An Efficient Delay Tolerant Networks Routing Protocol for Information-Centric Networking

Min Wook Kang¹, Dong Yeong Seo¹ and Yun Won Chung²,*

¹ Department of Information and Telecommunication Engineering, Graduate School, Soongsil University, Seoul 06978, Korea; goodlookmw@gmail.com (M.W.K.); seodong2da@gmail.com (D.Y.S.)
² School of Electronic Engineering, Soongsil University, Seoul 06978, Korea
* Correspondence: ywchung@ssu.ac.kr

Received: 19 April 2020; Accepted: 17 May 2020; Published: 19 May 2020

Abstract: Delay tolerant networks (DTN) is a good candidate for delivering information-centric networking (ICN) messages in fragmented networks due to disaster. In order to efficiently deliver ICN messages in DTN, the characteristics of multiple requester nodes for the same content and multiple provider nodes for the same request should be used efficiently. In this paper, we propose an efficient DTN routing protocol for ICN. In the proposed protocol, requester information for request packet, which is called an Interest in ICN, is shared by exchanging status table with requested Data ID, requester ID, and satisfaction flag, where satisfaction flag is defined to show the delivery status of Data, so that unnecessary forwarding of Data is avoided. Data is forwarded to a more probable node by comparing average delivery predictability to a set of requesters. Performance of the proposed protocol was evaluated using simulation from the aspect of Data delivery probability and Data overhead, for varying buffer sizes, number of relay nodes, and time-to-live (TTL) of Data. The results show that the proposed protocol has better Data delivery probability, compared to content distribution and retrieval framework in disaster networks for public protection (CIDOR) and opportunistic forwarding (OF) protocols, although there is a tradeoff from the aspect of Data overhead for varying buffer sizes and number of relay nodes.

Keywords: delay tolerant network; information-centric networking; delivery probability; overhead; ONE simulator

1. Introduction

Information-Centric Networking (ICN) [1–3] was proposed to overcome the current bottleneck problem of the Internet due to overwhelming content downloads. In the Internet, routing is carried out based on the IP address of destination node. If a node sends a content request packet, it is routed through intermediate routers based on the IP address of the destination node, i.e., content server. Therefore, if multiple users request the same content downloads, multiple unicast downloads from a content server are generated, even though the routing paths of those sessions may overlap significantly, and they consume significant network bandwidth.

In ICN, on the other hand, content name is used to search for a content instead of IP address, and content is cached in intermediate nodes as well as an original content server for efficient content delivery. In ICN, a node which requests a content is called a consumer, and a node which generates a content is called a provider. For name-based routing, Pending Interest Table (PIT), Content Store (CS), and Forwarding Information Base (FIB) are defined in ICN [1–3]. PIT stores input interface, which is called a face in ICN, of the received Interest packet. CS acts as the caching storage of an ICN node. FIB manages forwarding information for requested content, i.e., Data. If a consumer wants to download content, it transmits Interest for the requested content to neighbor nodes. A node which
receives the Interest records the input face of the Interest in its PIT and checks for its CS firstly to see if there is the requested content. If there is, it returns the content to the incoming face. Otherwise, it checks FIB table to find appropriate outgoing face for the requested Interest. If Interest is arrived at either a node with the requested content in its CS or an original content provider, the content is replied back to the consumer along the reverse path of forwarding path of Interest using PIT, and content may be cached in the CS of intermediate nodes. Therefore, ICN can solve the current bottleneck problem of the Internet, by using the name of content for routing and returning of content from nearby caching node.

Recently, the importance of an efficient communication in disaster environment has been increased, especially to save the life of the survivors effectively. The current cellular network or Internet may not be efficient in a disaster environment because survivors may not be aware of appropriate point of contacts, i.e., phone number or IP address. Also, communication between dynamically formed rescue team members is not efficient in current communication technology [4]. ICN is suitable in this disaster environment, since a name is used to send messages and responses are quickly returned from nearby caching nodes. Furthermore, communication between rescue team members can be achieved easily using a flexible name structure, and dynamic grouping of team members is very efficiently achieved [4].

Network infrastructure for ICN, however, is unlikely to be available in disaster environment due to the destruction of infrastructure. Thus, infrastructure-based networks such as cellular networks or Internet are not applicable. Since network connectivity may not be guaranteed due to the sparse density of nodes in disaster environment, mobile ad-hoc networks (MANET) is not applicable. Therefore, a new communication method should be applied to enable ICN message delivery in disaster environment, and delay tolerant networks (DTN) can be a good candidate for this situation. In DTN, communication is possible without infrastructure and a message can be delivered, even though there is no guaranteed end-to-end routing path between a source node and a destination node by using a ‘store-carry-forward’ approach [5–7]. A node with a message to deliver stores it in its buffer, which is operated in bundle layer. If it contacts another node while moving, it forwards the message to the contact nodes, based on a predefined criteria. By doing this hop-by-hop forwarding repeatedly, a message can be delivered to the destination node of the message successfully.

There have been some works on delivering ICN messages using DTN [8–18]. In these works, networks are generally fragmented due to disaster and DTN protocol is used to deliver ICN messages, where ICN message is encapsulated in DTN message. In [19], research challenge on interworking between ICN and DTN is presented in emergency scenario. It is, however, not efficient to apply DTN protocol to deliver ICN messages directly, since multiple requester nodes exist for the same content and multiple provider nodes exist for the same request in DTN. Therefore, it is necessary to use the characteristics of multiple requester nodes for the same content and multiple provider nodes for the same request in order to deliver Interest and content efficiently.

In our preliminary work on delivering ICN messages in DTN [20–22], we proposed schemes to efficiently deliver Interest and Data using the characteristics of both ICN and DTN. In [20], probabilistic routing protocol using history of encounters and transitivity (PRoPHET) protocol [23] is assumed for DTN protocol and it is extended for efficient ICN message delivery. In PRoPHET protocol, a message is forwarded to another contact node, if the contact node has a higher delivery predictability. In the proposed extension, since a requested Data may exist in multiple caching nodes, delivery predictability is newly calculated between a requester and a set of nodes which have the requested Data. Interests from different requesters with the same content request are merged for reducing overhead of Interest forwarding. In [21], the performance of the proposed protocol in [20] was analyzed using simulation. In [22], we extended our works in [20,21] to reduce message overhead by sharing successful delivery information of Interest and Data.

In this paper, we propose an efficient dissemination of Interest by sharing requester information between nodes efficiently, instead of forwarding Interests during contact. Also, we further improve
our preliminary work in [22] by using the information of already delivered Data in order to actively reduce the overhead due to unnecessary dissemination of already delivered Data, based on Vaccine scheme in [24]. In [24], a vaccine scheme was proposed to reduce unnecessary dissemination of already delivered message by actively sharing the successful delivery information, but it is defined for unicast communication with single source node and single destination node in DTN. Therefore, it is not efficient to apply the concept of Vaccine scheme for ICN in DTN directly, since multiple requesters and multiple caching nodes exist together. In this paper, we extended the Vaccine scheme for multiple requesters and multiple caching nodes environment. If Data is already delivered to a requester, the satisfaction flag for a pair of delivered Data and requester is set as 1. Then, if a node with a satisfaction flag of 1 contacts another node with a requester information for the Data corresponding to the satisfaction flag of 1, the requester information is removed from Data in order to avoid unnecessary forwarding of already delivered Data. Finally, Data forwarding between contact nodes is determined by using the average of delivery predictability to a set of requesters of the considered Data, in order to use the characteristics of multiple requesters of the delivered Data.

This paper is organized as follows: Section 2 reviews related works in detail. Section 3 illustrates proposed protocol for ICN in DTN. Performance of the proposed protocol is evaluated and compared in Section 4. Finally, Section 5 concludes this work.

2. Related Works

In this section, related works on ICN in DTN are surveyed in detail. In [8], the authors proposed an integrated ICN and DTN protocol. In DTN, routing based on the reverse path stored in PIT is not available due to the disconnection between nodes and the mobility of nodes. The authors propose to use DTN protocol for routing and forwarding of ICN in disconnected environment. ICN Interests are carried in payload of DTN messages, and if any node has a content for the Interest, the content is carried in payload of DTN messages to requesters. Bundle protocol query extension block is used for this interworking between ICN and DTN. In [9], agent-based content retrieval (ACR) scheme was proposed, where requesters delegate content retrieval to mobile agents and mobile agents retrieve contents on behalf of the requesters. If requesters and agents contact again, requesters retrieve contents from agents. The ACR scheme consists of three phases: (1) agent delegation, (2) content retrieval, and (3) content notification. In agent delegation phase, an agent is searched for and content retrieval is delegated to the agent. In content retrieval phase, agent finds and retrieves content for the requester by broadcasting Interest periodically. In content notification, if an agent retrieves content, it notifies it to the requester by performing either push or pull notification. In [10], the authors proposed content centric DTN network architecture, where multi-hop cellular network (MCN) is considered. In MCN, user devices communicate with each other using either cellular base station or device-to-device (D2D) communication. The proposed protocol consists of three planes: (1) control plane, (2) forwarding plane, and (3) routing decision plane. In control plane, meta-information is inserted in the DTN messages based on the packet type (Interest/Data). In forwarding plane, either ICN forwarding or DTN forwarding is selected. Mobile node uses DTN to forward packets in D2D way. Pending requester information table (PRIT) is defined to store requester information instead of arrival faces of Interest. In routing decision engine, request/response processing and content management are carried out.

In [11], the authors proposed an information centric delay tolerant network model based on the similarities between them, i.e., (1) in-network storage, (2) late binding, (3) data longevity, and (4) flexible routing. In order to show the advantages of information centric delay tolerant networks, the authors carried out simulations and compared two communication modes, i.e., information-centric and host-centric. In [12], the authors proposed a delay tolerant ICN for disaster management in fragmented networks due to disaster. Each fragmented network has a gateway which collects messages exchanged with external fragmented networks. Communication between fragmented networks is carried out by data mules such as ambulances, police-cars, and people, which have different movement patterns. Each data mule has encounter table which has delivery probability value with destinations.
and message, and each message has a priority. Based on probability value and message priority, transmission priority is determined and a message with a higher transmission priority is transmitted earlier. In [13,14], the authors proposed a data discovery and retrieval based on social cooperation in content-centric delay tolerant network, where relay nodes are selected based on social cooperation criterion, in order to reduce content discovery and recovery latency. They define social relationship factor between two nodes based on past contact history, where total number of encounters between two nodes and encounter function which measures the elapsed time since the last contact are used to calculate social relationship. The proposed protocol consists of three phases: (1) distribution phase, (2) search phase, and (3) return phase. In the distribution phase, content information is disseminated by either producer nodes or expert nodes who received content from producer nodes. In the search phase, content requests are generated in a distributed way. In the return phase, nodes are selected to forward content to requester node.

In [15], the authors proposed content searching by using random regret minimization, where the decision on Interest replication is based on several measurable attributes, and the lifetime of a message is repeatedly scaled down to reduce overhead. In [16], the authors proposed a reputation-based content caching strategy to achieve a logical trade-off between the dissemination rate and the energy efficiency, where DTN stations deliver content stored in ICN stations. Since intermediate relay nodes have limited buffer storage, ICN provider selects appropriate DTN stations as caching nodes based on the reputation mechanism, where a reputation of a DTN nodes is determined by buffer capacity, forward rate of content, and battery energy to deliver the content. In [17], the authors proposed opportunistic named functions which rely on both user-defined Interests and locally optimal decisions based on remaining battery lifetime and device capabilities in information-centric disruption-tolerant networks. In [18], the authors proposed a content distribution and retrieval framework in disaster networks for public protection (CIDOR) in interworked ICN and DTN network architecture. They proposed a new Data structure called CIDOR-PRIT and a new DTN forwarding mechanism is proposed. In CIDOR-PRIT table, the requester end point ID of the received Interest is stored. The requester end point ID is used as the destination end point ID. To reduce overhead, both duplicated Interest and Data packets are suppressed. For redundancy elimination, CIDOR nodes aggregates similar Interests in the CIDOR-PRIT table and it is used to return the Data to requesters efficiently.

3. Proposed Protocol

In this paper, we propose an efficient DTN routing protocol for ICN and focus on Interest and Data delivery between DTN nodes. In DTN, network topology is highly dynamic and thus, the operation PIT and FIB is not efficient, as originally proposed for wired network. Reverse path forwarding using PIT in ICN cannot be used efficiently too, since topology is not stable due to the mobility of DTN nodes [18]. Therefore, Interest and Data should be delivered opportunistically by store-carry-forward mechanism of DTN, where Interest and Data are encapsulated within DTN bundle.

We consider an ICN environment, where multiple Interests are generated for the same requested Data by multiple requesters. For efficient dissemination of Interest and Data in DTN, we propose a status table with requested Data ID, requester ID, and satisfaction flag, as shown in Figure 1. Satisfaction flag is defined for a pair of requested Data ID and requester ID to show the delivery status of Data to the requester node, so that unnecessary forwarding of Data for already satisfied Interests (delivered Data) is avoided, which results in the reduction of Data overhead. In Figure 1, Data $d_1$ is requested by requesters $R_1$ and $R_2$ but none of the requests is satisfied yet. Data $d_2$ is requested by requesters $R_2$, $R_3$, and $R_4$, and Data $d_3$ is successfully delivered to requesters $R_2$ and $R_4$ already. We note that the status table in Figure 1 is managed by each node by using locally obtained information from other nodes during contacts, and each node operates based on locally obtained up-to-date information.

Figure 2 shows the exchange of the status table information defined in Figure 1 between two nodes when they contact each other. When node A and node B contact each other, they exchange summary vectors which include the information of status table. Then, each node updates its status
table using the received summary vector. In Figure 2, node A updates a satisfaction flag for a pair of Data $d_1$ and requester $R_1$ as 1 and adds a new entry for a pair of Data $d_2$ and requester $R_3$ and sets its satisfaction flag as 1, which was received from node B. Also, node B adds an entry for a pair of Data $d_1$ and requester $R_2$ and sets its satisfaction flag as 0, which was received from node A. Also, node B adds a new entry for a pair of Data $d_2$ and requester $R_4$ and sets its satisfaction flag as 1. By updating the values of status table, each node can manage up-to-date information of Data ID, requester ID, and satisfaction flag. We note that although Interests are generated at original requesters, they are not exchanged between contact nodes, but the information for requesters is shared using the exchange of status table, which has little overhead.

![Figure 1. A status table.](image1)

![Figure 2. Exchange of status table information.](image2)

Figure 3 shows the update of requester ID in Data stored in buffer by using exchanged status table information between two nodes when they contact each other. When a node A receives Data ID, requester ID, and satisfaction flag information from another node, the requester ID is updated, if corresponding Data already exists in buffer. As shown in Figure 3, if a status table information of node A was updated after contact, requester information for Data stored in the buffer of node A is updated. In this example, it is assumed that only Data $d_1$ and $d_2$ are already stored in the buffer of
node A. Since requester ID $R_2$ with satisfaction flag of 0 for Data $d_1$ is not present in Data $d_1$ of buffer of node A, requester ID $R_2$ is newly added to the Data $d_1$. Also, requester ID $R_2$ with satisfaction flag of 0 for Data $d_2$ is not present in Data $d_2$ of buffer of node A, requester ID $R_2$ is newly added to the Data $d_2$. However, requester ID $R_3$ for Data $d_2$, which is already present in Data $d_2$ of buffer of node A, is removed from the Data $d_2$ to reduce overhead, since the satisfaction flag of requester ID $R_3$ for Data $d_2$ is 1, which means that Data $d_2$ was successfully delivered to requester ID $R_3$ already. Since Data $d_3$ is not present in the buffer of node A, the requester $R_2$ with satisfaction flag 0 for Data $d_3$ is not used for the update of requester ID of Data stored in the buffer of node A.

![Buffer of node A](image_url)

**Figure 3.** Requester ID update in Data using status table information.

In the proposed protocol, the decision of choosing a forwarding node for Data delivery is based on delivery predictability, which is proposed in PRoPHET protocol [23]. Delivery predictability between node A and node B, i.e., $P(A,B)$, has a value from 0 to 1, and is calculated as follows:

$$P(A, B) = P(A, B)_{old} + (1 - \Delta - P(A, B)_{old}) \times P_{\text{encounter}},$$

(1)

$$P(A, B) = P(A, B)_{old} \times \gamma^K,$$

(2)

$$P(A, C) = \max(P(A, C)_{old}, P(A, B) \times P(B, C) \times \beta),$$

(3)

where $P_{\text{encounter}}$ is a scaling factor to control the increasing rate of the delivery predictability after contact. $\Delta$ is a parameter limiting an upper bound of delivery predictability. $\gamma$ is an aging constant with a value between 0 and 1. $K$ is the number of elapsed time units after the last contact. Finally, $\beta$ is a parameter to control transitivity with a value between 0 and 1.

In conventional DTN, Data is delivered from a single source node to a single destination node. In ICN, however, Data corresponding to the requested Interest should be replied to any of requesters. Therefore, when two nodes contact, the decision of forwarding Data stored in the buffer of each node is determined based on the comparison of average delivery predictability of a set of requesters for considered Data ID $i$, which is defined as follows:

$$P(A, S(R(i))) = \frac{1}{N_{\text{requesters}(i)}} \sum_{r=1}^{N_{\text{requesters}(i)}} P(A, R_r(i)),$$

(4)

where $S(R(i))$, $N_{\text{requesters}(i)}$, and $R_r(i)$ are a set of requesters, number of requesters, and requester $r$ for Data ID $i$, respectively.
Figure 4 shows a flowchart for the proposed algorithm. When a node A wants to receive Data, it generates an Interest corresponding to the requested Data, and the Data ID and requester ID with satisfaction flag of 0 are stored in the status table. When a node A contacts with node B, it exchanges summary vector with node B, and then status table information is updated and also information for Data, which is stored in the buffer of node A, is also updated accordingly. If node A has Data, where the destination of the Data is node B, it is delivered to node B. Otherwise, node A compares the average delivery predictability of node A for each stored Data with that of node B. Then, if node B has larger average delivery predictability, the Data is forwarded to node B. If node A is the destination node of Data which is stored in node B, the Data is delivered to node A from node B, and Interest stored in node A is removed. Then, the requester information stored in the Data is removed, too and the satisfaction flag of the Data with requester ID A is updated as 1.

![Flowchart of the proposed algorithm.](image)

In this paper, the performance of the proposed algorithm is compared with that of CIDOR protocol [18] and an opportunistic forwarding (OF) protocol [20,21], which also deal with efficient ICN delivery in DTN. Table 1 compares the characteristics between CIDOR protocol, OF protocol, and the proposed protocol. In the CIDOR protocol, Epidemic protocol [25] is used for Interest forwarding. In Epidemic protocol, which is a flooding-based protocol, messages are forwarded to all contact nodes whenever a node contacts other nodes. In CIDOR protocol, it was assumed that the total number of Interest replication is limited to 10 copies by restricting the total number of forwarding. For Data forwarding, Spray and Wait protocol [26] is used. Spray and Wait protocol consist of Spray phase
and Wait phase for message delivery, where the total number of message copies are limited to L. In Spray phase, a node, which generates an original message, forwards one message copy to L-1 other nodes when it contacts them. Then, L copies of messages are delivered to the destination node of the message only in Spray phase. In CIDOR protocol, 10 copies of messages are disseminated in Spray phase. In CIDOR protocol, therefore, the overhead of Interest and Data dissemination is controlled by limiting the replication of Interest and Data to 10 copies. In OF protocol [20,21], if a node contacts another node which has different requester information for Interest, the requester information is merged in its Interest to reduce overhead of Interest forwarding. Similarly, requester information is merged in Data, too. Interest forwarding is carried out based on extended PRoPHET protocol by using the delivery predictability to a set of nodes caching the requested Data. For Data forwarding, extended PRoPHET protocol by using the average delivery predictability to a set of requesters is used. However, no overhead control of Data was proposed in OF protocol. In the proposed protocol, requester information for Interest is shared by exchanging status table. Data is forwarded to a more probable node by comparing average delivery predictability to a set of requesters. The overhead of Data is controlled by reducing unnecessary forwarding of Data using satisfaction flag in status table.

Table 1. Comparison of CIDOR and the proposed protocol.

| Interest forwarding | CIDOR Protocol | OF Protocol | Proposed Protocol |
|---------------------|----------------|-------------|-------------------|
| Epidemic protocol (total number of copies: 10 copies) | Extended PRoPHET protocol by using the delivery predictability to a set of nodes caching the requested Data | Sharing of requester information in status table |

Data forwarding

| Spray and Wait protocol (total number of copies: 10 copies) | Extended PRoPHET protocol by using the average delivery predictability to a set of requesters | Extended PRoPHET protocol by using the average delivery predictability to a set of requesters |

Overhead control of Data

| Limitation of replications to 10 copies in Spray phase | None | Reduction of unnecessary forwarding of Data using satisfaction flag in status table |

4. Performance Analysis

In this paper, the Data delivery probability and Data overhead are compared by varying the values of buffer sizes, TTL of Data, and number of relay nodes by simulation using opportunistic network environment (ONE) simulator [27]. The Data delivery probability and Data overhead are defined as follows:

\[
\text{Data delivery probability} = \frac{1}{N_{\text{total}}(D)} \sum_{i=1}^{N_{\text{total}}(D)} \frac{N_{\text{delivered}(i)}}{N_{\text{requesters}(i)}}
\]

\[
\text{Data overhead} = \frac{1}{N_{\text{total}}(D)} \sum_{i=1}^{N_{\text{total}}(D)} \frac{N_{\text{relayed}(i)}}{N_{\text{delivered}(i)}}
\]

where \(N_{\text{total}}(D), N_{\text{delivered}(i)}, \) and \(N_{\text{relayed}(i)}\) are total number of requested Data IDs, number of delivered Data for Data ID \(i\), and number of relayed Data for Data \(i\), respectively.

Table 2 shows parameter values used in the simulation.
Table 2. Parameter values.

| Parameter            | Value                        |
|----------------------|------------------------------|
| Simulation time (s)  | 43,200                       |
| Transmit speed (Mbps)| 2                            |
| Transmit range (m)   | 10                           |
| Buffer size (Mbytes) | 10–55 (default: 30)          |
| Requester            | 50                           |
| Original Data caching node: | 10                         |
| Relay node           | 10–100 (default: 50)         |
| Speed of nodes (m/s) | U [0.5, 1.5]                 |
| Mobility model       | Shortest Path Map Based Movement |
| Message size (Mbytes)| Interest: 0.001              |
|                      | Data: U [0.5, 1.0]           |
| Interest generation interval (s) | U [25, 35]    |
| Number of Data       | 100                          |
| TTL (minutes)        | Interest: 90                 |
|                      | Data: 15–150 (default: 90)   |
| Cache replacement policy of CS | FIFO                  |

Figure 5 shows Data delivery probability for varying buffer sizes. The delivery probability increases as buffer sizes increases, and it saturates when the buffer sizes is sufficient. This is because more Data can be accommodated in the buffer and more Data can be delivered without dropping due to buffer overflow, as buffer sizes increases. When the buffer sizes is sufficient, the effect of increasing buffer sizes is not significant and thus the delivery probability saturates. The proposed protocol has higher delivery probability than both CIDOR and OF protocols, especially for higher buffer sizes. The proposed protocol has maximum 34% of increase than CIDOR protocol and has maximum 105% of increase than OF protocol. This is because effective forwarding of Data based on efficient requester ID management and effective overhead control using satisfaction flag achieve efficient Interest and Data delivery.

![Figure 5](image)

Figure 5. Data delivery probability for varying buffer sizes.

Figure 6 shows Data overhead for varying buffer sizes. The Data overhead decreases as buffer sizes increases, and it saturates when the buffer size is sufficient. This is because more Data can be accommodated in buffer for larger buffer sizes and thus additional Data forwarding due to dropped Data by buffer overflow decreases more. The OF protocol has higher Data overhead in most of the
considered buffer sizes, since it does not control Data overhead. The proposed protocol has maximum 26% of decrease than OF protocol. Since the proposed protocol generates more Data dissemination than CIDOR protocol, the proposed protocol has higher Data overhead when the buffer size is small. For large values of buffer sizes, however, since the proposed protocol has much higher delivery probability than CIDOR protocol, the Data overhead of the proposed protocol is smaller than the CIDOR protocol.

Figure 6. Data overhead for varying buffer sizes.

Figure 7 shows delivery probability for varying the number of relay nodes. We note that the number of nodes generating Interest messages does not change and only the number of relay nodes changes for consistent message generation patterns even for the change of total number of nodes. Delivery probability increases as the number of relay nodes increases since more Data can be delivered because of the more contacts between nodes. The proposed protocol has higher delivery probability than both CIDOR and OF protocols for all the considered number of relay nodes because of the effective Data forwarding and overhead control. The proposed protocol has maximum 59% of increase than CIDOR protocol and has maximum 154% of increase than OF protocol. We note that the proposed protocol has much higher delivery probability when the number of relay nodes is small due to more message dissemination, which overcomes the limitation of Data forwarding due to less contacts between nodes.

Figure 7. Data delivery probability for varying the number of relay nodes.

Figure 8 shows Data overhead for varying the number of relay nodes. The result shows that Data overhead increases as the number of relay nodes increases since more contacts between nodes occur
and more Data forwarding are generated. The OF protocol has the highest Data overhead, and the proposed protocol has maximum 29% of decrease than OF protocol. The proposed protocol has lower overhead than CIDOR protocol when the number of relay nodes is small due to much higher delivery probability. However, it has higher overhead than CIDOR protocol when the number of relay nodes is largely due to the dominant effect of more message dissemination.

![Figure 8. Data overhead for varying the number of relay nodes.](image)

Figure 8. Data overhead for varying the number of relay nodes.

Figure 9 shows delivery probability for varying TTL of Data. The result shows that the delivery probability increases as TTL increases since more Data can be alive in the buffer. The delivery probability in both the proposed and CIDOR protocols decreases slightly as the buffer size increases when the TTL value is very high. This is because very high TTL results in message drop due to buffer overflow. The proposed protocol has maximum 42% of increase than CIDOR protocol and has maximum 269% of increase than OF protocol. This is because effective forwarding of Data based on efficient requester ID management and effective overhead control using satisfaction flag achieves efficient Interest and Data forwarding.

![Figure 9. Data delivery probability for varying time-to-live (TTL).](image)

Figure 9. Data delivery probability for varying time-to-live (TTL).

Figure 10 shows Data overhead for varying TTL of Data. The proposed protocol has the lowest Data overhead, since it has the highest delivery probability. The OF protocol has the highest Data overhead, and the proposed protocol has maximum 24% of decrease than OF protocol. Also, the proposed
protocol has maximum 15% of decrease than CIDOR protocol. The result shows that the Data overhead does not change significantly for the change of TTL of Data, compared to the change of buffer sizes and the number of relay nodes. This is because the effect of unnecessary message forwarding due to message drop by buffer overflow is not significant for the change of TTL of Data.

![Figure 10. Data overhead for varying TTL.](image)

5. Conclusions

In this paper, we proposed an efficient DTN routing protocol for ICN. In the proposed protocol, requester information for Interest was shared by exchanging status table with requested Data ID, requester ID, and satisfaction flag. The satisfaction flag was defined to show the delivery status of Data, so that unnecessary forwarding of Data is avoided. In the proposed protocol, Data is forwarded to a more probable node by comparing the average delivery predictability to a set of requesters for a considered Data. Performance of the proposed protocol was evaluated using ONE simulator from the aspect of Data delivery probability and Data overhead, for varying buffer sizes, number of relay nodes, and TTL of Data. Then the performance of the proposed protocol was compared with both CIDOR and OF protocols.

Based on the performance analysis results, it was shown that the proposed protocol has higher delivery probability than both CIDOR and OF protocols for all the considered parameter values for varying buffer sizes, number of relay node, and TTL of Data. The results show that effective forwarding of Data based on efficient requester ID management and effective overhead control using satisfaction flag achieve efficient Interest and Data delivery. By sharing status table, Interest can be shared efficiently without much overhead. Satisfaction flag can efficiently limit Data overhead by avoiding unnecessary forwarding of already delivered Data. There was tradeoff between the proposed protocol and CIDOR from the aspect of Data overhead, for varying buffer sizes and number of relay nodes. The proposed protocol has higher Data overhead than CIDOR protocol for small buffer sizes and higher number of relay nodes due to the more dominant effect of more Data dissemination. For large buffer sizes and smaller number of relay nodes, on the other hand, the effect of better Data delivery is more dominant, and the proposed protocol has smaller Data overhead than CIDOR protocol. It is concluded that the proposed protocol has better delivery probability, although there is a tradeoff between CIDOR protocol from the aspect of Data overhead for varying buffer sizes and number of relay nodes.

**Author Contributions:** M.W.K., D.Y.S., and Y.W.C. conceived and designed the proposed protocol; M.W.K. and Y.W.C. conceived and designed the simulations; M.W.K. performed the simulations; M.W.K. and Y.W.C. analyzed the data; M.W.K., D.Y.S., and Y.W.C. wrote the original draft; M.W.K., D.Y.S., and Y.W.C. reviewed and edited the revised paper. All authors have read and agreed to the published version of the manuscript.
**Funding:** This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2017-0-00613, Development of Content-oriented Delay Tolerant networking in Multi-access Edge Computing Environment). This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2017-0-01633) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

**Acknowledgments:** The preliminary version of this paper, entitled “An Efficient Opportunistic Routing Protocol for ICN”, was presented at the Posters & Demos session of the 6th ACM Conference on Information-Centric Networking, which was held in Macao, China in 2019.

**Conflicts of Interest:** The authors declare no conflicts of interest.

**References**

1. Pentikousis, K.; Ohlman, B.; Corujo, D.; Boggia, G.; Tyson, G.; Davies, E.; Molinaro, A.; Eum, S. *Information-Centric Networking: Baseline Scenarios*; RFC 7476; Internet Research Task Force: Fremont, CA, USA, 2015.

2. Xylomenos, G.; Ververidis, C.N.; Siris, V.A.; Fotiou, N.; Tsilopoulos, C.; Vasilakos, X.; Katsaros, K.V.; Polyzos, G.C. A Survey of Information-Centric Networking Research. *IEEE Commun. Surv. Tutor.* 2014, 16, 1024–1049. [CrossRef]

3. Fang, C.; Yao, H.; Wang, Z.; Wu, W.; Jin, X.; Yu, F.R. A Survey of Mobile Information-Centric Networking: Research Issues and Challenges. *IEEE Commun. Surv. Tutor.* 2018, 20, 2353–2371. [CrossRef]

4. Seedorf, J.; Tagami, A.; Arumaiturai, M.; Koizumi, Y.; Melazzi, N.B.; Kutscher, D.; Sugiyama, K.; Hasegawa, T.; Asami, T.; Ramakrishnan, K.K.; et al. The Benefit of Information Centric Networking for Enabling Communications in Disaster Scenarios. In Proceedings of the 2015 IEEE Globecom Workshops (GC Wkshps), San Diego, CA, USA, 6–10 December 2015.

5. Cerf, V.; Burleigh, S.; Hooke, A.; Durst, R.; Scott, K.; Fall, K.; Weiss, H. *Delay-Tolerant Networking Architecture*; RFC 4838; Internet Research Task Force: Fremont, CA, USA, 2007.

6. Cao, Y.; Sun, Z. Routing in Delay/Disruption Tolerant Networks: A Taxonomy, Survey and Challenges. *IEEE Commun. Surv. Tutor.* 2013, 15, 654–677. [CrossRef]

7. Wei, K.; Liang, X.; Xu, K. A Survey of Social-Aware Routing Protocols in Delay Tolerant Networks: Applications, Taxonomy and Design-Related Issues. *IEEE Commun. Surv. Tutor.* 2014, 16, 556–578.

8. Islam, H.M.A.; Lukyanenko, A.; Tarkoma, S.; Yla-Jaaski, A. Towards Disruption Tolerant ICN. In Proceedings of the 20th IEEE Symposium on Computers and Communication (ISCC), Larnaca, Cyprus, 6–9 July 2015.

9. Anastasiades, C.; Schmid, T.; Weber, J.; Braumann, T. Information-centric content retrieval for delay-tolerant networks. *Comput. Netw.* 2016, 107, 194–207. [CrossRef]

10. Islam, H.M.A.; Chatzopoulos, D.; Lagutin, D.; Hui, P.; Ylä-Jääski, A. Boosting the Performance of Content Centric Networking Using Delay Tolerant Networking Mechanisms. *IEEE Access* 2017, 5, 23858–23870. [CrossRef]

11. Tyson, G.; Bigham, J.; Bodanese, E. Towards an Information-Centric Delay-Tolerant Network. In Proceedings of the 2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Turin, Italy, 14–19 April 2013.

12. Monticelli, E.; Schubert, B.M.; Arumaiturai, M.; Fu, X.; Ramakrishnan, K.K. An information centric approach for communications in disaster situations. In Proceedings of the 2014 IEEE 20th International Workshop on Local & Metropolitan Area Networks (LANMAN), Reno, NV, USA, 21–23 May 2014.

13. Souza, C.D.; Ferreira, D.L.; Campos, C.A.V. A Protocol for Data Discovery and Retrieval in Content-Centric and Delay-Tolerant Networks. In Proceedings of the 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall), Toronto, ON, Canada, 24–27 September 2017.

14. Souza, C.D.T.D.; Ferreira, D.L.; Campos, C.A.V.; Júnior, A.C.D.O.; Cardoso, K.V.; Moreira, W. Employing Social Cooperation to Improve Data Discovery and Retrieval in Content-Centric Delay-Tolerant Networks. *IEEE Access* 2019, 7, 137930–137944. [CrossRef]

15. Saha, B.K.; Misra, S. Named Content Searching in Opportunistic Mobile Networks. *IEEE Commun. Lett.* 2016, 20, 2067–2070. [CrossRef]
16. Arabi, S.; Sabir, E.; Elbiaze, H. Information-centric networking meets delay tolerant networking: Beyond edge caching. In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018.

17. Graubner, P.; Lampe, P.; Höchst, J.; Baumgärtner, L.; Mezini, M.; Freisleben, B. Opportunistic named functions in disruption-tolerant emergency networks. In Proceedings of the 15th ACM International Conference on Computing Frontiers, Ischia, Italy, 8–10 May 2018.

18. Islam, M.A.; Lagutin, D.; Lukyanenko, A.; Gurtov, A.; Ylä-Jääski, A. CIDOR: Content Distribution and Retrieval in Disaster Networks for Public Protection. In Proceedings of the 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Rome, Italy, 9–11 October 2017.

19. Seedorf, J.; Arumaithurai, M.; Tagami, A.; Ramakrishnan, K.; Melazzi, N.B. Research Directions for Using ICN in Disaster Scenarios; ICNRG draft-irtf-icnrg-disaster-10; Internet Engineering Task Force: Fremont, CA, USA, 2020.

20. Kang, M.W.; Kim, Y.; Chung, Y.W. An opportunistic forwarding scheme for ICN in disaster situations. In Proceedings of the 2017 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 18–20 October 2017.

21. Kang, M.W.; Chung, Y.W. Performance Analysis of a Novel DTN Routing Protocol for ICN in Disaster Environments. In Proceedings of the 2018 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 17–19 October 2018.

22. Kang, M.W.; Seo, D.Y.; Chung, Y.W. An Efficient Opportunistic Routing Protocol for ICN. In Proceedings of the 6th ACM Conference on Information-Centric Networking, Macao, China, 24–26 September 2019.

23. Lindgren, A.; Doria, A.; Davies, E.; Grasic, S. Probabilistic Routing Protocol for Intermittently Connected Networks; RFC 6693; Internet Engineering Task Force: Fremont, CA, USA, 2012.

24. Haas, Z.J.; Small, T. A New Networking Model for Biological Applications of Ad Hoc Sensor Networks. IEEE/ACM Trans. Netw. 2006, 14, 27–40. [CrossRef]

25. Vahdat, A.; Becker, D. Epidemic Routing for Partially-Connected Ad Hoc Networks; Technical Report CS-200006; Duke University: Durham, NC, USA, 2000.

26. Spyropoulos, T.; Psounis, K.; Raghavendra, C.S. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, Philadelphia, PA, USA, 22 August 2005; pp. 252–259.

27. The Opportunistic Network Environment Simulator Home Page. Available online: https://akeranen.github.io/the-one/ (accessed on 18 April 2020).

© 2020 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 (http://creativecommons.org/licenses/by/4.0/).