Adapting Named Data Networking (NDN) for Better Consumer Mobility Support in LEO Satellite Networks

Zhongda Xia
Harbin Institute of Technology
Harbin, Heilongjiang, China
xiazhongda@hit.edu.cn

Yu Zhang
Harbin Institute of Technology
Harbin, Heilongjiang, China
Peng Cheng Laboratory
Shenzhen, Guangdong, China
yuzhang@hit.edu.cn

Teng Liang
Peng Cheng Laboratory
Shenzhen, Guangdong, China
liangt@pcl.ac.cn

Xinggong Zhang
Peking University
Beijing, China
zhangxg@pku.edu.cn

Binxing Fang
Harbin Institute of Technology
Harbin, Heilongjiang, China
Peng Cheng Laboratory
Shenzhen, Guangdong, China
fangbx@hit.edu.cn

ABSTRACT

Large low Earth orbit (LEO) satellite constellations provide low-latency and high-bandwidth Internet connectivity at the global scale. One major challenge is to handle frequent satellite handovers. Named Data Networking (NDN) adopts a pull-based communication model, which allows users to retrieve data that fail to come back because of satellite handovers by retransmitting the corresponding requests, hence simplifying mobility management when retrieving data. However, we find that relying on such retransmissions alone can be highly inefficient in typical LEO satellite constellations. Specifically, typical inter-satellite topologies and satellite handover strategies may produce bad cases for retransmissions, generating a significant amount of additional traffic. Motivated by this observation, this paper attempts to consolidate NDN's advantage in mobility management with the Data Recovery Link Service (DRLS), a shim layer service operating between the network and link layer in the NDN protocol stack. DRLS hides recurring satellite handovers from forwarding by recovering data from the previously connected satellite via alternative paths, thus ensuring the bidirectional request-response exchange of NDN without retransmitting requests. A prototype of DRLS is implemented in the reference NDN software forwarder and evaluated through simulations. Results prove the efficacy of the proposed mechanism at reducing the overall traffic volume.

MSWiM '21, November 22–26, 2021, Alicante, Spain

ACM ISBN 978-1-4503-9077-4/21/11…$15.00
https://doi.org/10.1145/3479239.3485699

CCS CONCEPTS

• **General and reference → Design**; *Empirical studies*; *Evaluation*; *Experimentation*; *Performance*; • **Networks → Cross-layer protocols**; **Intermediate nodes**; **Network performance evaluation**; **Network mobility**; **Mobile networks**.

KEYWORDS

Information-centric networking (ICN); Named-Data Networking (NDN); large LEO satellite constellations; NDN consumer mobility; NDN link service

ACM Reference Format:
Zhongda Xia, Yu Zhang, Teng Liang, Xinggong Zhang, and Binxing Fang. 2021. Adapting Named Data Networking (NDN) for Better Consumer Mobility Support in LEO Satellite Networks. In *Proceedings of the 24th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '21), November 22–26, 2021, Alicante, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3479239.3485699

1 INTRODUCTION

Into the 5G era, communication systems are expected to provide general-purpose and ubiquitous communication service connecting anyone, anything at anywhere, anytime. Satellite communication systems, with their inherent coverage and resiliency advantages, have been recognized as a powerful complement to their terrestrial counterpart for realizing this goal. Particularly, large *low Earth orbit (LEO)* satellite constellations [22, 23] containing up to tens of thousand of inter-connected satellites are proposed to provide global low-latency high-bandwidth Internet connectivity. Due to their very large scale and highly mobile nature, it is still unknown how to design the networking of large LEO satellite constellations.

Although the IP architecture gains great success in terrestrial networking, it faces significant challenges in this scenario, especially with mobility management [12]. A LEO satellite at 550 km altitude travels at 7.6 km/s and takes only about 95 minutes to orbit the Earth, and a user on the ground would experience a satellite handover approximately every 5 minutes, which is extremely challenging for IP mobility management at the global scale. Proposed future Internet architectures such as Named Data Networking (NDN) [25] are more friendly to mobility, and promise to be better solutions for satellite networking.

NDN follows the information-centric networking (ICN) communication paradigm, joining the efforts to revolutionize the Internet with a clean-slate design that prioritizes modern application needs such as content distribution and mobility. Among representative ICN proposals including MobilityFirst [24] and eXpressive Internet Architecture (XIA) [20], NDN has attracted the most attention from both industry and academia. With an active research community and a strong codebase [10], NDN is expected to see spin-offs in various application scenarios including IoT [1, 3, 4], disaster relief [21], and tactical communications [5]. The idea of applying NDN to satellite networking is also slowly gaining traction. In our another ongoing work [17], we briefly discussed the major incentives for such a combination. Native NDN features such as pull-based communication model, adaptive forwarding, in-network caching, and built-in security provide satellite networking with desirable traits including resilient forwarding, efficient bandwidth utilization, and native mobility support. Such incentives have also been discussed and verified to various degrees in several related works [8, 16].

Regarding mobility, NDN provides native support for the mobility of users requesting data (*consumers*), while supporting the mobility of users providing data (*producers*) still require additional mechanisms. In NDN's pull-based communication model, consumers proactively retrieve data by sending requests with names (*Interest*), which build transient per-packet forwarding paths for data to follow back. After a handover, consumers simply retransmit Interests that fail to bring data back. A retransmitted Interest will build a new path to retrieve data back, resuming communication without performing any form of mobility management. NDN producer mobility, as surveyed in [26], can be supported by tracking producer locations [27], or by moving data to compensate for producer movements [15].

This paper investigates NDN consumer mobility support in the LEO satellite constellation scenario, and finds that Interest retransmission at consumer suffers from high traffic overhead. More specifically, typical inter-satellite topology and satellite handover strategy collectively produce bad cases where a significant amount of traffic is wasted. Considering the fact that the satellites a user is connected to before and after a handover are likely within a couple of hops away, this paper proposes to provide efficient consumer mobility support by "recovering" data directly from the previously connected satellite. Such *data recovery* mechanisms operate in the shim layer between the network and link layer to hide recurring consumer mobility from forwarding, and deal with consumer mobility in a flexible and efficient manner without retransmitting Interests.

A architectural framework to support such shim layer mechanisms within the current NDN protocol stack is proposed. This framework changes how NDN nodes manage *faces*, the NDN abstraction of communication channels through which network layer packets are delivered. Such changes prevent a face from being removed after the associated link breaks, and instead allow such a face to send or receive data to finish pending data retrievals. Further, we propose the *Data Recovery Link Service (DRLS)*, an NDN *link service* that implements the shim layer mechanisms under the framework above. DRLS follows a light-weight design, and recovers data in a best-effort manner to complement NDN's native consumer mobility support, which remains as the last resort solution. The proposed shim layer mechanisms are transparent to both NDN forwarding and the LEO satellite constellation network environment, thus may serve as a buffer zone allowing both to evolve on their own, and a plug-and-play test field for trialing mechanisms that may be eventually integrated into upper or lower layers. A prototype of DRLS is implemented in the reference NDN software forwarder and evaluated via simulations. Results prove DRLS's advantage at reducing traffic overhead in large LEO satellite constellations.

This paper makes the following major contributions:

- Identifies an issue with NDN's native consumer mobility support in LEO satellite constellations.
- Defines an architectural framework to deal with consumer mobility within the shim layer between the network and link layer.
- Proposes a light-weight and best-effort protocol that implements efficient consumer mobility support under the proposed framework.
- Develops a solution for simulating an NDN network in LEO satellite constellations.

The rest of the paper is organized as follows. Section 2 gives a brief introduction to NDN and the LEO satellite constellation networking environment, then explains the design rationale of the shim layer mechanisms; a brief review of related works that apply NDN to satellite networking is also given. The data recovery framework which defines a new way to manage NDN faces is introduced in Section 3. The design of DRLS is proposed in Section 4, including discussions about key design choices and impacts. Section 5 presents evaluation results, and finally Section 6 concludes the paper.

2 BACKGROUND

2.1 A brief introduction to NDN forwarding

NDN defines two types of packets, *Interest* and *Data*. Interests are sent by *consumers* as requests for named data, while Data are generated by *producers* to represent named data. An Interest is forwarded by matching the specified name against a *Forwarding Information Base (FIB)* to determine the next hop, while where the Interest is received from is recorded in a *Pending Interest Table (PIT)*. A data source responds with a Data to satisfy a received Interest, then Data is forwarded according to PITs along the reverse forwarding path(s) of the corresponding Interest to eventually reach the consumer(s), thus concluding a successful data retrieval. Data may also be cached along the way in a *Content Store (CS)* to directly satisfy Interests.

2.2 NDN consumer mobility in LEO satellite constellations

Modern LEO satellite constellations are constructed by rapidly moving LEO satellites inter-connected with *inter-satellite links (ISLs)*. *Ground users*, e.g., ground stations, satellite phones, directly connect to satellites, i.e., their *access satellites*, over *user links*. The LEO satellites are uniformly distributed in various orbits sharing the same inclination to construct a relatively-stable inter-satellite geometry.

The lower the orbit height, the faster satellites move relative to ground, the more frequent ground users undergo satellite handovers. For ground user consumers, LEO satellite handovers would frequently break the reverse forwarding path of Data, leading to data retrieval failures. NDN natively supports such consumer mobility with Interest retransmission. A consumer may retrieve Data lost due to a handover simply by retransmitting the corresponding Interest after the handover. Retransmitted Interests are not necessarily forwarded all the way to the producer, they may be aggregated in PITs or hit caches at the nearest point where the forwarding paths of the initial and retransmitted Interest cross. The nearer this cross point is from the consumer's current network location, the less additional traffic will be produced as a result of the retransmission.

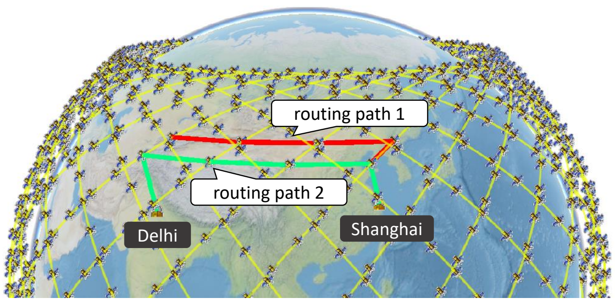


Figure 1: A case for late crossing of the forwarding paths of the initial and retransmitted Interest. The consumer is located at Delhi (left), while the producer is located at Shanghai (right). The red lines represent the forwarding path of the initial Interest, and the green lines represent the forwarding path of the retransmitted Interest.

However, we find that the grid-like inter-satellite topology and localized satellite handover pattern in LEO constellations may collectively produce bad cases for Interest retransmission, generating a significant amount of additional traffic with each handover. To study the relationship between Interest paths before and after a handover, we wrote a program to simulate the movement of constellations, and use Cesium [2], a 3D visualization platform for geospatial data, to visualize Interest forwarding paths. Visualization results are shown in Fig. 1. With a Starlink-like constellation, a consumer located at Delhi requests Data served by a producer located at Shanghai. Using Dijkstra algorithm to compute the shortest routing path, the forwarding paths of Interests before and after a consumer's handover are shown as routing path 1 (red) and routing path 2 (green), respectively. In this case, Data would travel over a total of 9 hops, while the consumer is only 5 hops away from the producer after the handover. Roughly 44.5% of Data traffic is wasted in this case. Considering the high handover frequency, and potentially large number of outstanding Interests, the phenomenon above may lead to significant performance degradation in realistic setups. Such observations hin the need for additional measures to augment NDN's native support for consumer mobility in LEO constellations.

2.3 Related Work

To the best of the authors' knowledge, a very limited number of attempts have been made to apply NDN to satellite networking. Among such attempts, only a few investigated the consumer mobility support issue.

Chen et al. [8] applies NDN to a federated Satellite-5G network. A novel Sat5G-ICN networking paradigm is proposed to incorporate satellite, ICN, and 5G features in an integrated architecture. Considering the feasibility of adopting NDN, a "reverse path broken problem" is identified, which is essentially the NDN consumer mobility problem. Two solutions are proposed to ensure continuous ICN-based communication service: the former initiates Interest retransmission from the edge of the ground segment of the network; the latter superimposes a stable virtual topology on top of the physical topology by adopting the Virtual Node network model. The first solution improves reactivity to handovers, but still suffers from wasted traffic due to retransmissions. As to the second solution, although the Virtual Node network model [14] completely masks satellites' mobility, it also requires state transitions which might be complex and expensive in the face of NDN stateful forwarding. Also, it remains unclear how the Virtual Node topology may be efficiently applied to non-Iridium-like constellations.

Li et al. [16] proposes to empower the Satellite-Terrestrial Integrated Network (STIN) [13] paradigm with ICN and Software-Defined Networking (SDN). To enhance content distribution performance via centralized control, a cooperative caching scheme is proposed, which effectively reduces data retrieval delay and enhances the overall network capacity. The Multiple-Layer Satellite Network (MLSN) environment exhibits more complex behavior than the LSN environment considered in this paper, and points to an interesting research direction. We believe that the proposed mechanisms in both papers may be combined to further enhance content distribution performance.

Liu et al. [18] proposes a routing mechanism for NDN in MLSN. The proposed mechanism requires substantial changes to the NDN architecture, including adding additional fields to Interest and Data, and using source routing for both Interest and Data forwarding. The impact of such fundamental changes to NDN forwarding behavior remains unclear. Our approach is transparent to NDN forwarding, and an architectural view is provided to clarify the introduced changes to the NDN protocol stack.

Cha et al. [7] studies NDN consumer mobility when NDN is deployed as an overlay over IP. Similar to our work, they also extends the functionality of NDN link service to directly retrieve Data back from a recently disconnected NDN gateway. However, this work studies a very restrained mode of NDN deployment, and thus lacks universality to be applied to other network environments where, unlike when deployed over IP, not every NDN node is only one hop away.

Another work of this paper's authors [17] also studies NDN consumer mobility in LEO satellite constellations. The two work share similar motivations, and employ similar evaluation methods and setups. However, their focus and approaches are different. Liang et al. [17] serves more as an introduction to a promising novel research direction for relevant research communities, and may thus be considered a prelude to this work. Solution-wise, Liang et al. [17] employs native NDN features, while our work attempts to address the problem below the NDN network layer. The two papers complement each other in the sense that they explore different directions to integrating NDN and satellite networks: by evolving the NDN architecture itself, or by bridging the gap with a shim layer. We believe that both directions deserve further investigation, and the efforts would jointly advance the research on incorporating satellite networking as a critical part of the future Internet.

3 THE DATA RECOVERY FRAMEWORK

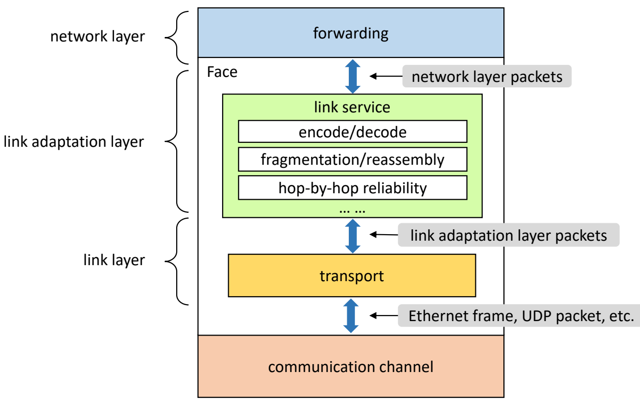


Figure 2: The NDN protocol stack architecture based on the NFD design

NDN forwarding is built on the *face* abstraction. According to the design of the reference NDN forwarder *NDN Forwarding Daemon (NFD) [9]*, as shown in Fig. 2, face abstracts various types of underlying communication channels (e.g., physical links, TCP/UDP sockets, and Unix stream sockets) to provide a uniform communication interface for network layer packet delivery. The data recovery framework further extends the face abstraction with a *data recovery* capability, which further unleashes the power of the face abstraction to bridge the gap between the expectation of forwarding and the characteristics of underlying links. Specifically, when a user link breaks during a satellite handover, the associated face on the satellite/ground user may still send/receive Data packets to complete pending Data retrievals over alternative paths, bypassing NDN forwarding on intermediate nodes.

To support this new capability, the data recovery framework defines a new face status, and updates face lifecycle management accordingly, as shown in Fig. 3. In a simplified version of face lifecycle management, a new face is created when a new communication channel becomes available, and the status is set to "UP", indicating that this face may be reliably used for packet delivery. When the communication channel breaks, the status of the associated face changes to "DOWN". The face as well as related states in FIB and PIT are then removed. In the data recovery framework, however, if the broken communication channel is a user link (e.g., the user link between GT-A and SAT-1), the face enters the *"FINALIZING"* state. In this state, data recovery mechanisms attempt to "recover" lost Data from the previous access satellite (e.g., SAT-1) over alternative paths to the ground user (e.g., GT-A), and related states in PIT are kept untouched. Further, since a face in "FINALIZING" state does not have an associated active communication channel, the data recovery framework also allows faces to directly exchange information without going through forwarding. This allows a "FINALIZING" face to send/receive over active links associated with "UP" faces. Note that the data recovery framework is transparent to the routing plane. "FINALIZING" faces only serve to deliver Data from the previous access satellite to the ground user, the routing plane should still consider the user link broken and update routes accordingly.

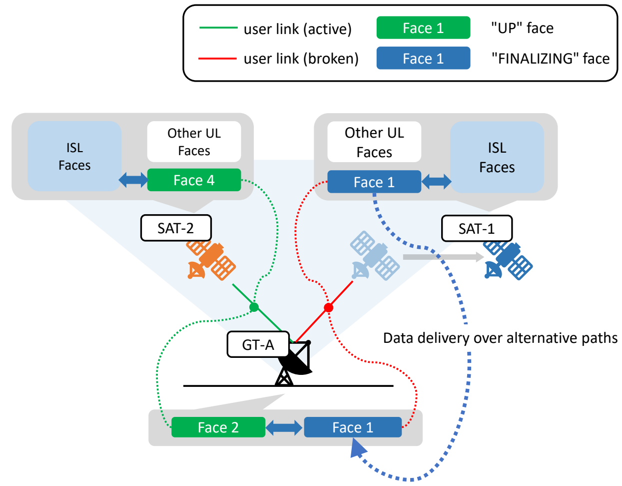


Figure 3: Illustration of the data recovery framework

Under this framework, various shim layer data recovery mechanisms in the form of NDN *link service* may be designed to provide the data recovery service, as will be exemplified by the design of DRLS.

4 THE DATA RECOVERY LINK SERVICE

As shown in Fig. 2, NDN link service constitutes the upper part of the NDN *face* abstraction to provide the uniform communication interface for network layer packet delivery, while *transport* wraps link layer protocols to compose the lower part. Architecturally, link service operates as the shim layer, i.e., the link adaptation layer, between the network and link layer to translate packets between layers, and provide additional services on top of the basic delivery service provided by underlying links to facilitate forwarding operations. Within the data recovery framework defined in Section 3, DRLS provides a data recovery service to bridge the gap between NDN's stateful data retrieval process and recurring consumer mobility caused by rapidly moving LEO satellites.

The DRLS design is based on the default link service in NFD, namely the *Generic Link Service (GLS)*, which provides general-purpose services including fragmentation/reassembly and hop-by-hop reliability. As shown in Fig. 4, in accordance with the internal stacked architecture of GLS, DRLS adds a *data recovery module* at the top to provide the data recovery service. The data recovery module would set additional fields, namely *data recovery fields*, in link adaptation layer packets. With the support from the data recovery framework, such packets may be directly exchanged between data

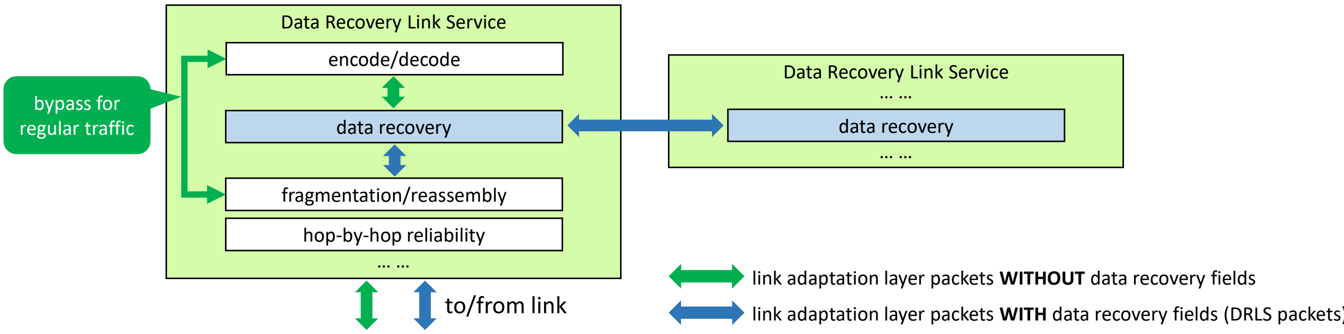


Figure 4: DRLS architecture

recovery modules of different faces. Specifications of DRLS include a set of data recovery fields as an extension to the NDNLPv2 [11] packet format, additional data structures, and a data recovery workflow that defines the operations of a broadcast-based data path discovery protocol, and a stateless data forwarding protocol.

4.1 Packet format

GLS follows the NDNLPv2 packet format specifications. As an extension to GLS, DRLS extends the NDNLPv2 packet format specifications. In NDNLPv2, a link adaptation layer packet is represented by an `LpPacket`. The `LpHeaderField` is a repeatable optional structure storing various types of headers, the standard way to add NDNLP features is to add new header types by extending the definition of `LpHeaderField`. DRLS follows this practice and adds a `DataRecoveryHeader` type. A `DataRecoveryHeader` may carry five fields:

- `OpCode` is mandatory, it indicates the role of a packet and determines how DRLS processes the packet. Its value is an enum type of three possible values "Discovery", "Response", and "Payload"; correspondingly, the `LpPacket` may be referred to as a *discovery packet*, *response packet*, or *payload packet*.
- `UserLinkId` is mandatory, its value is the identifier of a user link.
- `Path` is mandatory. For discovery packet, it records the path information; for response and payload packet, it provides information for forwarding.
- `Nonce` is required in discovery and response packet to distinguish different attempts to establish paths for the same user link.
- `HopLimit` is required only in discovery packet to control its broadcast scope.

Additionally, if `OpCode` is "Payload", the LpPacket must carry a `Fragment` field, which is where (fragments of) network layer packets (Data) are stored.

4.2 Data structure

DRLS defines two data structures: *User Link Table (ULT)* and *Data Path Table (DPT)*. ULT stores the *one-to-one* mapping between identifier of user links and faces. A ULT entry is added for each newly established user link to record the mapping. DPT stores information about discovered paths, which are sequences of face names as will be explained in details later.

4.3 Workflow

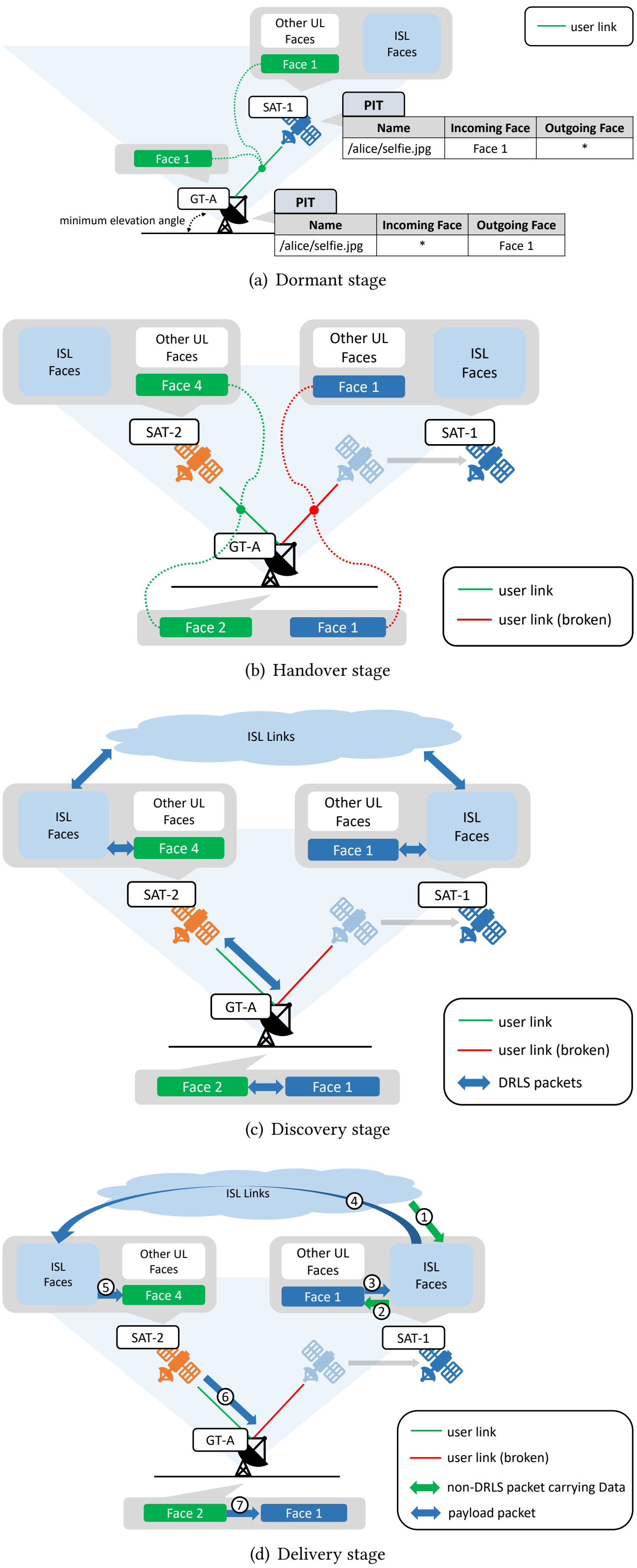
Based on the definitions above, we proceed to explain how DRLS performs data recovery. For ease of exposition, we refer to `LpPackets` with `DataRecoveryHeader` as DRLS packets, and only consider `DataRecoveryHeader` fields which are exclusively processed by data recovery modules, and the `Fragment` field that carries the network layer packet. Thus we ignore underlying link service modules such as packet fragmentation and reliability, which are transparent to DRLS operations.

For a single ground user consumer that experiences a handover, the data recovery workflow consists of four stages, namely *dormant stage*, *handover stage*, *discovery stage*, and *delivery stage*. An illustration of these stages is given in Fig. 5 in a representative case, while details are shown in Fig. 6.

4.3.1 Dormant stage (Fig. 5(a)). Consider a ground terminal GT-A and its access satellite SAT-1, a face associated with the user link between GT-A and SAT-1 is created on both sides (both named "Face 1" in this case). An identifier *ID* (in this case, $ID = SAT-1_GT-A$) for the user link between GT-A and SAT-1 is agreed upon by both parties and stored in ULT (in the `User Link ID` field) alongside the identifier of the associated face (in the `Face ID` field). On GT-A, Interests are sent out through "Face 1", creating PIT entries that indicate "Face 1" as the outgoing face, which means that Data are expected to come back through "Face 1". Similarly, PIT entries are created on SAT-1 indicating "Face 1" as the face through which Data should be sent back. In this stage, data recovery modules are bypassed for the traffic between GT-A and SAT-1, and DRLS behaves just like GLS.

4.3.2 Handover stage (Fig. 5(b)). GT-A handovers from SAT-1 to SAT-2, breaking the user link between GT-A and SAT-1. The associated faces enter the "FINALIZING" state. Into the handover stage, GT-A and SAT-1 will both mark the user link as recently broken for a limited period of time by updating the `User Link Status` field of the corresponding entry to "BROKEN".

4.3.3 Discovery stage (Fig. 5(c)). In the discovery stage, once other user link(s) become available, GT-A will broadcast a discovery packet. The `UserLinkID` field is set to *ID*, `HopLimit` field is set



(a) Dormant stage

(b) Handover stage

(c) Discovery stage

(d) Delivery stage

Figure 5: Stages in the data recovery workflow

according to the configured broadcast scope (2 in this case), and `Path` field is set to empty. The current access satellite SAT-2 would receive this packet. The data recovery module in "Face 4" would process this packet by first matching the `UserLinkID` field against the `User Link ID` field in ULT. If no match is found, which is the case for SAT-2, then this satellite is the not the access satellite of this user link, thus this packet will be further broadcasted over all ISLs. If a match is found, then the packet will trigger other operations as will be later explained when SAT-1 receives this packet. Before forwarding, the `HopLimit` field is decremented, and for each outgoing packet, local names of the incoming and outgoing face is appended to the `Path` field. Satellites connected to SAT-2 with ISLs will receive the broadcasted packet, and match ULT in the same way. If the `Hop Limit` value is one, the packet will not be further broadcasted. Assume that SAT-1 is connected with SAT-2 with an ISL, SAT-1 will receive the broadcasted DRLS packet. The `UserLinkID` field would match a ULT entry, triggering SAT-1 to send back a response packet, whose `UserLinkId` field is set to *ID*, and `Path` field is set according to the `Path` field of the discovery packet, through the same face that the discovery packet was received from. SAT-1 will also update its DPT by adding an entry for *ID*, which now becomes the identifier of the data path from SAT-1 to GT-A. The `Data Path` of this entry will be set according to the `Path` field of the received discovery packet. The `User Link Status` field of the ULT entry for *ID* should then be updated to "RECOVERED" to prevent from responding to subsequent discovery packets whose `UserLinkId` is set to *ID*, as well as enabling sending Data out through the associated face. The response packet will be forwarded by SAT-2 by checking and updating the `Path` field, the head of the sequence in `Path` determines the face to forward this response through, and is popped out from `Path` before forwarding. The response packet will reach GT-A. Like SAT-1, GT-A also match ULT to determine whether this response packet is expected. If matched, as is the case here, the corresponding ULT entry is updated by setting the `User Link Status` field to "RECOVERED". Such an update indicates that DRLS is successful in finding an alternative path, and Data is expected to come back soon afterwards.

4.3.4 Delivery stage (Fig. 5(d)). When Data arrive at SAT-1 to satisfy pending Interests from GT-A (Step 1), they would be processed by forwarding to be sent out through "Face 1" (Step 2 and 3). The data recovery module in "Face 1" would generate a payload packet carrying the data in the `Fragment Field`, and whose `Path` field is set by checking DPT for the data path associated with the corresponding broken user link. Since "Face 1" no longer associates with an active link, this payload packet will be passed to other faces to be sent out over ISL(s) (Step 4). Between NDN nodes, this payload packet would be forwarded according to the `Path` field in the same way as response packets, and eventually reach GT-A's current access satellite, SAT-2, and traverse the user link to reach GT-A (Step 5 and 6). Finally, the payload packet is passed to "Face 1" and the carried Data is extracted and passed on to the network layer. NDN forwarding would observe that Data are received from "Face 1" to satisfy pending Interests, and complete the data retrievals as if no handover occurred.

4.4 Discussion

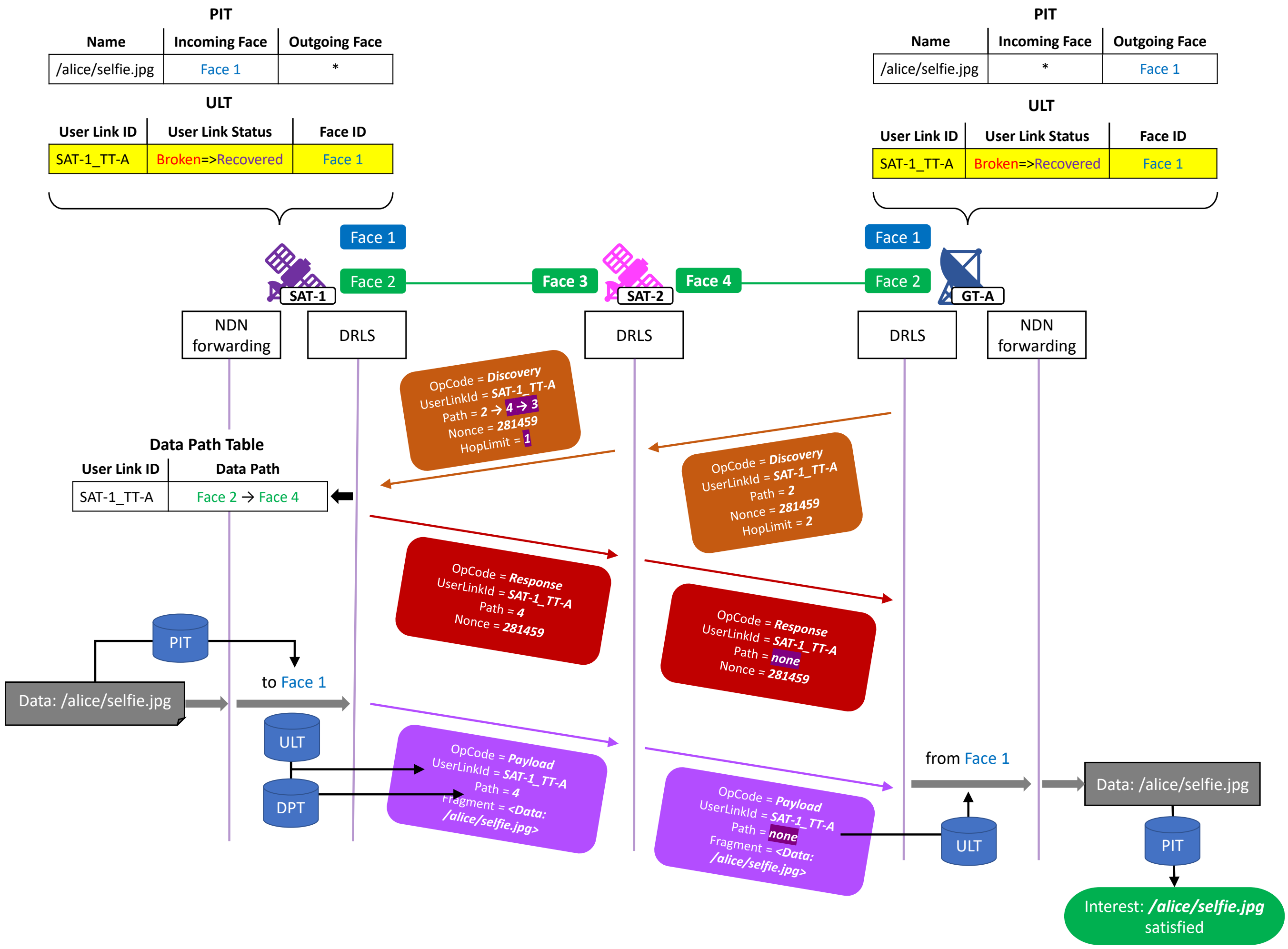


Figure 6: Detailed DRLS operations in the four stages of the data recovery workflow

4.4.1 *Setting the broadcast scope.* The best-effort broadcast-based approach does not guarantee the discovery of data path(s). The broader the broadcast scope, the higher the probability a discovery packet will reach the previous access satellite, while signaling overhead would also increase exponentially. However, in typical LEO constellations, it is highly likely that a current access satellite is adjacent or rather near to a recently disconnected one, thus even a small broadcast scope of a couple of hops should suffice to allow T-Req to reach the previous access satellite. For less deterministic access satellite handover patterns, a scope-restricted routing protocol may be employed to discover the previous access satellite with controllable overhead.

4.4.2 *Network layer packet buffering.* There can be cases when a network layer packet to be forwarded through a "FINALIZING" face comes before the corresponding data path is discovered. In such cases, network layer packets should preferably be buffered in per user-link queues to allow them to be sent out later when a data path is discovered.

4.4.3 *Interaction with the forwarding plane.* The power of DRLS may be further unleashed by interacting with the forwarding plane. For example, DRLS may inform the forwarding plane to retransmit pending Interests sent through a "FINALIZING" face through other "UP" faces associated with active user links, if DRLS fails to discover a data path for the corresponding broken user link.

5 EVALUATION

DRLS is implemented in NFD, and evaluated in a consumer mobility scenario over the *ndnSIM* 2.x [19] platform. We developed a helper program that deals with satellite networking environment setups, the support for which is currently not built in ndnSIM. This program takes satellite constellation parameters and ground user locations as input, to generate a series of snapshots of the network topology, while ndnSIM runs an NDN network over the dynamic LEO constellation topology to collect metrics that reflect data retrieval performance, including *data retrieval delay*, *data retrieval speed*, and *traffic waste ratio*.

Simulations are conducted in an experimental Starlink-like constellation, consumers and producers are placed at various cities, and the data retrieval performance of DRLS is compared with that of two Interest retransmission mechanisms 5.1: 1) application Interest retransmission (AIR), and 2) in-network Interest retransmission.

5.1 Benchmark mechanisms

Both AIR and IIR are natively supported reliability mechanisms in NDN. With AIR, a consumer would retransmit Interests after waiting for an arbitrary period according to observed RTTs, as will be further explained in Section 5.3. With IIR, as studied by Carofiglio et al. in [6], the NDN forwarder of a consumer would immediately retransmit all pending Interests sent through a broken user link through other available user links after a handover, this would produce the ideal case for Interest retransmission in terms of data retrieval delay. Neither mechanism attempts to optimize the forwarding path of retransmitted Interest, thus should suffer from the consumer mobility support issue identified in this paper.

5.2 Experiment setup

5.2.1 Constellation parameters. The simulated LEO constellation is based on the Starlink phase-1 deployment. A total number of 1584 LEO satellites are running at the height of $550km$ on 24 circular inclined orbits, whose inclination are set to 53.8°. The elevation angle, which determines the visibility of satellites, is set to 25°, which is a typical setting for open areas under ideal weather condition.

5.2.2 Link assignments. Each satellite is set to four persistent ISLs with two adjacent satellites in the same orbit plane, and one closest satellite in each of the two neighboring orbit planes. For user links, a ground user may set up only a single link with one satellite at a time, while a satellite may establish multiple user links to different ground users. ISLs and user links are modeled as perfect point-to-point links with a delay of 10 milliseconds.

5.2.3 Satellite handover strategy.

We consider a simple strategy: always prefer the nearest satellite in the same orbit plane as the current access satellite. This strategy implies that most of the time the previous and current access satellites would be within a couple of hops away.

5.2.4 Routing. A hypothetical virtual topology routing mechanism is adopted. The topology is considered static within a short time slot (1 minute). The topology in each time slot is pre-calculated based on satellite movements, ground user locations, and satellite handover strategy. Routes to each producer are calculated using the Dijkstra algorithm over the topology in each time slot, and updated instantaneously on each node.

5.3 Consumer behavior

We use constant-bit-rate (CBR) consumer sending Interests out at a constant speed (consumer sending speed, measured with Interests/second), with even time spans between transmissions. A retransmission timer is set according to recently observed RTTs, and once a timeout occurs, a retransmission is scheduled in roughly 2 RTTs. Both new and retransmitted Interests are queued before sending out at the constant speed.

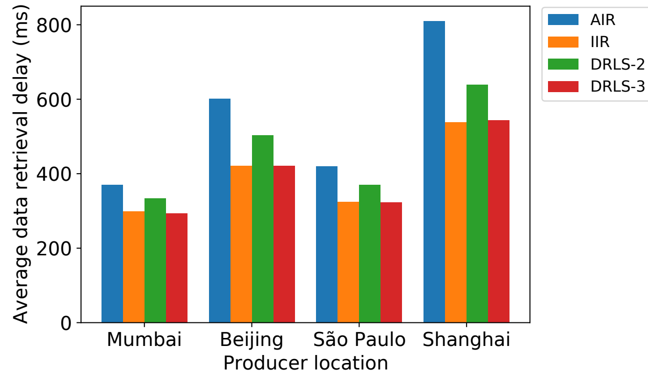


Figure 7: Average data retrieval delay vs. producer location (consumer sending speed = 300 Interests/second)

5.4 Results

In the simulations, a CBR consumer is placed at New York, while four producers are placed in Beijing, Shanghai, Mumbai, and SÃčo Paulo, respectively. The CBR consumer sending Interest our at between 50 to 300 Interests/second with steps of 50. For DRLS, we experiment with two broadcast scope settings of 2 (DRLS-2) and 3 (DRLS-3) hops.

With the satellite handover strategy described in Section 5.2.3, the consumer in New York would experience 40 handovers during one orbit period of 95 minutes, and the average topological distance (hops) between the previous and current access satellite across all handovers is 1.82.

We focus on the impact of satellite handovers, and thus only evaluate the data retrieval process within a short time period (2 seconds in total) before and after each handover.

5.4.1 Data retrieval delay. Data retrieval delay refers to the elapsed time from sending the first Interest to finally receiving the requested data, and thus indicates a mechanism's ability to smoothen the handover process. The average data retrieval delay across all successful data retrievals under different producer location settings are shown in Fig. 7. Because results are similar under different consumer sending speed settings, we present here the results under the highest setting of 300 Interests/second. AIR performs the worst as the retransmission (triggered by the timeout) under this mechanism is not conducted promptly after a handover. IIR performs better than DRLS when the broadcast scope setting is 2 hops, this is because DRLS-2 may fail and fall back to AIR when the previous and access satellite are not directly connected. However, when the broadcast scope setting is 3 hops, DRLS performs almost as well as IIR. This is because the previous and current access satellite across all handovers are seldomly more than 2 hops away from each other, and DRLS rarely falls back to AIR, allowing Data to come back as soon as possible.

We further evaluated how different mechanisms affects the delay jitter during handovers. A CDF of data retrieval delays for different producer locations is shown in Fig. 8. It can be seen that IIR produces the narrowest delay distribution with the smallest fluctuations. DRLS still manages to perform better than AIR, while a few cases where DRLS fails still produce significantly longer delays, leading

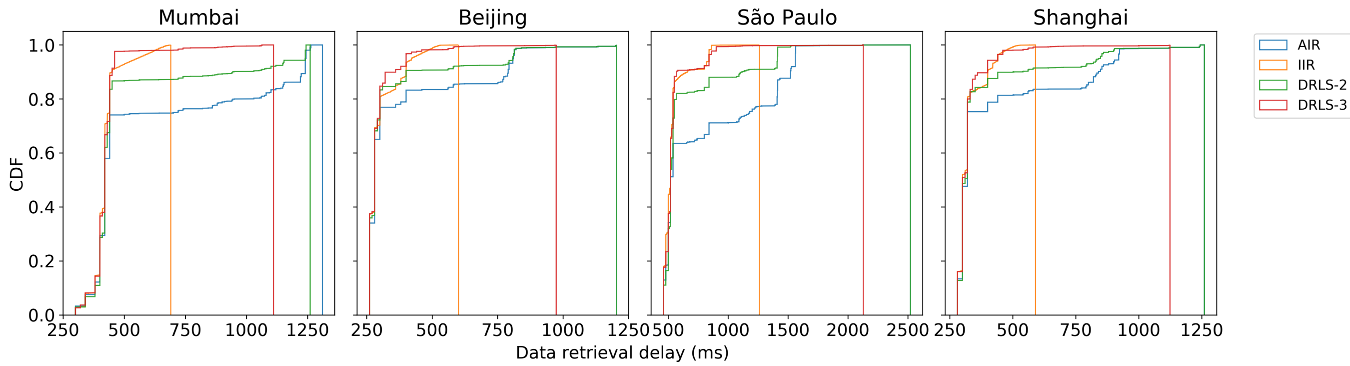


Figure 8: CDF for data retrieval delay for different producer locations (consumer sending speed = 300 Interests/second).

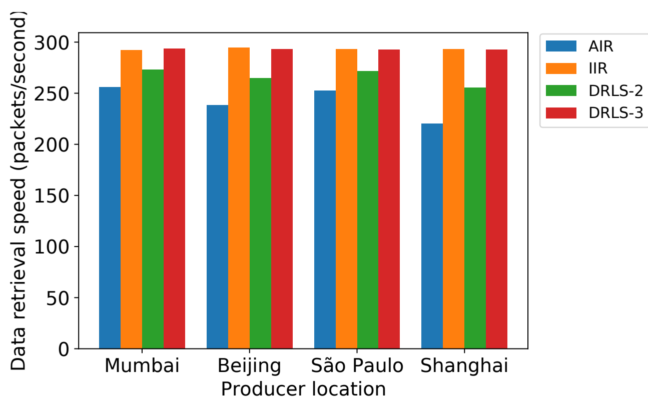


Figure 9: Data retrieval speed vs. producer location (consumer sending speed = 300 Interests/second).

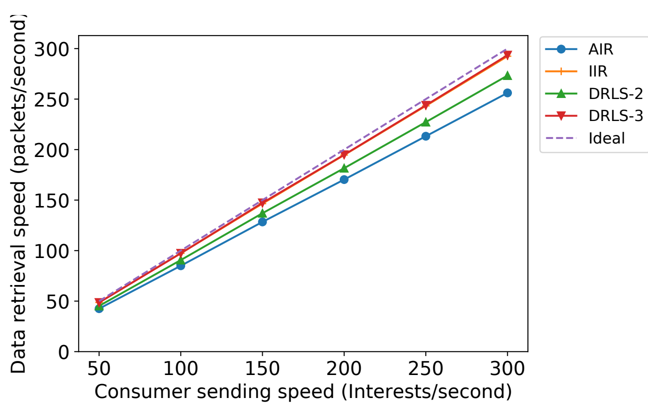


Figure 10: Data retrieval speed vs. consumer sending speed (producer at Beijing)

to higher jitters compared with IIR. But when the broadcast scope is increased to 3 hops, DRLS is almost identical to IIR, although there still exists a few cases where data recovery fails due to too many hops between the previous and current access satellite, in which case AIR is performed to retrieve data back with significantly longer delays.

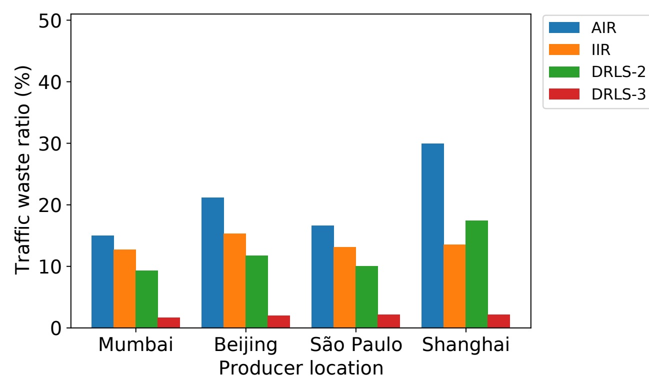


Figure 11: Goodput ratio vs. producer location (consumer sending speed = 300 Interests/second). Y-axis stops at 50%.

5.4.2 Data retrieval speed. Data retrieval speed is measured with the total number of successfully retrieved data averaged over the total amount of observed time, and reflects a mechanism's ability to mitigate the impact of handovers on application-level congestion control.

The performance of three evaluated mechanisms when consumer sending speed is 300 Interests/second is shown in Fig. 9. Across all producer locations, IIR and DRLS-3 perform significantly better than AIR, and only slightly worse than the ideal case where no handover occurs (where Data retrieval speed would be equal to consumer sending speed). The performance of DRLS is significantly improved when the broadcast scope setting is increased from 2 to 3 hops, but still outperforms AIR at all time. This is because IIR require no application retransmissions, while DRLS requires only a few (and fewer as the broadcast scope expands), and thus the consumer application can keep a steady data retrieval rate. It can also be observed that the performance of both AIR and DRLS (DRLS-3, and to a lesser degree) is noticeably affected by producer locations. This is because a longer RTT (e.g., between New York and Shanghai) means that more Interests would be affected by a handover, and would thus produce more retransmissions and further reduce data retrieval speed.

The relationship between consumer sending speed and data retrieval speed for the Beijing producer is shown in Fig. 10. It can be observed that the amount of retrieved data grows linearly with consumer sending speed, and each mechanism grows at a different rate, which leads to higher download rate differences at higher consumer sending speeds (e.g., DRLS-3 allows retrieving up to about 15% more data packets with consumer sending speed of 300 Interests/second compared to AIR). Because the results for each producer location show similar patterns, results for other locations are not presented here.

5.4.3 Traffic waste ratio. Traffic waste ratio reflects the amount of additional traffic generated due to consumer mobility. It is defined as the ratio between the volume of "wasted" traffic to the volume of overall data traffic in the network. "Wasted" traffic refers to the data traffic generated outside the path between the current locations of producer and consumer. Therefore, a lower traffic waste ratio means that less bandwidth is wasted for delivering data to the consumer. Results for this metric across different producer locations is given in Fig. 11, which shows the results under the highest consumer sending speed setting of 300 Interests/second. DRLS-3 has the lowest traffic waste ratio, with an average value below 9% and exhibit trivial deviations across different locations. DRLS-2 still out performs AIR and IIR in most cases with about 15% except for Shanghai. IIR averages about 18% with small fluctuations across locations. AIR performs the worst, with a value up to 32% for Shanghai. Such results clearly show that, pulling Data back from the previous access satellite can effectively reduce additional traffic under the considered satellite handover strategy. It can also be observed that even the smallest broadcast scope of 2 hops provides improvement, while a reasonably larger setting leads to major gains.

6 FINAL REMARKS

This paper identifies a challenge facing NDN's native consumer mobility support in a typical satellite networking environment. To address the wasted traffic issue caused by retransmissions, we take an architectural approach by extending the capability of NDN link adaptation layer operations. A new NDN link service is proposed under a general framework that extends the NDN face abstraction. The proposed mechanism is proven effective through simulations. For our immediate future work, in-depth analysis and experiments in more versatile and realistic settings will be performed. We also plan to work with the NDN community to further polish the proposed framework by discussing the proposed architectural changes, and considering other kinds of dynamic network environments such as vehicular networks.

ACKNOWLEDGMENTS

We would like to thank the invaluable comments from anonymous reviewers. This material is based upon work supported by the Peng Cheng Laboratory Funds (No. PCL2021A02). Any findings, discussions, and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsor.

REFERENCES

[1] Ahed Aboodi, Tat-Chee Wan, and Gian-Chand Sodhy. 2019. Survey on the Incorporation of NDN/CCN in IoT. *IEEE Access* 7 (2019), 71827–71858.

[2] AGI. 2021. The Cesium project homepage. https://cesium.com/

[3] Marica Amadeo, Claudia Campolo, Antonio Iera, and Antonella Molinaro. 2014. Named data networking for IoT: An architectural perspective. In *2014 European Conference on Networks and Communications (EuCNC)*. IEEE, 1–5.

[4] Emmanuel Baccelli, Christian Mehlis, Oliver Hahm, Thomas C Schmidt, and Matthias Wählisch. 2014. Information centric networking in the IoT: Experiments with NDN in the wild. In *Proceedings of the 1st ACM Conference on Information-Centric Networking*. ACM, 77–86.

[5] Jeff Burke, Alex Afanasyev, Tamer Refaei, and Lixia Zhang. 2018. Ndn impact on tactical application development. In *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*. IEEE, 640–646.

[6] Giovanna Carofiglio, Luca Muscariello, Michele Papalini, Natalya Rozhnova, and Xuan Zeng. 2016. Leveraging ICN in-network control for loss detection and recovery in wireless mobile networks. In *Proceedings of the 3rd ACM Conference on Information-Centric Networking*. ACM, 50–59.

[7] Jung-Hwan Cha, Ju-Ho Choi, Ji-Yong Kim, Youn-Hee Han, and Sung-Gi Min. 2018. A mobility link service for NDN consumer mobility. *Wireless Communications and Mobile Computing* 2018 (2018), 8 pages.

[8] Jiadi Chen, Wei Mi, Zongzhen Liu, Jingguo Ge, et al. 2020. Exploitation of Information Centric Networking in federated satellite: 5G network. *Wireless Networks* 26, 5 (2020), 3159–3167.

[9] NDN Community. 2021. Named Data Networking Forwarding Daemon (NFD) project homepage. https://named-data.net/doc/NFD/current/.

[10] NDN Community. 2021. The NDN platform. https://named-data.net/codebase/platform/.

[11] NDN Community. 2021. NDNLPv2 NDN wiki page. https://redmine.named-data.net/projects/nfd/wiki/NDNLPv2.

[12] Tasneem Darwish, Gunes Kurt, Halim Yanikomeroglu, Guillaume Lamontagne, and Michel Bellemare. 2021. Location Management in IP-based Future LEO Satellite Networks: A Review. *arXiv* 2021 (2021), 24 pages. arXiv:arXiv:2101.08336

[13] Barry Evans, Markus Werner, Erich Lutz, Michel Bousquet, Giovanni Emanuele Corazza, Gerard Maral, and Robert Rumeau. 2005. Integration of satellite and terrestrial systems in future multimedia communications. *IEEE Wireless Communications* 12, 5 (2005), 72–80.

[14] Thomas R Henderson and Randy H Katz. 2000. On distributed, geographic-based packet routing for LEO satellite networks. In *Globecom'00-IEEE. Global Telecommunications Conference. Conference Record (Cat. No. 00CH37137)*, Vol. 2. IEEE, 1119–1123.

[15] V. Jacobson et al. 2012. Custodian-based Information Sharing. *IEEE Communications Magazine* 50, 7 (2012), 38–43.

[16] Jian Li, Kaiping Xue, Jianqing Liu, Yongdong Zhang, and Yuguang Fang. 2019. An ICN/SDN-based network architecture and efficient content retrieval for future satellite-terrestrial integrated networks. *IEEE Network* 34, 1 (2019), 188–195.

[17] Teng Liang, Zhongda Xia, Guoming Tang, Yu Zhang, and Beichuan Zhang. 2021. NDN in Large LEO Satellite Constellations: A Case of Consumer Mobility Support. In *Proceedings of the 8th ACM Conference on Information-Centric Networking*. ACM, 11 pages. https://doi.org/10.1145/3460417.3482970

[18] Di Liu, Chuanhe Huang, Xi Chen, and Li Xin. 2017. Distributed Dynamic Routing Method for Multi-layered Satellite Networks Based on Named Data Networking. *Acta Electronica Sinica* 45, 11 (2017), 2771–2778.

[19] Spyridon Mastorakis, Alexander Afanasyev, and Lixia Zhang. 2017. On the evolution of ndnSIM: An open-source simulator for NDN experimentation. *ACM SIGCOMM Computer Communication Review* 47, 3 (2017), 19–33.

[20] David Naylor, Matthew K Mukerjee, Patrick Agyapong, Robert Grandl, Ruogu Kang, Michel Machado, Stephanie Brown, Cody Doucette, Hsu-Chun Hsiao, Dongsu Han, et al. 2014. XIA: architecting a more trustworthy and evolvable internet. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 50–57.

[21] Takeo Ogawara, Yoshihiro Kawahara, and Tohru Asami. 2013. Information dissemination performance of a disaster-tolerant NDN-based distributed application in disrupted cellular networks. In *IEEE P2P 2013 Proceedings*. IEEE, 1–5.

[22] OneWeb. 2021. The OneWeb project homepage. https://www.oneweb.world/

[23] SpaceX. 2021. The Starlink project homepage. https://www.starlink.com/

[24] Arun Venkataramani, James F Kurose, Dipankar Raychaudhuri, Kiran Nagaraja, Morley Mao, and Suman Banerjee. 2014. MobilityFirst: a mobility-centric and trustworthy internet architecture. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 74–80.

[25] Lixia Zhang, Alexander Afanasyev, Jeffrey Burke, Van Jacobson, KC Claffy, Patrick Crowley, Christos Papadopoulos, Lan Wang, and Beichuan Zhang. 2014. Named data networking. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 66–73.

[26] Yu Zhang, Alexander Afanasyev, Jeff Burke, and Lixia Zhang. 2016. A survey of mobility support in named data networking. In *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 83–88.

[27] Yu Zhang, Zhongda Xia, Spyridon Mastorakis, and Lixia Zhang. 2018. KITE: producer mobility support in named data networking. In *Proceedings of the 5th ACM Conference on Information-Centric Networking*. ACM, 125–136.