Article

OPMSS: Optimal Producer Mobility Support Solution for Named Data Networking

Muktar Hussaini [1], Muhammad Ali Naeem [2] and Byung-Seo Kim [3,*]

[1] Hussaini Adamu Federal Polytechnic, Kazaure 705101, Nigeria; intaiium@gmail.com
[2] School of Science, Guangdung University of Petrochemical Technology, Maoming 525000, China; malinaeem7@gmail.com
[3] Department of Software and Communications Engineering, Hongik University, Seoul 04066, Korea
[*] Correspondence: jsnbs@hongik.ac.kr

Abstract: Named data networking (NDN) is designed as a clean-slate Internet architecture to replace the current IP Internet architecture. The named data networking was proposed to offer vast advantages, especially with the advent of new content distributions in IoT, 5G and vehicular networking. However, the architecture is still facing challenges for managing content producer mobility. Despite the efforts of many researchers that curtailed the high handoff latency and signaling overhead, there are still some prominent challenges, such as non-optimal routing path, long delay for data delivery and unnecessary interest packet losses. This paper proposed a solution to minimize unnecessary interest packet losses, delay and provide data path optimization when the mobile producer relocates by using mobility update, broadcasting and best route strategies. The proposed solution is implemented, evaluated and benchmarked with an existing Kite solution. The performance analysis result revealed that our proposed Optimal Producer Mobility Support Solution (OPMSS) minimizes the number of unnecessary interest packets lost on average by 30%, and an average delay of 25% to 30%, with almost equal and acceptable signaling overhead costs. Furthermore, it provides a better data packet delivery route than the Kite solution.

Keywords: handoff latency; mobility support; named data networking; optimal data path; producer mobility support; signaling cost

1. Introduction

Based on Internet protocol (IP) architecture, the Internet has achieved exceptional success since its establishment, particularly with the advent of overlays such as peer-to-peer networking, web applications, wireless sensor networking, mobile networking, the Internet of Things (IoT), vehicular networking and content distribution networking. However, the nature of Internet use is rapidly changing from being host-based and emphasizing point-to-point connection to being content-centric and emphasizing the sharing of information. Moreover, the IP architecture is becoming complex due to integration overlays that result in ineffective content distribution, delays, insecurity, and high bandwidth consumption attributable to increased user demand.

Named data networking (NDN) is the prominent architecture under the umbrella of information-centric networking (ICN), which is designed as a clean-slate Internet architecture to replace the current IP Internet architecture [1]. The named data networking was proposed to offer vast advantages, especially with the advent of new content distributions in IoT, 5G and vehicular networking. Fundamentally, NDN supports content consumer mobility due to the consumer-driven design principles and nature of in-network catching capabilities. Named data networking (NDN) architecture is recognized as a potential content-centric paradigm for wired and wireless domains. Its novel concepts, such as security, in-network caching, hierarchical naming or namespace, named content, and name-based routing, suit the requirements of IoT [2]. Moreover, the architecture has the potential

for efficient data dissemination and interconnecting billions of heterogeneous objects in an IoT network. Despite the benefits of NDN in an IoT network, the mobility perspective must be studied, and a concrete solution to support mobile IoT must be found [2]. If mobile nodes are no longer reachable in IoT, the problem will worsen, as the mobile frequencies in intelligent transportation and eHealth applications are very important [2]. In addition, the services that are based on NDN with blockchain technology can be applied in IoT for secured data management.

Decoupling senders and receivers, in-network caching, and the hop-by-hop transmission characteristics of NDN enable it to support 5G networks [3]. The result of the potentiality is shown by information-centric networking (ICN), such as security, mobility, energy efficiency, and in-network caching, which are considered important in the 5G telecommunication network [4]. In addition, ICN is promoted as a framework for virtual function provisioning and named content retrieval in 5G [3]. The encouraging results of ICN have motivated certain technical specification organizations to promote ICN as an enabler of 5G, which evolved from 4G core network architecture focus on radio access networks, as an overlay on network function virtualization, and software-defined networking infrastructure [5]. However, in adapting 4G, the drawbacks of IP architecture, which include tunneling technology to support mobility, high signaling cost, security challenges, and the lack of multihoming support, were inherited. Therefore, the inherent benefit of ICN with the support of name-based networking, security, in-network storage, in-network caching, and mobility is the ability to solve the inherited problems of the 5G network. In addition, the mobility management involved in network slicing-based 5G is challenging due to the high density and high mobility of devices [6]. Consequently, the mobility support and in-network caching characteristics of NDN emerged in the architecture as promising candidates to integrate into 5G [7].

NDN supports consumer mobility without additional mechanisms by using certain characteristics, such as its consumer-driven nature and in-network caching [8–10]. NDN design principles inherently support content consumer mobility; for example, when the physical location of a consumer changes, there is no effect on the data plane, unlike with IP [7]. The retransmission of requests can take place without additional signaling to the network. However, challenges arise in relation to real-time group communication and content producer mobility, depending on the content lifetime, latency requirements, and frequency of the producer's movements [7]. Zhu, Afanasyev, and Zhang [11], the creators of NDN, note that NDN does not support the mobile producer. In addition, they recommended that to provide a solution to producer mobility support, the separation of name prefix identifier and locator from the single hierarchical name prefix in NDN is necessary. Moreover, the prefix identifier and locator can be mapped using the broadcasting method or DNS servers [11].

The problem of producer mobility can be lessened via caching and providing a mobile producer handoff solution. In [12], the author emphasizes the natural support of mobile consumers and states that alternative cached copies can be used by consumers after relocation. The authors report that consumer mobility is intrinsically supported. However, there are many challenges with producer mobility that started with the initial design of NDN, such as long handoff latency, high handoff signaling costs, table size scalability problems, unnecessary interest packet loss, and the high cost of bandwidth utilization. Therefore, producer mobility is not supported in NDN, as the initial proposal of NDN claimed the missing of producer mobility support and passive consumers [13], hence, a sustainable solution is required to make NDN a promising architecture that can supplement the future Internet with the integration of 5G [14], IoT [15], and other network overlays that require mobility support. By solving the mobility problem inherited from IP architecture, NDN can successfully become future Internet architecture [16]. Hence, the researchers are motivated to conduct the highest research records in NDN mobility support between 2017 and 2018, due to the anticipation of NDN capability to incorporate other networks without additional mechanisms. Cisco proposed hybrid ICN (CCN and NDN) in a 5G network [17], which

encouraged many researchers to investigate integrating ICN into 5G networks [5,18], NDN-5G-SDN support [18,19], ICN with IoT networking [20–22], and NDN-based vehicular ad hoc networking [23].

Therefore, much research was conducted from 2012 to present, to find the solution to producer mobility using, for example, the location-based approach, the mapping-based approach [24], the tracing-based (TB) approach [25], the indirection-based (IB) approach [26], and the anchor-based (AB) approach [27]. The mapping approach utilizes a mapping point, such as DNS server [28], a rendezvous server (RS), or a resolution handler server [24], to update the network about the new name prefix of the mobile producer to provide optimal data packet delivery after the handoff. However, the approach has a high signaling overhead cost due to the server query and update, which may result in long handoff latency. The IB approach uses an indirection point of the home router, a rendezvous point, or an anchor node [29] to store the new location and traced information for the request forwarding to the mobile producer. The approach provides normal handoff signaling and latency, but triggers a path stretching problem due to triangular routing. In addition, the representation of indirection points can become a single point of failure.

Moreover, this paper aims to provide a solution to producer mobility in NDN, reduce the handoff signaling and latency, provide an optimal path by addressing the path stretching problem, addressed the interest packets loss, and avoid the single point of failure from the indirection-based approach. In fact, is proposed as the improvement of our previous work [30] that introduced the concept of using a broadcasting strategy to provide an optimal route after handoff, which was analyzed using an initial and reference model as evident limitation. In addition, analytical mobility model was used to validate the conceptual model presented in the previous work, and the result is limited to numerical analysis on handoff latency, signaling cost and data delivery. Further, among the benchmark schemes, there was a Kite scheme that was limited to analytical evaluation method, whereby the result obtained cannot be generalized [30]. However, this paper proposed a novelty for the design and implementation of an updated broadcasting strategy, which includes the design of mobility interest (MI) packets to update mobile producers' new information. We further used the broadcast method to design a broadcasting strategy to update the forwarding information-based (FIB) tables of the intermediate routers. Similarly, a best-route strategy was used after FIB updates to guarantee data path optimization. Moreover, the restricted domain router (RDR) was introduced to curtail the problem of broadcasting storm during intra and inter domain mobility. Correspondingly, the proposed OPMSS is designed and implemented using ndnSIM simulator, that have the standard NDN stack as a new network-layer protocol model. The result is not limited to only handoff signaling, latency and data delivery, but throughput, data packets and interest packets losses are considered.

The paper is organized into seven sections, apart from the introductory part, Section 2 consists of a review of the existing approach. In addition, we highlight the research gap that this paper addresses. Section 3 discusses the design processes of the proposed optimal producer mobility support solution. Section 4 presents the formulation analysis of the proposed solution. Section 5 presents the details of the implementation of OPMSS in ndnSIM, performance evaluation of both the proposed and the existing producer mobility solutions. Section 6 consists of the overall discussion, followed by the conclusion.

2. Related Works

To achieve the routing and forwarding of packets, NDN uses named-base routing and possesses of two different types of packets, namely interest and data packets. Additionally, the NDN node can be represented as client consumer, producer or a router, which maintains three aspects—data structure pending interest table (PIT), forwarding information base (FIB) and content store (CS)—and determines when and where to forward data and interest [1,31]. For the purpose of PIT records and storing any incoming interest information, FIB maintained the forwarding strategy and decided when and where to forward interest and CS; this is a temporary cache of data stored based on NDN caching policy. The network

was set up and the client node (consumer) was established as a connection in the network. When the consumer intends to retrieve data from the network, it will send an interest to the NDN router. On arrival of the interest, the router checks the CS for matching data; if found, it forwards the data packets back to the consumer. Otherwise, the router looks up the name content in its PIT for matching entries—if the entry is not found, it will record the name content and incoming interface and forward the interest to the next hop through FIB, otherwise it will aggregate and record the interface only. The same process takes place up towards the content producer. Interest can be dropped on certain circumstances depending on the forwarding strategy, e.g., the congestion of upstream links or security breaches by suspecting interest to be part of DoS. Once the data producer receives an interest request from the home router and its PIT, and finds the required content, it will forward the data packets back through the interface that received the interest to the downstream interface recorded in PIT [32,33].

2.1. Producer Mobility Support Approach

NDN was introduced as promising Internet architecture to replace IP Internet, particularly from the perspective of mobility support. Many researchers adopted the mobile IP concept called the indirection-based mobility approach (IBMA) to solve the mobility problem in ICN, particularly NDN. The indirection-based concept is explored more, as it is inherited from the IP mobility solution concept. In addition, the concept does not attract additional network infrastructure, which is one of the benefits of NDN architecture. Our study tackles the problems related to the IB approach's path stretching or data packet delivery optimization problem. Moreover, many areas that are booming, such as Internet-of-Things (IoT), vehicular Internet-of-Things (VIoT) and vehicular named data networking (VNDN), are employing NDN architecture for a proper solution related to mobility and data dissemination.

Due to the relevance and important features of NDN for mobility support of the field Internet-of-Things [34], vehicular Internet-of-Things (VIoT) [35] and Internet-of-Things Cloud (IoTC) [15], it employs the NDN architecture for the solution of data retrieval and node mobility. Wang [15] proved that deploying NDN in IoT is quite challenging, due to the different architectures and working mechanisms and the effective mobility support strategies in NDN being lacking. The author [15] proposed an efficient framework for NDN-based IoTC, aiming to provide effective and improve data retrieval success, by employing a unicast method to obtain data from the nearest node and support node as a consumer or producer on mobile. However, the proposed framework is not solely for NDN mobile producers that have clear differences with the consumer. This means that the name prefix announcement is not considered as a unique feature of the producer. Moreover, the solution did not evaluate the handoff latency and signaling performance. Vehicular named data networking (VNDN) is a new paradigm proposed to facilitate communication between vehicles [36]. NDN proved to be a promising future Internet under the ICN paradigm which supports any vehicular networking [36,37]. Al-qutwani in [37] stressed some challenges, such as node mobility related to packet forwarding, which still need to be addressed. The broadcast strategy is used for the proposal of request/advertise-based content forwarding approach to solve the issue of node mobility and forwarding performance [37]. The content consumer broadcasts a message named request and content provider advertise contents in the same way as the consumer, by using an advertisement message. The solution gives the provider or consumer the opportunity to find desired data packets with minimal delay and overhead. In addition, the proposed solution can help a mobile producer to change location and use an advertisement message to announce its current position [37]. However, the solution may cause a very high signaling overhead when the number of mobile producers increases; this is due to the usage of request and advertisement messages in line with data and interest packets.

Recently, a number of producer mobility schemes were proposed as in [34]. The solution is an NDN producer mobility scheme that requires no use of anchor; there is no

need to announce the producer's movement to the nodes over NDN architecture and the scheme does not modify the current NDN router. It exchanges a pair of data and interest packets during a normal transaction. The proposed scheme support producer mobility introduces a dual connectivity to both hide the link breakage to the NDN, and a new NDN link service is created separately which is called the producer mobility link service (PMLS), which is used to repair the old link so that the connectivity with the previous NAR can be maintained and force the router to buffer the interests arriving during the movement transition period, forwarding them to the producer after the completion of the handoff [34]. However, in the process of buffering the interest packets, latency and signaling will be high before the old link is repaired due to the exchange of update and status messages other than normal interest packages. Moreover, the buffer zone requires a high amount of storage, especially in the bigger, and denser, mobile producers' environment. Furthermore, a mechanism using the spatial locality of moving producers was designed to support producer mobility. The solution maintained the reverse paths of data similar to the original ICN of NDN architecture [38]. The mechanism is modeled using hop count forwarding strategy that considered a weighted sum of the round trip time (RTT) as the average service time between the consumer and every intermediate node [38]. However, the sole use of RTT cannot determine if there is congestion among the intermediate nodes which allow the consumer to keep on sending the unsatisfied interest packets. Therefore, for ICN to support multi-access edge computing for 5G technology of anywhere–anytime connected services for mass mobile connectivity, the movement pattern of mobile content producers needs to be investigated [14].

2.2. Anchor-Based and Tracing-Based Methods

The AB and TB methods are typical examples of IBMA solutions. In the AB method, the immobile anchor router or resolution server (RS) is placed in the network when the mobile producer moves to a new location and obtains a new prefix name. The new prefix is then sent to the anchor router or the RS. Next, the consumer sends a pending interest packet to the known location of the anchor router, and the anchor router tunnels it to the new location of the mobile producer. Unlike in the TB method, the anchor router or RS utilizes the traces of the mobile producer stored in pending interest tables (PITs) or FIB tables. When the interest packet arrives, the anchor router or RS forwards the interest along the traces without tunneling.

Zhang et al. [25] proposed a TB solution called Kite that utilizes the PITs and FIBs to store mobility traces and reach the new location of the mobile producer through routable anchors [13]. This scheme is also called an AB approach [27] or a PIT-based approach [39], where a mobile producer sends traced interest or trace interest [25] packets to the immobile anchor or the RS and establishes a trace route in the PITs of the intermediate routers, and sends the new location of the mobile producer to the anchor router or RS. The traced or trace interest packets, as presented in a report about the Kite mobility solution, can be sent to the immobile server or the data deport. The corresponding node, consumer, immobile server, or data deport requesting content sends tracing interest packets as in [13] or traced data as in [25] back to the anchor router or RS. In the latest Kite report, the RS announces a routing prefix and the mobile producer issues a trace interest packet with a special tag to the RS. The RS verifies this and responds with trace data to update the intermediate routers as trace setup processes [25]. The consumer sends an interest packet to the RS and then forwards the trace to the mobile producer. These processes are similar to those in a previous Kite solution, except for the change of tracing and traced interest packets that are replaced with trace interest (*TI*) packets and trace data (*TD*) in addition to the consumer's interest packet. However, the Kite solution results in high signaling to keep the traces active, along with the intermediate router in PITs, by sending *TI* packets and *TD* in a timely fashion. Moreover, the solution results in path stretching, where the optimal routing path is not guaranteed because the consumer's interest packets pass through the RS or anchor point.

An attempt was made by Kim and Ko [27] and Do and Kim [39] to improve the Kite solution by providing an optimal routing path and reducing the signaling overhead cost. Do and Kim [39] proposed a scheme for a large-scale NDN to solve the path stretching and routing problem of the Kite solution and to reduce the signaling overhead cost. The author [39] introduced a fast handoff mechanism that directs the subsequent interest packets to the new location of the mobile producer and creates an optimal data path. A special mobility management packet is used to make the update before the handoff occurs. The scheme uses a forwarding plane of NDN to establish an optimal data delivery path and predict producer movements. However, the solution is not guaranteed for large deployment, as claimed by the author. In addition, due to the processes of fast handoff, the signaling overhead cost becomes higher, and the addition of a binding cache and neighbor information tables increases the look-up time.

Kim and Ko [27] proposed an on-demand anchor-based solution to producer mobility that uses a mechanism called adaptive interest packets forwarding. When the producer moves to a new location, the FIB entries become invalid, and the interest packets looking for the content from the mobile producer are redirected towards the anchor node from the previously known location of the producer, instead of being dropped. The mobile producer keeps on updating the anchor node with mobility update packets whenever there is movement. When the interest packet has been redirected to the anchor node, it uses the traces in the mobility update and forwards the pending interest packet. The solution aimed to reduce handoff delay and signaling overhead costs caused by the blind flooding of *TI* and *TD* in the Kite solution [27]. However, the solution ends with a path stretching longer than that of the Kite solution, which provides perfect triangular routing. Therefore, path optimization is not considered, and the anchor's placement becomes critical [7] and becomes a single point of failure.

In the Kite solution [25], we can assume that a mobile producer is disconnected from the current router and directly reconnected to the new router. The mobile producer sends a *TI* message with a special trace tag to the known location of the RS or the anchor router. The RS verifies and responds with a *TD* message along the path that *TI* is received, and the trace created along with the intermediate routers by updating the FIB tables between new router and the RS. When a consumer sends an interest packet to the RS requesting new or unsatisfied content, the RS forwards it to the new location of the mobile producer (new router) via the trace. The process of trace set-up continues whenever the producer moves to a new location. The Kite solution allows the mobile producer to keep the trace alive by exchanging *TI* and *TD*.

In the proposed OPMSS, when the mobile producer relocates to a new location, it configures and sends an *MI* packet for a new name prefix update. The mobility tag is attached to the name prefix and is broadcasted to the FIB table's update domain. Upon receipt by the restricted domain control router, the router will determine the domains that need to rebroadcast the *MI* packet based on the old prefix information. The routing plane then creates an optimal path between the consumer and the mobile producer.

2.3. Broadcasting Method

The broadcasting of interest packets is a practical aspect of NDN; therefore, much of the research has used the broadcasting method in different sub-fields of networking to enhance the performance of NDN. Vehicle-to-NDN [40], vehicle-to-vehicle NDN [41,42], and wireless NDN [43] utilize the broadcasting method for data and interest communication to support moving contents, although some additional methods, techniques, or mechanisms have been added to mitigate broadcasting storms. Siris in [44] proposed a popularity-aware intra-domain content mobility management solution that uses broadcasting and a name resolution service. The model utilizes a name resolution system (NRS) to update the network about the new location of the mobile producer when there is high popularity content and low mobility, while broadcasting of the location queries is used when the content popularity is low and there is high mobility [44]. The solution provides

mobility support and significantly reduces signaling overhead costs. Asif and Kim [45] proposed a location-aware on-demand multipath protocol that uses a broadcast transmission mechanism to transmit data and interest packets. The packets are transmitted via multiple routes to the new location of the mobile producer. In addition, a relay node is used to control high signaling and to prevent intermittent connectivity and packet flooding that negatively affects network performance.

Shi, Newberry, and Zhang [46] showed how broadcast-based self-learning is applied to solve the trust and name-prefix granularity problem in NDN networks. The broadcast-based self-learning mechanism is used to find the delivery path of packets when content is moving within the network [46]. Therefore, in this paper, a broadcasting strategy is designed to utilize *MI* packets to update intermediate routers when the producer relocates to a new point of attachment (PoA). The *MI* packets are similar in function to the traced and tracing interest in [25] the binding update or forwarding hint in [47], and the mobility management packet in [39]. In addition, a network domain is restricted as in [11] to control broadcasting storms and to prevent high signaling.

2.4. Overview of Kite Scheme Operation

Kite is a popular NDN mobility support proposed by Zhang et al. as part of the architectural development of the NDN project. In this paper, we categorized Kite as tracing-based mobility support approach reviewed and briefly explained in Section 2.2. To gain a clear understanding of the Kite operation, the following steps describe the process before, during, and after handoff, as shown in Figure 1. Moreover, the name resolution for tracing of mobile producer is normally take place by the anchor router or RS in Kite scenario, however, in Figure 1, the anchor router is used.

Steps before the handoff, when the consumer is connected to CR1 and the producer is connected to CR4

- Step 1: Normally, a consumer sends an interest packet requesting data to CR1. CR1 then checks if the content is available; otherwise, it floods the NDN network with the interest packet.
- Step 2: The prefix data name within the interest packet passes through routers to the location of the producer. If the data are cached by any router in the network, the router will immediately respond with the data back to the consumer. Otherwise, the routers along the way will record the interest information in the PIT and the FIB table as entries and forward it until the content producer is reached. The producer then sends the data to the consumer in a breadcrumb style.

Steps when handoff starts

- Step 3: The producer suddenly decides to move and connects to CR4.
- Step 4: A new content name prefix is formed after the connection, and the producer gets ready to update the anchor router (i.e., CR1 from Figure 1) or RS about the new name prefix.
- Step 5: The producer floods the network with the *TI* packet via the anchor router or the RS to update the system about the new name prefix.
- Step 6: The anchor router or the RS respond with trace data packets and establish a trace between the mobile producer and the anchor router or the RS through CR2, CR3, and CR4.
- Step 7: The content consumer could not trace the new location of the producer after sending the pending interest packet.
- Step 8: The consumer will send the pending interest packet to the well-known location of the anchor router or the RS.

Steps after handoff

- Step 9: The anchor router or the RS forward the consumer's interest packet via the established trace to the mobile producer.

- Step 10: The mobile producer replies to the consumer with the requested data packets via the anchor router or the RS.
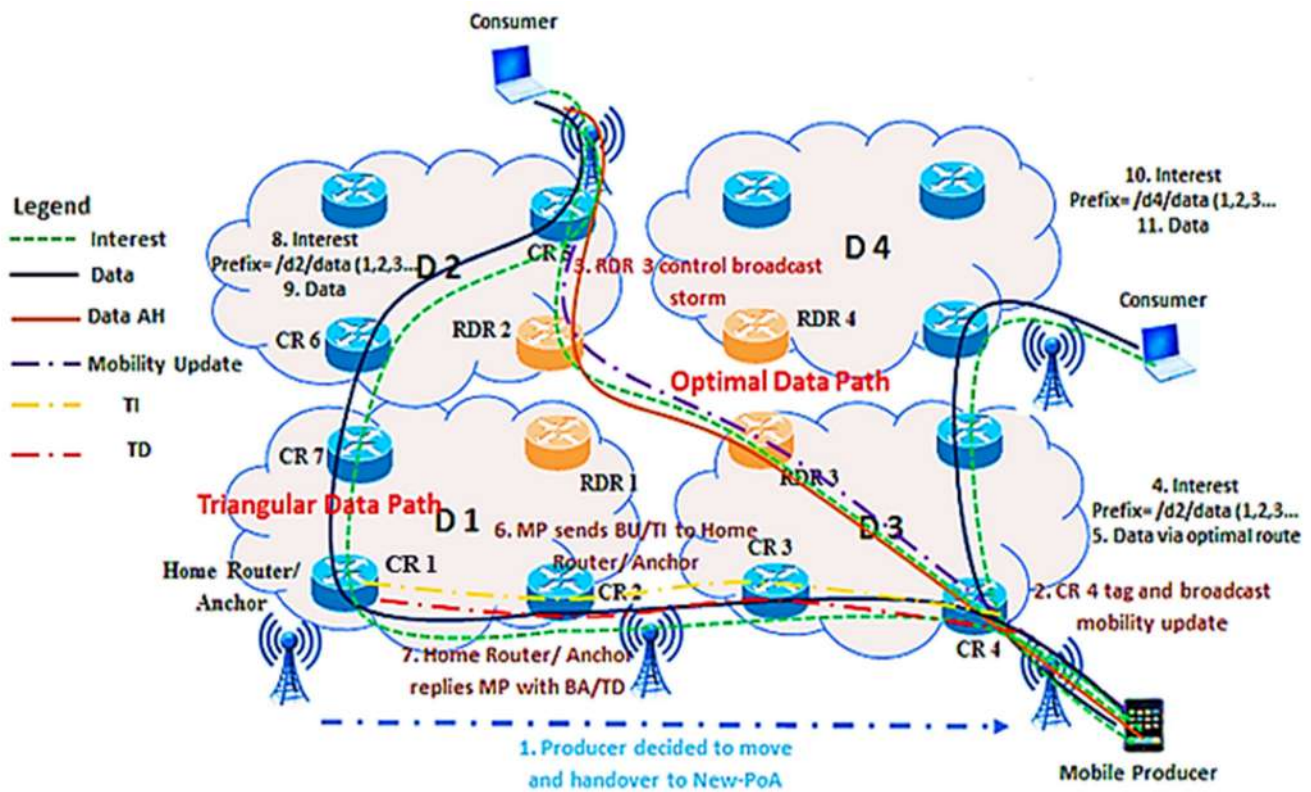

Figure 1. Overview of Kite Operation.

Despite the advantages of the Kite scheme that outperforms the mapping-based approach in terms of handoff delay with similar signaling overhead when a mobile producer moves frequently, the optimal data path is not guaranteed in such a way that consumer's interest packets always pass through the immobile anchor or RS. Hence, the Kite scheme falls into the category of an indirection-based approach that can make the forwarding path longer than the path computed in the normal NDN routing plane. This is called the path stretching problem, as the provision of the optimal path was not considered, consequently, making the handoff latency higher. Additionally, there is no improvement in signaling cost minimization compared to the mapping-based scheme for both the pull and upload pattern of Kite solution.

The proposed OPMSS have more advantages over the existing Kite solution, in such a way that *MI* packets alone can easily represent the functionality of *TI* and *TD* packets of Kite, hence the high signaling caused by the exchange of *TI* and *TD* can be minimized. Furthermore, the broadcasting strategy proposed in OPMSS provides a solution to the problem of the single point of failure caused by immobile anchor or RS in Kite solution. When *MI* packet is broadcasted, there is no need for an RS or an immobile anchor router in the processes of data and interest exchange. Conclusively, the *MI* packet and broadcasting strategy solve the path stretching the problem of Kite; once the FIBs of the intermediate routers are updated with the new routing name prefix, the interest can find the shortest route the new location of mobile producer without passing through the RS or immobile anchor router.

3. Design of Proposed Optimal Producer Mobility Support Solution

The OPMSS works immediately when the mobile producer relocates from the old PoA to the new PoA. The routing prefix name that is already stored in the FIB tables of the

intermediate routers becomes outdated due to the location change of the mobile producer. Therefore, there is a need for a mechanism to update the FIB tables about the new prefix of the mobile producer. The OPMSS consists of four components—mobility interest packet design, broadcast strategy design, restricted domain router configuration, and a best-route strategy to address producer mobility. Together, the four OPMSS components can solve the known handoff performance problems and provide an optimal data path after handoff.

3.1. Mobility Interest Packet

The *MI* packet is designed as an extension of the producer's application. The packet comprises three fields—current prefix name, new prefix name, and mobility status tag, as shown in Figure 2. The namespace of the prefix name contains the hierarchy of the producer name, the access point name, the router, and the domain name. For example, let us assume we have a producer *p1* attached to access point *ap1* and content router (CR) *cr1* in domain *d1*; the name prefix will be *p1/ap1/cr1/d1/*. For content called movie, the prefix is *p1/ap1/cr1/d1/movie*.

Figure 2. Mobility interest packet.

Figure 2 shows a representation of our newly designed *MI* packets that have the current prefix . . . */d1/data(1,2,3..n)* in domain 1 (D1) and the new prefix . . . */d2/data(1,2,3..n)* when the mobile producer moves to the new domain called domain 2 (D2). The mobility flags representing the normal prefix when the producer is at rest, move within the same domain, and moving to another domain are 00, 01, and 11, respectively. The mobility flag is a data bits used to indicate the movement status of the mobile producer when it is at rest or moving. The *MI* packets are tagged with a mobility flag to make the intermediate router aware of the movement and decide whether to make the update or ignore it. The update is ignored when the mobility status tag is 00.

3.2. Broadcast Strategy

The mobility interest packets are designed to carry the producer's new location information and update the intermediate routers. Hence, the broadcasting strategy is designed in such a way that, when the content producer moves to the new location, the new CR that accommodates a connected mobile producer will configure the new prefix and set the mobility flag, tag the *MI* packets, and broadcast to the available interface connections within the domain. Once *MI* packets are received by the next router, the look-up is performed to check if there are any prefix records before handoff that need to be updated with the new prefix after handoff. Otherwise, the router will record the new prefix in the FIB table and forward it to available interface connections. Figure 3 shows the flowchart of how the broadcasting strategy utilizes *MI* packets to update the intermediate routers. The aim of *MI* packet broadcasting is to update the entire network about the mobility information or a new data name prefix of the mobile producer. These can be described in steps for a proper understanding of the operational processes of the proposed OPMSS. The following steps are described in three phases—the process before, during, and after the handoff.
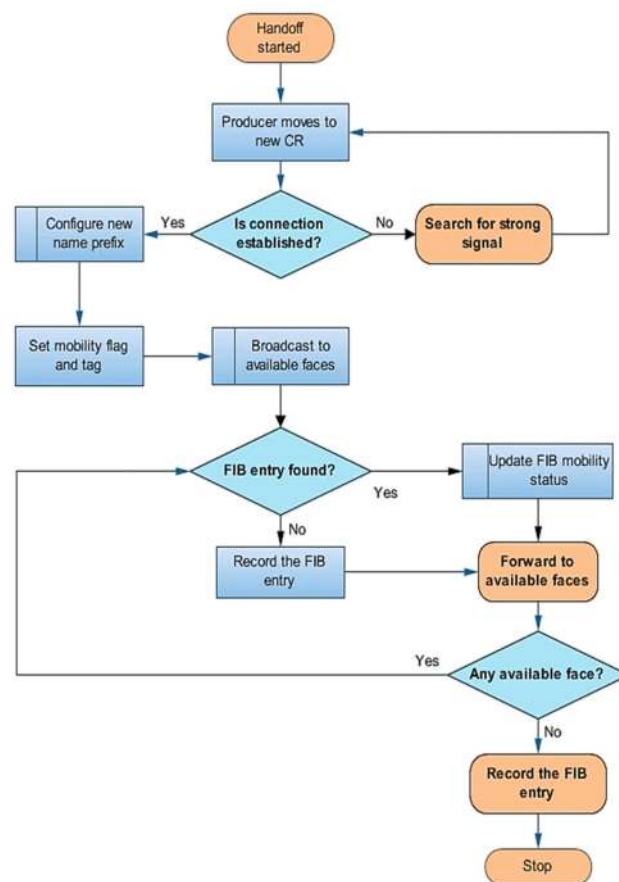
Figure 3. Mobility update by broadcasting strategy.

Steps before the handoff, when the consumer is connected to CR5 and the producer is connected to CR1

- Step 1: Normally, a consumer sends an interest packet requesting data to CR5. CR5 then checks if the content is available; otherwise, it floods the NDN network with the interest packet.
- Step 2: The interest packet passes through routers until the content producer is reached. The producer then sends the requested data to the consumer in a breadcrumb style. The steps before handoff are normal behavior in NDN and are the same as in the Kite solution.

Steps when handoff starts

- Step 3: The producer suddenly decides to disconnect from CR1 and search for strong signal and connect to CR2 in another domain.
- Step 4: The new content name prefix and the *MI* packet are configured after connection with CR2 for an update.
- Step 5: The CR2 broadcasts the *MI* packets within the domain to update all FIB tables, including the FIB table of the restricted domain router (RDR).
- Step 6: The RDR placed in each domain to control the broadcasting storm will check the old and new prefix information. If the prefix hierarchy contains a neighbor domain, the RDR will re-broadcast the *MI* packet to the neighboring domain.
- Step 7: The content consumer re-sends the pending interest packet to the network as a normal transmission.

Steps after handoff

- Step 8: The routing plane of NDN uses the best route strategy to establish the optimal route and forward the pending interest packet to the new location of the mobile producer.

- Step 9: The mobile producer replies to the consumer with the requested data packets via the optimal path. Steps 3–9 will be repeated when the producer moves to the next access point.

3.3. Interest and Mobility Interest Packet Transmission

Normally in NDN, if a content consumer sends an interest packet requesting data, when the Interest packet reaches a content producer, the data is sent back immediately, by using NDN routing and forwarding plan E [48]. The interest packet can be dropped if the data and producer are not found due to the change of location. A new routing prefix is needed to locate the mobile producer as a result of the location change due to the movement. In the OPMSS, when the producer moves, the procedure called process handoff will take place by updating the intermediate routers about the new routing prefix by using the proposed *MI* packets and broadcasting strategy, as explained in the previous section. Once the new routing prefix is updated among the intermediate routers, the NDN routing plane uses the best route strategy to create an optimal path for the data packet to reach the consumer from the new PoA of the mobile producer, once the handoff is completed. A pending Interest packet can then be forwarded by the consumer via the optimal route to the new location of the mobile producer. The interest packet and data exchange processes take place with the help of three data structure mechanisms—the content store (CS), the PIT, and the FIB table. The PIT is used to record the name prefixes of all interest packets and incoming faces, the CS table records the name prefixes and corresponding data, and the FIB table records the name prefixes and outgoing faces. The processes of interest packets transmission is illustrated in the flowchart of Figure 4.
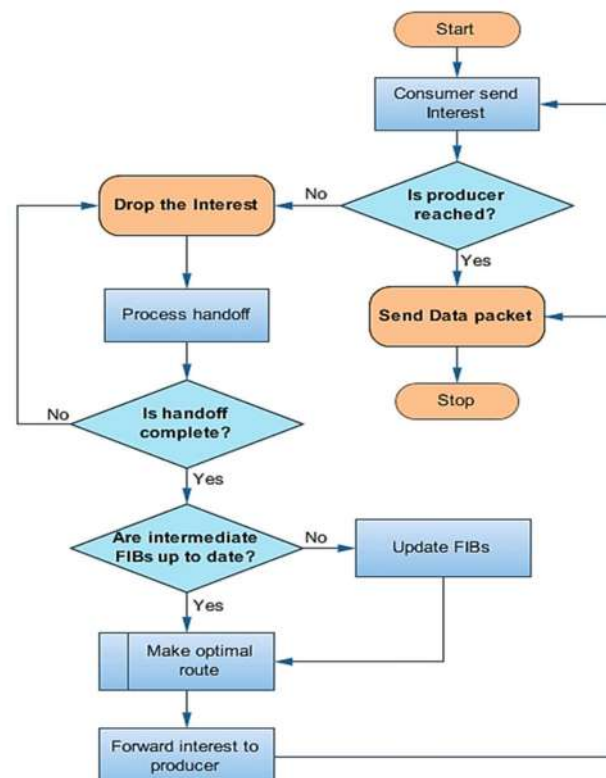

Figure 4. Interest packet transmission in OPMSS.

4. Formulation Analysis of OPMSS

The formulations of handoff performance are derived from the network setup in Figure 5 for both the proposed OPMSS and the benchmarked Kite solution. The network analysis model shown in Figure 5 contains links between the intermediate routers, the consumer, and the producer. The wireless link or hop between the consumer or producer

and the intermediate router is labeled with a, and the wired links or hops between routers are labeled with *b*. For every link between two nodes, there is a compounded link delay *Ldw* and *Ldwl* associated with wired or wireless, a queuing delay *Qd*, and required bandwidths *Bw* and *Bwl* for certain transmissions of data S_{int/TI_TD}, where *S* represents a packet for data, interest, update, *MI*, *TI* or *TD*, etc. The total delay for wired or wireless links between two nodes can be represented in Equations (1) and (2) with respect to the probability of link failure *q*.

$$Lwl_{int/TI\_TD} = \left(\frac{S_{int/TI\_TD}}{(Bw)} + (Ldw + Qd) \right) \tag{1}$$

$$Lwl_{int/TI\_TD} = \left(\frac{1+q}{1-q} \right) \times \left(\frac{S_{int/TI\_TD}}{Bwl} + Ldwl \right) \tag{2}$$

Therefore, the transmission latency between the consumer, the producer, and the CRs can be expressed as:

$$TLat_{wl} = a \times Lwl_{int/TI\_TD}$$

$$TLat_w = b \times Lw_{int\_MI/TI\_TD.}$$

Looking at Figure 5, three scenarios for the analysis of numerical handoff latency, handoff signaling cost, and data packet delivery cost can be formulated. The formulation for the three scenarios is covered in Equations (3)–(14) for both the Kite solution and our proposed solution. From Figure 5, the label *a* represents a wireless link between the producer or consumer and the router, while the *b* represents the link between routers. The D1 and D2 represent the two different domains, and CR is a content router and RDR is the restricted domain router, which can serve as a content router at the same time.
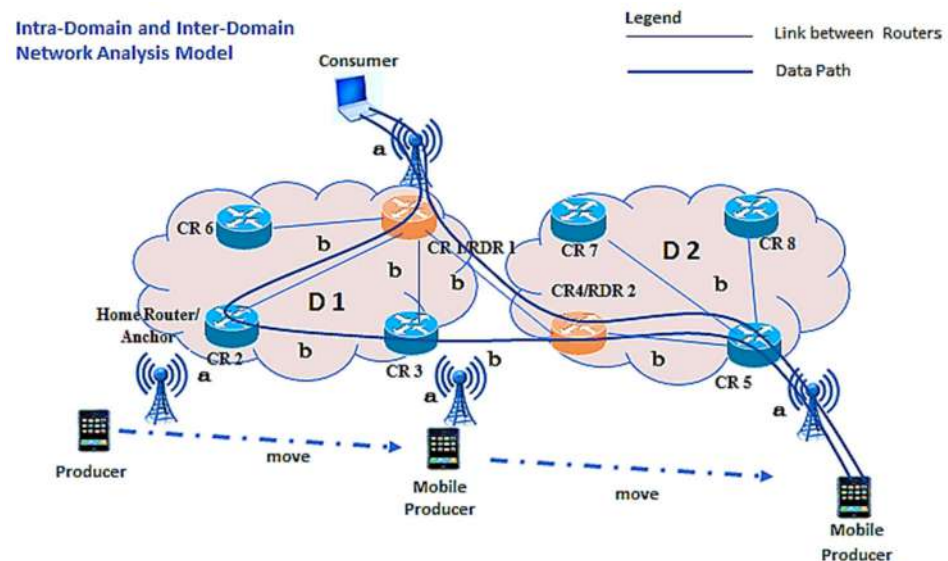

Figure 5. Network analysis model.

4.1. Scenario I: Handoff Latency

The total handoff latency of the Kite solution can be generated when the mobile producer disconnects from CR2 and moves to CR3 within D1. Before the resumption of communication, the consumer must wait for the update signaling to take place after the disconnection and reconnection. The mobile producer exchanges the *TI* and *TD* with the anchor router or the RS via CR2 to CR1. Then, the consumer's interest packet can be redirected from CR1 to CR3 to the mobile producer. The transmission latency between the consumer, the producer, and router is $3 \times a$ hops, and one between routers is $3 \times b$ hops for sending the *TI*, *TD*, and consumer's interest before the resumption of communication after the handoff, which can be formulated in Equation (3). In addition, if the mobile producer moves to D2, the total latency covers additional $4 \times a$ and $6 \times b$ hops, as presented in

Equation (5). Hence, the numerical formulation of handoff latency for the intra-domain of the Kite solution and OPMSS can be presented in Equations (3) and (4), respectively.

$$HLat_{Kite} = ((\ TLat_{wl} + TLat_w) + (2TLat_{wl} + 2\ TLat_w)) \tag{3}$$

$$HLat_{opmss} = ((\ TLat_{wl}) + (2TLat_{wl} + TLat_w)) \tag{4}$$

The total handoff latency of the proposed OPMSS solution can be generated when the mobile producer disconnects from CR2 and moves to CR3 within D1, the same as in the Kite solution. CR3 broadcasts the *MI* packet within the domain to locate the mobile producer. The total handoff latency between the consumer, the producer, and router is $3 \times a$, and one between routers is b hops, for sending *MI* packet and new interest to the new location of mobile producer; these can be formulated in Equation (4). In addition, if the mobile producer moves to D2, the total latency covers additional $3 \times a$ and $3 \times b$ hops due to *MI* broadcasting for both D1 and D2, as presented in Equation (6). Equations (5) and (6) present the numerical formulation of handoff latency for inter-domain as:

$$\begin{aligned} HLat_{Kite} = ((\ TLat_{wl} + TLat_w) + (2TLat_{wl} + 2\ TLat_w)) \\ + ((2TLat_{wl} + 3TLat_w) + (2TLat_{wl} + 4TLat_w)) \end{aligned} \tag{5}$$

$$\begin{aligned} HLat_{opmss} = ((\ TLat_{wl}) + (2TLat_{wl} + TLat_w)) \\ + ((TLat_{wl} + 2TLat_w) + (2TLat_{wl} + 2TLat_w)) \end{aligned} \tag{6}$$

4.2. Scenario II: Handoff Signaling Cost

The total handoff signaling cost of the Kite solution can be formulated when the mobile producer disconnects from CR2 and moves to CR3 within D1. Before the resumption of communication, a number of messages are exchanged between the mobile producer, the anchor router or the RS, and the consumer to update the FIB table before transmission takes place. The mobile producer exchanges the *TI* and *TD* messages or signals with the anchor router or the RS via CR2 to CR1. Then, the consumer sends a message of interest from CR1 to CR3 to the mobile producer. The total number of *TI*, *TD*, and interest messages sent between the consumer, the producer, and the router is $3 \times a$, and the one between routers is $3 \times b$ per hops. The signaling cost of the Kite solution is formulated in Equation (7). In addition, if the mobile producer moves to D2, the total signaling that takes place is added with $5 \times a$ and $7 \times b$ per hop, as presented in Equation (9). For the intra-domain, the two equations are generated to analyze the handoff signaling cost for the Kite solution and OPMSS, as presented in Equations (7) and (8), respectively.

$$HCost_{kite} = \gamma \times (S_{TI/TD} \times (a + b) + S_{int} \times (2a + 2b)) \tag{7}$$

$$HCost_{opmss} = \gamma \times (S_{MI} \times (a) + S_{int} \times (2a + b)) \tag{8}$$

The total handoff signaling cost of the proposed OPMSS solution can be obtained the same way as in the Kite solution. The number of messages is exchanged as signaling between the mobile producer and the consumer for the FIB table update by broadcasting *MI* packets before transmission takes place. For a single domain, the consumer resends the Interest packet after the relocation of the mobile producer from CR1 to CR2, and the signaling cost covers are $3 \times a$ and b per hops, as shown in Equation (8). For inter-domain mobility, if a mobile producer moves to D2, the total signaling added is $3 \times a$ and $9 \times b$ per hop, due to the *MI* broadcasting to both D1 and D2, as presented in Equation (10). Equations (9) and (10) are generated for the signaling cost for the inter-domain mobility of the Kite solution and the OPMSS, respectively.

$$\begin{aligned} HCost_{Kite} = \gamma \times ((S_{TI/TD} \times 2(a + b) + S_{int} \times (2a + 2b)) \\ + (S_{TI/TD} \times 2(a + 2b) + S_{int} \times (2a + 4b)) \end{aligned} \tag{9}$$

$$HCost_{opmss} = \gamma \times (S_{MI} \times (a) + S_{int} \times (2a+b) + S_{MI} \times (a+7b) + \\ S_{int} \times (2a+2b)) \tag{10}$$

4.3. Scenario III: Packet Delivery Cost

The packet delivery cost of the Kite solution and the proposed OPMSS can be determined when handoff processes are completed and the first interest packet after the handoff reaches the mobile producer. In the Kite solution, the data path is created through the anchor router or the RS, while in OPMSS the best route is created between the consumer and the mobile producer to avoid triangular routing. Therefore, Equations (11) and (13) are generated for both the intra-domain and inter-domain mobility analysis of the Kite solution. The data path is created along the trace left by *TI* and *TD* through CR1 to CR3 via CR2 for intra-domain mobility and through CR1 to CR5 via CR2 and CR3 for inter-domain mobility.

$$DCost_{Kite} = \mu \times S_{data} \times (2a+2b) \tag{11}$$

$$DCost_{opmss} = \mu \times S_{data} \times (2a+b) \tag{12}$$

To obtain OPMSS packet delivery cost after the handoff process, the best route strategy creates an optimal path between the consumer and the producer. The path is created from CR1 to CR3 directly for intra-domain mobility and from CR1 to CR5 via CR4 for inter-domain mobility, which is presented in Equations (12) and (14), respectively.

$$DCost_{Kite} = \mu \times (S_{data} \times (2a+2b) + S_{data} \times (2a+4b)) \tag{13}$$

$$DCost_{opmss} = \mu \times (S_{data} \times (2a+b) + S_{data} \times (2a+2b)) \tag{14}$$

5. Performance Evaluations

Numerical evaluation is used, as in [49–51], for the handoff performance evaluation. However, the numerical evaluation only is not enough to cover the peculiarities of producer mobility. Hence, both numerical and simulation-based investigations are used for the evaluation of handoff performance, as in [7,25]. Therefore, the performance of OPMSS is evaluated using both numerical and simulation investigation, specifically for the evolution of unnecessary interest packet loss, the provision of data path optimization, reduced handoff latency, signaling cost, and good data delivery. The numerical evaluation considers handoff performance, such as handoff latency, signaling cost, and data delivery via the optimal path.

5.1. Numerical Evaluation

The parameters in Table 1 are used to investigate OPMSS in terms of minimal handoff latency, signaling cost, and path optimization. A numerical evaluation is conducted as in previous research [8,51] using a network analysis model based on Equations (3)–(14) presented in Section 4.

The handoff latency is the time it takes a mobile producer to receive a pending interest packet from the content consumer after completion of the handoff. In OPMSS, the time that the *MI* message takes to update the intermediate FIB tables is measured based on the number of hops between the content consumer and the mobile producer. In the Kite solution, the time taken for the exchange of *TI* and *TD* messages between the anchor router or RS and the mobile producer to create traces in the FIB tables, and in addition, the time taken for the pending interest to reach the new location of mobile producer in both PMSS and Kite solution. Figure 6 shows the summary of the numerical handoff latency results for OPMSS and the Kite solution that covered both inter-domain and intra-domain motilities that are measured against wireless link failure. To discuss the intra-domain and inter-domain handoff latency performance of the proposed OPMSS and the Kite solution, observations, justification, a quantitative analysis, and inferences are needed to determine the best solution.

Table 1. Numerical Analysis Parameters and Values.

Notation	Parameters	Value
a	Transmission latency btw client and router	1
b	Transmission latency between routers	5
Ldw	Wired link delay	2 ms
$Ldwl$	Wireless link delay	10 ms
Qd	Queuing delay	5 ms
Bw	Wired bandwidth	100 mbps
Bwl	Wireless bandwidth	11 mbps
$Sname$	Size of the signaling packet	+16 bytes
$Sdata$	Size of data packets	2000 bytes
$Siint$	Size of Interest packet	40 bytes
q	The probability of link failure	$0 \leq q \leq 1$
γ	Interest arrival rate	0–10
μ	Subnet crossing rate	0–100%

The handoff latency results are achieved by plotting total handoff latency measured in milliseconds against the probability of the link failure of the mobile producer. Figure 6a shows that OPMSS has 11% lower latency than the Kite solution for intra-domain mobility. The minimum handoff latency is 150 ms for OPMSS and 169 ms for the Kite solution. Furthermore, the handoff latency of OPMSS is less than 20% compared to the Kite solution for inter-domain mobility, with a minimum of 229 ms for OPMSS and 289 ms for the Kite solution. Therefore, OPMSS has better handoff latency performance for both intra- and inter-domain mobility compared to the Kite solution. However, at $q = 0.9$ onward, we can observe that the latency of OPMSS increases by 10% compared to the Kite solution. This is because the Kite solution is able to manipulate the network by keeping the trace update alive. The OPMSS needs to reconnect and broadcast the *MI* packet to the domain, because when the link fails, the mobile producer may not be reached. Overall, despite the possibility of total link failure, OPMSS has better handoff latency than the Kite solution.

The OPMSS solution is implemented in an NDN simulator called ndnSIM that operates based on an open source NS-3. It offers reliable simulation results, particularly for a name-based architecture popularly known as ICN. The ndnSIM allows the transfer of simulated experiments to real practical applications. The simulator is implemented using different C++ programming methods to model NDN architectural behavior, such as the FIB table, CS, PIT, and interfaces of communication with other nodes. The benchmark solution known as TB or the Kite solution is already implemented in ndnSIM, and the codes are available to reproduce the results. The Kite solution is rerun in the same environment with OPMSS to evaluate and compare their performance.

The operating system used for the simulation implementation of OPMSS is Linux distribution Ubuntu 18.04 LTS, which is open source and compatible with many open source simulators. This is followed by NS-3, a discrete-event network simulator that puts the emphasis on layers 2, 3, and 4. NS-3 is commonly used to model Internet systems and communication networks. NS-3 is integrated with ndnSIM 2.3, and both are configured in Ubuntu 18.04 LTS for a successful simulation experiment.

5.2. Simulation Parameters

The parameters used in this simulation relate to the mobility model, speed, number of mobile producers, topologies, and NDN set-up. Two different topologies, Abilene and the 4 × 4 grid, are used for two different scenarios. The speed of the mobile producer ranges between 0 m/s and 100 m/s, depending on the scenario set-up. A summary of the parameters with their respective values is presented in Table 2.

Table 2. Simulation parameters and values.

Category	Parameters	Value
Mobility	Mobility model	Random waypoint mobility
	Mobility speed	1–100 m/s
	Number of mobile nodes	1–5
	Network	802.1 1n
NDN	Forwarding strategy	BestRoute
	Cache replacement	LRU, 100 items on each node
	Cache size	1000 objects
	Interest rate	10/s
	Data packet size	1024 bytes
	Interest packet size	24 bytes
	Application	Consumer and producer
Topology	Abilene	1 Gbps link
	Grid 4 × 4	1 Gbps link
	Link delay	10 ms

5.3. Simulation Scenarios

In this experiment, there are two simulation scenarios for the Kite solution and OPMSS. The two scenarios are based on the 4 × 4 grid and Abilene topology. In scenario I, the infrastructural topology is a 4 × 4 grid that consists of 19–23 nodes, with a distance of 100 m between them. Among the nodes, there are three consumers and one mobile producer by default, and the simulation runs for 100 s. The content mobile producer moves randomly within the 400 × 400 square area covered by a different access point (AP) at a varying constant speed of 10 m/s –100 m/s. The grid nodes serve as routers and APs. Scenario II differs from Scenario I, as it consists of 12–13 nodes in an Abilene topology. Among the nodes, the mobile producer's number varies from 1 to 5 and the simulation runs for 100 s, while producers move randomly at constant speed varying from 1 m/s –100 m/s by taking the interval of 10 m/s into consideration. The scenarios are run for the Kite solution and OPMSS.

The handoff performance of OPMSS is evaluated and benchmarked with the Kite solution using numerical investigation and a simulation experiment to ensure our proposed solution reduces unnecessary Interest packet loss and provides optimal packet delivery, good throughput, and a good packet delivery ratio.

5.4. Simulation Evaluation

Simulations for OPMSS, Kite upload, and pull pattern are run for 100 s, under two scenarios with different infrastructural topology. The mobile producer is randomly moving within the bounds of the random walk mobility model for each simulation executed. In OPMSS, the restricted domain routers are placed at nodes 5, 6, 9, and 10. The consumer node is connected to node 15 for the 4 × 4 grid topology and is connected to nodes 3 and 7 for the Abilene topology. For the Kite upload and pull pattern, the server or consumer node is connected to node 0 for the grid topology and to node 15 for the Abilene topology, while the RS node is connected to node 0 and node 1 for the grid topology and the Abilene topology, respectively. The simulation is conducted for different speeds of the mobile producer; the results are presented in Figure 6b.

The results of handoff latency and data path stretching can be determined by observing the hop count distances between the consumer and the mobile producer. The average hop count result is obtained against the speed of mobile producer that is varied for the two scenarios. The results are plotted in a graph with average hop count on the *y*-axis against the speed of the mobile producer on the *x*-axis as presented in Figure 6a,b, respectively. The figures present the average hop count results that prove the solution to the path stretching problems of the Kite solution compared with the proposed OPMSS.

The results of the grid topology scenario show that when the speed of the mobile producer increases, the average hop count decreases for both OPMSS and the Kite pull pattern, while that of the Kite upload remains unstable. The overall average hop counts of OPMSS are 46% and 42% lower than those of the Kite solution, as shown in Figure 6b, and are 34% and 37% lower in Figure 7a for both the upload and pull pattern. The hop count in the Kite upload pattern is higher than in OPMSS and the Kite pull pattern; this is because in the Kite upload, the consumer must establish the trace from end to end, without a half-way trace due to the absence of an RS. OPMSS has a lower average hop count due to the advantage of *MI* packets being broadcast, which updates all the FIB tables of the intermediate routers and establishes an optimal route between the consumer and the mobile producer. The low average hop count proves that the proposed OPMSS has better handoff latency and solves the path stretching problem of the Kite solution for both the grid and Abilene topologies.
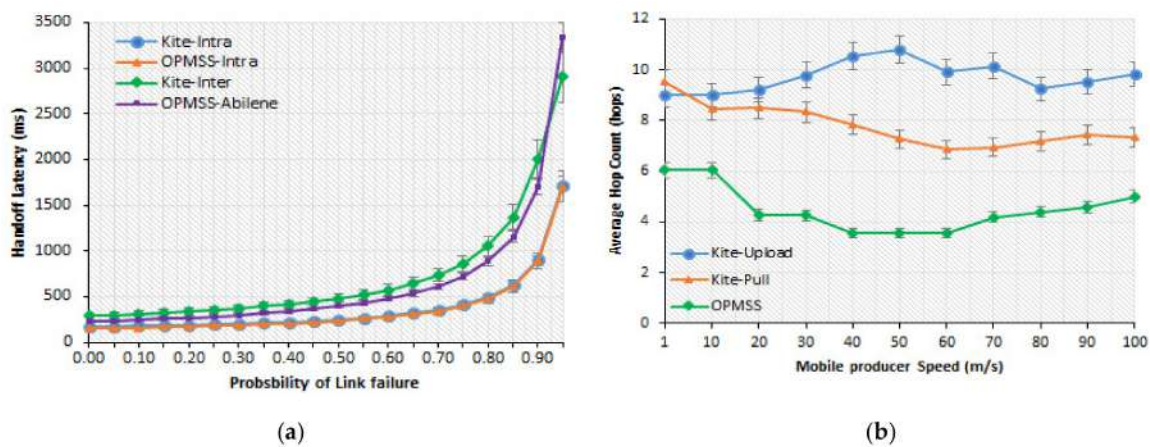

Figure 6. (**a**) Inter- and intra-domain handoff latencies (**b**) Average hop count vs. speed in the grid scenario.
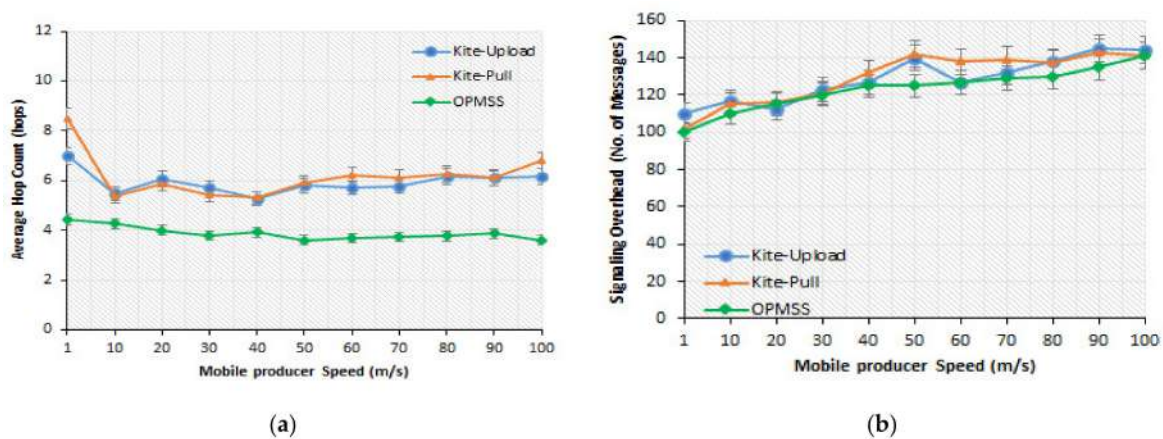

Figure 7. (**a**) Average hop count vs. speed in the Abilene scenario, (**b**) Signaling overhead vs. speed in the Abilene scenario.

The handoff signaling cost results for the two different scenarios are plotted on the graph; the *y*-axis represents the handoff signaling overhead cost, while the *x*-axis represents the speed of the mobile producer. Figures 7b and 8a present the average signaling overhead cost of OPMSS and the Kite solution for both the upload and the pull pattern. The results show that despite the broadcasting nature of OPMSS and the help of the domain restriction application, the outcome is encouraging. The Kite solution is considered very good in terms of having minimal handoff signaling overhead compared to the mapping-based approach and the control/data plane-based approach. The results shown in Figures 7b

and 8a reveal that OPMSS has nearly equal or less handoff signaling overhead than the Kite solution, with about 4% signaling overhead compared to the Kite upload and pull pattern. The OPMSS results can be considered good, even though there is no significant improvement against the Kite solution.

The average delay refers to the time it takes for a packet transmitted from the mobile producer to reach the consumer, which is measured across the number of consecutive hops. The average packet delay describes the significant effect of path stretching and the handoff latency of the proposed OPMSS and the Kite solution. In the Kite solution, the handoff delay is accumulated between the distances that the producer moves to an anchor or RS point. However, the mobile producer moves in a random direction; even if the consumer and producer are nearby, the interest packet must pass through the RS, thus causing the path to stretch. In the proposed OPMSS, by broadcasting the *MI* update message, the consumer interest packet can reach the mobile producer via a single hop. Figure 8b shows the overall average packet delay in OPMSS compared to the Kite solution. The vertical *y*-axis of the graph represents average packet delay in milliseconds, and the horizontal *x*-axis represents the speed of the mobile consumer measured in seconds. By general observation, the OPMSS has 30% fewer average packet delays for Scenario I and 25% fewer average packet delays for Scenario II compared to the Kite solution. This indicates that OPMSS reduces the delay between half and three-quarters of the average packet delay and reduces the handoff latency compared to the Kite solution. These results prove the effectiveness of OPMSS in solving the path-stretching problem and providing data path optimization.
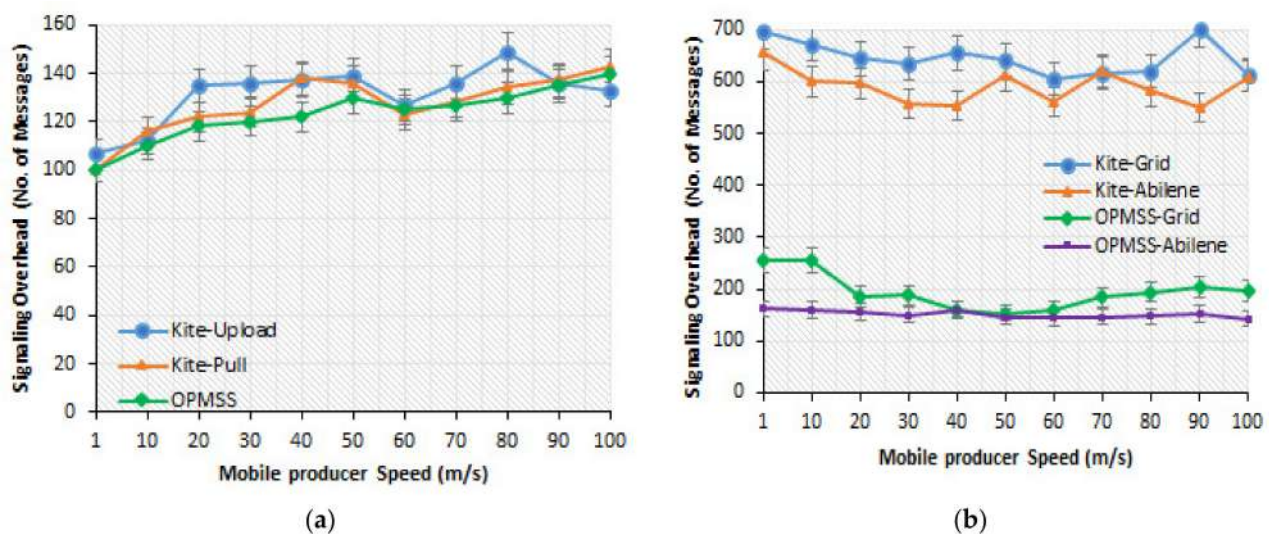

Figure 8. (**a**) Signaling overhead vs. speed in the grid scenario, (**b**) Average packet delay vs. speed.

Interest packet loss is one of the major concerns in solving producer mobility problems, especially when the movement frequency and speed is high. Figure 9 shows the performance of OPMSS in terms of controlling interest packet loss compared to the Kite solution. The OPMSS regulates and stabilizes the network by broadcasting *MI* packets to update the intermediate routers to reduce the number of lost interest packets. The overall results indicate that OPMSS regulates and minimizes the level of Interest packet loss from 15–40% compared to the Kite solution across the two scenarios. This result is observed when the speed of the mobile producer increases from 40 m/s to 100 m/s, which means that OPMSS performed better than the Kite solution.
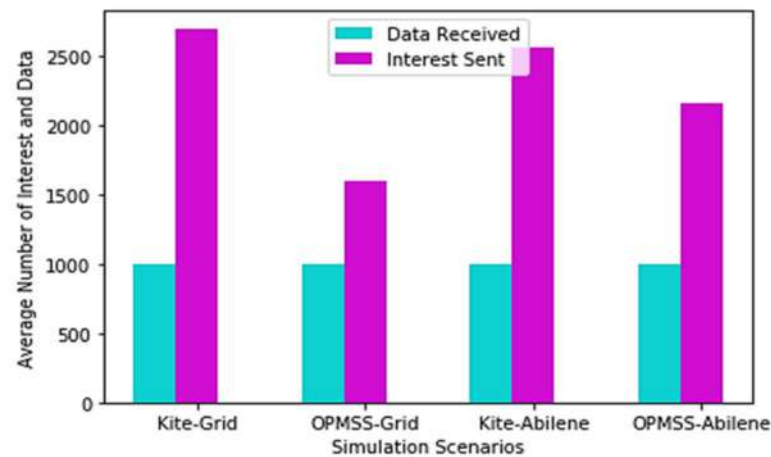
Figure 9. Interest packet loss in different scenarios.

The two scenarios are compared based on equal data received for OPMSS and the Kite solution. Figure 9 shows the average amount of interest packets and data sent per scenario; OPMSS has minimal interest packets sent for an equal amount of data compared to the Kite solution. Specifically, the grid scenario shows that about 1500 interest packets were required to be sent for 1000 data packets, while more than 2500 interest packets were sent in the Kite solution. Therefore, controlling the loss of interest packets will improve the packet delivery level, reduce the signaling cost, and provide better packet throughput and bandwidth consumption.

Throughput is the average number of packets received per unit simulation time, and a higher value of throughput indicates better performance in terms of producer mobility. Hence, the packet throughputs of OPMSS and the Kite solution are measured to determine the level of successful packet delivery after the handoff. Figure 10a shows that, from an earlier time point in the simulation, that is, between 20 and 50 s, the Kite solution has a higher throughput. At the start, as early as 17 s, and when the simulation time goes beyond 50 s, OPMSS performs better than the Kite solution. Overall, we can conclude that the proposed OPMSS performed better than the Kite solution by at least 20% for both scenarios, as presented in Figure 10a,b, respectively. The high throughput performance of OPMSS is because the data are delivered via an optimal path; the best route is established by the NDN routing plane immediately after the FIB tables are updated using the *MI* packets. The good performance of the proposed OPMSS is due to the lower hop count for data packet delivery and better handoff latency when the mobile producer relocates to a new PoA.
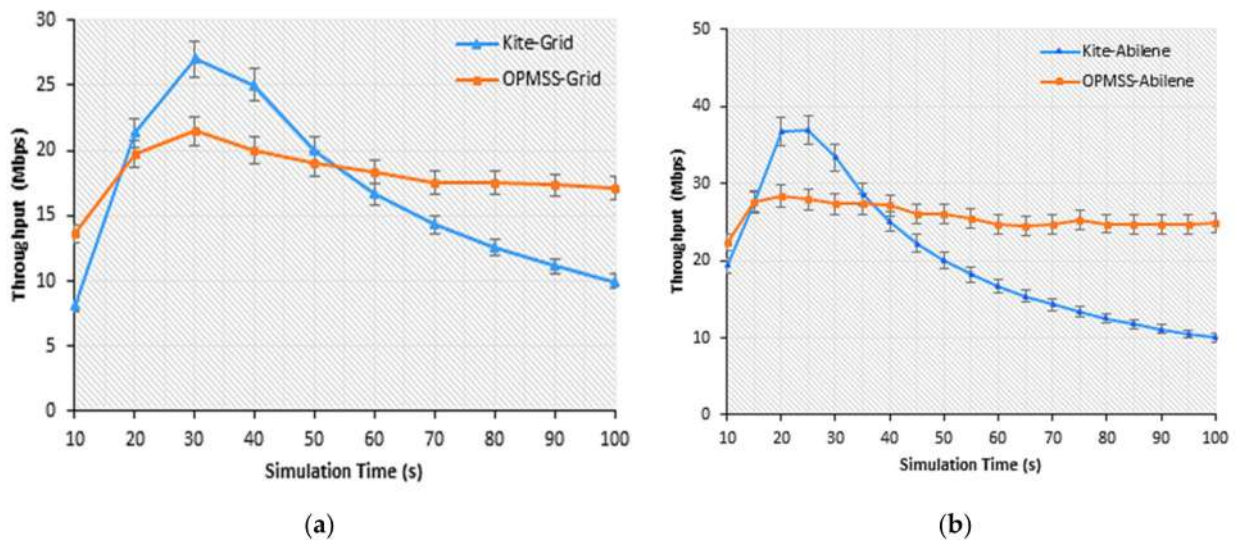
Figure 10. (**a**) Throughput vs. simulation time in the grid scenario, (**b**) Throughput vs. simulation time in the Abilene scenario.

6. Discussion

The results obtained from Section 5 for the evaluation of the proposed OPMSS and benchmarked with the existing Kite solution proved that mobility support in the future Internet is feasible. The OPMSS solution is proposed to bridge the gap and support the smooth integration of NDN with 5G, WSN, IoT, and any other recent technology that requires mobility support upon integration with NDN. Our solution provides an avenue to rapidly update the local FIB tables when the content producer moves to a new location, thus ensuring the reachability of the mobile producer. The solution does not change the architecture of NDN but rather introduces a little functionality at the application level of the producer, the domain control router, and the expansion of FIB tables. The *MI* packet and broadcasting strategy are implemented at the application and forwarding level of the NDN producer. Therefore, the routing plane of NDN is still maintained. Once the FIB tables are updated, the network uses the routing plane of NDN to create an optimal path to reach the new location of the mobile producer after handoff. The solution minimizes handoff latency, signaling costs, and unnecessary interest packet loss by providing support for seamless producer mobility.

The differences between Kite and OPMSS solution are obvious, as Kite has as much as three additional data packets, such as trace data (*TD*), trace interest (*TI*) and consumer interest, while the OPMSS has mobility interest (*MI*) and consumer interest. The exchange of three different packets may result in high signaling in the network. In addition, the Kite has a rendezvous server (RS) or immobile anchor router for interest forwarding when the mobile producer changes to a new location. The OPMSS used broadcasting strategy to update the intermediate routers with the new routing prefix to prevent the incidence of single point of failure as a result of using RS or immobile anchor. Moreover, with the help of NDN data plane, broadcasting strategy and *MI* packet, the OPMSS provides an optimal data path after handoff, which is not possible in Kite, as some data exchange passes through the RS or immobile anchor that causes path stretching, as shown in Table 3.

Table 3. Numerical Analysis Parameters and Values.

Functionalities	Kite	OPMSS
Packets	Trace Data (TD), Trace Interest (TI) and Consumer Interest	Mobility Interest (MI) and Consumer Interest
Server	Rendezvous Server (RS) or Immobile anchor	Nil
Forwarding Strategy	Flooding and Interest Forwarding	Broadcasting strategy
Path Optimization	Not Optimal	Optimal
Single point of failure	Yes	No
Routing	NDN data plane	NDN data plane

7. Conclusions

Seamless mobility support is the key feature of NDN, especially as the most prominent ICN architecture that has genuine possibilities to exploit the design of IoT, wireless sensor network and 5G networks. Mobility support was provided by many researchers to make sure that NDN can successfully replace IP Internet in the future. Mobile producer support is provided by using different approaches such as an anchor-based, indirection-based, mapping-based, DNS-like, rendezvous-based approach, and many more. However, these research papers employ additional infrastructures that make NDN architecture a little bit complicated to handle. Therefore, a prominent Kite solution was proposed to make the data retrieval and routing transparent, while achieving location freeness by using a routable rendezvous or immobile anchor. The Kite solution performed better than prominent mapping-based and location-based approach solutions in terms of handoff signaling cost and latency. However, the Kite solution introduces path stretching problem and rendezvous, or immobile anchors can serve as a single point of failure, in such a case that the data exchange must pass through them.

In this paper, an OPMSS solution is proposed to curtail the path stretching problem and ensure that a single point of failure is avoided. We contribute the utilization of broadcasting method and mobility interest packet to update the FIBs along with the intermediate routers, to ensure that a mobile producer can be reached. To avoid the problem of the broadcasting storm, a restricted domain is provided, using a domain control router to make sure that *MI* packet broadcasting is within the concerned domain. The performance of OPMSS has been evaluated using both analytical and simulation investigation; the evaluation result shows that OPMSS provides an optimal data path delivery with minimum handoff latency. In addition, the solution avoids the single point of failure with equal and acceptable signaling overhead compared to Kite. Moreover, the result ascertains that the OPMSS provision of mobility support solution can influence the perfect integration of NDN with other networks that support packets broadcasting, such as IoT, 5G and wireless sensor network. OPMSS performed well in smaller and semi-larger networks with scalable and acceptable signaling overhead cost. However, further research can be carried out to thoroughly investigate the scalability issue and mitigation of broadcasting storm in case of a very large network.

Author Contributions: Conceptualization, M.H. and M.A.N.; writing—original draft preparation, M.H.; writing—review and editing, B.-S.K. and M.H.; funding acquisition, B.-S.K. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The topologies used in this study are publicly available at https: //github.com/KITE-2018, accessed on 28 April 2021.

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

References

1. Zhang, L.; Estrin, D.; Burke, J.; Jacobson, V.; Thorton, J.D.; Smetters, D.K.; Zhang, B.; Tsudik, G.; Claffy, K.; Krioukov, D.; et al. Named Data Networking. *NDN Tech. Rep. NDN-0001* **2010**. [CrossRef]
2. Amadeo, M.; Campolo, C.; Iera, A.; Molinaro, A. Named data networking for IoT: An architectural perspective. In Proceedings of the EuCNC 2014—European Conference on Networks and Communications, Bologna, Italy, 23–26 June 2014; pp. 1–5.
3. Amadeo, M.; Campolo, C.; Molinaro, A. Empowering 5G network softwarization through Information Centric Networking. *Internet Technol. Lett.* **2018**, *1*, e30. [CrossRef]
4. Conti, M.; Hassan, M.; Lal, C. BlockAuth: BlockChain based distributed producer authentication in ICN. *Comput. Netw.* **2019**, *164*, 106888. [CrossRef]
5. Ravindran, R.; Chakraborti, A.; Amin, S.O.; Azgin, A.; Wang, G. 5G-ICN: Delivering ICN Services over 5G Using Network Slicing. *IEEE Commun. Mag.* **2017**, *55*, 101–107. [CrossRef]
6. Zhang, H.; Liu, N.; Chu, X.; Long, K.; Aghvami, A.H.; Leung, V.C.M. Network Slicing Based 5G and Future Mobile Networks: Mobility, Resource Management, and Challenges. *IEEE Commun. Mag.* **2017**, *55*, 138–145. [CrossRef]
7. Augé, J.; Carofiglio, G.; Grassi, G.; Muscariello, L.; Pau, G.; Zeng, X. MAP-Me: Managing anchor-less producer mobility in Information-Centric Networks. *IEEE Trans. Netw. Serv. Manag.* **2018**, *15*, 596–610. [CrossRef]
8. Hussaini, M.; Nor, S.A.; Ahmad, A. PMSS: Producer mobility support scheme optimization with RWP mobility model in Named Data Networking. *Int. J. Commun. Netw. Inf. Secur.* **2018**, *10*, 329–339.
9. Hussaini, M.; Nor, S.A.; Ahmad, A. Producer Mobility Support for Information Centric Networking Approaches: A Review. *Int. J. Appl. Eng. Res.* **2018**, *13*, 3272–3280.
10. Hussaini, M.; Nor, S.A.; Ahmad, A. Producer mobility support schemes for named data networking: A survey. *Int. J. Electr. Comput. Eng.* **2018**, *8*, 31–42. [CrossRef]
11. Zhu, Z.; Afanasyev, A.; Zhang, L. *Named-Data Networking Project, Technical Report NDN-0013*; NDN Project; University of California: Los Angeles, CA, USA, 2013; pp. 1–6.
12. Ahlgren, B.; Dannewitz, C.; Imbrenda, C.; Kutscher, D.; Ohlman, B. A Survey of Information-Centric Networking. *IEEE Commun. Mag.* **2012**, *50*, 26–36. [CrossRef]
13. Zhang, Y.; Zhang, H.; Zhang, L. *NDN, Technical Report NDN-0020*; NDN Project; University of California: Los Angeles, CA, USA, 2014; pp. 1–10.
14. Gur, G.; Porambage, P.; Liyanage, M. Convergence of ICN and MEC for 5G: Opportunities and Challenges. *IEEE Commun. Stand. Mag.* **2020**, *4*, 64–71. [CrossRef]
15. Wang, X. An Efficient Named-Data-Networking-Based IoT Cloud Framework. *IEEE Internet Things J.* **2020**, *7*, 3453–3461. [CrossRef]
16. Saxena, D.; Raychoudhury, V.; Suri, N.; Becker, C.; Cao, J. Named Data Networking: A Survey. *Comput. Sci. Rev.* **2016**, *19*, 15–55. [CrossRef]
17. Carofiglio, G. Mobile Video Delivery with Hybrid ICN IP-Integrated ICN Solution for 5G. 2017. Available online: https://www. cisco.com/c/dam/en/us/solutions/collateral/service-provider/ultra-services-platform/mwc17-hicn-video-wp.pdf (accessed on 16 April 2021).
18. Li, Z.; Xu, Y.; Liu, K.; Wang, X.; Liu, D. 5G with B-MaFIB Based Named Data Networking. *IEEE Access* **2018**, *6*, 30501–30507. [CrossRef]
19. Din, S.; Paul, A.; Rehman, A. 5G-enabled hierarchical architecture for Software-Defined Intelligent Transportation System. *Comput. Netw.* **2019**, *150*, 81–89. [CrossRef]
20. Mars, D.; Mettali Gammar, S.; Lahmadi, A.; Azouz Saidane, L. Using Information Centric Networking in Internet of Things: A Survey. *Wirel. Pers. Commun.* **2019**, *105*, 87–103. [CrossRef]
21. Arshad, S.; Azam, M.A.; Rehmani, M.H.; Loo, J. Recent advances in Information-Centric Networking based Internet of Things (ICN-IoT). *IEEE Internet Things J.* **2018**, *14*, 1–31. [CrossRef]
22. Meddeb, M.; Dhraief, A.; Belghith, A.; Monteil, T.; Drira, K.; Gannouni, S. AFIRM: Adaptive forwarding based link recovery for mobility support in NDN/IoT networks. *Futur. Gener. Comput. Syst.* **2018**, *87*, 351–363. [CrossRef]
23. Khan, J.A.; Ghamri-doudane, Y. ROVERS: Incentive-based recruitment of connected vehicles for urban big data collection. *IEEE Trans. Veh. Technol.* **2019**, *68*, 5281–5294. [CrossRef]
24. Jiang, X.; Bi, J.; Wang, Y.; Lin, P.; Li, Z. A content provider mobility solution of Named Data Networking. In Proceedings of the Proceedings—2012 20th IEEE International Conference on Network Protocols (ICNP), Austin, TX, USA, 2 November 2012; pp. 1–2.
25. Zhang, Y.; Xia, Z.; Mastorakis, S.; Zhang, L. KITE: Producer mobility support in Named Data Networking. In Proceedings of the ICN '18: 5th ACM Conference on Information-Centric Networking (ICN '18), Boston, MA, USA, 21–23 September 2018; ACM: Boston, MA, USA, 2018; pp. 1–12.
26. Yan, Z.; Zeadally, S.; Zhang, S.; Guo, R.; Park, Y.-J. Distributed mobility management in Named Data Networking. *Wirel. Commun. Mob. Comput.* **2016**, *16*, 1773–1783. [CrossRef]

27. Kim, D.; Ko, Y. On-demand anchor-based mobility support method for Named Data Networking. In Proceedings of the 19th International Conference on Advanced Communication Technology (ICACT), Pyeongchang, Korea, 19–22 February 2017; pp. 19–23.

28. Gao, S.; Zhang, H. Scalable mobility management for content sources in Named Data Networking. In Proceedings of the 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 9–12 January 2016; pp. 79–84. [CrossRef]

29. Cha, J.H.; Choi, J.H.; Kim, J.Y.; Min, S.G.; Han, Y.H. A mobility link service in NDN face to support consumer mobility service. In Proceedings of the International Conference on Ubiquitous and Future Networks, ICUFN, Milan, Italy, 4–7 July 2017; pp. 444–449.

30. Hussaini, M.; Nor, S.A.; Ahmad, A. Optimal broadcast strategy-based producer mobility support scheme for Named Data Networking. *Int. J. Interact. Mob. Technol.* **2019**, *13*, 4–19. [CrossRef]

31. Hussaini, M.; Naeem, M.A.; Kim, B.-S.; Maijama'a, I.S. Efficient producer mobility management model in Information-Centric Networking. *IEEE Access* **2019**, *7*, 42032–42051. [CrossRef]

32. Hussaini, M.; Nor, S.A.; Bello-Salau, H.; Hadi, H.J.; Gumel, A.A.; Jahun, K.A. Mobility Support Challenges for the Integration of 5G and IoT in Named Data Networking. In Proceedings of the 2019 2nd International Conference of the IEEE Nigeria Computer Chapter, NigeriaComputConf 2019, Zaria, Nigeria, 14–17 October 2019.

33. Hussaini, M.; Bello-Salau, H.; Hadi, H.J.; Nor, S.A.; Ahmad, A.; Hassan, I.M. Tools for Analytical and Simulation Investigation of Named Data Networking. In Proceedings of the 2019 2nd International Conference of the IEEE Nigeria Computer Chapter, NigeriaComputConf 2019, Zaria, Nigeria, 14–17 October 2019.

34. Choi, J.; Cha, J.; Han, Y.; Min, S. A Dual-Connectivity Mobility Link Service for Producer Mobility in the Named Data Networking. *Sensors* **2020**, *20*, 4859. [CrossRef]

35. Kerrche, C.A.; Ahmad, F.; Elhoseny, M.; Adnane, A.; Ahmad, Z.; Nour, B. Internet of Vehicles Over Named Data Networking: Current Status and Future Challenges. In *Emerging Technologies for Connected Internet of Vehicles and Intelligent Transportation System Networks*; Springer International Publishing: Cham, Switzerland, 2020; pp. 83–99. ISBN 9783030227739.

36. Xu, Y.; Tong, S.; Zhang, T. COMPASS: Directing Named Data Transmission in VANETs by Dynamic Directional Interfaces. *IEEE Access* **2020**, *8*, 8418–8435. [CrossRef]

37. Al-qutwani, M. Request/Advertise-Based Content Forwarding in Vehicular Named Data Networking. *IEEE Access* **2021**, *9*, 226–236. [CrossRef]

38. Sivaraman, V.; Guha, D.; Sikdar, B. Towards Seamless Producer Mobility in Information Centric Vehicular Networks. In Proceedings of the (VTC2020-Spring) 2020 IEEE 91st Vehicular Technology Conference, Antwerp, Belgium, 25–28 May 2020; pp. 1–5.

39. Do, T.X.; Kim, Y. Optimal provider mobility in large-scale Named- Data Networking. *KSII Trans. Internet Inf. Syst.* **2015**, *9*, 4054–4071. [CrossRef]

40. Saxena, D.; Becker, C. Implementation and Performance Evaluation of Name-based Forwarding Schemes in V-NDN. In Proceedings of the 18th International Conference on Distributed Computing and Networking, Hyderabad, India, 1 January 2017; pp. 3–6. [CrossRef]

41. Duarte, J.M.; Braun, T.; Villas, L.A. Addressing the Effects of LowVehicle Densities in Highly Mobile Vehicular Named-Data Networks. In *MSWiM '17: Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*; Association for Computing Machinery: New York, NY, USA, 2017; pp. 1–8. [CrossRef]

42. Duarte, J.M.; Braun, T.; Villas, L.A. Receiver Mobility in Vehicular Named Data Networking. In *MobiArch '17: Proceedings of the Workshop on Mobility in the Evolving Internet Architecture*; Association for Computing Machinery: New York, NY, USA, 2017; pp. 43–48. [CrossRef]

43. Lee, J.; Kim, D. Partial path extension scheme for mobile content source in content-centric networking (CCN). *EURASIP J. Wirel. Commun. Netw.* **2015**, *2015*, 212. [CrossRef]

44. Siris, V.A. Popularity-aware Intra-domain Mobility Management. In Proceedings of the Workshop on Mobility in the Evolving Internet Architecture; ACM: New York, NY, USA, 2017; pp. 13–18.

45. Rehman, R.A.; Kim, B.-S. LOMCF: Forwarding and Caching in Named Data Networking-based MANETs. *IEEE Trans. Veh. Technol.* **2017**, *66*. [CrossRef]

46. Shi, J.; Newberry, E. On Broadcast-based Self-Learning in Named Data Networking. In Proceedings of the IFIP Networking, Stockholm, Sweden, 12–16 June 2017; pp. 1–9.

47. Rao, Y.; Luo, H.; Gao, D.; Zhou, H.; Zhang, H. LBMA: A novel Locator Based Mobility support Approach in Named Data Networking. *China Commun.* **2014**, *11*, 111–120. [CrossRef]

48. Afanasyev, A.; Moiseenko, I.; Zhang, L. *ndnSIM: NDN Simulator for NS-NDN, Tech. Rep. NDN-0005*; NDN Project; University of California: Los Angeles, CA, USA, 2012; pp. 1–7.

49. Hussaini, M.; Nor, S.A.; Ahmad, A. Analytical modelling solution of producer mobility support scheme for named data networking. *Int. J. Electr. Comput. Eng.* **2019**, *9*, 3850–3861. [CrossRef]

50. Hussaini, M.; Nor, S.A.; Ahmad, A.; Mustapha, R.; Abdulateef, A.F. A Conceptual Model of Producer Mobility Support for Named Data Networking using Design Research Methodology. *IAENG Int. J. Comput. Sci.* **2019**, *46*, 1–11.

51. Rui, L.; Yang, S.; Huang, H. A producer mobility support scheme for real-time multimedia delivery in Named Data Networking. *Multimed. Tools Appl.* **2018**, *77*, 4811–4826. [CrossRef]