



Article

On the Potential of Enhancing Delay-Tolerant Routing Protocols via Age of Information

Georgios Kallitsis ¹, Vasileios Karyotis ^{2,*} and Symeon Papavassiliou ^{1,*}

¹ School of Electrical & Computer Engineering, National Technical University of Athens, 15780 Athens, Greece; giorgoskallitsis99@gmail.com

² Department of Informatics, Ionian University, Tsirigoti Sq. 7, 49100 Corfu, Greece

* Correspondence: karyotis@ionio.gr (V.K.); papavass@mail.ntua.gr (S.P.);
Tel.: +30-266-108-7869 (V.K.); +30-210-772-2550 (S.P.)

Abstract: In this paper, we study the potential of using the metric of Age of Information (AoI) for enhancing delay-tolerant routing protocols. The latter have been proposed for alleviating the impact of long roundtrip time in networks operating in harsh environments, e.g., in distributed applications deployed in a desert/sparsely populated area without infrastructure, a space network, etc. Delay-tolerant routing protocols can prevent excessive packet timer expiration, but do not provide any packet delivery time guarantee. Thus, they are unsuitable for time-sensitive applications that are more intensely desired nowadays in the next generation networking applications. By incorporating AoI into the operation of delay-tolerant routing protocols, we aim at devising routing protocols that can cope with both long propagation times and challenges related to time-sensitivity in packet delivery. More specifically, in this work, we modify the operation of a well-known delay-tolerant routing protocol, namely FRESH, to make AoI-based packet forwarding decisions, aiming at achieving specific delay guarantees regarding the end-to-end delivery time. We investigate the advantages and disadvantages of such an approach compared to the traditional FRESH protocol. This work serves as a cornerstone for successfully demonstrating the potential of exploiting AoI in delay-tolerant routing and its applications.

Keywords: age of information; delay-tolerant networks; distributed routing protocols; packet delivery time



Citation: Kallitsis, G.; Karyotis, V.; Papavassiliou, S. On the Potential of Enhancing Delay-Tolerant Routing Protocols via Age of Information. *Future Internet* **2022**, *14*, 242. <https://doi.org/10.3390/fi14080242>

Academic Editors: Valderi R. Q. Leithardt, Paul Crocker and Raúl García Ovejero

Received: 21 July 2022

Accepted: 13 August 2022

Published: 17 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Delay-tolerant (DT) routing protocols [1,2] have been proposed to alleviate the negative effect of long end-to-end delay emerging in harsh communications environments, e.g., in space communications, sparsely populated areas, networks deployed in a desert, etc. Under the TCP/IP protocol stack, long propagation times cause packet timers to expire, leading to unnecessary packet retransmissions and consequently reducing the throughput of the system. The delay-tolerant approach employs a store-carry-forward strategy [3,4], compared to the traditional store-and-forward one [5,6]. It relies on exploiting the mobility of the network by having nodes carry a packet until they get closer to the end destination. Typically the packet is forwarded to a node closer to the end destination or the destination itself if found in proximity. This means that even though packet timer expiration can be prevented, at least to a satisfactory degree, by relaxing the packet expiration intervals the average end-to-end packet delivery delay is prolonged, and in principle, it could become unbounded. Consequently, it is necessary to address such deficiency, in order to allow proper operation of such protocols for a larger gamut of applications in harsh environments.

Several works have been proposed that attempt to address the deficiencies of the traditional DT-routing protocols, mainly trying to improve the algorithms proposed in the literature in the fields of energy consumption [7–10] and congestion control [11]. Some

are trying to take advantage of the social link awareness [12], while others are focused mainly on Vehicular Delay-Tolerant Networks (VDTNs), being concentrated on the field of Messages' Delivery Time [13]. However, none of the aforementioned works focuses on providing packet delivery time guarantees for delay-tolerant routing protocols, which is the main novelty of the current work.

In this paper, we take a different approach. We capitalize on the metric of the Age of Information [14,15], a time-based metric proposed in the last decade to measure the "freshness" of a piece of information, e.g., a sensor reading, a packet, etc. We suggest using this metric to enhance the operation of DT routing protocols, by bounding the end-to-end (e2e) packet delivery delay, and eventually the average end-to-end packet delivery delay. This feature can potentially enhance the operation of traditional DT routing protocols, allowing them to be applied even in time-sensitive applications.

We demonstrate the above approach utilizing a specific DT routing protocol, namely FRESH [16]. We utilize AoI in the traditional operation of the above routing protocol, but also in two variations proposed in this work. We compare the results of the operation of all protocols with respect to several metrics and identify an emerging tradeoff: the enhanced protocols can achieve a notably bounded average packet delivery delay, at a very small cost of increased packet drop rate, compared to the zero packet drop rate and unbounded packet delivery delay of FRESH. Thus, the proposed enhancement demonstrates the efficacy of utilizing AoI for such routing protocols and paves the way for a broader spectrum of delay-tolerant protocol enhancements.

The rest of this paper is organized as follows. Section 2 reviews relevant literature, while Section 3 presents the protocols of interest and their enhancement. Section 4 presents evaluation results and discusses their impact, while Section 5 provides an overall discussion of the outcomes from this work. Finally, Section 6 concludes the paper.

2. Related Work

2.1. Age of Information

The concept of Age of Information (AoI) [17] was introduced in the early 2010s, in order to quantify the freshness of the knowledge that exists about a process observed from distance. More specifically, let a system consist of two nodes, namely a source and a destination node. For ease of tracking the mathematical symbols used in the following, Table 1 summarizes the symbols used in the paper. Assume a stochastic process $X(t)$ is observed by the source node, extracting samples, which carry information about the state of the stochastic process at the source node. Each one of them must be transmitted via a communication channel to the destination node. In the transmitter of the source node, there is a buffer that stores the samples in the form of packets. Each packet contains not only the value of the stochastic process $X(t_i)$ at time t_i where the sample was extracted, but also the timestamp t_i itself.

The transmission of status updates can be studied according to the model shown in Figure 1. Suppose that the creation of a status update is modeled as a stochastic process with average rate λ and packets are transmitted at an average service rate μ to the destination. The concept of AoI refers to the freshness of the knowledge possessed by the destination about the status of the source node. This freshness measurement is quantified as the time elapsed since the last status update received from the destination. A typical definition of *Age of Information* is as follows:

Definition 1 (Age of Information). *Suppose a system consists of a source-destination communication pair. Let t'_k be the times at which status updates are received from the destination. At the current time ξ , the index of the most recent update is:*

$$N(\xi) = \max\{k | t'_k \leq \xi\}, \quad (1)$$

where the timestamp of the most recent update is:

$$u(\xi) = t_{N(\xi)}. \tag{2}$$

The Age of Information (AoI) of some information of the source, e.g., a packet carrying a value update, at destination is defined as the random process:

$$\Delta(t) = t - u(t). \tag{3}$$

Table 1. List of mathematical symbols used in this work.

Symbol	Interpretation
$X(t)$	Stochastic process observed by the source node
t_i	Timestamp of extraction of packet i
λ	Average rate of creation of status updates in the source node
μ	Average rate of of transmission of status updates to the destination node
t'_i	Timestamp at which status update i is received from the destination node
$N(t)$	Index of the most recent update at time t
$u(t)$	Timestamp of the most recent update at time t
$\Delta(t)$	Random process of Age of Information
Y_i	The i -th interarrival time, i.e., the time elapsed between the creation of update i and the previous update
T_i	The i -th system time, i.e., the time required to generate status update i from source until the receipt of this update from the destination node
Δ_T	Average age of a status information system, considering an observation period $(0, T)$
Δ	Average Age of Information

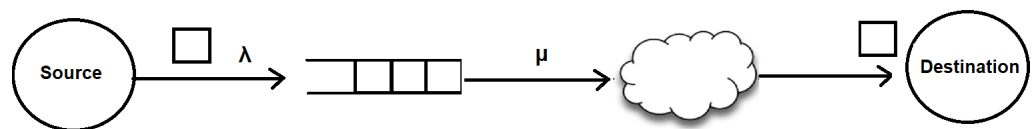


Figure 1. The basic model for Age of Information.

A sample path of the random process defined in Equation (3) exhibits a typical sawtooth form, illustrated in Figure 2. In our case, the AoI is a parameter defined for each separate packet and it describes the elapsed time, from the generation of the packet at the packet source until the current time instance Equation (3), wherever it is measured (the term destination actually refers to the point where AoI is measured). It shows the freshness of the packet from the same source at a point received in the network. AoI can be computed by inserting a timestamp when the packet is generated at the source, and each node holding the packet can compute the time difference between the current time and the packet’s timestamp (in DTN implementations such as DTN2, only one node holds a packet copy—such node is denoted as custodian). Figure 2 provides an indicative sample of such quantity for a single packet, measured at the destination. The term destination here denotes any destination computing the difference between the current time and the packet’s timestamp, namely AoI. Thus, each node computes the AoI of each packet in custody (received packets). Based on this, it will make a routing decision, as to whether it will further forward the packet or drop it because it is no longer considered as “fresh” enough. The intuition behind this is that if AoI grows too much, it might be more convenient to drop the packet for the sake of another fresher “packet” from the same source transferred in the network, thus saving resources, here saving on end-to-end delay, but also other metrics can be positively impacted, such as energy consumption. Given the above, Figure 2 shows a sample of the Age of Information from the perspective of a node receiving update packets from the same source, i.e., successive packets of, e.g., a sensor measurement, demonstrating the characteristic sawtooth pattern expected for AoI.

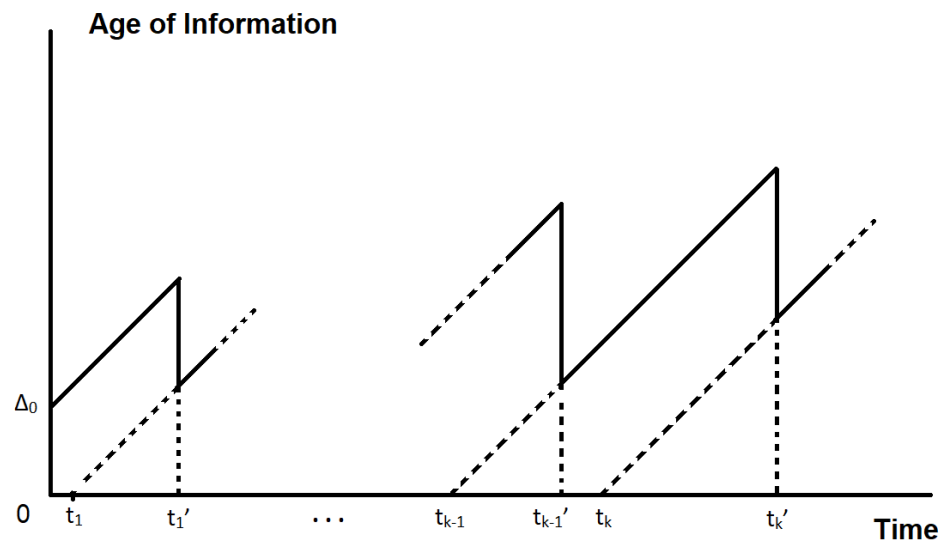


Figure 2. An indicative sample path of the process $\Delta(t)$ showing how AoI grows and drops upon receipt of an update. This would be the AoI measured by a node for packets from a specific source. Multiple such samples would be obtained at the same node for packets belonging to other flows from different sources (or the same source).

In order to understand more about the process $\Delta(t)$, we provide some more definitions: the i -th interarrival time Y_i is defined as the time elapsed between the creation of update i and the previous update, i.e., update $i-1$:

$$Y_i = t_i - t_{i-1}. \tag{4}$$

In addition, system time i , denoted by T_i , corresponds to the sum of queue time and service time of update i , and it is defined as the time required to generate status update i from source until the receipt of this update from the destination:

$$T_i = t'_i - t_i. \tag{5}$$

Assuming that the procedure $\{\Delta(t)\}$ is ergodic (a stochastic process is characterized as ergodic if the ensemble average of an observable sample equals the time average. In such case, any collection of random samples from the stochastic process represents the average statistical properties of the entire process), the average age can be calculated using a time average. Considering an observation period $(0, T)$, the average age of a status information system is calculated as:

$$\Delta_T = \frac{1}{T} * \int_0^T \Delta(t) dt. \tag{6}$$

Taking the length of the observation interval T to infinity and observing Figure 2, it is proven that the expression for the average Age of Information is:

$$\Delta = \frac{E[Y * T] + E[Y^2]/2}{E[Y]}, \tag{7}$$

where it is stated that $\lambda = \frac{1}{E[Y]}$ is the steady state rate of creating status updates and $E[.]$ is the mean value operator.

2.2. Delay Tolerant Networks

A special category of networks, referred to as Delay Tolerant Networks (DTNs), describes networks capable of managing long end-to-end delays and potential data losses more effectively in a variety of disrupted scenarios and environments [1,2,18]. Such envi-

ronments could be emergency communication infrastructures following a disaster, space communications, battlefield communication networks, earthquake and extreme terrestrial environments, etc. The alternative term disruption-tolerant network is also occasionally used. The main feature of such networks is that they may not have permanent connectivity between any two nodes.

DTN differs from the traditional Internet in terms of the transmission of the information. More specifically, in the traditional Internet a store-and-forward mechanism [5,6] is used in order to transmit the desired information, in which packets are forwarded to an intermediate node where they are stored and sent at a later time either to the final destination or to another intermediate node. At the intermediate stop, the packet integrity is verified before further forwarding (by analyzing packet headers). On the other hand, the mobility in DTN essentially imposes a different approach called store-carry-forward mechanism [3,4], in which a node sending a packet stores it in its buffer. The packet is forwarded to another node when the two nodes are in communication proximity, and if a pre-defined forwarding condition, e.g., distance to final destination is reduced, is met. Through opportunistic contacts and a series of intermediate forwarding, the packet can be delivered to its final destination.

Some of the key features of DTN are the following [1,18]:

1. **Disconnection and variety in mobility patterns.** In the DTN setting, it is frequently impossible to establish an end-to-end connection or observe deterministic mobility patterns. Stochastic node mobility means that any two nodes may not eventually meet for a long time.
2. **Long delay in queues.** Due to the fact that link disruption is very common in DTN, nodes must temporarily store messages, and queuing time can vary from seconds to minutes, even days.
3. **Limited resources.** DTN nodes typically have limited memory, as well as limited processing power. This results in the use of special routing protocols, which are significantly different from the routing protocols used in the traditional Internet.
4. **High rate of data transmission errors.** The fact that the environment in DTN is changing so dynamically as well as the unknown mobility of DTN nodes results in a higher probability of packet loss than in traditional communication networks.

Routing can be a major challenge in DTNs due to the first feature mentioned above. Various works proposing routing solutions have been proposed to date. Considering them, DTN routing protocols can be segregated into four main categories, on the basis of the employed knowledge to determine the next hop node: *encounter-based*, *time-based*, *infrastructure-based* and *hybrid and other types of routing* [1]. They are explained briefly in the following:

- **Encounter-based routing:** This category includes protocols that exploit the meeting history between nodes for making packet forwarding decisions. When two nodes are in communication range, they are considered to have an “encounter”. In the protocols of this category, each node maintains the history of such encounters. Indicative examples are the Epidemic Routing Protocol [19–21], PROPHET [22,23] and FRESH [16].
- **Time-based routing:** The corresponding protocols determine the next hop node based on any time-related metric, such as time interval, contact duration, time between encounters, contact time, etc. Protocols which fall into the above category are Contact Duration Based Routing (CDBR) [24,25], SEDUM [26] and Seek-and-Focus Protocol [27].
- **Infrastructure-based routing:** These protocols exploit infrastructure information to determine the next hop. Typically they use node location, determined routes, map information, or node traffic direction, to make the forwarding decisions. Protocols which fall into the above category are MOVement of VEHICLE (MOVE) Protocol [28], Distance Aware Epidemic Routing (DAER) [29] and Source Routing [30].
- **Hybrid and other types of routing:** This category includes routing protocols that may combine the approaches of the above categories (hybrid), or other protocols following

approaches that cannot be classified in the above. Such protocols are Neighborhood contact history routing (NECTAR) [31], Erasure Coding [32] and Minimizing Relay node and Hop count (MRH) [33].

3. Delay-Tolerant Protocols Enhanced via Age of Information

3.1. Fresh Protocol

The FRESH protocol is a routing protocol widely used in DTNs. More specifically, it is a protocol belonging to the broader class of Mobile Ad-hoc Networks (MANET) [34] protocols, and due to the fact that it relies primarily on encounter times, it belongs to the Encounter-based routing protocol category mentioned above. Its operation is relatively simple and is briefly described in the following. Figure 3 summarizes its operation with a flowchart.

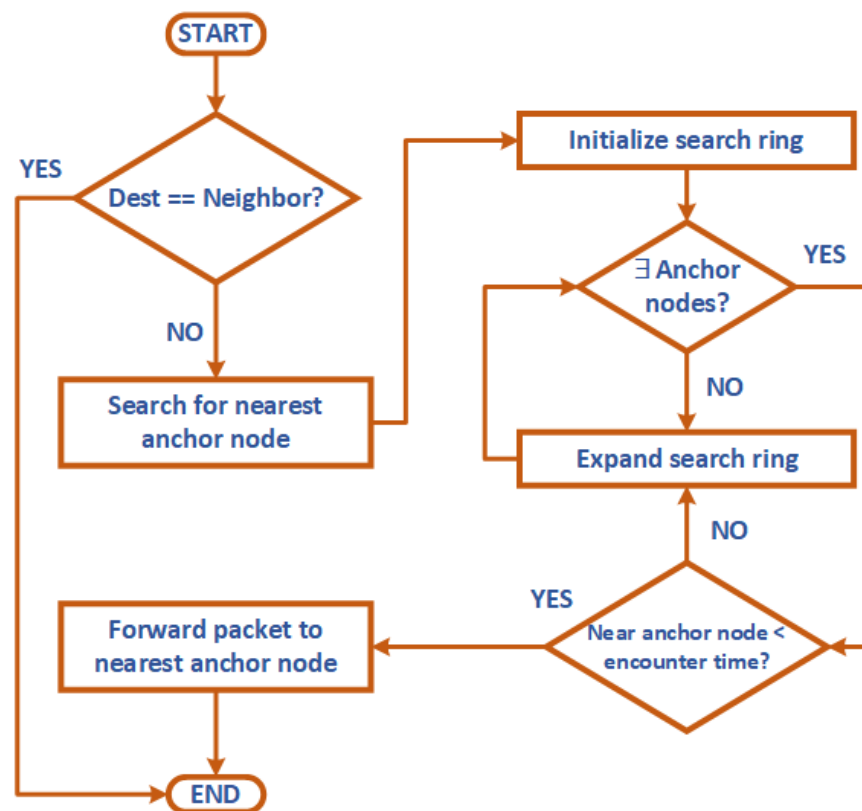


Figure 3. Flowchart of FRESH protocol.

Each node of the DTN network maintains a file (vector), which contains the most recent encounter times of this node with all other nodes in the network. In each step, instead of searching only for the destination node, the source node searches for “near anchor nodes” (the meaning of “near anchor nodes” will be better explained below) that had contact with the destination node more recently than the corresponding contact of the source node with the destination node. This step is then repeated until the destination successfully receives the message. Dubois-Ferriere et al. [16] have shown that the total path that each packet will follow does not contain a loop (loop-free), as at each step of the algorithm, the encounter age decreases monotonically until the destination node is found in a finite number of steps, with the sole condition that the source node and the destination node belong to a connected subset of nodes. In addition, it is obvious that the algorithm does not require any general knowledge of the network, so it can easily be implemented as a distributed system, because at each step, each node is only interested in its own local encounter table.

An encounter between two nodes occurs when these nodes are one-hop neighbors. Since the one-hop neighborhood depends on the link layer, the exact condition in order for an encounter to take place varies, as it depends on the respective wireless technology employed. For example, the 802.11b [35] connectivity range can be over 250 m, while Bluetooth allows only a few meters transmission range. The encounter age between 2 nodes is equal to the time that has elapsed since the most recent meeting of these nodes. The way to specify neighbors at any given time at a practical level varies again, depending on the choice of link layer, e.g., sending normal or Hello packets between neighbors (A Hello packet is a special control packet (message) periodically sent by a router in order to establish and verify network proximity/connectivity relationships with other routers. Hello packets are used primarily in the implementation of the OSPF (Open Shortest Path First) routing protocol, and more generally where there is a need to confirm proximity relationships between nodes of a network).

An interesting aspect of FRESH is the definition of the term “near anchor nodes” in each step of the algorithm. This term does not necessarily indicate that we are dealing only with neighboring nodes. More specifically, the algorithm implements the following operation: when a source node searches for a near anchor node that should have a smaller encounter time with the destination node than the corresponding encounter time of the source node with the destination node, it forms a ring of a specific radius centered on the source node and those nodes inside this ring are examined by the algorithm. If the radius of the ring is small enough so that the source node has no near anchor nodes or in the general case, no near anchor node satisfies the property of having an encounter time with the destination less than that of the source with the destination, then the search continues, appropriately increasing the radius of the ring, until the node that satisfies the above property is found. The way in which the ring radius grows, as well as the choice of the initial ring radius, vary depending on the occasion and the characteristics of the DTN network. It should be emphasized that the selection of the appropriate near anchor node is independent of the direction and is affected only by the distance of the source node from the candidate near anchor node. This procedure is depicted in Figure 4.

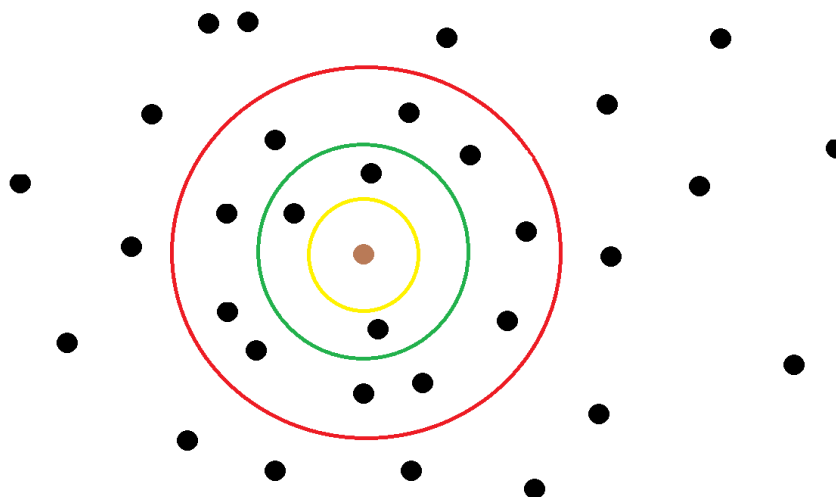


Figure 4. The procedure of formation of the rings for the finding of the “near anchor nodes”. The source node is coloured brown, while the initial radius of the ring is the yellow one, the second one is green and the biggest of all is red. In the first ring, the source node has no near anchor nodes at all, while in the second and third ring, it has 3 and 13 anchor nodes, respectively.

The pseudocode of the above algorithm is given below in Algorithm 1:

Algorithm 1 Pseudocode of FRESH Protocol [16].

```

Procedure FRESH(D):
if thisnode.ID = D then
    replyToSource()
else
     $T \leftarrow \text{prevEncounterAge}(D)$ 
     $A \leftarrow \text{findNextAnchor}(D, T)$ 
    if  $A \neq D$  then
         $\text{notifyNextAnchor}(A, D)$ 
    end if
end if

```

In the above, D is the destination node, and the pseudo-command call $\text{prevEncounterAge}(D)$ returns the encounter time of the node to which it was called with the node with an ID equal to D , i.e., how much time has elapsed since the last time the node with an ID equal to D was a direct neighbor of the node to which the **FRESH**(D) function is called. The $\text{findNextAnchor}(D, T)$ pseudo-command then returns the node with an ID equal to A that has most recently met with destination D relative to the value T , which is the encounter time of the source with the destination. This search will be conducted as described above. Finally, calling $\text{notifyNextAnchor}(A, D)$ will trigger the **FRESH**(D) function on the node with an ID equal to A . The $\text{notifyNextAnchor}(A, D)$ pseudo-command does not require the T parameter, as the node with an ID equal to A only needs its own encounter time with destination D .

In addition to the above original functionality of the FRESH routing protocol, in this paper, we suggest some variations, depending on the choice of the so-called “near anchor nodes” on a practical level:

FRESH-near-v1

An alternative design implementation has the source node to choose to deliver its packet to its “optimal” close neighbor, that is, the one within the specific radius ring that would have formed around the source node and have the smallest encounter time with the destination from all near anchor nodes of the source (and of course less than the encounter time of the source with the destination).

FRESH-near-v2

Another alternative is to perform the exact same procedure as before for ring formation but to deliver the packet to its first near anchor node that satisfies the above property and not necessarily to its optimal near anchor node.

FRESH-neigh

In addition, a different variation is the following: the selection of the next hop node to be based on the current neighbors of the source (based on the encounter times at the link level) and not based on the process of forming the circular rings, as explained above.

Naturally, tradeoffs are expected to be found between the above design implementations, as observed in the Simulation section, where the above variants are compared, as well as the modified version of the FRESH protocol, in which the concept of AoI is utilized.

At this point, we need to make an important note regarding the energy consumption of the studied protocols. Energy is a key (enabling) factor for mobile networks and DTNs in particular. Energy consumption depends on the transmission range of each device. The larger the transmission range, the greater the associated energy consumption. Thus, protocols such as FRESH-near-v2 will suffer from increased energy consumption, since the increase in the search ring diameter will lead to more energy consumption. This is something that should be taken into account explicitly, and will be deferred for an appropriate extension of this work (where AoI can be exploited for jointly improving energy and delay performance), as it is a problem of its own merit.

3.2. Age of Information Enhanced Routing

In this work, a variant of the FRESH protocol is proposed, as explained in detail. More specifically, it is obvious that during the implementation of this routing algorithm, the so-called “age” of the packet is not taken into account at all, i.e., the time interval from the moment of its production to the time of observation. Depending on the choice of various system parameters (neighborhood radius, number of nodes in the DTN network, initial ring formation radius when delivering packets, etc.), system packets are likely to circulate indefinitely in the network, wasting available resources, as well as gradually increasing their age. Nevertheless, in the real world of DTNs, it makes sense for packets to have a specific Time-To-Live (TTL) available, which acts indirectly as an expiration date. Therefore, if a packet exceeds the expiration date value without reaching the destination, then it is considered useless and should be dropped from the network. Furthermore, this is necessary in time-sensitive applications.

This idea is promoted in this work, where we incorporate in the FRESH protocol the concept of AoI, in order to impose a specific upper bound for the age of each packet. Thus, in the proposed modified protocol, the routing algorithm is similar to the one explained above. However, at the end of each time slot, the age of each packet remaining in the network increases by one time unit, and if the age of a packet exceeds the value of a specific threshold for AoI, the packet is dropped from the network. At first glance, with the modified FRESH protocol, one does not expect an increase in throughput, namely the ratio of successfully transmitted packets to the total number of packets generated in the network, as the proposed modified protocol does not differ substantially from the original FRESH protocol’s main function. Any differentiation of the throughput could be observed in more complex cases, such as in the case of specific buffer space constraints for each node. Here, for simplicity, it has been assumed that each node has an infinite buffer. The expected difference is in the age of the packets and is reflected in metrics, such as the average age of the packets, the maximum observed age, etc. Eventually, it is expected to find a tradeoff between the throughput and the average age of the packets.

4. Protocol Evaluation

In this section, we evaluate the performance of the proposed extensions and new protocols, and compare it with the original FRESH protocol. The goal is to demonstrate the feasibility of using AoI for meaningful purposes in DTN and quantify the obtained benefits/costs.

4.1. Evaluation Setup

For the simulation, the Python programming language was used in a Google Colab environment. All experiments were performed on a PC with an Intel(R) Core(TM) i5-7200U CPU@2.50 GHz, 6 GB available RAM and Windows 10 operating system. In order to make it easier to reproduce and verify the measurements made, the entire simulation code is available online in the Github repository [36].

In this work, three different versions of the FRESH protocol are simulated and for each of them the original and the modified version of the protocol are compared with each other by incorporating the concept of AoI, as explained in Section 3.2. The common elements of all six simulations performed are summarized below:

Initially, in each time slot 1–3 new packets are generated with equal probability and with random nodes as their source and destination. Subsequently, all nodes move according to the Random Waypoint Mobility Model [37] and for each node its current neighbors are found (it is checked if it is more or less than a predefined proximity radius value, and the Euclidean distance is used to measure the distance) and the encounter vectors are updated accordingly. For all simulations, the common assumption made is that in each time slot, each node can send at most one of the packets that it currently has stored in its buffer. More specifically, for the three versions concerning the original FRESH, for each node one of the packets it owns at that moment is randomly selected, while for the three modified versions

of FRESH, for each node it is chosen to transfer its packet to the network that has the highest AoI value at this time, in order to make use of the property of Age of Information (the older the age of a packet, the more critical it is to be transported in the network and delivered as soon as possible to the destination). Afterwards, each node in each of the versions of the algorithm chooses according to separate criteria what to do with the packet it chose in the previous step. Thus, it applies a certain reasoning in order to find the next hop of the packet to transmit at this time. At the end, it delivers this packet to the node it found from the above process and after this has been implemented by all nodes in the network, all packets that have not yet been delivered or dropped increase their age by 1. In the three versions of the Modified Protocol, if the age of a packet exceeds the threshold value, the respective packet is dropped from the network.

The three variants of the FRESH protocol, which are simulated both in their original version and in the modified version with the introduction of the AoI variable, are the following:

- (i) **Relay node selection based on neighbors:** in this variant, the node selects the relay node of the packet it has selected based on its current neighbors at the link layer. More specifically, if the destination of the packet belongs to the set of its neighbors at this time, then this is the best case, as it delivers the packet directly to its destination. Otherwise, if the destination does not belong to the set of its neighbors at this time, it looks at the encounter vectors of its neighbors and delivers the packet to the neighbor that had the most recent encounter with the destination. In case it has no neighbors at all, it keeps the packet stored in its buffer. This variant is hereafter denoted as **FRESH-neigh** or **Modified-neigh** depending on whether the original FRESH protocol is used or the one modified by introducing the AoI metric.
- (ii) **Relay node selection based on “best near anchor” node:** in this variant, when each node chooses the packet to forward in the network, it forms a circular ring of a certain radius centered on itself. Then, it searches for the nodes residing inside the ring it has just formed (we call these nodes from now on “near anchor nodes”) and then, it searches if there is a node close to it that has a more recent encounter than its own of the encounter with the destination of the packet. If no near anchor node satisfying this property is found, it forms a new larger ring and repeats the process until at least one near anchor node is found satisfying this property. In the present implementation, an initial ring radius equal to 10 has been chosen and if no node is found that satisfies the aforementioned property, the radius is doubled at each step. This variant is hereafter denoted as **FRESH-near-v1** or **Modified-near-v1** depending on whether the original FRESH protocol is used or the one modified by introducing the AoI metric.
- (iii) **Relay node selection based on “first near anchor” node:** this variant is exactly the same as the above with the difference that here the source node only needs to find one near the anchor node that has an encounter time with the destination more recent than its corresponding time with the destination. On the contrary, in the above variant, when a near anchor node is found that satisfies this property, all the near anchor nodes of the source (with the specific ring radius) are searched and among them the “optimal” one is selected, i.e., the nearest anchor node that has the most recent encounter time with the destination compared to all others near anchors. This variant is hereafter denoted as **FRESH-near-v2** or **Modified-near-v2**, depending on whether the original FRESH protocol is used or the one modified by introducing the AoI metric.

4.2. Simulation Parameterization

In this section, some initial results from the simulations are presented, which serve the purpose of properly dimensioning the parameters of the system. More specifically, starting from very small networks and increasing their size progressively, various parameters of the problem are modified accordingly, such as the number of DTN nodes, the proximity radius (at the link layer), the size of the square grid, the number of time slots of simulation

time and the Upper Bound for the AoI for the three versions of the modified protocol. For all cases, the metrics mainly considered are throughput (i.e., the percentage (%) of the successfully transmitted packets in the whole simulation), average age of packets and maximum age observed, while also keeping the total number of packets produced, the total number of successfully transmitted packets, the number of dropped packets (due to exceeding the AoI bound in the three versions of the modified protocol), the number of remaining packets in the network and finally the total execution time of each simulation. The results are presented in Tables 2–7 (it is emphasized that in the first column—Number of Nodes—the star superscript denotes that the Grid of the topology has size 100×100 , otherwise the size is 1000×1000):

Table 2. Initial results for Modified-neigh.

Number of Nodes	Time Slots	Radius of Proximity	Bound of AoI	Produced Packets	Dropped Packets	Successful Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
50 *	500	20	30	1012	132	860	20	2.55	0.849	12.5	30
50 *	1000	15	70	2021	380	1563	78	7.89	0.773	40.45	70
100 *	1000	15	70	1964	131	1780	53	17.24	0.906	23.76	70
100	1000	149	70	2001	148	1795	58	17.52	0.897	26.15	70
100	2000	149	140	4030	13	3965	52	58.63	0.983	27.12	140
100	5000	149	200	10,032	6	9961	65	356.28	0.992	29.87	200

Table 3. Initial results for Modified-near-v1.

Number of Nodes	Time Slots	Radius of Proximity	Bound of AoI	Produced Packets	Dropped Packets	Successful Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
50 *	500	20	30	998	686	257	55	3.564	0.257	25.07	30
50 *	1000	15	70	1997	105	1838	54	10.398	0.92	28.87	70
100 *	1000	15	70	1963	793	1083	87	25.2	0.551	47.83	70
100	1000	149	70	2004	848	1048	108	32.78	0.522	51.08	70
100	2000	149	140	4029	1548	2252	229	97.76	0.558	102.08	140
100	5000	149	200	10,014	4145	5558	311	457.35	0.555	150.45	200

Table 4. Initial results for Modified-near-v2.

Number of Nodes	Time Slots	Radius of Proximity	Bound of AoI	Produced Packets	Dropped Packets	Successful Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
50 *	500	20	30	998	681	267	50	3.32	0.267	25.51	30
50 *	1000	15	70	1993	349	1573	71	9.36	0.789	42.07	70
100 *	1000	15	70	1992	975	895	122	23.12	0.449	54.26	70
100	1000	149	70	1987	1069	798	120	30.06	0.4	54.29	70
100	2000	149	140	3947	2052	1680	215	84.69	0.42	111.77	140
100	5000	149	200	10,041	5514	4203	324	407.18	0.418	163.45	200

Table 5. Initial results for FRESH-neigh.

Number of Nodes	Time Slots	Radius of Proximity	Produced Packets	Successfull Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
50 *	500	20	1009	975	34	2.395	0.966	14.91	87
50 *	1000	15	1974	1888	86	7.328	0.956	47.48	438
100 *	1000	15	1983	1923	60	17.02	0.969	26.49	215
100	1000	149	2039	1993	46	17.3	0.977	26.4	226
100	2000	149	4049	3996	53	57.29	0.986	27.59	248
100	5000	149	9921	9870	51	324.78	0.994	27.739	315

Table 6. Initial results for FRESH-near-v1.

Number of Nodes	Time Slots	Radius of Proximity	Produced Packets	Successfull Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
50 *	500	20	1010	397	613	3.85	0.39	139.08	500
50 *	1000	15	2037	1959	78	9.52	0.961	25.9	347
100 *	1000	15	1971	1253	718	25.77	0.635	197.07	969
100	1000	149	1981	1128	853	37.92	0.569	234.57	999
100	2000	149	3983	2178	1805	100.29	0.546	421.88	1997
100	5000	149	10,141	5281	4860	484.86	0.52	1152.04	4986

Table 7. Initial results for FRESH-near-v2.

Number of Nodes	Time Slots	Radius of Proximity	Produced Packets	Successfull Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
50 *	500	20	997	299	698	3.482	0.299	158.49	498
50 *	1000	15	1970	1496	474	9.445	0.759	115.86	885
100 *	1000	15	2009	898	1111	23.67	0.446	295.49	999
100	1000	149	2016	726	1290	34.09	0.36	314.75	991
100	2000	149	3358	1563	2395	89.82	0.394	618.34	2000
100	5000	149	10,080	3706	6374	426.18	0.367	1571.19	5000

4.3. Larger-Scale Simulations

In this subsection, we present the results obtained from more extensive simulations, in which the behavior of the six versions studied in the work is examined by changing parameters, such as the number of DTN nodes, the number of time slots, as well as the value of the Bound for the AoI (for the three versions of the modified protocol). The results are shown below in Tables 8–13:

Table 8. Larger scale simulation for Modified-neigh.

Number of Nodes	Time Slots	Radius of Proximity	Bound of AoI	Produced Packets	Dropped Packets	Successful Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
100	5000	149	200	10,032	6	9961	65	356.28	0.992	29.87	200
250	5000	149	200	10,007	3	9955	49	943.18	0.994	22.98	200
500	5000	149	200	10,019	7	9965	47	2333.63	0.994	23.69	200
100	10,000	149	200	19,955	5	19,891	59	1374.09	0.996	29.87	200
100	15,000	149	200	30,130	16	30,056	58	3033.45	0.997	30.14	200
100	5000	149	150	10,023	37	9920	66	348.76	0.989	29.42	150
100	5000	149	100	9992	266	9667	59	344.61	0.967	29.07	100
100	5000	149	50	9891	1499	8343	49	3451.41	0.843	23.13	50

Table 9. Larger scale simulation for Modified-near-v1.

Number of Nodes	Time Slots	Radius of Proximity	Bound of AoI	Produced Packets	Dropped Packets	Successful Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
100	5000	149	200	10,014	4145	5558	311	457.35	0.555	150.45	200
250	5000	149	200	9976	7785	1817	374	1316.71	0.182	180.11	200
500	5000	149	200	9973	8857	731	385	3305.67	0.073	189.11	200
100	10,000	149	200	19,841	8223	11,310	308	1513.17	0.57	152.03	200
100	15,000	149	200	29,867	12,617	16,948	302	3285.21	0.567	153.8	200
100	5000	149	150	10,033	4225	5568	240	397.27	0.554	112.54	150
100	5000	149	100	10,081	4388	5525	168	401.49	0.548	74.95	100
100	5000	149	50	9975	5247	4641	87	379.72	0.465	38.48	50

Table 10. Larger scale simulation for Modified-near-v2.

Number of Nodes	Time Slots	Radius of Proximity	Bound of AoI	Produced Packets	Dropped Packets	Successful Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
100	5000	149	200	10,041	5514	4203	324	407.18	0.418	163.45	200
250	5000	149	200	9955	8074	1515	366	1458.19	0.152	183.85	200
500	5000	149	200	10,001	8974	634	393	3079.35	0.063	190.69	200
100	10,000	149	200	19,830	11,151	8325	354	1793.91	0.419	166.15	200
100	15,000	149	200	30,024	17,028	12,671	325	3090.86	0.422	166.93	200
100	5000	149	150	10,004	5522	4251	231	397.36	0.425	124.1	150
100	5000	149	100	10,012	5478	4365	169	393.27	0.435	82.42	100
100	5000	149	50	9981	5931	3968	82	397.98	0.397	41.22	50

Table 11. Larger scale simulation for FRESH-neigh.

Number of Nodes	Time Slots	Radius of Proximity	Produced Packets	Successfull Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
100	5000	149	9921	9870	51	324.78	0.994	27.739	315
250	5000	149	10,058	10,015	43	1131.49	0.995	22.54	228
500	5000	149	10,110	10,062	48	2156.49	0.995	23.63	296
100	10,000	149	20,104	20,052	52	1276.08	0.997	28.16	302
100	15000	149	29,957	29,911	46	2792.84	0.998	28.15	297

Table 12. Larger scale simulation for FRESH-near-v1.

Number of Nodes	Time Slots	Radius of Proximity	Produced Packets	Successfull Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
100	5000	149	10,141	5281	4860	484.86	0.52	1152.04	4986
250	5000	149	10,040	1790	8250	1289.87	0.178	2001.59	5000
500	5000	149	9981	835	9146	3361.1	0.08	2289.78	5000
100	10,000	149	20,121	10,592	9529	1640.18	0.526	2336.56	9995
100	15,000	149	29,851	15,719	14,132	3362.13	0.526	3514.33	14,995

Table 13. Larger scale simulation for FRESH-near-v2.

Number of Nodes	Time Slots	Radius of Proximity	Produced Packets	Successfull Packets	Remaining Packets	Execution Time (s)	Throughput	Mean AoI (Time Units)	Max AoI (Time Units)
100	5000	149	10,080	3706	6374	426.18	0.367	1571.19	5000
250	5000	149	10,078	1247	8831	1074.49	0.123	1294.85	5000
500	5000	149	9970	600	9370	3415.69	0.06	2360.14	5000
100	10,000	149	20,001	7296	12,705	1538.44	0.364	3111.45	9998
100	15,000	149	29,949	10,645	19,304	3275.26	0.355	4807.08	15,000

4.4. Cumulative Comparison

In this section, the main results of the above simulations are presented for the three versions of the modified protocol proposed in the work and the three versions of the original FRESH protocol that were studied. The two metrics that are examined are the throughput and the average age of the network packets. The execution time of the simulations is also studied. Initially, in order to analyze the effect of the number of time slots on the simulations, Figure 5 is constructed, capturing the dependence of the throughput on the change in the number of time slots for the four versions of FRESH (where each node delivers its packet to some “near anchor”).

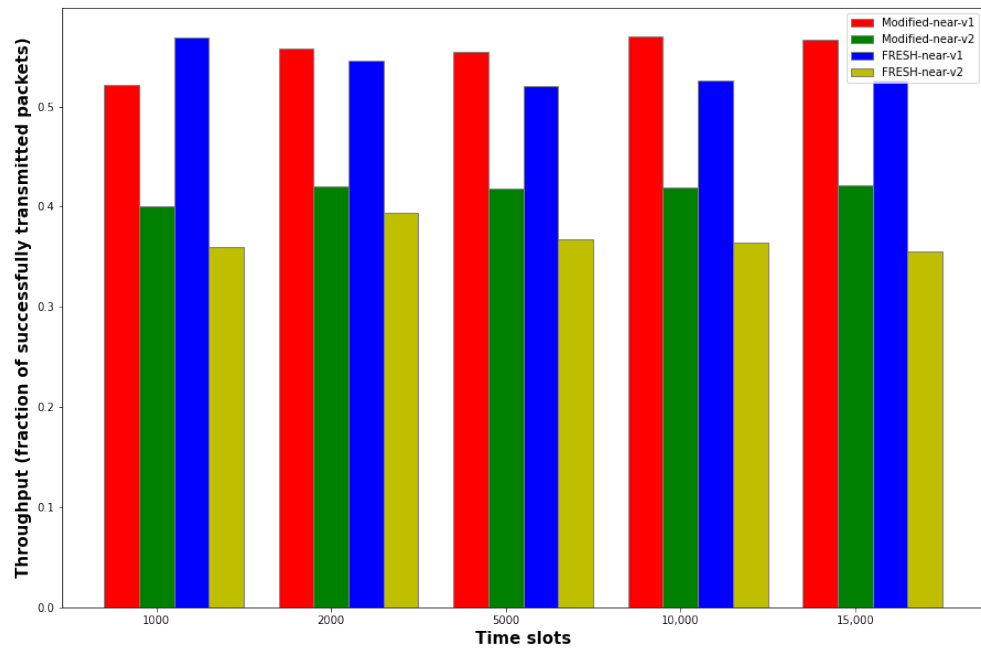


Figure 5. Dependence of throughput versus number of time slots for Modified-near-v1, Modified-near-v2, FRESH-near-v1 and FRESH-near-v2 protocols.

It is noted that the ratio of the Bound value for the AoI for the two modified versions of the FRESH protocol to the number of time slots selected in each case equals 7% for the first two time slots values, 4% for the price of 5000 time slots, 2% for the price of 10,000 time slots and 1.33% for the price of 15,000 time slots. From the above graph, it is possible to extract the number of time slots above which the so-called “transient effect” has completely disappeared, resulting in the throughput tending asymptotically to a constant value. For all four protocols, it seems that for a number of time slots greater than 5000, the throughput does not change significantly, so it is safe to conclude that the value of 5000 time slots is sufficient to eliminate the transient phenomena. Furthermore, for all time slots values, the Modified-near-v1 and FRESH-near-v1 protocols demonstrate the best performance, since in them, by definition, nodes search for their optimal “nearest anchor

node” and not just the first better that they encounter, therefore the chances of successful transmission of the packets are increased in these two protocols. In fact, it is observed that in 4/5 cases, the Modified-near-v1 protocol outperforms the FRESH-near-v1 even for a while, a result which certainly needs justification, as the modified versions of the FRESH protocol were proposed in this paper in order to improve the packet age domain and not (necessarily) the throughput, a result which is confirmed by conducting many of the same simulations and averaging the results. Furthermore, we should not forget the dependence of the throughput on the AoI Bound in the two versions of the modified protocol proposed in the work. A clearer conclusion about this dependence is drawn below by citing a related graph. However, a similar superiority is observed in the Modified-near-v2 protocol over FRESH-near-v2 in the field of performance.

Similarly, in Figure 6, we demonstrate the dependence of the average age of the packets with the change in the number of time slots for the four protocols.

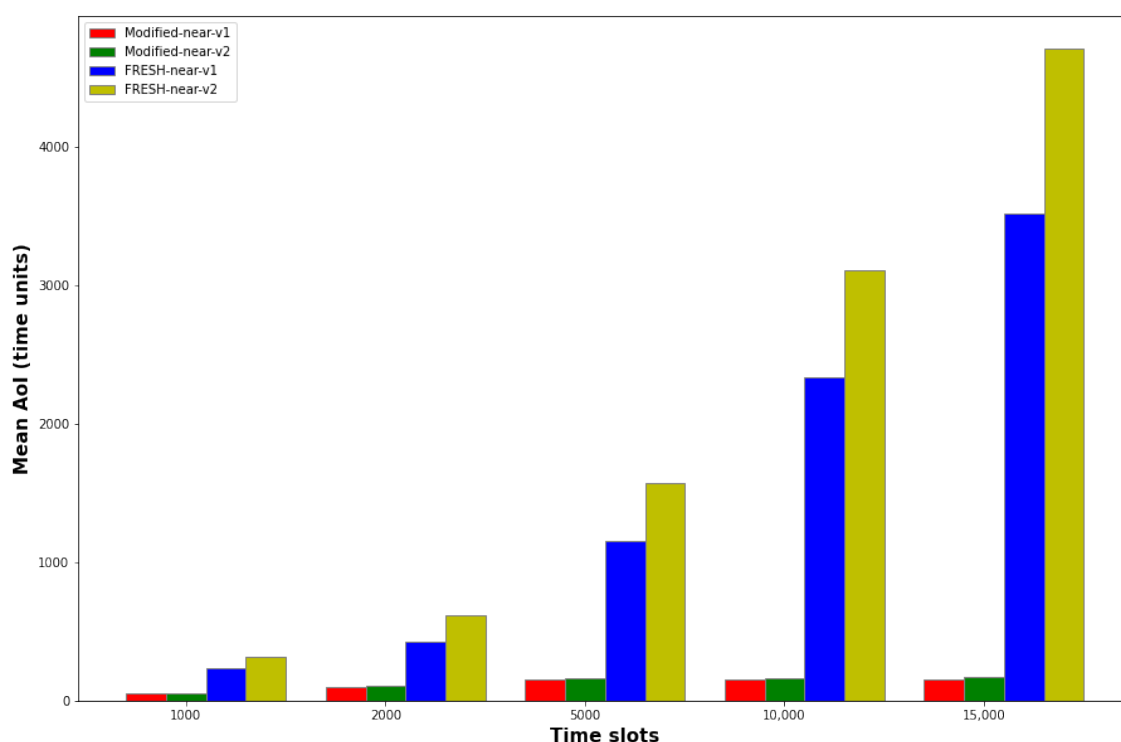


Figure 6. Dependence of AoI versus number of time slots for Modified-near-v1, Modified-near-v2, FRESH-near-v1 and FRESH-near-v2 protocols.

From this figure, the superiority of the two modified versions of the FRESH protocol over the original versions is evident. More specifically, especially for the large values of the number of time slots, the average age of the packets in the two FRESH-near protocols is almost 30-fold higher than the corresponding age of the packets in the two Modified-near protocols. Therefore, choosing the modified protocol over the original one is an effective option, since the difference in throughput is almost insignificant, while on the other hand, the benefit in packet age is considerable.

Another outcome regards the effect of throughput with respect to how dense the DTN network is, i.e., with respect to the number of its nodes. This effect is depicted in Figure 7.

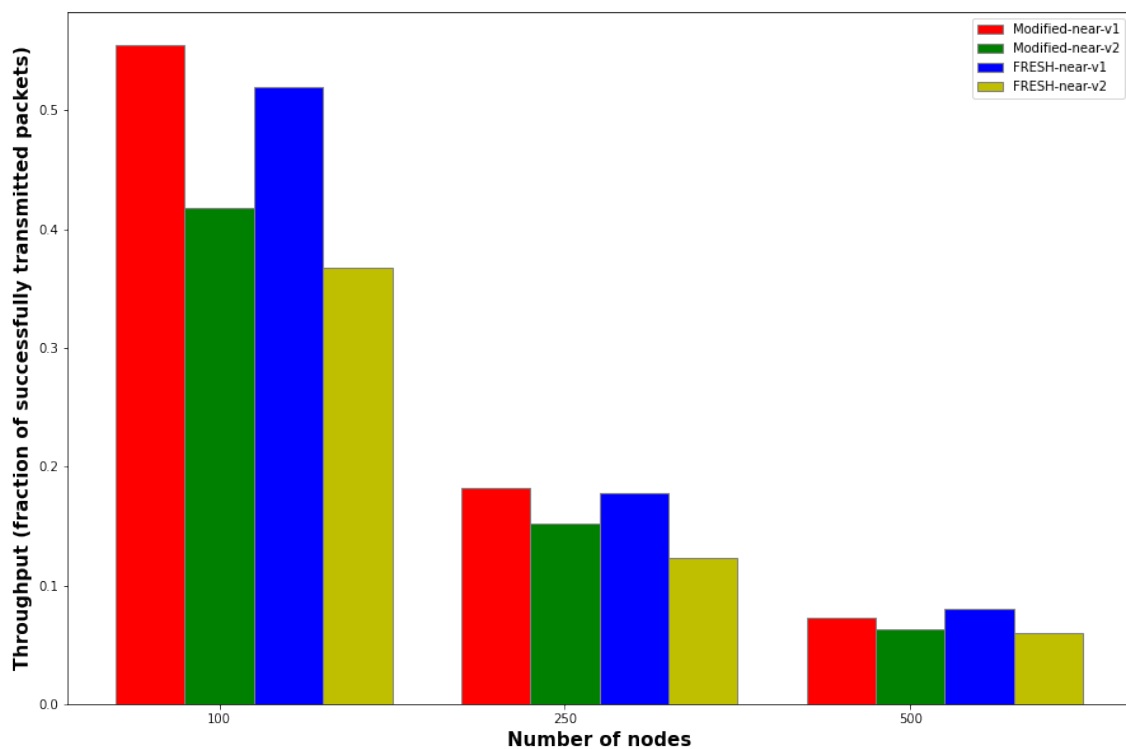


Figure 7. Dependence of throughput versus number of DTN nodes for Modified-near-v1, Modified-near-v2, FRESH-near-v1 and FRESH-near-v2 protocols.

It can be observed that for all protocols the throughput decreases almost exponentially with the increase in the network density. This result is expected, as with the increase in the number of nodes, the available “near” nodes also increase, making it more difficult to transmit a packet to its destination. This observation is directly related to a well-known and general result of P. Gupta and P. R. Kumar [38] that the throughput is $\Theta\left(\frac{W}{\sqrt{n}}\right)$ bits per second for each node for a destination that is not too far away, assuming that n nodes are optimally placed in a disk of unit area, traffic patterns are optimally assigned, the range of each transmission is optimally chosen, and each node is capable of transmitting at W bits per second. In addition, it is observed that the best performance is again achieved by the two protocols that choose their “nearest anchor node” as a relay node, while again there is a small superiority of the modified protocols over the original ones even in the throughput part.

In Figure 8, the behavior of the average age of the packets compared to the network density is similarly shown.

Again, the overwhelming superiority of the two modified protocols over the two original ones is evident in the part of the average age of the packets, while overall regarding the part of the network density, it is concluded that the use of the modified protocol FRESH on the one hand does not significantly affect the part of throughput, on the other hand it offers a significant gain in the AoI of the packets, which here are much fresher than the corresponding packets in the original FRESH protocol.

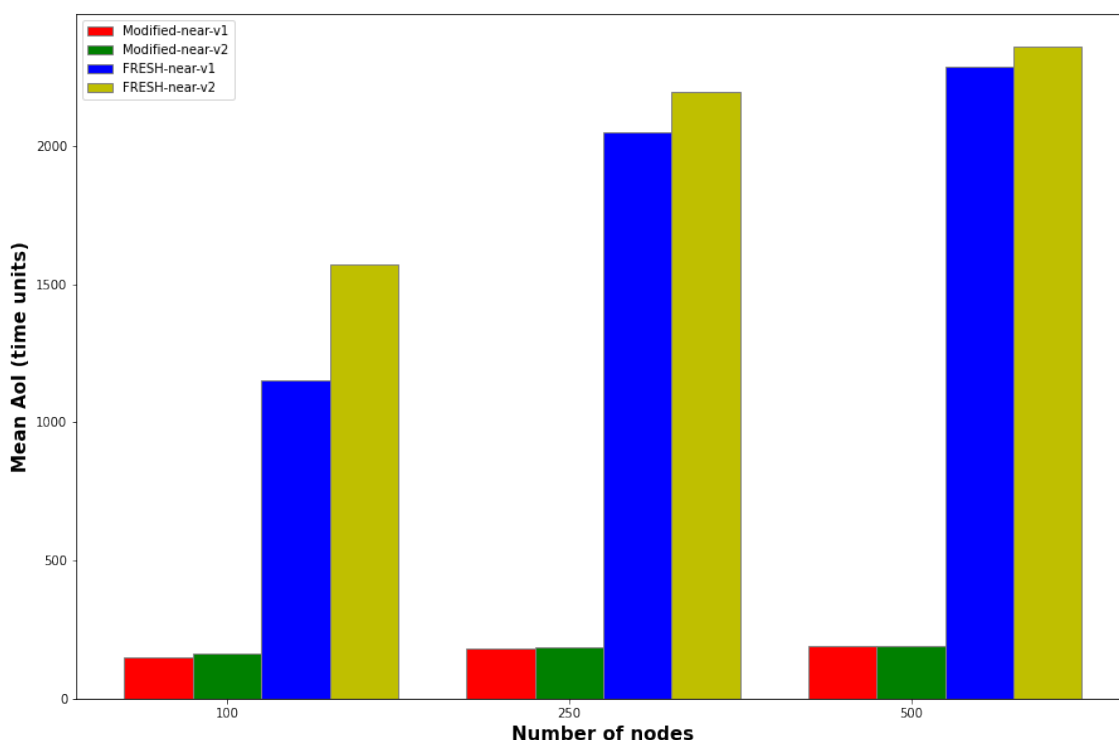


Figure 8. Dependence of AoI versus number of DTN nodes for Modified-near-v1, Modified-near-v2, FRESH-near-v1 and FRESH-near-v2 protocols.

At the same time, an interesting observation to be made is whether the two metrics considered (throughput and average packet age) are affected by the AoI Bound in the modified protocol. Intuitively, if the threshold value for the AoI is reduced enough, on the one hand the packets have a younger age, and on the other hand this leads to more drops, therefore possibly to a reduction in throughput. The figures below (Figures 9 and 10) show the effect of AoI Bound on throughput and average packet age for the three modified versions of the FRESH protocol. It is noted that in all cases, the number of time slots is equal to 5000.

As expected, lowering the AoI Bound leads to a decrease in throughput and average packet age in all cases. It is observed that the reduction of the average age is steeper for the Modified-near-v1 and Modified-near-v2 protocols than for the Modified-near-v1 protocol. At the same time, it seems that in terms of throughput, for all protocols, reducing the AoI Bound up to the value of 100 does not cause serious reductions in throughput, while there is a noticeable reduction when the Bound is set equal to 50. Possibly, the value of 100 is a balance point (for a number of time slots equal to 5000). Therefore, it can be concluded that the reduction of the upper bound of the AoI up to the value of 100 (for the number of time slots equal to 5000) does not significantly change the throughput; therefore, it is considered a profitable option, as it achieves a very sharp reduction in the average age of the packets. The throughput is almost unchanged of the AoI Bound up to the value 100, which is justified by the fact that for large barrier values (800, 600) we have fewer packet drops but much more remaining packets in the system, while for smaller barrier values (200, 100) we have more drops but much less remaining packets, with the result that the final throughput of the system remains almost unchanged, as the number of successfully transmitted packets remains almost constant.

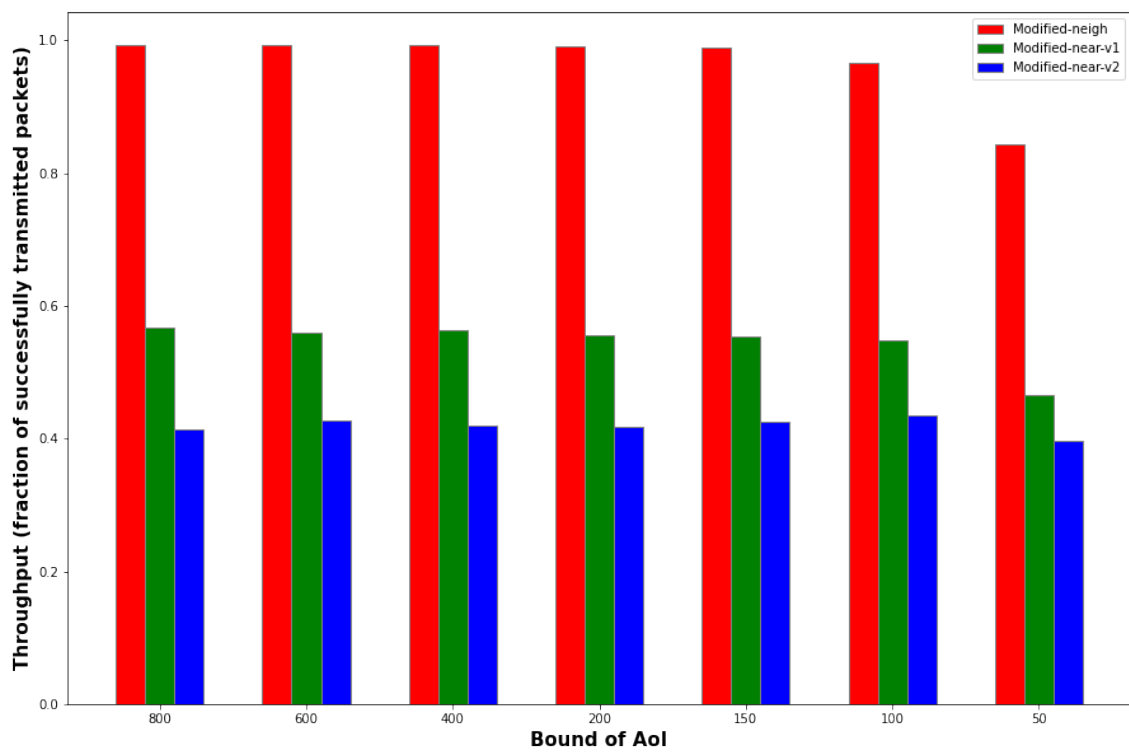


Figure 9. Dependence of throughput versus Bound of AoI for Modified-neigh, Modified-near-v1 and Modified-near-v2 protocols.

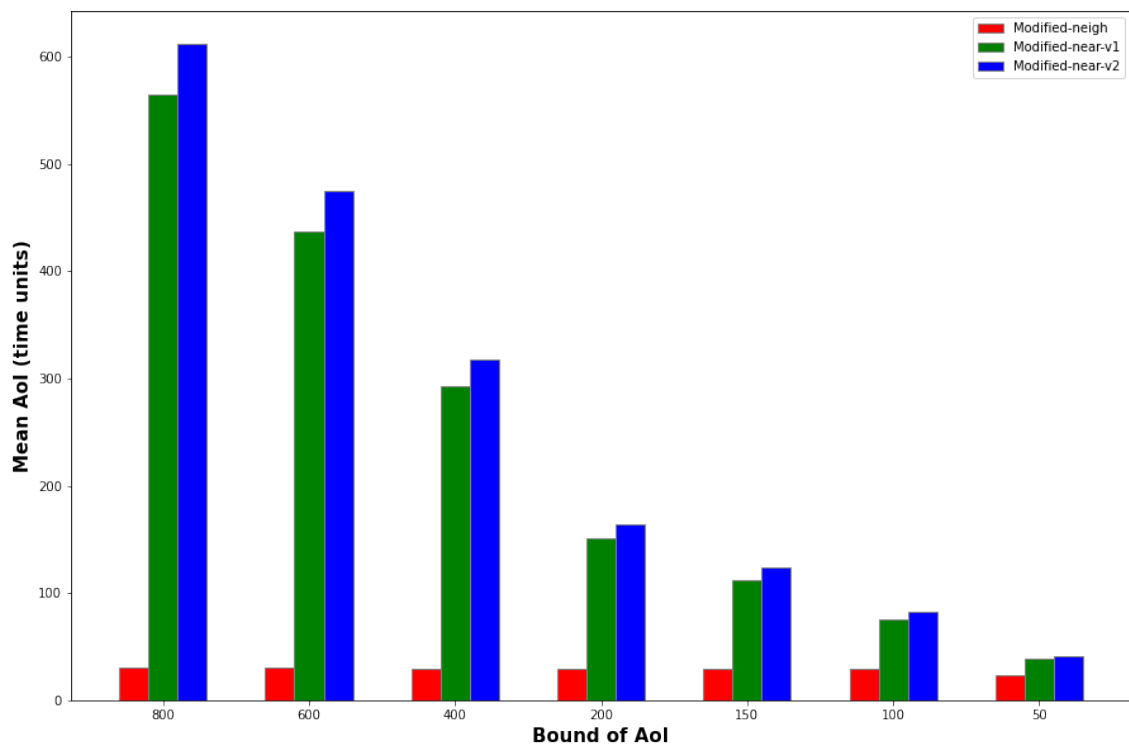


Figure 10. Dependence of AoI versus Bound of AoI for Modified-neigh, Modified-near-v1 and Modified-near-v2 protocols.

Finally, an undoubtedly important metric in any kind of simulation is the time performance (execution time). More specifically, Figure 11 shows the time required to complete

the simulations for the protocols Modified-near-v1, Modified-near-v2, FRESH-near-v1 and FRESH-near-v2 for various values of the system parameters.

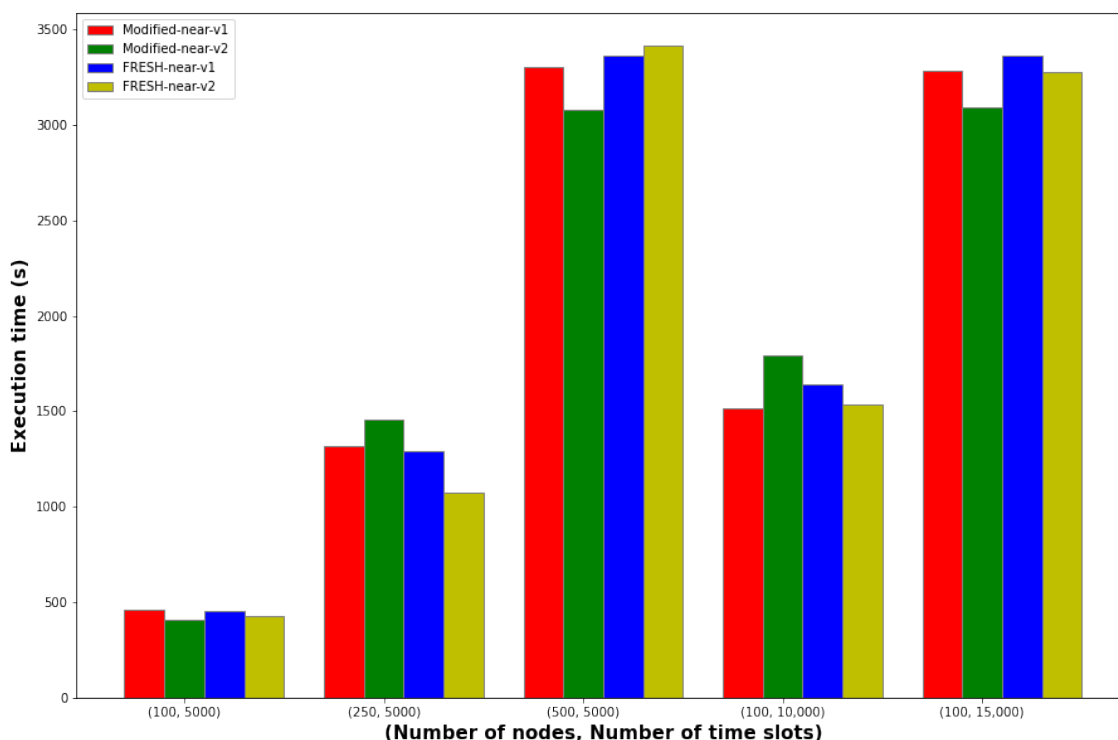


Figure 11. Running time of simulations for various values of system parameters for Modified-near-v1, Modified-near-v2, FRESH-near-v1 and FRESH-near-v2 protocols.

Understandably, in the majority of cases, the near-v2 version requires less execution time compared to the near-v1 version in the respective protocol, since there each node only searches for one best “neighbor” and not the best “neighbor”. In addition, it is observed that the increase in the number of time slots brings about a much smoother increase in the execution time than the corresponding increase brought about by the doubling of the number of nodes in the network, which could be characterized as almost exponential.

Furthermore, Figure 11 shows that the simulation time in the instance (100, 10,000) is reduced compared to the previous simulation (500, 5000), despite ostensibly being a more complex problem. Regarding this, we should note that we have taken a discrete-event simulation approach for the evaluation of the systems under investigation, which means that in each time slot considered, we account for all events taking place among nodes in the system. The two main parameters of the technique that significantly affect the execution time of the simulation are the size of the DTN network (i.e., practically the number of network nodes) and the number of time slots that will be selected for each simulation. Each simulation produces events between nodes at random, for the duration of the simulation, i.e., the number of slots simulated. Between the above two scenarios, one parameter of the problem increases in value (the number of time slots), while on the contrary the other parameter of the problem decreases in value (the density of the DTN network). Indeed, it is clear that the latter is significantly reduced (sub-fivefold), while the number of time slots is doubled. Furthermore, the number of events taking place in each time slot depends on the number of nodes of the scenario, since the events essentially represent interactions between nodes (and the more the density of the network, the more interactions will emerge on average). Thus, the complexity in the simulation can be essentially attributed more to the developing events in each time slot that require proper handling in each time slot loop, and lesser to the time slot loops. Experimentation confirmed that the increase in the network density leads to a more rapid increase in the execution time than the corresponding increase

in the number of time slots (corresponding observations also apply in case of a decrease in these two parameters).

5. Discussion

In this section, we provide an overall discussion of the obtained outcomes.

- Regarding the larger-scale experiments (large number of nodes; large number of time slots), it is observed that the two modified Modified-near protocols decisively limit both the average age of the packets and the maximum age observed in the system (which naturally equals with the threshold value set in each simulation, after which the packets are dropped). On the other hand, the two FRESH-near protocols do not deal with the age of the packets, with the result that in almost all experiments, packets that are practically constantly circulating in the network (for example, for a number of time slots equal to 5000, the maximum age observed in the system is equal to 5000, that is, one of the first packets of the system has not yet been delivered at the end of the simulation. Correspondingly, for a number of time slots equal to 15,000, Max AoI 14,995 or even 15,000 are observed). It is therefore noticeable that in these cases, the packets remaining in the network offer absolutely nothing, as they are constantly circulating from node to node or are stored in the buffers of the DTN nodes, with the possibility of their successful delivery being significantly reduced. The situation is worse, since not only do these packets offer absolutely nothing, but they also waste the available network nodes. In other words, in a more realistic simulation where the nodes would have buffers of finite capacity, these packets would occupy space in the buffers unnecessarily, a fact which would probably have a significant impact on the system throughput itself.
- A reasonable observation that can be made is that in the three modified protocols, the number of remaining packets at the end of the simulations is relatively small, as the packets are limited to the part of their age, resulting in being dropped. On the other hand, in the three original FRESH protocols, the number of remaining packets is very large (especially in FRESH-near), which is justified by the fact that, as mentioned before, even packets produced in the first moments of the simulation may be circulating on the network.
- In the simulations, it appears that the increase in the size of the square Grid from 100×100 to 1000×1000 did not bring about any significant change in the metrics considered (throughput, average age of the packets). However, this is not true, as in the simulations the value of the neighborhood radius has been increased accordingly (for both 100×100 and 1000×1000 an appropriate neighborhood radius has been chosen, so that the expected number of neighbors of each node is approximately equal to 7% of the number of network nodes).
- Regarding the execution time, obviously the implementations of the algorithms can be improved further, in order to reduce their time complexity and possibly run simulations in even larger networks and with an even greater number of time slots.

6. Conclusions

This section presents a summary of the results of this work and the conclusions from the application of the proposed routing protocol in DTNs. Finally, some possible future extensions of the research work are suggested.

6.1. Summary and Conclusions

In the context of this work, protocols were studied and developed that can be applied to Delay Tolerant Networks, including the concept of the Age of Information. First, the FRESH protocol was described in detail, which is one of the most widespread protocols in DTNs, especially due to its simplicity in terms of implementation. More specifically, three variants of the protocol were analyzed, depending on the way each node chooses to find the next relay node for a specific packet. Then, it was explained that the original version of

this protocol does not help to maintain a relatively young age of transmitted packets in the DTN network, resulting in packet freshness fading more and more over time. Subsequently, the modified FRESH protocol proposed in this work was analyzed, with the addition of the concept of the Age of Information. More specifically, in order for the packets not to be constantly transferred to the DTN network (thus wasting its available resources), as soon as they exceed a certain threshold value in their age domain, they are automatically dropped from the network. The results of the conducted simulations demonstrate the enormous benefit of applying the modified protocol proposed in the paper in terms of the average packet age metric. At the same time, it was experimentally proven that imposing the Bound on the age of the packets did not significantly affect the part of the system throughput. In fact, it was demonstrated that the value of the AoI barrier can be set quite low, without an appreciable change in the throughput and at the same time achieving a huge increase in the freshness of the transmitted packets.

6.2. Future Directions

This work has interesting extensions that could be studied in the future. Some suggested ones are listed below:

- More realistic experimental simulation of FRESH protocol versions. More specifically, in this work, one key assumption made was that the storage space of the DTN nodes (buffer) has an infinite capacity, which can be relaxed for a more realistic implementation. Understandably, nodes in real DTN systems have limited memory and storage resources. Therefore, a future simulation where nodes have finite buffer space may further highlight the benefits of using the proposed modified FRESH protocol not only in terms of packet age and freshness but also in system throughput.
- Conducting simulations of the proposed protocol in real Delay Tolerant Networks. In the present work, for the experimental evaluation of the modified FRESH protocol, synthetic networks were constructed, in which the widely used Random Waypoint Model mobility model was applied for the mobility of the nodes. It would be very interesting to conduct simulations on actual real-life Delay Tolerant Networks in order to study the effectiveness of the proposed protocol, as well as to further demonstrate the benefits of using it.
- Study of Age of Information in DTNs in order to improve the energy consumption of such networks as well, and probably jointly provide delay guarantees. Since the main topic of this paper is not to produce an energy optimal operation for DTNs, but rather demonstrate the efficacy of employing AoI in order to provide delay guarantees in delay-tolerant routing protocols, this is a topic of its own merit and we will explore it in our future work.
- Introduction of the Age of Information concept to other routing protocols in DTN. In the relevant literature, a multitude of routing protocols have been proposed that aim to make this type of subnets run as smoothly as possible. Possibly, the concept of AoI can match and co-exist smoothly with many of these protocols, especially those that belong to the Encounter-based Protocols and Time-based Protocols categories. In this way, it is possible to improve the quality metrics of DTNs with what this implies for the future evolution of the quality of life itself in the world.

Author Contributions: Conceptualization, all authors; methodology, G.K. and V.K.; software and validation, G.K.; investigation, all authors; writing, all authors; visualization, G.K.; supervision, V.K. and S.P.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not Applicable, the study does not report any data.

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

Abbreviations

The following abbreviations are used in this manuscript:

AoI	Age of Information
CDBR	Contact Duration-Based Routing
DAER	Distance Aware Epidemic Routing
DT	Delay-Tolerant
DTN	Delay-Tolerant Network
FRESH	FResher Encounter Search
MANET	Mobile Ad hoc Networks
MOVE	MOvement of VEhicle
MRH	Minimizing Relay node and Hop count
NECTAR	Neighbourhood contact history routing
PROPHET	Probabilistic ROuting Protocol using History of Encounters and Transitivity
SEDUM	Social Network Oriented and Duration Utility-based Distributed Multi-copy routing
TCP/IP	Transmission Control Protocol/Internet Protocol
TTL	Time-To-Live
VDTNs	Vehicular Delay-Tolerant Networks

References

- Verma, A.; Savita; Kumar, S. Routing Protocols in Delay Tolerant Networks: Comparative and Empirical Analysis. *Wirel. Pers. Commun.* **2021**, *118*, 551–574.
- Ali, S.; Qadir, J.; Baig, A. Routing protocols in Delay Tolerant Networks—A survey. In Proceedings of the 2010 6th International Conference on Emerging Technologies (ICET), Islamabad, Pakistan, 18–19 October 2010; pp. 70–75.
- Wang, Y.; Liu, Y.; Zhang, J.; Ye, H.; Tan, Z. Cooperative Store-Carry-Forward Scheme for Intermittently Connected Vehicular Networks. *IEEE Trans. Veh. Technol.* **2017**, *66*, 777–784. [[CrossRef](#)]
- Nordemann, F.; Tönjes, R. Transparent and autonomous store-carry-forward communication in Delay Tolerant Networks (DTNs). In Proceedings of the 2012 International Conference on Computing, Networking and Communications (ICNC), Maui, HI, USA, 30 January–2 February 2012; pp. 761–765. [[CrossRef](#)]
- Chuah, M.C.; Yang, P.; Davison, B.D.; Cheng, L. Store-and-Forward Performance in a DTN. In Proceedings of the 2006 IEEE 63rd Vehicular Technology Conference, Melbourne, Australia, 7–10 May 2006; pp. 187–191. [[CrossRef](#)]
- Snowwell, C.; Finnane, A.; Janda, M.; Soyer, H.P.; Whitty, J.A. Cost-effectiveness of Store-and-Forward Teledermatology: A Systematic Review. *JAMA Dermatol.* **2016**, *152*, 702–708. [[CrossRef](#)]
- Bista, B.B. Improving Energy Consumption of Epidemic Routing in Delay Tolerant Networks. In Proceedings of the 2016 10th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Fukuoka, Japan, 6–8 July 2016; pp. 278–283. [[CrossRef](#)]
- dos Anjos, J.C.S.; Gross, J.L.G.; Matteussi, K.J.; González, G.V.; Leithardt, V.R.Q.; Geyer, C.F.R. An Algorithm to Minimize Energy Consumption and Elapsed Time for IoT Workloads in a Hybrid Architecture. *Sensors* **2021**, *21*, 2914. [[CrossRef](#)]
- Alusta, F.; Öz, G.; Ulusoy, A.H. Investigation of Delay Tolerant Network Routing Protocols with Energy Consumption Analysis. *Adhoc Sens. Wirel. Netw.* **2020**, *46*, 53–82. [[CrossRef](#)]
- Bhattacharjee, S.; Bandyopadhyay, S. An Energy Efficient-Delay Aware Routing Algorithm in Multihop Wireless Sensor Networks. *Adhoc Sens. Wirel. Netw.* **2019**, *43*, 1–32. [[CrossRef](#)]
- Li, Y.; Cheng, X.; Liu, Q.; You, X. A Novel Congestion Control Strategy in Delay Tolerant Networks. In Proceedings of the 2010 Second International Conference on Future Networks, Sanya, China, 22–24 January 2010; pp. 233–237. [[CrossRef](#)]
- Wang, K.; Guo, H. An Improved Routing Algorithm Based on Social Link Awareness in Delay Tolerant Networks. *Wirel. Pers. Commun.* **2014**, *75*, 397–414.
- de Jesus, V.N.D.G.; Rodrigues, J.J.P.C.; Ferreira, P.S.; Nogueira, A.M.D. Improvement of Messages Delivery Time on Vehicular Delay-Tolerant Networks. In Proceedings of the 2009 International Conference on Parallel Processing Workshops, Vienna, Austria, 22–25 September 2009; pp. 344–349. [[CrossRef](#)]
- Moltafet, M.; Leinonen, M.; Codreanu, M. On the Age of Information in Multi-Source Queueing Models. *IEEE Trans. Commun.* **2020**, *68*, 5003–5017. [[CrossRef](#)]
- Costa, M.; Codreanu, M.; Ephremides, A. On the Age of Information in Status Update Systems with Packet Management. *IEEE Trans. Inf. Theory* **2016**, *62*, 1897–1910. [[CrossRef](#)]
- Dubois-Ferriere, H.; Grossglauser, M.; Vetterli, M. (2003). Age matters: Efcient route discovery in mobile ad hoc networks using encounter ages. In Proceedings of the 4th International Symposium on Mobile ad Hoc Networking and Computing, Annapolis, MD, USA, 1–3 June 2003; pp. 257–266. [[CrossRef](#)]
- Kosta, A.; Pappas, N.; Angelakis, V. Age of Information: A New Concept, Metric, and Tool. *Found. Trends Netw.* **2017**, *12*, 162–259. [[CrossRef](#)]

18. Zhang, J.; Wang, G.; Liu, C.; Zhao, F.; Zhang, X. Delay Tolerant Network and the Algorithms of DTN Routing. *J. Phys. Conf. Ser.* **2019**, *1169*, 012058. [[CrossRef](#)]
19. Lu, X.; Hui, P. An Energy-Efficient n-Epidemic Routing Protocol for Delay Tolerant Networks. In Proceedings of the 2010 IEEE Fifth International Conference on Networking, Architecture, and Storage, Macau, China, 15–17 July 2010; pp. 341–347. [[CrossRef](#)]
20. Vahdat, A.; Becker, D. *Epidemic Routing for Partially Connected Ad Hoc Networks*; Technical Report CS-2000-06; Duke University: Durham, NC, USA, 2000. [[CrossRef](#)]
21. Feng, Z.; Chin, K. A Unified Study of Epidemic Routing Protocols and their Enhancements. In Proceedings of the 2012 IEEE 26th International Parallel and Distributed Processing Symposium Workshops & PhD Forum, Shanghai, China, 21–25 May 2012; pp. 1484–1493. [[CrossRef](#)]
22. Lindgren, A.; Doria, A.; Schelen, O. Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* **2003**, *7*, 19–20.
23. Bhattacharjee, S.; Basu, S.; Roy, S.; Das Bit, S. Best-effort delivery of emergency messages in post-disaster scenario with content-based filtering and Priority-enhanced PROPHET over DTN. In Proceedings of the 2016 8th International Conference on Communication Systems and Networks (COMSNETS), Bangalore, India, 5–10 January 2016; pp. 1–7.
24. Ayaz, M.; Abdullah, A.; Low, T. Dynamic Cluster Based Routing for Underwater Wireless Sensor Networks. In Proceedings of the 2010 International Symposium on Information Technology, Kuala Lumpur, Malaysia, 15–17 June 2010.
25. Pirzadi, S.; Pourmina, M.A.; Safavi-Hemami, S.M. A novel routing method in hybrid DTN–MANET networks in the critical situations. *Computing* **2022**. [[CrossRef](#)]
26. Li, Z.; Shen, H. SEDUM: Exploiting Social Networks in Utility–Based Distributed Routing for DTNs. *IEEE Trans. Comput.* **2013**, *62*, 83–97. [[CrossRef](#)]
27. Spyropoulos, T.; Psounis, K.; Raghavendra, C. Efficient routing in intermittently connected mobile networks: The single-copy case. *IEEE/ACM Trans. Netw.* **2008**, *16*, 63–76. [[CrossRef](#)]
28. LeBrun, J.; Chuah, C.N.; Ghosal, D.; Zhang, M. Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In Proceedings of the 2005 IEEE 61st Vehicular Technology Conference, Stockholm, Sweden, 30 May–1 June 2005.
29. Huang, H.Y.; Luo, P.E.; Li, M.; Li, D.; Li, X.; Shu, W.; Wu, M.Y. Performance evaluation of suvnet with real-time traffic data. *IEEE Trans. Veh. Technol.* **2007**, *56*, 3381–3396. [[CrossRef](#)]
30. Shen, J.; Moh, S.; Chung, I. Routing Protocols in Delay Tolerant Networks: A Comparative Survey. In Proceedings of the 23rd International Conference on Circuits/Systems, Computers and Communications, (ITC-CSCC 2008), Shimonoseki, Japan, 6–9 July 2008. [[CrossRef](#)]
31. De Oliveira, E.C.; De Albuquerque, C.V. Nectar: A dtn routing protocol based on neighborhood contact history. In Proceedings of the 2009 ACM Symposium on Applied Computing, Honolulu, HI, USA, 9–12 March 2009.
32. Wang, Y.; Jain, S.; Martonosi, M.; Fall, K. Erasure-coding based routing for opportunistic networks. In Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, Philadelphia, PA, USA, 26 August 2005.
33. Farahmand, F.; Cerutti, I.; Patel, A.; Zhang, Q.; Jue, J. Relay node placement in vehicular delaytolerant networks. In Proceedings of the IEEE GLOBECOM 2008—2008 IEEE Global Telecommunications Conference, New Orleans, LA, USA, 30 November–4 December 2008.
34. Alameri, I.; Komarkova, J.; Ramadhan, M.K. Conceptual analysis of single and multiple path routing in MANET network. In Proceedings of the 2021 International Conference on Information and Digital Technologies (IDT), Zilina, Slovakia, 22–24 June 2021; pp. 235–244.
35. Wong, S.H.Y.; Yang, H.; Lu, S.; Bharghavan, V. Robust rate adaptation for 802.11 wireless networks. In Proceedings of the 12th Annual International Conference on Mobile Computing and Networking, Los Angeles, CA, USA, 23–29 September 2006; pp. 146–157.
36. giorgoskallitsis99/DTN-AoI. Available online: <https://github.com/giorgoskallitsis99/DTN-AoI> (accessed on 20 July 2022).
37. Hyytiä, E.; Virtamo, J. Random waypoint mobility model in cellular networks. *Wirel. Netw.* **2007**, *13*, 177–188.
38. Gupta, P.; Kumar, P.R. The capacity of wireless networks. *IEEE Trans. Inf. Theory* **2000**, *46*, 388–404.