Article

Location-Based Deferred Broadcast for Ad-Hoc Named Data Networking

Meng Kuai *,† and **Xiaoyan Hong** *

Department of Computer Science, The University of Alabama, Tuscaloosa, AL 35487, USA
* Correspondence: mkuai@crimson.ua.edu (M.K.); hxy@cs.ua.edu (X.H.); Tel.: +1-205-454-6260 (M.K.); +1-205-348-4042 (X.H.)
† Current address: GetHired, Inc. 540 University Ave, Palo Alto, CA 94301, USA.

Abstract: The emerging connected and autonomous vehicles (CAVs) challenge ad hoc wireless multi-hop communications by mobility, large-scale, new data acquisition and computing patterns. The Named Data Networking (NDN) is suitable for such vehicle ad hoc networks due to its information centric networking approach. However, flooding interest packets in ad-hoc NDN can lead to broadcast storm issue. Existing solutions will either increase the number of redundant interest packets or need a global knowledge about data producers. In this paper, a Location-Based Deferred Broadcast (LBDB) scheme is introduced to improve the efficiency and performance of interest broadcast in ad-hoc NDN. The scheme takes advantage of location information to set up timers when rebroadcasting an interest. The LBDB is implemented in V-NDN network architecture using ndnSIM simulator. Comparisons with several existing protocols are conducted in simulation. The results show that LBDB improves the overhead, the average number of hops and delay while maintaining an average satisfaction ratio when compared with several other broadcast schemes. The improvement can help offer timely data acquisition for quick responses in emergent CAV application situations.

Keywords: Named Data Networking; Ad-Hoc Networking; Broadcast Storm Problem

1. Introduction

The emerging of connected and autonomous vehicles (CAVs) challenge ad hoc wireless multi-hop communication by mobility induced dynamic topologies (e.g., highways vs urban environments), large-amount of mobile devices (e.g., vehicles and the embedded sensors), and large-scale, in-network data acquisitions and computations (e.g., collaborative platooning, smart transportation systems). Recent research has pointed out that the Named Data Networking (NDN) is suitable for such vehicle ad hoc network scenarios due to its information centric networking approach, where data names are directly used in network architecture. It solves the difficulty in managing the host-based IP addresses and matching the content-centric application needs more naturally [1,2].

NDN is a reactive communication architecture. In NDN [2], data pieces are named; communications are based on the names of the data instead of host IP addresses. When there is a request for data, a consumer sends an interest packet with the name of the data to the network. The interest packet either follows a routing path to a provider or floods to the network in an exhaustive attempt to find a convenient provider. Any provider who has the name-matching data will return a data packet. For wired Internet, providers run routing protocols to advertise their name prefixes. The features of NDN, such as connection-free, multicast communication nature, and in-path caching can offer great advantages for mobile ad-hoc networks (termed ad-hoc NDN) [1]. The ad-hoc NDN has been further studied for vehicle ad-hoc network (i.e., vehicular named data networking (V-NDN) [3]). Three main modifications to baseline NDN are suggested for V-NDN to tackle the topology dynamics and the data

consumption pattern, which are that routing protocols are not necessary due to topology dynamics, ubiquitous cache is desired due to ubiquitous data sources, and caching data regardless of network connectivity is helpful due to potential intermittent connectivity. The three modifications apply well to general ad-hoc NDN. Following this line, our work targets at general ad-hoc networking applications. The proposed scheme builds on the software available for vehicular named data networking where the three modifications are implemented but the scheme itself applies to general ad-hoc NDN scenarios.

In ad-hoc NDN, flooding interest packets, while being an effective way to spread the interest for a piece of data, suffer from the broadcast storming issue. This issue arises due to the lack of routing information about the data sources (recall that a routing protocol is not used). Thus, when receiving an interest, a node will rebroadcast the pending interest to its neighbors if it does not have the data to satisfy the interest itself. This leads to two issues: first, packet collisions will occur frequently if neighboring nodes rebroadcast the same interest at almost the same time, especially in dense networks; second, blind broadcast will propagate the interest to all directions and flood the network, leading to unnecessary transmissions, further adding to bandwidth overhead and data retrieval latency due to contention.

The broadcast storm problem in ad-hoc NDN calls for careful research. Grassi et al. proposed a collision avoidance solution by adding a collision avoidance timer with adaptation [3]. Other existing works have proposed broadcast schemes to reach the destination quicker. For example, using the location information to select forwarding nodes either farther away from the original sender, or closer to the data sources [4–6] but they will either increase the number of redundant interest packets or need global knowledge about data producers. As such, we believe the efficiency and performance of an interest broadcast scheme can be further improved by taking advantage of location information.

This paper presents a Location-Based Deferred Broadcast (LBDB) scheme for interest forwarding in ad-hoc NDN. The scheme takes advantage of location information to set up timers when rebroadcasting an interest. New algorithms are introduced in calculating the values for the timers to deal with the collision problem and set the forwarding priority for different potential forwarders. Our work, different from the early work, gives priorities to nodes that are closer to the straight lines between the previous forwarders and the data producer. By taking advantage of the location information, the nodes that are between data consumer and data producer will have higher priority to forward the interest packet. As such, the interest packets are accurately forwarded to the data producer with less overhead and shorter delay.

The impact of the collision-avoidance timer on interest broadcast in ad-hoc NDN is separately evaluated in the wireless testbed ORBIT. The proposed protocol LBDB is evaluated using an NDN simulator (i.e., ndnSIM) with wireless additions. We have fully implemented LBDB and integrated it with the default NDN forwarding mechanism in ndnSIM. The performance of LBDB is compared with several existing protocols. The simulation results show that LBDB generates the least transmission overhead, shortest path, and shorter delivery delay than other broadcast schemes while maintaining a relatively high satisfaction ratio. This improvement is particularly important for CAVs because timely data acquisitions lead to quick responses in emergent live-death situations [7,8].

The rest of the paper is organized as follows: In Section 2, we analyze the differences and similarities of broadcast in traditional MANET and ad-hoc NDN; we then give a brief survey on different approaches in addressing the broadcast storm issue in ad-hoc NDN; Section 3 introduces our proposed Location-Based Deferred Broadcast scheme in detail and the ad-hoc NDN architecture with the LBDB scheme; Sections 4 and 5 evaluate the performance of LBDB and the trade-off in timer settings; and Section 6 concludes the paper.

2. Related Work

Both types of MANET routing protocols, namely, proactive routing and reactive routing, deal with the issue of redundant routing packet broadcasts. The proactive routing protocols can use the available global routing states in tackling the issue, for example, using the selected multiple point

relays as in Optimized Link State Routing Protocol (OLSR) [9] or landmark nodes as in LANMAR [10]. In reactive routing protocols, most efforts focus on the routing discovery phase where flooding is used to search for the destination. Since Ad-hoc NDN is reactive, this work shares common aspects with our work. Among them, two categories are closely related to our work in terms of mitigating broadcast storm. One category involves selecting a subset of neighbors, while the other category uses probability in selecting nodes [11–15]. Works to mitigate broadcast storm in ad-hoc NDN can benefit from the work on solving traditional MANETs broadcast problems, yet with a focus on their unique issues. The existing work generally falls into three approaches, namely: selective broadcast, probabilistic broadcast and deferred broadcast.

2.1. Broadcast Storm in Ad-Hoc NDN

The storm issue of broadcast has been deeply studied in routing protocols in traditional MANETs [16]. The consequences of the broadcast storm include: redundant broadcast generates much transmission overhead and costs much network bandwidth; contention and collision lead to packet loss. Although ad-hoc NDN bears similarity with traditional MANETs [16] when interests are flooded, fundamental differences exist between the two. First, their purposes and broadcast actions are different. Routing discovery in MANETs only uses broadcast in routing request flooding to find the unique destination. The destination then replies via unicast to the source node to set up a routing path. Only after this phase, is the source able to forward data packets to the destination along the path via unicast. In ad-hoc NDN, broadcasting is used in flooding an interest from a consumer to find matching data. The matching data can exist at the providers as well at the caching nodes. The matching data are returned to the consumer along the reverse path via broadcast. This allows the nodes on the path or close to the path (via overhearing) to cache the data for future interests. Second, MANETs and ad-hoc NDNs deal with the broadcast issue with a different layering consideration in the protocol stack. Flooding route request in MANET is part of routing protocol, handled at the network layer. In NDN architecture, forwarding data chucks is the network layer. NDN is a universal overlay that can run over strategies such as IP, UDP, P2P and broadcast [17]. The wireless ad hoc network is a broadcast strategy at the strategy layer below the network layer. Methods to handle the broadcast problem are thus implemented below the network layer. Implementation-wise, ad-hoc NDN uses the *network-face-over-ad-hoc* in NDN architecture when sending interest and receiving data. While the native NDN forwarding daemon is in the network layer, the implementation of the proposed work will interface both the network and MAC layers.

2.2. Existing Work to Mitigate Broadcast Storm in Ad-Hoc NDN

Selective Broadcast: Protocols typically pre-select eligible forwarders so that the total number of forwarders can be reduced, lead to the reduction of the overall transmission overhead. A direction-selective forwarding strategy [18] is proposed to reduce overhead generated by interest packet forwarding process. Each node chooses the farthest neighbor in each direction to be the next forwarder. The scheme can reduce broadcast overhead significantly. However, it could decrease the hit ratio. Additional control packets are also needed to exchange neighbor information. The latter brings more overhead. Grassi et al. proposed Navigo [4], a location based packet forwarding mechanism for Vehicular Named Data Networking (V-NDN) [3]. The scheme guides interests packets towards data producers using a specialized shortest path over the road topology by mapping data names to data locations and forwarding interest packets along the best path. Special selection of forwarders at the intersections allows the interest to cover a large area with the smallest number of hops. Ahmed et al. proposed a forwarding strategy to unicast instead of broadcast by selecting forwarder based on neighboring forwarding information [19]. Kim and Ko proposed a similar strategy [20]. Kuang and Yu proposed a broadcast scheme combining two strategies: a mobility-based forward node selection algorithm selecting less mobile nodes as the forwarders and an available link capacity-based forwarding scheme selecting nodes with larger available link capacities [21].

Probabilistic Broadcast: Each node will have a certain probability to determine whether to be a forwarder. As such, not all the nodes will participate in broadcast. For example, the protocol "Neighborhood-Aware Interest Forwarding (NAIF)" achieves a lower bandwidth usage [22] by adjusting the percentage of interest package to be forwarded (forwarding rate) dynamically. It calculates the estimated missed rate based on the overheard forwarding packets. If the missed rate is low or zero, the relay node will continue to decrease forwarding rate. Otherwise, it will increase forwarding rate. In this way, each node reduces forwarding overhead by sharing workload with its neighbors. In another work of E-CHANET [23], each packet is forwarded by considering a counter-based suppression mechanism to reduce the collision probability and limit the broadcast storm. It requires each node, be it a consumer or a forwarder, to keep track of reachable content providers and the distance to them.

Deferred Broadcast: In this category, each node will wait on a timer before rebroadcasting an interest. The one with higher forwarding priority will have a shorter waiting timer. While waiting, if a node overhears an incoming interest/data packet that has the same data name as its already scheduled transmission, it cancels its own scheduled transmission because the same interest has been forwarded by another node that has a higher forwarding priority. As a result, only a part of nodes that have higher forwarding priorities actually forward the interest. Thus, the deferred broadcast (or timer-based rebroadcast) mechanism reduces the probabilities of packet collisions and lowers the bandwidth usage. Listen First Broadcast Later (LFBL) [24] is a scheme adopted by some wireless NDN forwarding strategies to address physical mobility of nodes and logical mobility of data for highly dynamic wireless networks. In LFBL, only the nodes closer to a packet destination than the previous sender are eligible to be the forwarders. Eligible forwarders set up waiting timers before forwarding [25], which serves two purposes: collision avoidance and forwarding prioritization. In LFBL, the value of the timer is determined by the distance on the best path to the destination plus a random value. Thus, a receiver closer to the best path will have a shorter waiting timer. A rapid traffic information dissemination scheme is proposed for VANET applications to disseminate traffic information rapidly [5,26]. It inherits the ideas about broadcasting and overhearing from LFBL adding adjustments to the vehicular environments. Similar to LFBL, it has a collision-avoidance timer to avoid packet collision on MAC layer. It also uses a push timer to push traffic information rapidly. The timer is set based on the distance from the previous transmitter [27]. In particular, receivers farther from the previous transmitter will wait for a shorter time to forward the packet than closer receivers. During the waiting time, if a node receives an interest or data packet with the same name, it will check if the overhead packet is from the neighbor farther away from the origin. If so, it will cancel its own scheduled forwarding. Another work by Amadeo et al. proposed a provider-blind forwarding strategy and a provider-aware strategy through a defer timer and overhearing to deal with flooding [28]. Li et al. proposed a forwarding strategy using the angle information to decide forwarding eligibility in addition to the distance when setting up the deferred timer [29].

2.3. Comparisons and Summary

The three aforementioned approaches mitigate broadcast storm with different strategies. Selective broadcast pre-selects some neighbors as eligible forwarders, which generates extra transmission overhead when exchanging neighborhood information. In an ad-hoc network, the topology can be very dynamic thus maintaining neighborhood information is costly. Probabilistic broadcast sets up a certain rate to rebroadcast the interest at each node, which doesn't incur additional overhead but may lower packet delivery ratio. Deferred broadcast defers rebroadcast by overhearing packets from its neighbors first. It does not introduce extra transmission overhead. Unlike probabilistic broadcast that uses forwarding rate to indicate forwarding priority, deferred broadcast uses a deferred timer to indicate forwarding priority. In probabilistic broadcast, each node decides its own forwarding process based on its forwarding priority, while in deferred broadcast, nodes' forwarding priority can affect their neighbors' forwarding process. Deferred broadcast takes advantage of the shared wireless

medium to overhear packets in-the-air and decides to cancel the pending interest if it overhears an interest/data packet with the same name. It reduces more unnecessary transmissions than probabilistic broadcast. In addition, unlike probabilistic broadcast which may decrease delivery ratio because it uses a fixed forwarding rate, nodes with higher forwarding priority in deferred broadcast will have 100% probability to forward the interest. Therefore, deferred broadcast suits ad-hoc NDN better in terms of achieving higher delivery ratio and lower transmission overhead.

The key part of designing a deferred broadcast scheme is the selection of the metric to indicate the forwarding priority. LFBL [24] uses a distance metric, which can be calculated using signal strength or hop count. Rapid forwarding [5] uses the distance to previous forwarder to rank each forwarder. The nodes that are farther from the previous forwarders will have higher priorities. DASB [29] uses the distance and angle to set up deferred timers. Those works are challenged in cases that the nodes carrying the desired data is unknown when the interest has to be broadcasted to all directions to reach the potential data carrier, thus reducing interest coverage ratio and bring extra transmission overhead. In this work, we focus on the CAV applications where the location information is available. Our work takes the opportunity that data location can be used in interest forwarding. Our deferred broadcast scheme takes consideration of both intermediate nodes' locations and the data carriers' locations. This will lead to a balance in terms of the number of rebroadcasts, the delay and the interest coverage in vehicular scenarios.

3. Location-Based Deferred Broadcast Scheme

In this section, we describe the design of our proposed Location-Based Deferred Broadcast (LBDB) scheme in the order of application name design, ad-hoc NDN basic scheme, deferred timers in ad-hoc NDN, LBDB forwarding priority timer design and broadcast algorithms.

3.1. Application Name Design

The first step for ad-hoc NDN to support CAV applications is to design a naming convention for the application data. Take a vehicular named data network as an example, the traffic information needs to be categorized and represented (such as emergencies, accidents or signals). Given these data are tightly coupled with the locations relating to the information, we consider the traffic information is critically categorized in terms of geographical locations and time. Thus, the naming structure can contain the location, being a point of interest or a region of interest. An example of using a point of interest would be in this format */traffic/(latitude,longitude)/sequence-in-time*. If data are generated during movement, new data pieces with new locations and times may be generated per the preference of the application. Additional attributes of the data can be further appended to the structure. This name structure will be used in all the network packets.

GPS devices have been popularly equipped in smart phones, vehicles, and so forth. Data tagged with location information is also abundant. When these data are expressed including their location attribute in the names, the locations of the data sources will be included in the interest packets and data packets when consumers retrieve the data by names. Such location information is ready to be used in our deferred broadcast scheme.

3.2. Ad-Hoc NDN Basic Scheme

NDN is a proposed future Internet architecture that adopts data-oriented communication paradigm [2]. In NDN, all communications are based on the names of data. There are two types of packets in NDN: (1) interest packets that include the requested data name; (2) data packets that contain a data name, metadata and data. Consumers initiate communication by sending an interest packet. Producers reply the requested data upon receiving an interest packet with the same name. The data name field uses the naming structure introduced earlier. Thus, LBDB can retrieve the the location information directly from the received interest packets. Each node maintains three data structures in NDN: (1) Content Store (CS): it caches generated or cached data. The content store is managed based

on the local cache policy and replacement policy; (2) Forwarding Information Table (FIB): it guides the interest request to the next forwarder towards the provider. The FIB table is constructed by routing protocols, where the data providers advertise name prefixes; (3) Pending Interest Table (PIT): it keeps track of the received and forwarded interest packets that are not yet satisfied by a returned data packet. The NDN data retrieval process has two phases: forwarding interest and flowing data. In the first phase, a consumer initiates the data retrieval by sending an interest packet containing data name. When a node receives an interest packet, it will first check the CS using prefix matching. If it already has the requested data, it will reply by sending back the data. If the match is missed, it will check the PIT to see if this node has sent a request with the same name. If yes, it will discard the interest packet because the same request packet has been forwarded. Otherwise, it will forward the interest packet according to the matched entry in FIB. In the flowing data phase, when a node receives a data packet, its name is used to look up the PIT. If a matching entry is found, the node sends the data packet to the interface according to the PIT entry. Then it will delete the entry in PIT. In addition, it will cache the data packet in CS if desired. If there is no matching entry in PIT, the data packet is considered unsolicited and will be dropped.

Wireless networks, such as vehicular networks, differ from wired Internet in these aspects: the ad-hoc connection, the intermittent connectivity and the ability of physically store, carry and forward data. Thus, V-NDN [3] has some modifications to the baseline NDN operations for the vehicular environment. First, to take advantage of the nature of wireless broadcast, a vehicle caches all received data regardless of whether it has a matching PIT entry or whether it needs the data for itself. This opportunistic caching strategy can be advantageous in facilitating rapid data dissemination in highly dynamic environments. Secondly, to take advantage of vehicles' physical movement, packets can be carried by moving vehicles when there is no network connectivity. It enlarges data spreading areas and increases rendezvous opportunity between consumers looking for a specific piece of data. Finally, due to the dynamics of connectivity among moving vehicles, it is infeasible to run a routing protocol to build and maintain the FIB. Therefore, V-NDN adopts an interest flooding strategy instead of following FIB when forwarding interest packets. Our work targets ad-hoc NDN, which is based on V-NDN but can be further applied to multiple scenarios including vehicular networks and underwater sensor networks.

3.3. Deferred Timers in Ad-Hoc NDN

Timers are effective mechanisms for addressing issues relating to the broadcast wireless medium by delaying the times of transmissions. Typically, a transmission occurs only when a timer expires after a pre-set waiting time value. The timer-based deferred broadcast schemes can achieve the goals of reducing packet losses and bandwidth usage. In our scheme LBDB, two types of timers are used, namely, Forwarding-Priority (FP) timer and Collision-Avoidance (CA) timer. The FP timer determines the transmission priority. A node with higher transmission priority should send with a shorter delay, i.e., a smaller value for the FP timer. The CA timer is to prevent collision. Thus, it is set to a random number so that the neighboring nodes will delay randomly before they transmit.

For interest flooding, among all nodes that need to rebroadcast an interest, the one with higher forwarding priority will have a shorter waiting timer before transmission. If a node overhears an interest packet that has the same name as its already scheduled transmission, it cancels this transmission for that interest because the interest has been forwarded by another node with a higher forwarding priority than itself. Eventually, only a part of the nodes with higher forwarding priorities actually forward the interest. In some cases different nodes may have same forwarding priority. Then CA timer will add randomness.

Figure 1 illustrates how deferred timer works in a vehicular network scenario. In this example, the consumer at the east side requests traffic information near the location of the producer at the west side. It broadcasts an interest containing the location in the data name. Car A, B and C all receive the interest from the consumer and decide to forward the interest because they do not have the data

name. They set up a pending transmission using a deferred timer that can be calculated based on each car's forwarding priority. In this example, car A has the highest forwarding priority thus its deferred timer is the shortest. Therefore, A is the first to actually transmitting interest. Car B and C will receive the interest from A. They will know the interest has been forwarded by some other cars with higher forwarding priority. So they cancel their own pending transmission.
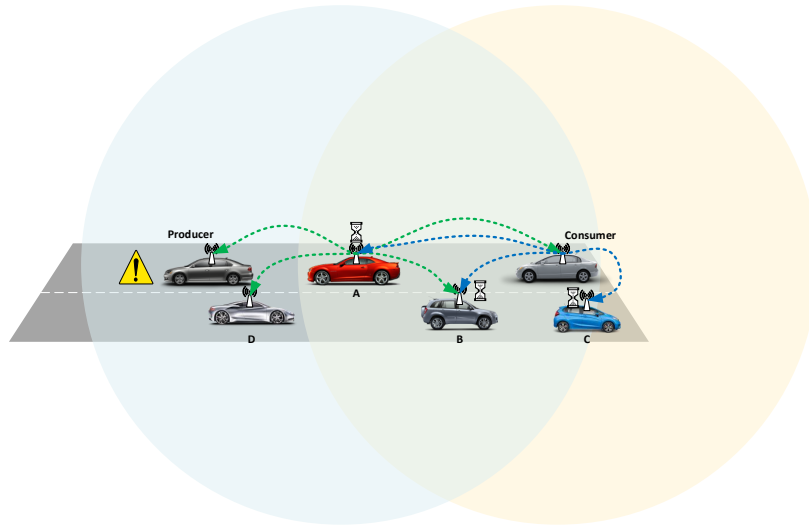

Figure 1. Deferred Timer.

3.4. LBDB Forwarding Priority Timer Design

As introduced in Section 2, the key part of designing a deferred broadcast scheme is the metric to indicate the forwarding priority. We consider two factors in calculating forwarding priority via the FP timer. On the one hand, the nodes farther away from the last hop should have higher forwarding priority so that the interest can be propagated faster with fewer hops. On the other hand, the nodes closer to data location should have higher forwarding priority thus the interest could have more chance to be forwarded to its desired area. To consider these two factors, we use a reference node, which is defined as the nearest possible geo-coordinate to the data location within the transmission range of the last hop [30]. For example, in Figure 2, node A receives an interest packet coming from node B and the name indicates data location is D. So R is the reference node. In our design, a node closer to the reference node has a shorter rebroadcast deferred timer because of the higher forwarding priority. We use the following equation to compute the timer:

$$T_{rebroadcast} = T_{max} \frac{D_R}{2D_{max}} + T_{random} \tag{1}$$

where T_{max} is the maximum deferred time, D_{max} is the transmission range and D_R indicates the distance to reference node. The value range of $\frac{D_R}{2D_{max}}$ is from 0 to 1 so that the range of deferred time is from 0 to T_{max}. A random CA timer T_{random} is appended to the equation to avoid packet collision when two calculated timers are close to each other.

Both CA timer and FP timer play significant roles in the performance of a deferred broadcast scheme. We use both experiment and simulation to evaluate our proposed LBDB. To investigate the impact of CA timer, we first design an experiment running V-NDN code on real wireless testbed when FP timers are the same. Then, in simulation, we compare LBDB with other schemes and evaluate their performance in a vehicular environment.
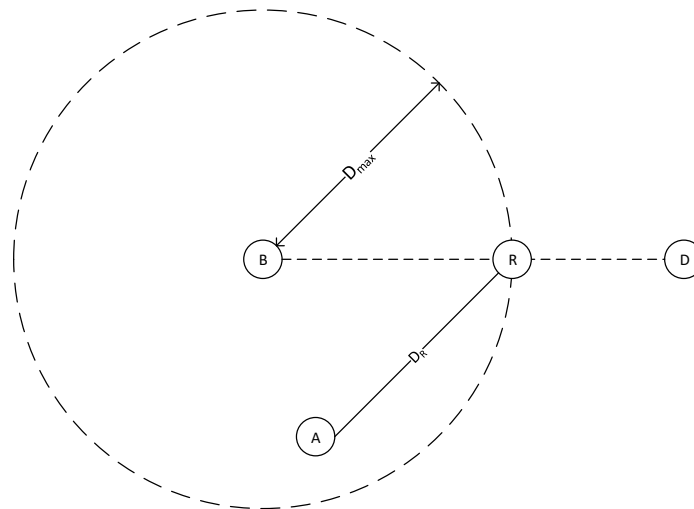
Figure 2. Rebroadcast Forwarding Priority.

3.5. LDBD Broadcast Algorithms

The broadcast issue of wireless links in ad-hoc NDN is handled using an intermediate layer between NDN network layer and MAC layer via *network face over ad-hoc*. The purpose of using this layer is to enhance the functionalities of NDN layer by adjusting ad-hoc scenario without changing default NDN forwarding behavior. The intermediate layer, called Link Adaptation Layer (LAL) is illustrated in Figure 3, where the proposed intelligent broadcast scheme LBDB is implemented. Both interest and data packets will be processed by Link Adaptation Layer.
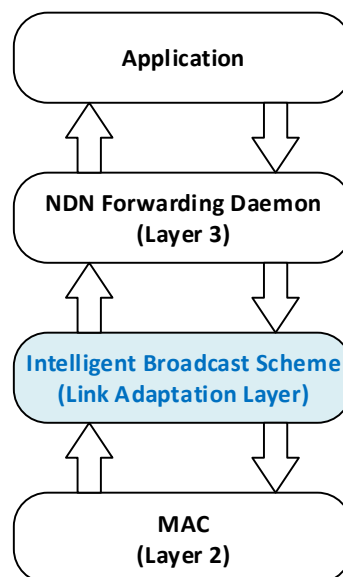
Figure 3. Ad-Hoc Named Data Networking (NDN) Architecture.

The Link Adaptation Layer interfaces with NDN network layer via the NDN data structures of CS and PIT. Different from default wired NDN, in ad-hoc NDN, FIB is not used because the interest is forwarded by broadcasting instead of using routing information. The Link Adaptation Layer maintains two additional queues, namely, *Interest Tx Queue* and *Data Tx Queue*. The two queues are used to conduct deferred broadcast by buffering the pending interest and data chunks that are waiting to be passing to the MAC layer for transmissions.

The workflows of the LBDB algorithm for handling interest and data packets are given in Figure 4. The left of Figure 4 shows the steps in forwarding an interest. After a node receives an interest, it will first check its *Interest Tx Queue* by matching the data name carried in the interest packet. If a matched entry is found in *Interest Tx Queue*, it indicates that the same interest has been received earlier and is waiting to be rebroadcast by this node. Receiving the same interest again means the interest is now being broadcasted by another node with a higher forwarding priority. As such, this node will discard the received interest. If a matching entry is not found in the *Interest Tx Queue*, *Data Tx Queue* will be checked by matching the data name. If a matching entry is found in the *Data Tx Queue*, it indicates that the requested data has been received by this node and is waiting to be broadcast (note, this is because data is sent via broadcast in ad-hoc NDN). As such, forwarding the received interest is no longer necessary. The interest is then dropped. If a matching entry is not found in the *Data Tx Queue*, the interest is passed to the upper layer. Following native NDN forwarding process, PIT and CS will be checked. The data forwarding workflow is similar. It is shown in the right of Figure 4. These steps show the difference in processing an interest or a data packet before actually sending them between the native NDN and the ad-hoc NDN, that is, adding them to interest or *Data Tx Queue* and waiting for additional time.
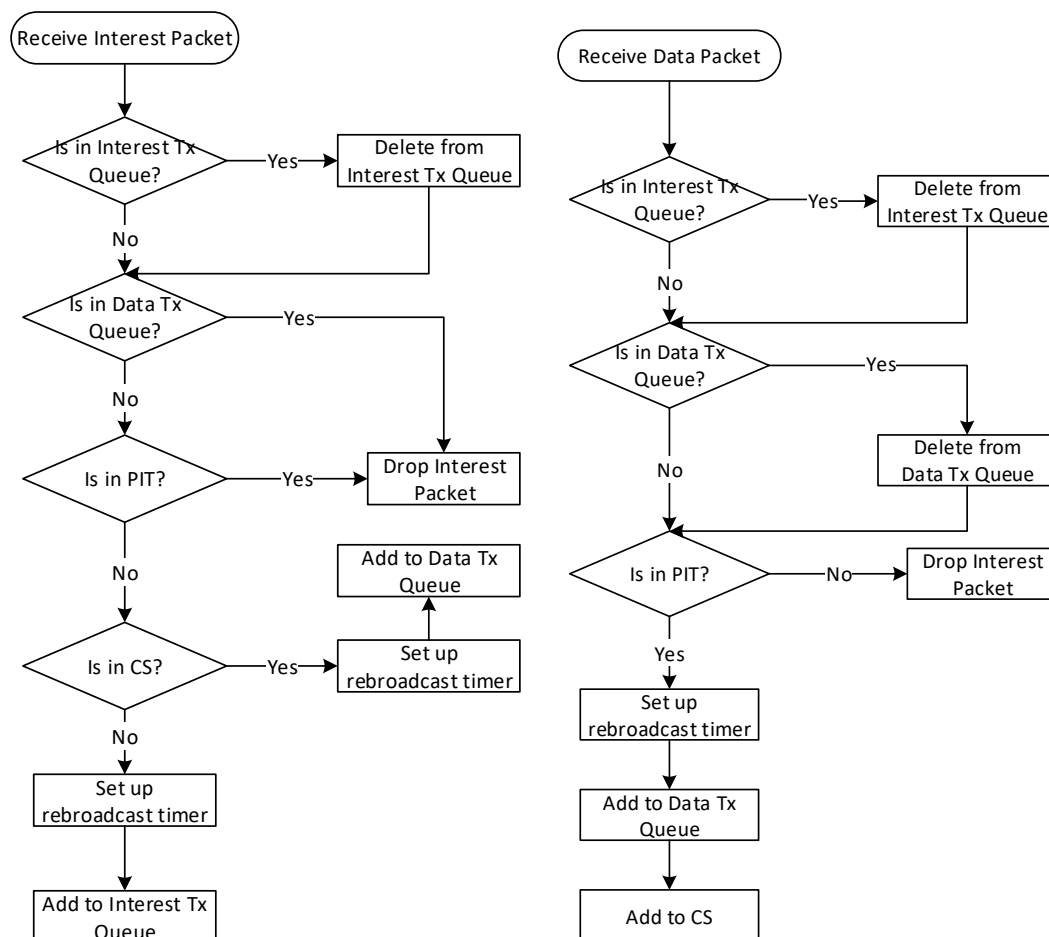

Figure 4. Interest and Data Forwarding Workflows.

4. Experimental Evaluation

The purpose of this experiment is to study the impact from the collision-avoidance timer on the performance of interest delivery in a deferred broadcast scheme. The impact is measured as the interest-receiving gain, which is the ratio of the total received number of interest packets over the total

transmitted number of interest packets. In a dense network, collisions could happen if the timer value is not appropriate, leading to low interest-receiving gain.

The experiments are performed using ORBIT (Open-Access Research Testbed for Next-Generation Wireless Networks) [31]. All the active nodes from the main grid according to the Testbed Portal were installed with *baseline-ubuntu-12-04-32bit.ndz* image and MadWiFi [32]. Each node is equipped with at least one wireless card, and the transmittable signal power can be adjusted. By changing the Wi-Fi signal power, we are able to emulate network density. Since our goal is to evaluate the broadcasting performance in dense network scenarios, and the signal propagation speed is much faster than vehicles' moving speed, it is reasonable to emulate a static environment. Figure 5 gives the grid availability and the topology of our experiment.

The code we run was V-NDN, which was the basic ad-hoc NDN code and has been tested [33,34]. It contains NDN daemon that works as a router and defines the network faces required by NDN. It is responsible for maintaining both CS and PIT tables as well as sending and receiving network packets. If a consumer application requests the road traffic information about a given region of interest (in our example, it is the top-right corner in Figure 5), it sends an interest packet and waits to receive the data. The nodes in ORBIT didn't have GPS devices. But ORBIT supports experiments with mobility by emulating locations using software and connectivity. We use this mechanism to provide continual GPS data during the experiment. The experiment mimics a broadcast of an interest packet from the center node *node10-10* for traffic information about the region. All of the neighboring nodes inside its transmission range receive the interest packet and rebroadcast it after a deferred time. Specifically, we give one case in Figure 5 where node7-6 and node6-7 will receive the interest packet at the same time and schedule to rebroadcast at the same time because they have the same distance to node10-10 and the same distance to the region of interest. Their FP timers will be the same. Therefore, the CA timer will play an important role in avoiding collision when the two nodes rebroadcast the interest.

Figure 5. Open-Access Research Testbed (ORBIT) Experiment Topology.

During the experiment, we measure how interest-receiving gain changes when the range of the random numbers that the CA timer will draw from changes. A large range will yield smaller collision probabilities. The interest-receiving gain is the ratio of received interests over the transmitted interests. Note that the ratio may be larger than 1 (100%) because one transmitted interest could be received by many neighbors. Figure 6 presents the result from our experiment. The maximum value of each range

increases as shown in x axis. It shows that small ranges up to 1000 lead to low interest-receiving gain, which suggests more packet collisions. When the range increases, the interest-receiving gain increases. The figure shows a critical increase of the interest-receiving gain at the range maximum of 1500 μs, and a relatively flat level afterwards. This means that given the density used in the experiment, 1500 μs is large enough to absorb the potential collisions.

The result of this experiment can be used as a guideline in setting up the collision-avoidance timer. Considering that a larger timer value will result in a prolonged delivery delay, for the network density in this experiment, the best maximum number of the range of the random numbers for the collision-avoidance timer is around 1500 microseconds. This preliminary result guides the configuration in the simulations where the value of CA timer should be carefully considered to achieve high success ratio and short delivery delay.
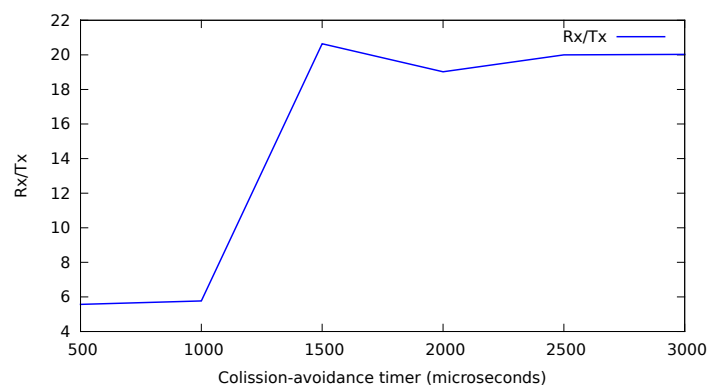

Figure 6. Interest Broadcast Evaluation Result.

5. Simulation Evaluation

5.1. Simulation Configuration

Our simulation map is an urban area spanning about 1×1 km^2 in downtown, San Francisco, as shown in Figure 7. We imported the map into SUMO [35] and generated a certain number of vehicles randomly distributed on the map. During the 500-second simulation, each vehicle picks up a random point on the map as a destination and calculates a shortest path. Once it reaches the point, it will randomly choose the next destination and keep doing so until the end of simulation.

We have chosen two points on the area as the centers of Regions of Interest, as shown Figure 7, where a Road Side Unit (RSU) is deployed. RSU can detect nearby traffic information and generate data in response to interest packets. In our simulation, the two RSUs are running NDN producer applications and replying data upon interest. A subset of vehicles are running NDN consumer applications by issuing sequential interest at a certain frequency (10 s per interest). Each interest is requesting traffic data generated by one of the RSUs and will be expired in 50 s. Other vehicles are playing the role of forwarders to forward the interest if needed. This configuration is to mimic the situation in VANETs that vehicles periodically request sequential traffic information in their interested area.

Two variables will be used to simulate different network scenarios: number of cars and number of consumers. The first one mimics the network density. More cars means denser network. The second one simulate request load. More interests will be sent if there are more consumers.

We fully implemented our proposed LBDB and compared protocols in a modified version of ndnSIM 2.0 [36,37]. The broadcast schemes were added to NetDevice interface without making any changes to the default NDN Forwarding Daemon (NFD). All vehicles and RSUs are equipped with Wi-Fi ad-hoc interfaces and installed with NDN stack.
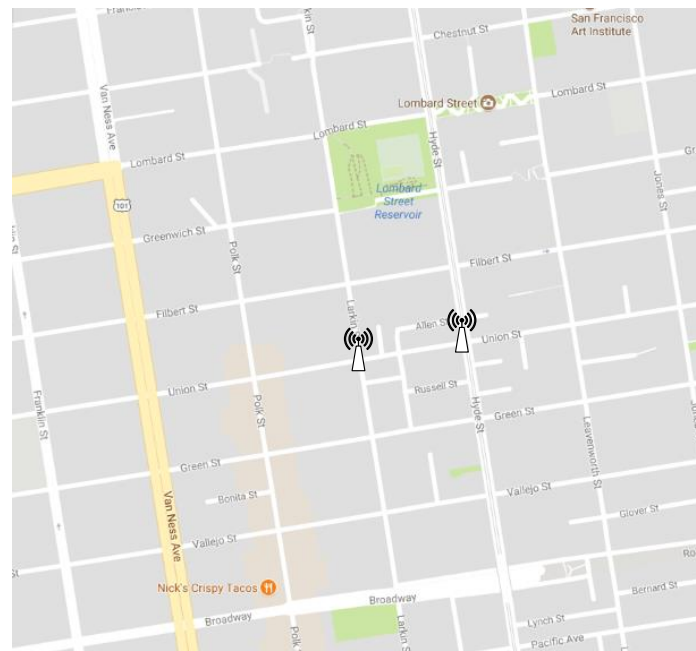
Figure 7. Simulation Scenario.

5.2. Compared Protocols

To study and evaluate our proposed LBDB, we have chosen other four broadcast schemes from the current literature that use different approaches to address broadcast storm issue. We have run the simulation 50 times with different random seeds to obtain the average value of these metrics.

1. **No FP Timer**: in this broadcast scheme, no forwarding priority timer is used. Each potential forwarder only sets up CA timer to avoid collision.
2. **Rapid Forwarding** [5]: the protocol assigns forwarding priority based on the distance to the previous hop. A node with farther distance from the previous hop will wait for shorter time, which could spread interest rapidly.
3. **DASB** [29]: this is an enhanced version of Rapid Forwarding. Both of them use the distance to the previous hop to set up deferred timer. Difference is that Rapid Forwarding cancels the pending transmission if a node overhears an interest that has the same name as its pending interests, while DASB considers the node's forwarding eligibility first based on the locations before it determines whether to cancel the pending transmission or not. If the node is in a proper location that may help to spread the interest, it will not cancel its own pending transmission.
4. **NAIF** [22]: this is a probabilistic broadcast scheme, which adjusts a forwarding rate dynamically based on forwarding statistic information it overhears locally, and the estimates a missed rate. If the missed rate is 0, this relay node will continue to decrease the forwarding rate. Otherwise, it will increase the forwarding rate. Each node decides whether to forward the interest according to its forwarding rate.

5.3. Performance Metrics

In the following representation, we use m to indicate the total number of interest issued by consumers during the whole simulation, n to indicate the number of satisfied interest that consumer obtains the requested data.

1. *Satisfaction Ratio*: it is defined as $\frac{n}{m}$. This metric indicates the ratio of issued interested and satisfied interest at the application level. Higher satisfaction ratio means more interested data are received, i.e., the interests are satisfied.

2. *Average Hops*: it is defined as $\frac{\sum_{i=1}^{n} h_i}{n}$, where h_i indicates the count of interest hops and data hops for a satisfied interest. This metric reflects how many times interest and data packets need to be forwarded to get an interest satisfied. Less hops means a shorter path.

3. *Normalized Transmission Overhead*: it is defined as $\frac{\sum_{i=1}^{m} Int_i + \sum_{i=1}^{n} Data_i}{n}$, where Int_i is the count of transmitted interest generated by each issued interest, and $Data_i$ is the count of transmitted data generated by each returned data. Then their sum is normalized by the total number of satisfied interest to indicate the overhead cost for a certain satisfaction ratio. The normalization shows the average transmission overhead for each request. This metric can measure the transmission cost and bandwidth usage of the network. Less normalized transmission overhead means an interest can get satisfied at a lower cost.

4. *Average Delay*: it is defined as $\frac{\sum_{i=1}^{n} t_i}{n}$, where t_i indicates the delay between when the interest is issued by the consumer and when the returned data arrives at the consumer. Shorter average delay means interest can be satisfied faster, which is important for those time-critical applications.

5.4. Evaluation Results

At first, we evaluate the performance of satisfaction ratio of those broadcast schemes. Figure 8 shows that the satisfaction ratio of LBDB is lower than DASB and Rapid Forwarding, but higher than No FP timer and NAIF. DASB achieves the highest satisfaction ratio, which fits in with expectation. It is because some nodes in DASB may not cancel their pending interests upon receiving an interest which has the same name as the pending interests while other schemes will cancel its own pending interest transmission. Thus in DASB, interests have more opportunities to be forwarded to the potential data carrier. Satisfaction ratio of LBDB is a little lower than that of Rapid Forwarding. It indicates that in an urban area, a shortest path to data may not exist sometimes. Rapid Forwarding and DASB may choose father nodes from data to forward the interest. Thus they will have a little higher satisfaction ratio. However, LBDB does not lose too much satisfaction ratio because of this. It has much higher satisfaction ratio than No FP Timer and NAIF. We observe that more cars contribute to a higher satisfaction ratio, which is reasonable because the interests have more chances to be forwarded to the producer. We also find that more consumers result in lower satisfaction ratio. It is because more interests are issued when there are more consumers. Even though some interests can be satisfied by cached data, the total number of lost interest is still increasing.
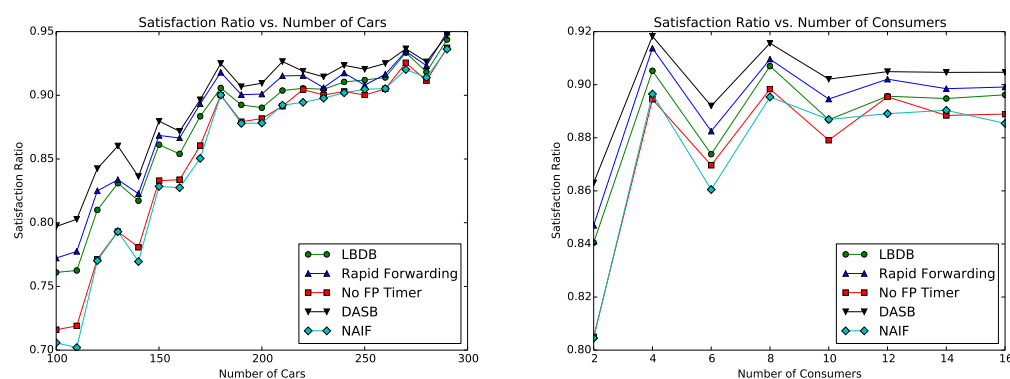

Figure 8. Satisfaction Ratio.

From the above analysis, we find that LBDB does not achieve highest satisfaction ratio among all broadcast schemes. The reason is that LBDB forwards interest toward data but sometimes the node nearest to the data is not on the path to the data. The actual path may be longer than the shortest path. Other protocols should not have this issue because the interest may be forwarded to all directions.

To verify our analysis, we measure the average of hops, which indicates how long the path is from consumer to produce then back the consumer. Figure 9 shows the results of average hops. We find that LBDB always results in the least hop count, which proves that LBDB always try to find the shortest path to the data. And interesting finding is that as the number of consumers increases, there comes more hops. It indicates that even though more interests can be satisfied by cache, increased interests need more hops to get satisfied.
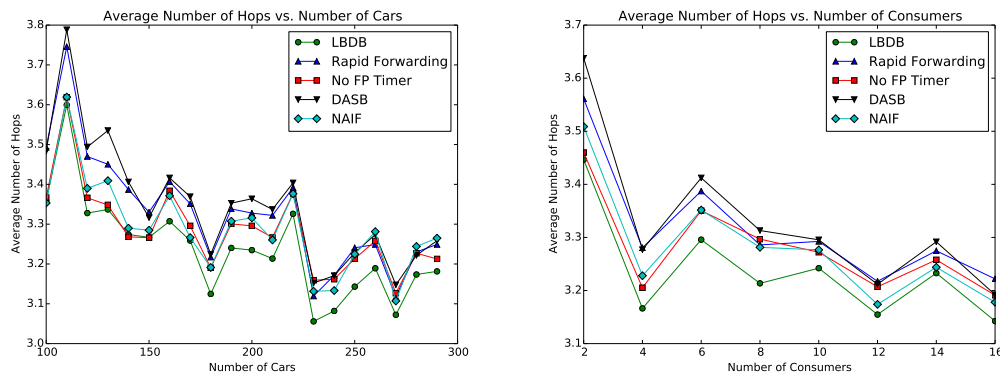
Figure 9. Average Hops.

Next, we study the performance in terms of normalized transmission overhead. Figure 10 shows that LBDB has the least normalized transmission overhead among all the evaluated protocols. It means LBDB uses the least amount of bandwidth to provide the same data retrievals as the other protocols. The figure also shows that such advantage is more significant when the network is dense or there are less consumers. The reason is that LBDB takes consideration of data's location so the interest can be forwarded toward the data so redundant forwarding is reduced. The figure also shows that more cars (i.e., denser networks) incur more transmission overhead, which is expected as more cars participate in the forwarding process. On the other hand, when the number of consumers increases, transmission overhead is reduced. This is because more interests are satisfied by the caches. The result suggests that LBDB reduces the transmission overhead.
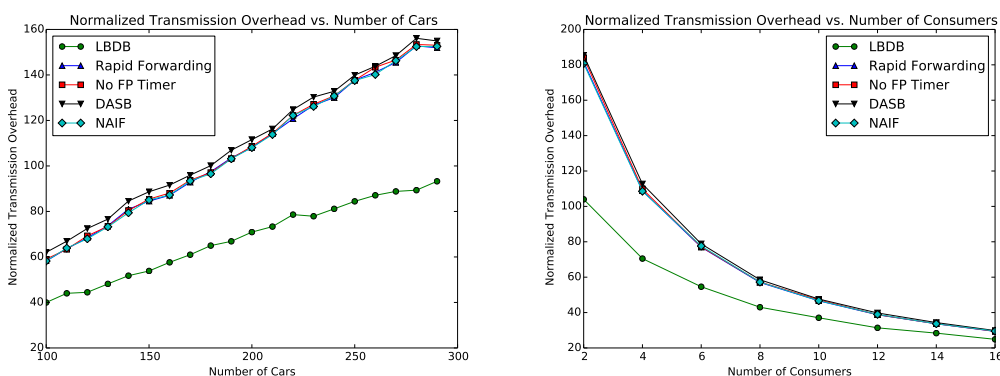

Figure 10. Normalized Transmission Overhead (interest and data).

Figure 11 presents the results of average delay. We notice that No FP Timer and NAIF show the shortest delays. This is because the two do not use FP timer, leading to reduced total waiting time. Rapid Forwarding and DASB use the same method to calculate FP deferred timer so their performance are similar. Our proposed LBDB has shorter average delay than Rapid Forwarding and DASB, which indicates that LBDB takes shorter time to satisfy interest than the existing works.
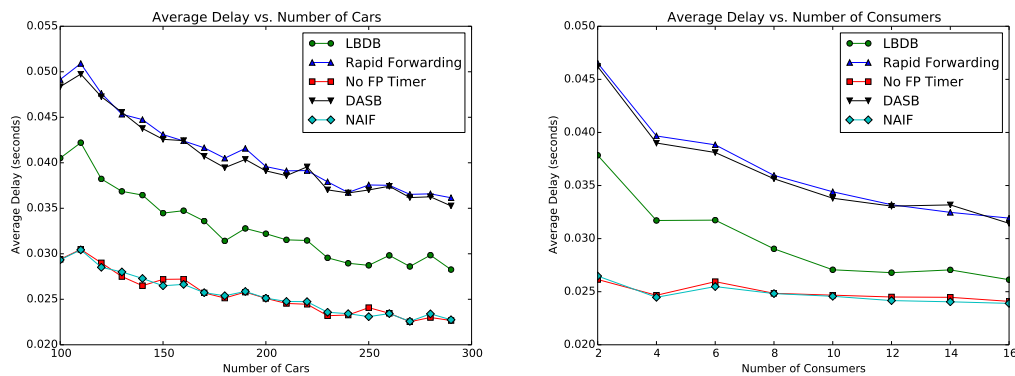
Figure 11. Average Delay.

5.5. Trade-Off Study

As aforementioned, both CA timer and FP timer use a number from a range according to our defined calculation method. The maximum value of a range determines not only the delay but also other performance such as transmission overhead. In this subsection, we study the impact of maximum CA timer and FP timer, separately.

We first perform a similar study as Section 4. The difference is that in this simulation, use dynamic topology and mobility are configured. Figure 12 shows the impact of max CA timer to the normalized transmission overhead and average delay. The results indicate that the transmission overhead increases as the increment of max CA timer and remains similar values when max CA timer is larger than a certain value. Also, the results show that there is a clear linear positive correlation between max CA timer and average delay.

Figure 13 gives the results of max FP timer's impact on the performance. We observed that longer max FP timers contribute to less normalized transmission overhead, but leads to longer average delay. This is a trade-off when assigning the value of max FP timers.
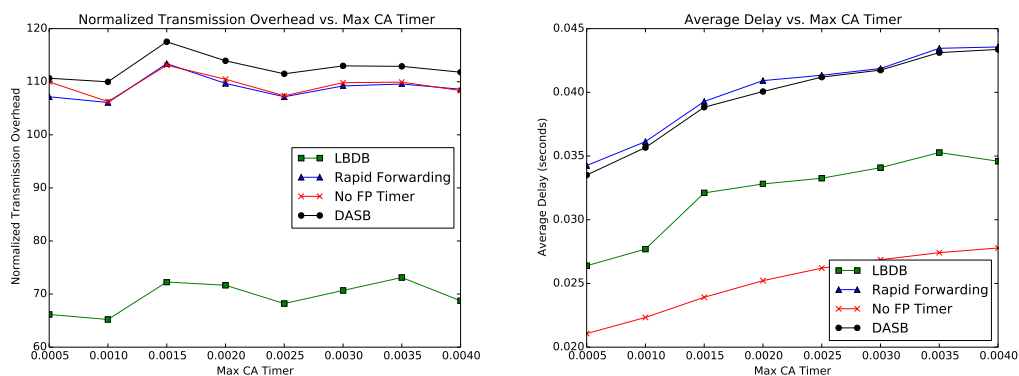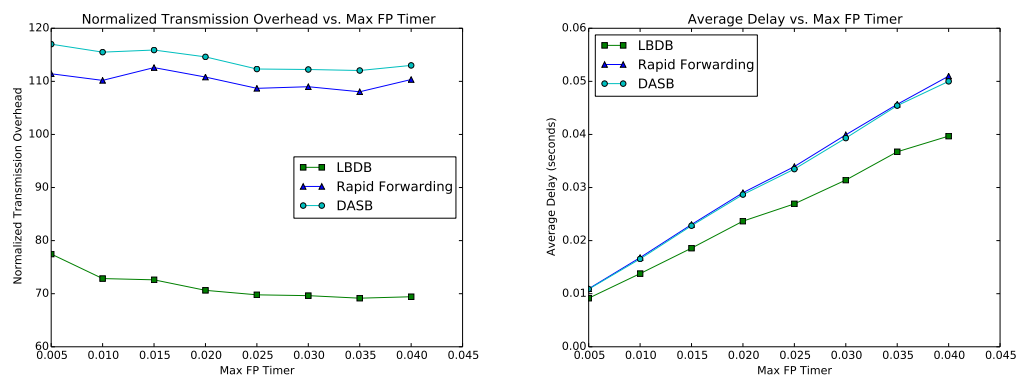

Figure 12. Impact of Max CA Timer.

Figure 13. Impact of Max Forwarding Priority (FP) Timer.

6. Conclusions

In this paper, we proposed a Location-Based Deferred Broadcast (LBDB) scheme for interest forwarding in ad-hoc NDN. Location information of the forwarding node and the data sources are used to derive waiting time for the timers that set up the forwarding priority and reduce broadcast collisions. As such, the proposed scheme selects the forwarding nodes to reduce data delivery delay by reducing transmission overhead. The effectiveness of timer mechanism is validated through ORBIT testbed experimentation. The LBDB is fully implemented in V-NDN network architecture using ndnSIM. The simulation results show that the use of two timers has proven to improve the overhead, the average number of hops and delay while maintaining an average satisfaction ratio. Such an improvement can be particular important for CAVs in live-death situations in terms of the data acquisition delay. The future work will extend this method for mobile scenarios in IoT where energy consumption is a priority.

Author Contributions: Conceptualization, M.K. and X.H.; Data curation, M.K.; Formal analysis, M.K.; Funding acquisition, X.H.; Investigation, M.K.; Methodology, M.K.; Project administration, X.H.; Resources, X.H.; Software, M.K.; Supervision, X.H.; Writing—original draft, M.K.; Writing—review & editing, X.H.

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

References

1. Meisel, M.; Pappas, V.; Zhang, L. Adhoc networking via named data. In Proceedings of the Fifth ACM International Workshop on Mobility in the Evolving Internet Architecture, Chicago, IL, USA, 20–24 September 2010; pp. 3–8.
2. Zhang, L.; Afanasyev, A.; Burke, J.; Jacobson, V.; Claffy, K.; Crowley, P.; Papadopoulos, C.; Wang, L.; Zhang, B. Named data networking. *SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 66–73. [CrossRef]
3. Grassi, G.; Pesavento, D.; Pau, G.; Vuyyuru, R.; Wakikawa, R.; Zhang, L. VANET via named data networking. In Proceedings of the IEEE INFOCOMM 2014 NOMEN Workshop, Toronto, ON, Canada, 27 April–2 May 2014.
4. Grassi, G.; Pesavento, D.; Pau, G.; Zhang, L.; Fdida, S. Navigo: Interest forwarding by geolocations in vehicular named data networking. In Proceedings of the 2015 IEEE 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Boston, MA, USA, 14–17 June 2015; pp. 1–10.
5. Wang, L.; Afanasyev, A.; Kuntz, R.; Vuyyuru, R.; Wakikawa, R.; Zhang, L. Rapid traffic information dissemination using named data. In Proceedings of the 1st ACM Workshop on Emerging Name-Oriented Mobile Networking Design—Architecture, Algorithms, and Applications, Hilton Head, SC, USA, 11–14 June 2012; pp. 7–12.

6.　Yu, X.; Coutinho, R.W.L.; Boukerche, A.; Loureirol, A.A.F. A Distance-Based Interest Forwarding Protocol for Vehicular Information-Centric Networks. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017; pp. 1–5.

7.　Shladover, S. Connected and automated vehicle systems: Introduction and overview. *J. Intell. Transp. Syst.* **2018**, *22*, 187–197. [CrossRef]

8.　Sovani, S. Top 3 Challenges to Produce Level 5 Autonomous Vehicles. Available online: https://www.ansys.com/blog/challenges-level-5-autonomous-vehicles (accessed on 21 June 2019).

9.　Qayyum, A.; Viennot, L.; Laouiti, A. Multipoint Relaying: An Efficient Technique for Flooding in Mobile Wireless Networks. Research Report RR-3898, INRIA. 2000. Available online: https://hal.inria.fr/inria-00072756 (accessed on 24 June 2019).

10.　Pei, G.; Gerla, M.; Hong, X. LANMAR: Landmark Routing for Large Scale Wireless AdHoc Networks with Group Mobility. In Proceedings of the IEEE/ACM MobiHoc 2000, Boston, MA, USA, 6–11 August 2000.

11.　Călinescu, G.; Mandoiu, I.I.; Wan, P.-J.; Zelikovsky, A.Z. Selecting forwarding neighbors in wireless adhoc networks. *Mob. Netw. Appl.* **2004**, *9*, 101–111. [CrossRef]

12.　Peng, W.; Lu, X.-C. On the reduction of broadcast redundancy in mobile adhoc networks. In Proceedings of the 1st ACM International Symposium on Mobile AdHoc Networking & Computing, Boston, MA, USA, 11 August 2000; pp. 129–130.

13.　Kim, J.S.; Zhang, Q.; Agrawal, D.P. Probabilistic broadcasting based on coverage area and neighbor confirmation in mobile adhoc networks. In Proceedings of the IEEE Global Telecommunications Conference Workshops, GlobeCom Workshops 2004, Dallas, TX, USA, 27 November–3 December 2004; pp. 96–101.

14.　Stojmenovic, I.; Seddigh, M.; Zunic, J. Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE Trans. Parallel Distrib. Syst.* **2002**, *13*, 14–25. [CrossRef]

15.　Zhang, X.M.; Wang, E.B.; Xia, J.J.; Sung, D.K. A neighbor coverage-based probabilistic rebroadcast for reducing routing overhead in mobile adhoc networks. *IEEE Trans. Mob. Comput.* **2013**, *12*, 424–433. [CrossRef]

16.　Tseng, Y.-C.; Ni, S.-Y.; Chen, Y.-S.; Sheu, J.-P. The broadcast storm problem in a mobile adhoc network. *Wirel. Netw.* **2002**, *8*, 153–167. [CrossRef]

17.　Named Data Networking: Motivation & Details. Available online: https://named-data.net/project/archoverview/ (accessed on 21 June 2019).

18.　Lu, Y.; Zhou, B.; Tung, L.-C.; Gerla, M.; Ramesh, A.; Nagaraja, L. Energy-efficient content retrieval in mobile cloud. In Proceedings of the Second ACM SIGCOMM Workshop on Mobile Cloud Computing, Hong Kong, China, 16 August 2013; pp. 21–26.

19.　Ahmed, S.H.; Bouk, S.H.; Kim, D. Rufs: Robust forwarder selection in vehicular content-centric networks. *IEEE Commun. Lett.* **2015**, *19*, 1616–1619. [CrossRef]

20.　Kim, D.; Ko, Y.-B. A novel message broadcasting strategy for reliable content retrieval in multi-hop wireless content centric networks. In Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication, Bali, Indonesia, 8–10 January 2015; p. 108.

21.　Kuang, J.; Yu, S.Z. Broadcast-based content delivery in information-centric hybrid multihop wireless networks. *IEEE Commun. Lett.* **2017**, *21*, 889–892. [CrossRef]

22.　Yu, Y.-T.; Dilmaghani, R.B.; Calo, S.; Sanadidi, M.; Gerla, M. Interest propagation in named data manets. In Proceedings of the 2013 International Conference on Computing, Networking and Communications (ICNC), San Diego, CA, USA, 28–31 January 2013; pp. 1118–1122.

23.　Amadeo, M.; Molinaro, A.; Ruggeri, G. E-chanet: Routing, forwarding and transport in information-centric multihop wireless networks. *Comput. Commun.* **2013**, *36*, 792–803. [CrossRef]

24.　Meisel, M.; Pappas, V.; Zhang, L. Listen first, broadcast later: Topology-agnostic forwarding under high dynamics. In Proceedings of the Annual Conference of International Technology Alliance in Network and Information Science, September 2010; pp. 1–8. Available online: http://web.cs.ucla.edu/~lixia/papers/10ITA-LFBL.pdf (accessed on 24 June 2019).

25.　Ye, F.; Chen, A.; Lu, S.; Zhang, L. A scalable solution to minimum cost forwarding in large sensor networks. In Proceedings of the Tenth International Conference on Computer Communications and Networks (Cat. No. 01EX495), Scottsdale, AZ, USA, 15–17 October 2001; pp. 304–309.

26. Wang, L.; Wakikawa, R.; Kuntz, R.; Vuyyuru, R.; Zhang, L. Data naming in vehicle-to-vehicle communications. In Proceedings of the 2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Orlando, FL, USA, 25–30 March 2012; pp. 328–333.

27. Kaul, S.; Gruteser, M.; Onishi, R.; Vuyyuru, R. Geomac: Geo-backoff based co-operative mac for v2v networks. In Proceedings of the 2008 IEEE International Conference on Vehicular Electronics and Safety, Columbus, OH, USA, 22–24 September 2008; pp. 334–339.

28. Amadeo, M.; Campolo, C.; Molinaro, A. Forwarding strategies in named data wireless adhoc networks: Design and evaluation. *J. Netw. Comput. Appl.* **2015**, *50*, 148–158. [CrossRef]

29. Li, Y.; Su, X.; Lindgren, A.; Shi, X.; Cai, X.; Riekki, J.; Que, X. *Distance Assisted Information Dissemination with Broadcast Suppression for ICN-Based VANET*; Springer International Publishing: Cham, Switzerlane, 2016; pp. 179–193.

30. Yu, Y.-T.; Li, Y.; Ma, X.; Shang, W.; Sanadidi, M.; Gerla, M. Scalable opportunistic VANET content routing with encounter information. In Proceedings of the 2013 21st IEEE International Conference on Network Protocols (ICNP), Göttingen, Germany, 7–10 October 2013.

31. ORBIT Testbed. Available online: http://www.orbit-lab.org (accessed on 21 June 2019).

32. MadWifi. Available online: http://madwifi-project.org (accessed on 21 June 2019).

33. V-NDN. Available online: https://github.com/named-data/vndn (accessed on 21 June 2019).

34. Grassi, G.; Pesavento, D.; Wang, L.; Pau, G.; Vuyyuru, R.; Wakikawa, R.; Zhang, L. Vehicular inter-networking via named data. *arXiv* **2013**, arXiv:1310.5980.

35. Krajzewicz, D.; Erdmann, J.; Behrisch, M.; Bieker, L. Recent development and applications of SUMO—Simulation of urban MObility. *Int. J. Adv. Syst. Meas.* **2012**, *5*, 128–138.

36. Afanasyev, A.; Moiseenko, I.; Zhang, L. *ndnSIM: NDN Simulator for NS-3*; Technical Report NDN-0005; NDN: 2012. Available online: https://named-data.net/publications/techreports/trndnsim/ (accessed on 24 june 2019).

37. Mastorakis, S.; Afanasyev, A.; Moiseenko, I.; Zhang, L. *ndnSIM 2.0: A New Version of the NDN Simulator for NS-3*; Technical Report NDN-0028; NDN: 2015. Available online: https://named-data.net/publications/techreports/ndn-0028-1-ndnsim-v2/ (accessed on 24 june 2019).