyanagar, India, 21–23 February 2013; pp. 21–23.
8. Bekmezci, I.; Sahingoz, O.K.; Temel, Ş. Flying ad-hoc networks (FANETs): A survey. *Ad Hoc Netw.* **2013**, *11*, 1254–1270. [CrossRef]
9. Sahingoz, O.K. Networking models in flying ad-hoc networks (FANETs): Concepts and challenges. *J. Intell. Robot. Syst.* **2014**, *74*, 513–527. [CrossRef]

10. Radenkovic, M.; Huynh, V.S.H.; John, R.; Manzoni, P. Enabling real-time communications and services in heterogeneous networks of drones and vehicles. In Proceedings of the 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Barcelona, Spain, 21–23 October 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6.
11. Mangrulkar, R.; Atique, M. Routing protocol for delay tolerant network: A survey and comparison. In Proceedings of the 2010 International Conference on Communication Control and Computing Technologies, Nagercoil, India, 7–9 October 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 210–215.
12. Fall, K. A delay-tolerant network architecture for challenged internets. In Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Karlsruhe, Germany, 25–29 August 2003; pp. 27–34.
13. Huang, C.M.; Lan, K.C.; Tsai, C.Z. A survey of opportunistic networks. In Proceedings of the 22nd International Conference on Advanced Information Networking and Applications Workshops (Aina Workshops 2008), Gino-wan, Japan, 25–28 March 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 1672–1677.
14. Gupta, B.; Agrawal, D.P. *Handbook of Research on Cloud Computing and Big Data Applications in IoT*; IGI Global: Hershey, PA, USA, 2019.
15. Rawat, A.; Sharma, S.; Sushil, R. VANET: Security attacks and its possible solutions. *J. Inf. Oper. Manag.* **2012**, *3*, 301.
16. Radenkovic, M.; Crowcroft, J.; Rehmani, M.H. Towards low cost prototyping of mobile opportunistic disconnection tolerant networks and systems. *IEEE Access* **2016**, *4*, 5309–5321. [CrossRef]
17. Buchegger, S.; Mundinger, J.; Le Boudec, J.Y. Reputation systems for self-organized networks. *IEEE Technol. Soc. Mag.* **2008**, *27*, 41–47. [CrossRef]
18. Zakhary, S.R.; Radenkovic, M. Reputation-based security protocol for MANETs in highly mobile disconnection-prone environments. In Proceedings of the 2010 Seventh International Conference on Wireless On-Demand Network Systems and Services (WONS), Kranjska Gora, Slovenia, 3–5 February 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 161–167.
19. Gautham, P.S.; Shanmughasundaram, R. Detection and isolation of Black Hole in VANET. In Proceedings of the 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kerala, India, 6–7 July 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1534–1539.
20. Cong, L.W.; He, Z. Blockchain disruption and smart contracts. *Rev. Financ. Stud.* **2019**, *32*, 1754–1797. [CrossRef]
21. Grover, J. Security of Vehicular Ad Hoc Networks using blockchain: A comprehensive review. *Veh. Commun.* **2022**, *34*, 100458. [CrossRef]
22. Lin, I.C.; Liao, T.C. A survey of blockchain security issues and challenges. *Int. J. Netw. Secur.* **2017**, *19*, 653–659.
23. Dhurandher, S.K.; Singh, J.; Nicopolitidis, P.; Kumar, R.; Gupta, G. A blockchain-based secure routing protocol for opportunistic networks. *J. Ambient. Intell. Humaniz. Comput.* **2022**, *13*, 2191–2203. [CrossRef]
24. Ku, I.; Lu, Y.; Gerla, M.; Gomes, R.L.; Ongaro, F.; Cerqueira, E. Towards software-defined VANET: Architecture and services. In Proceedings of the 2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET), Piran, Slovenia, 2–4 June 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 103–110.
25. Dressler, F.; Pannu, G.S.; Hagenauer, F.; Gerla, M.; Higuchi, T.; Altintas, O. Virtual edge computing using vehicular micro clouds. In Proceedings of the 2019 International Conference on Computing, Networking and Communications (ICNC), Honolulu, HI, USA, 18–21 February 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 537–541.
26. Hagenauer, F.; Sommer, C.; Higuchi, T.; Altintas, O.; Dressler, F. Vehicular micro clouds as virtual edge servers for efficient data collection. In Proceedings of the 2nd ACM International Workshop on Smart, Autonomous, and Connected Vehicular Systems and Services, Snowbird, UT, USA, 16 October 2017; pp. 31–35.
27. Yi, S.; Li, C.; Li, Q. A survey of fog computing: Concepts, applications and issues. In Proceedings of the 2015 Workshop on Mobile Big Data, Hangzhou, China, 21 June 2015; pp. 37–42.
28. Nobre, J.C.; de Souza, A.M.; Rosário, D.; Both, C.; Villas, L.A.; Cerqueira, E.; Braun, T.; Gerla, M. Vehicular software-defined networking and fog computing: Integration and design principles. *Ad Hoc Netw.* **2019**, *82*, 172–181. [CrossRef]
29. Vahdat, A.; Becker, D. *Epidemic Routing for Partially Connected Ad Hoc Networks*; Technical Report CS-200006; Duke University: Duhram, NC, USA, 2000.
30. Spyropoulos, T.; Psounis, K.; Raghavendra, C.S. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, Philadelphia, PA, USA, 26 August 2005; pp. 252–259.
31. Sok, P.; Tan, S.; Kim, K. PRoPHET routing protocol based on neighbor node distance using a community mobility model in delay tolerant networks. In Proceedings of the 2013 IEEE 10th International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing, Zhangjiajie, China, 13–15 November 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1233–1240.
32. Burgess, J.; Gallagher, B.; Jensen, D.D.; Levine, B.N. MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. In Proceedings of the IEEE INFOCOM 2006. 25th IEEE International Conference on Computer Communications, Barcelona, Spain, 23–29 April 2006; Volume 6, pp. 1–11.
33. Ali, A.H.K.; Lenando, H.; Chaoui, S.; Alrfaay, M.; Tawfeek, M.A. A Dynamic Resource-Aware Routing Protocol in Resource-Constrained Opportunistic Networks. *CMC-Comput. Mater. Contin.* **2022**, *70*, 4147–4167.
34. Singh, A.K.; Bera, T.; Pamula, R. PRCP: Packet replication control based prophet routing strategy for delay tolerant network. In Proceedings of the 2018 4th International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, India, 15–17 March 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–5.

35. Han, S.D.; Chung, Y.W. An improved PRoPHET routing protocol in delay tolerant network. *Sci. World J.* **2015**, *2015*, 623090. [CrossRef]
36. Huang, T.K.; Lee, C.K.; Chen, L.J. Prophet+: An adaptive prophet-based routing protocol for opportunistic network. In Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, WA, Australia, 20–23 April 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 112–119.
37. Das, M.; Sarkar, S.; Iqbal, S.M.A. TTL based MaxProp routing protocol. In Proceedings of the 2016 19th International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 18–20 December 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 7–12.
38. Radhika, M.; Raj, D.; Ramesh, M.V. Comparison and analysis of opportunistic delay tolerant network protocols for off-shore communication. *Int. J. Comput. Sci. Inf. Secur.* **2016**, *14*, 338.
39. Quddus, M.A.; Wang, C.; Ison, S.G. *Road Traffic Congestion and Crash Severity: Econometric Analysis Using Ordered Response Models*; © American Society of Civil Engineers: Reston, VA, USA, 2010.
40. Wang, C.; Quddus, M.; Ison, S. A spatio-temporal analysis of the impact of congestion on traffic safety on major roads in the UK. *Transp. A Transp. Sci.* **2013**, *9*, 124–148. [CrossRef]
41. Türkoğlu, M.; Polat, H.; Koçak, C.; Polat, O. Recognition of DDoS Attacks on SD-VANET Based on Combination of Hyperparameter Optimization and Feature Selection. *Expert Syst. Appl.* **2022**, *203*, 117500. [CrossRef]
42. Arun Raj Kumar, P. Detection and Mitigation of Smart Blackhole and Gray Hole Attacks in VANET Using Dynamic Time Warping. *Wirel. Pers. Commun.* **2022**, *124*, 931–966.
43. Dhanaraj, R.K.; Islam, S.K.; Rajasekar, V. A cryptographic paradigm to detect and mitigate blackhole attack in VANET environments. *Wireless Netw.* **2022**, 1–16. [CrossRef]
44. Yasin, A.; Abu Zant, M. Detecting and isolating black-hole attacks in MANET using timer based baited technique. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 9812135. [CrossRef]
45. Keränen, A.; Ott, J.; Kärkkäinen, T. The ONE Simulator for DTN Protocol Evaluation. In Proceedings of the 2nd International Conference on Simulation Tools and Techniques (Simutools '09), Rome, Italy, 2–6 March 2009; ICST (Institute of Computer Sciences, Social-Informatics and Telecommunications Engineering): Brussels, Belgium, 2009; pp. 1–10. [CrossRef]
46. Radenkovic, M.; Milic-Frayling, N. RasPiPCloud: A light-weight mobile personal cloud. In Proceedings of the 10th ACM MobiComWorkshop on Challenged Networks, Paris, France, 11 September 2015; pp. 57–58.
47. Zakhary, S.; Benslimane, A. On location-privacy in opportunistic mobile networks, a survey. *J. Netw. Comput. Appl.* **2018**, *103*, 157–170. [CrossRef]
48. Radenkovic, M.; Benslimane, A.; McAuley, D. Reputation aware obfuscation for mobile opportunistic networks. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *26*, 230–240. [CrossRef]
49. Eswaran, S.; Misra, A.; Bergamaschi, F.; La Porta, T. Implementation of utility-based resource optimization protocols on ITA Sensor Fabric. In Proceedings of the Ground/Air Multi-Sensor Interoperability, Integration, and Networking for Persistent ISR, Orlando, FL, USA, 6–9 April 2010; SPIE: Bellingham, WA, USA, 2010; Volume 7694, pp. 235–242.
50. Das, S.K.; Bhattacharya, A.; Roy, A.; Misra, A. Managing Location in "Universal" Location-Aware Computing. In *Wireless Internet Handbook: Technologies, Standards, and Application*; CRC Press, Inc.: Boca Raton, FL, USA, 2003; pp. 407–425.