Computer Communications 156 (2020) 25–34

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Improving mobile ad hoc networks using hybrid IP-Information Centric Networking

Yiannis Thomas *, Nikos Fotiou, Stavros Toumpis, George C. Polyzos

Mobile Multimedia Laboratory, Department of Informatics, School of Information Sciences and Technology, Athens University of Economics and Business, Greece

ARTICLE INFO

Keywords:
Mobile Ad hoc NETworks (MANETs)
Information-Centric Networking (ICN)
Named-Data Networking (NDN)

ABSTRACT

Routing protocols for Mobile Ad hoc NETworks (MANETs) constitute a research area with numerous evolved and well-investigated solutions that typically still rely on IP. Although IP-based protocols for MANETs can effectively handle network changes, thus enhancing connectivity while offering scalability, they lack support for advanced communication paradigms, which would enable richer applications and improve network goodput. Information-Centric Networking (ICN) is an inter-networking paradigm that natively supports functions that can enhance the performance of MANETs, such as multicast, multisource communication, and on-path caching. However, ICN raises significant scalability and feasibility concerns in the presence of intense node mobility. In this paper, we propose a hybrid architecture that combines the best of both worlds: it uses IP for maintaining network topology and it relies on ICN for content lookup and dissemination. To this end, we adapt the Named-Data Networking (NDN) ICN architecture to use legacy IP MANET routing protocols, such as Babel. Using a prototype implementation of our design and the Mininet Wi-Fi network emulator, we experimentally investigate the performance of our hybrid design in several scenarios, finding that the data delivery to network traffic ratio is increased up to 250% and 300% compared to pure NDN and IP solutions, respectively.

1. Introduction

Mobile Ad Hoc NETworks (MANETs) constitute a mature field of research that is expected to grow further due to the continuing evolution of networking devices and services. Wi-Fi chips have now become a commodity and more and more devices, such as smartphones, vehicles, and drones, have wireless connection capabilities. New applications such as file sharing, location-based services, community networks, and vehicle-to-vehicle communication, can be supported efficiently by local, ad hoc networks, that do not rely on some infrastructure, are not necessarily connected to the Internet and could even be partitioned during their operation. All these trends suggest that MANETs may finally find their way to the "masses".

Recent studies have shown that MANETs can benefit from advanced networking functions, such as consumer-driven multicast, multipath, multisource and on-path caching, that are not natively supported by IP-based protocols [1]. Specifically, MANETs can exploit on-path caching to address network partitioning, multisource and multipath transport to handle source, and node failures and consumer-driven transfers to handle consumer mobility [2]. The Information-Centric Networking (ICN) paradigm offers natively the aforementioned networking functions, by decoupling content names from their location and by providing content name-oriented network functions [3]. However, this new networking paradigm has not yet proven its value, facing concerns regarding its feasibility, mostly in terms of processing and memory overhead at network routers [4]. These concerns are even greater in environments that include highly dynamic networks, where the ICN protocols can present reduced scalability. Furthermore, the lack of efficient means of realistic evaluation of ICN-based architectures constitutes an overwhelming barrier to the adoption of this paradigm [2].

In this design, we enhance the performance of MANETs by designing a hybrid IP-ICN system that leverages IP ad hoc routing, in order to manage the changes of network topology, and ICN network functions, in order to enhance content lookup and delivery. Our solution provides "information-centric" methods that enhance content resolution and delivery, such as native multisource, multipath and on-path caching. Moreover, our solution deploys "host-centric" methods that manage mobility effectively and offer reasonable scalability.

The first part is implemented by modifying the Named-Data Networking (NDN) ICN architecture [5], while the latter exploits an IP-based proactive routing protocol, Babel [6].

In order to jointly leverage the IP-based routing protocol and the ICN-based architecture, we modified the NDN's forwarding in order to monitor (and adapt to) the routes that are discovered by Babel. Our hybrid design combines the gains of both worlds: it significantly mitigates

* Corresponding author.
E-mail addresses: thomasi@aueb.gr (Y. Thomas), fotiou@aueb.gr (N. Fotiou), toumpis@aueb.gr (S. Toumpis), polyzos@aueb.gr (G.C. Polyzos).

[1] The emulator is available at https://github.com/intrig-unicamp/mininet-wifi.

https://doi.org/10.1016/j.comcom.2020.03.029
Received 1 October 2019; Received in revised form 25 February 2020; Accepted 17 March 2020
Available online 24 March 2020

the control plane overhead (compared to existing ICN-based solutions), and enhances content delivery (compared to IP-based solutions). At the same time, our design contributes to the "incrementally deployable ICN" concept, where ICN functions find their way to the masses through the integration with IP networks. Using a prototype implementation of our solution and the Mininet Wi-Fi network emulator,[1] we validate the advantages of our design compared to pure IP or ICN solutions.

The structure of the remainder of this paper is as follows. In Section 2 we discuss the related work and the solutions that form the basis of our approach. Furthermore, in the same section we discuss how our approach improves the state of the art. In Section 3 we provide details of our system design. In Section 4 we present a thorough evaluation of our solution using network emulation. Finally, we present our conclusions and plans for future work in Section 5.

2. Background and related work

2.1. Routing in MANETs

Typically, MANET routing is comprised of three steps: firstly, any changes in the link state of the nodes must be discovered; secondly, the changes must be disseminated to the network; finally, the routing tables have to be updated accordingly. The efficiency of a routing protocol relies in discovering changes quickly and disseminating them with low traffic overhead, thus unveiling the trade-off between the size of the control traffic and the freshness of the routes [7]. The existing solutions are often categorized by the approach to achieving a good trade-off. In the following, we briefly discuss the two most popular categories and some indicative designs that we include in the evaluation of Section 4, referring the reader to [8] for a complete survey.

The first category of routing protocols for MANETs is the *proactive* protocols. In proactive protocols the routing tables of nodes are updated periodically or whenever a change in the topology is sensed by broadcasting topology-related control plane messages. The main advantage of these solutions is the timely update of routing information, which offers low latency (as a route is available before it is requested by a data source).

Babel [6] is one of the most popular proactive routing protocols. It is a distance-vector protocol designed for cases where some of the links are wired, allowing various metrics for route selection, such as hop-count, packet loss, radio diversity and delay. The control plane consists of periodical advertisements, known as *hello* messages. Using these messages, Babel nodes update their local link state, remembering all neighbors from which hello messages has been recently received. Babel nodes also periodically advertise the routes they have stored locally through *route update* messages to their neighbors, thus allowing the dissemination of routing information in a distributed fashion. Finally, unscheduled, or triggered, route updates and route requests are permitted in order to quickly respond to a significant change in the network topology, but are proposed only for critical occasions, such as to avoid a routing loop.

The second class of routing protocols for MANETs is the *reactive* protocols. In reactive routing protocols the routing tables of nodes are updated on-demand, i.e., only when communication between two nodes must take place. Solutions of this category are known to overcome the overhead problem of proactive designs, making them more bandwidth-efficient and scalable. On the other hand, the establishment of a path induces significant latency, since the route finding process precedes the actual transfer. The impact of latency is amplified in case of rapidly changing topologies, where routes must be frequently rediscovered during the transfer, thus increasing the idle periods during which a node waits for the new route to be discovered.

2.2. Content discovery in IP-based MANETs

Content discovery in MANETs is complicated due to the fluidity of the network topology [9]. We can categorize the available solutions into three general categories. In *directory-based* architectures, directories receive and store content advertisements and queries, thus resolving all content requests in the network. The directories can be centralized, located at a specific node, or distributed, spanning over multiple network nodes (or clusters of nodes). In *directory-less* architectures, there are no directories to mediate communication; content queries and advertisements are simply broadcasted. In case nodes store the broadcasted information, this architecture is similar to a directory-based architecture where all nodes act as (non-synchronized) directories. Finally, in *hybrid* architectures, nodes exploit directories when they are offered and fall back to broadcasting otherwise. This design can enhance content availability since content queries may be answered by content providers and directories.

There is no type of architecture that maximizes content availability and at the same time minimizes traffic overhead and latency. The tuning of essential parameters, such as the period of advertisements and the number of directories in the network, greatly affects the performance of each solution. However, assuming similar tuning of these parameters, when the content request frequency is low, a directory-less architecture can be more efficient than a directory-based one, since it enhances availability and reduces latency (all network nodes receive and can store advertisements and queries, thus managing a "local directory") while the traffic overhead due to broadcasting messages is bearable. In case of frequent content requests, the clients of a directory-less architecture frequently flood the whole network with their queries, thus creating more overhead than the traffic that is produced in a directory-based solution for keeping the directories consistent and delivering unicasting queries to directories. Finally, it is experimentally found that hybrid architecture can offer the highest service availability, especially in proactive routed MANETs [9].

2.3. Named-Data Networking (NDN)

Named-Data Networking (NDN) [5] is one of the earliest proposals for an ICN architecture, offering natively multipath, multicast, multisource transport and on-path caching. NDN introduces mechanisms that enable both content discovery and routing based solely on content names. Each content item is identified by a unique *name*; if two or more items have the same *name* then they are considered copies of the same information item. *Names* are location-independent, hence a *name* resolution process is required in order to discover *where* or by *whom* a content item is delivered.

In a nutshell, in NDN a *consumer*[2] expresses its interest for a content item by sending an *Interest* packet that contains the *name* of the desired item; Interests are routed by *Content Routers* (CRs) and eventually reach a *producer* or an *in-network* storage, which in return forwards the desired item back to the consumer by sending an *Data* packet.

2.3.1. Routing, forwarding and content discovery

The processes of routing, forwarding and content discovery present significant differences when studied in NDN instead of IP. First, the name-based NDN routing is integrated with the process of content discovery, in contrast to the IP routing which is clearly separated from the content discovery. Second, although routing and forwarding are hard to differentiate in IP, in NDN they are clearly separated by being implemented through different data structures in the CRs and by allowing the nodes to make routing decisions. In NDN, routing is based on the *Forwarding Information Base* (FIB), a lookup table that maps *name* to output interfaces, known as *faces*. A FIB entry can map *name* to multiple faces, thus supporting multipath and multisource transport

[2] Terminology is based on [10].

natively. For example, when multiple faces are available, the CR can dynamically forward packets over different faces (based on various metrics), thus actively participating in network routing. The algorithm that populates the FIB entries is not specified in the original NDN paper, allowing the implementation of different algorithms in order to address different network setups.[3] The use of a special message, called *Advertisement*, is used for updating the FIB of CRs. The NDN architecture follows the directory-less content discovery paradigm, where producers broadcast Advertisements (proactively) or consumers broadcast Interests (reactively) without the need of a Directory service.

Forwarding is based on another data structure, the *Pending Interest Table* (PIT), which is used for the reverse process, i.e., for forwarding *Data* packets back to the consumers. Every time a CR forwards an Interest, it updates the PIT by storing the *name* of the Interest and the face it received it from. Interests with the same *name* are aggregated in PIT, thus storing multiple faces for the same Interest and supporting natively multicast. A PIT entry is consumed when the requested Data packet is received or when its timer expires.

2.3.2. NDN and MANETs

MANETs can benefit from the advanced networking functions offered by ICN, such as consumer-driven multicast, multipath, multisource transport and on-path caching [1]. Specifically, MANETs can exploit NDN's on-path caching to address network partitioning, multisource and multipath to handle source, and node failures and consumer-driven transfers to handle consumer mobility. However, NDN faces concerns regarding its feasibility, mostly in terms of processing and memory overhead at network routers [4,13]. The concern is amplified in environments that include highly dynamic networks, when the routing protocols can saturate the network links with control plane traffic, thus greatly penalizing the scalability of the architecture. A comprehensive survey of research efforts that study the application of ICN in MANETs can be found in [2]. In what follows, we briefly discuss the two most popular categories of protocols, the *proactive* and *reactive*, as well as some indicative designs that we include in the evaluation section.

The proactive protocols rely on periodic control plane messages for populating FIBs. Producers advertise their content list periodically by flooding Advertisements. The Advertisements usually carry the *name* of a content (or the prefix of a content) and they are used by CRs in order to keep their FIBs updated; CRs store in their FIB the incoming interface along with the *name* of the content in the Advertisement. FIB entries expire after some period, hence, if a CR receives an Interest for which there is no FIB entry, the packet is dropped. The cost of flooding in NDN is significantly higher than the corresponding cost in IP, since Advertisements are emitted per content, instead of per node. Various studies try to mitigate the impact of flooding through complicated techniques, known as *controlled flooding*, however those designs raise serious concerns in terms of effectiveness and practicability [2].

The reactive protocols make little, or no, use of the FIB. With these approaches, when a consumer needs a content it issues an Interest packet which is flooded to the network; Data packets are forwarded over the reverse path using the PIT structure. When combined with controlled flooding techniques, the control plane overhead of these solutions can be little, compared to proactive solutions.

2.3.3. Related protocols

E-CHANET [14] is probably the most popular NDN-based architecture tailored for multihop wireless networks. E-CHANET uses packet broadcasting in order to take advantage of the wireless links, introduces an specific congestion control algorithm, and provides some special producer-mobility mechanisms. Being a reactive solution, it floods Interest packets to discover content items but, then, uses controlled flooding based on hop-counters in order to reduce the traffic overhead. Specifically, the packets are extended to carry a *hops to destination* (htd) counter that is reduced at every hop, thus reflecting the remaining hops of the producer. CRs store all known consumers for each content and a similar counter for each known consumer; the counter determines if an incoming packet must be forwarded (using broadcast), in case the packet's counter is larger than the CR's, or dropped. Flooding takes place only in two occasions: firstly, when there is no local knowledge of the distance to any producer for a *name* (usually when downloading a content for the first time) and, secondly, when the producer is no longer reachable (when the path to it fails due to mobility and the expiration of PIT entries).

A recent reactive solution which also mitigates the cost of Interest flooding is *Data reAchaBility BasEd Routing* (DABBER) [15], a protocol proposed for supporting communication between vehicles with intermittent connectivity as well as between vehicles and stationary infrastructure. DABBER avoids flooding by pushing the Interests to a subset of destination nodes that optimize three parameters: centrality, reliability and similarity, thus exploiting a MANET-specific routing cost metric. The performance of DABBER is promising, however it is not evaluated yet, therefore the properties of this design remain to be explored. In addition, DABBER does not conform totally to the ICN principles, introducing topology-based content names, where the naming scheme reflects the way mobile roaming works, i.e., it identifies the operator and home network of each node.

The *Internet of Things* (IoT) sensor networks and the *Vehicular Ad hoc NETworks* (VANETs) revived the need for enhanced connectivity in case of mobility. The infrastructure of these networks has specific characteristics, such as stationary gateways, or fixed node trajectories (due to the nodes having to stay in roads), such as driving in physical roads, thus the developed protocols cannot be used in an MANET, but some of the introduced techniques can be exploited in the context of general MANETs. In [16], the authors address the producer mobility issue in IoT networks with AFIRM, a content-driven, fully distributed, adaptive forwarding solution. AFIRM introduces a network setup phase in which flooding is used to forward requests in order to explore the entire network and update the FIBs of nodes. The flooding is repeated only when the FIBs are empty, following a reactive model. In order to tackle producer mobility, the gateways, where IoT sensors are attached, periodically send a ping message to check if the sensors are still reachable, thus also exhibiting proactive behavior. AFIRM is found to enhance connectivity, thus offering higher packet delivery rates. In [17], the authors provide a systematic view of the research on IoT communication in ICN. Regarding mobility support, authors state that the IoT-ICN designs focus on producer mobility by exploiting stationary network infrastructure, such as anchors and NDN routers, thus not being suitable for pure ad hoc networks. In [18], the authors propose a novel proactive caching design for ICN VANET networks. It is assumed that users connect to different cache-enabled stationary access points as they travel to their destination, hence the next access point can be "predicted" and the piece of data that the user will need when it gets there can be sent proactively. In [19], the authors provide a thorough presentation of the published literature in the area of VANETs in NDN. Authors conclude that, as compared to the general MANETs, mobility patterns in VANET are relatively predictable due to the known road topology, thus less challenging to the network operation.

Finally, it is worth mentioning some studies that are not mobility-specific but target essential problems of NDN. In [20], the authors introduce a content-based routing protocol that considers various metrics, including bandwidth, load, delay, and reliability. The *reliability* metric defines which path is more stable and reliable to retrieve data. Although this study does not consider mobility, this novel routing metric can be exploited in MANETs to enhance connectivity. Similarly, in [21] the authors introduce N-FIB, a proficient NDN forwarding design that reduces the memory and processing cost of NDN routers,

[3] The NLSR [11] protocol, as well as hyperbolic routing [12] are the current candidates for default route disseminating protocol of NDN, however we omit them from this discussion since they are not designed for MANETs.

thus delivering a more practical solution. N-FIB exploits Patricia tries in order to reduce the size of the forwarding state while supporting wire-speed operation and strong correctness in forwarding decisions. Although this design does not enhance mobility support, it can relieve the computation and memory stress of NDN routers in MANETs. In [22], the authors reduce the traffic footprint and the communication delay of IoT services in NDN networks by prioritizing the delivery of delay-intolerant IoT applications. While the design does not extend the mobility support of NDN, it is found to reduce the application delay by approximately 30%.

2.4. Incrementally deployable ICN

The feasibility concerns of ICN architectures as well as the upgrade challenges, that have hindered ICN adoption, recently drove numerous research studies to explore the possibility of an incremental, seamless integration of ICN within the IP world.

In [23], the authors demonstrate the real-life implementation of IPTV in a hybrid, IP-over-ICN, architecture: IP networks are preserved at the edge, connected to each other over an ICN core. This exploits the key benefits of ICN, enabling individual network operators to improve the performance of their IP-based services, without changing the rest of the Internet.

Similarly, in [24] the authors present *Hybrid-ICN* (hICN), an ICN integration inside IP architecture that combines the goods of both worlds in order to deliver a incrementally deployable ICN architecture. As with ICN, the design of hICN supports native consumer mobility through NDN's name-based addressing. Nevertheless, hICN is not focused in MANETs, thus not constituting a sophisticated solutions for detecting topology transitions faster while minimizing control plane overhead. Although the name prefixes coexist with normal IP addresses in the FIB, the population of FIB entries is based on standard IP routing protocols, such as *Border Gateway Protocol* (BGP), thus not enhancing mobility support (compared to plain NDN). In addition, the support of multipath and multisource in hICN is doubtful since these techniques are supported poorly by IP routing.

In [25], the authors present a solution for supporting topic-based publish–subscribe communication in MANETs. The introduced design extends the NDN architecture in order to support the publish–subscribe paradigm, which involves keeping track of the active subscriptions and publications at content level, and exploit the host-based proactive *Optimized Link State Routing* (OLSR) protocol in order to manage network mobility. The design also supposes that names are "topics" and each publisher may publish different data even under the same topic, hence two items published using the same name can be different. In addition, content names are location-dependent, including the producer's identifier, thus disallowing multisource.

2.5. Discussion

- In both IP and NDN, proactive and reactive solutions entail a fundamental trade-off between scalability and performance, hence different protocols perform better under different conditions.
- The NDN-based protocols present increased control plane overhead that questions their scalability and creates feasibility concerns.
- In IP, content discovery and node mobility are separated, thus being handled by different protocols, each optimized for its particular usage. For example, node mobility can be managed proactively by a routing protocol, such as Babel, while content discovery can be handled reactively by a separate directory-based content resolution protocol [9]. In NDN, the absence of the concept of "location" disallows such sophisticated setups.
- The impact of flooding can be addressed by controlled flooding techniques, that either take advantage of additional information, such as social networks, user preferences, GPS data and other, or induce memory overhead to the CRs. While the exploitation of additional information offers performance efficiency, it limits the application scope only to scenarios that this information is available. Similarly, the memory overhead of the CRs can be critical for the already stressed NDN networks.
- NDN proposes the exploitation of prefix advertisement (and longest-prefix-match) in order to mitigate the overhead of storing the FIB entries in CRs. The effectiveness of this technique is indisputable, being essential for deploying NDN in large-scale networks. Nevertheless, the memory savings of FIB come with an apparent cost, namely, multisource is disallowed when the same content is advertised under multiple prefixes, i.e., when content providers use individual node-specific prefixes. The native support of multisource in ICN is known to increase resilience to node failures, bandwidth aggregation and network load balance [26], thus making it important for MANETs where content reachability is heavily challenged.
- It is important for ICN research to propose solutions that can be seamlessly integrated into today's networks. A gradual shift of the communication paradigm towards information-centrism can be practical and can enhance network operation significantly while keeping the cost of the shift at a reasonable level.

3. System design

We introduce a hybrid IP-ICN system that leverages IP ad hoc routing, in order to maintain network topology, and ICN network functions, in order to enhance content lookup and delivery. To this end, we exploit the Babel proactive routing protocol for updating the node-level routing information, and an NDN-based reactive protocol to discover and deliver content. Our design is governed by the following design choices:

- Our design should not violate core ICN principles, including location-independence of content names. Hence, our system does not impose on the selected ICN system any restrictions on the content name format, semantics, etc.
- The ICN substrate of our architecture should be oblivious to the protocols used at the IP-level. For this reason, although in this paper we focus on Babel, any proactive routing protocol could have been used instead. Similarly, our solution can be deployed over IPv4, IPv6 and MAC protocols.
- The proposed system should target deployability. Consequently, our system does not require any modification to the IP substrate. Furthermore, in a network where nodes that support our system and legacy nodes co-exist, the latter nodes do not experience any connectivity issue due to our solution.

Our solution integrates *proactive* routing for handling node mobility and *reactive* directory-less content discovery. Proactive routing is supported through Babel, offering quick mobility management and, in turn, enhanced connectivity, while avoiding the control plane overhead that the NDN proactive protocols can present. Reactive directory-less content discovery is supported by NDN, where consumers broadcast the content queries and the producers respond accordingly, thus offering low control plane overhead with bearable content discovery latency.[4] The work presented in this paper is an improvement to the NDN-based solution included in our previous work published in [27]. The solution presented in that paper considered location-based content names, hence it did not support all ICN functions.

[4] We assume that content is not as mobile as nodes, i.e., the availability of content at a location changes less frequently than the network topology, hence the latency, that a reactive solution induces, is less frequently presented.

3.1. Entities, identifiers, and assumptions

Our design assumes network nodes that are interconnected using a wireless technology, such as Wi-Fi, and can be mobile or stationary. Each node is uniquely identified by a network-specific identifier referred to as $nodeID$. The $nodeID$s can be of arbitrary form, but in our implementation we have selected IPv4 addresses for this purpose. Additionally, all nodes are configured with a *broadcast* address (the same for all nodes). Nodes establish direct connections with each other, thus forming a network graph, and a routing protocol is used for routing packets among nodes through the network graph; in our implementation we have selected Babel.

The information-centric data plane is implemented using a modified NDN network stack, which maintains the name modeling principles of NDN, the PIT and the Content Storage structures, but adapts the structure of the FIB to the node-level routing rules of Babel.

3.2. Data structures

Each node maintains the following data structures:

- **Routing Table** (RT): RTs are used for routing packets. They map a destination $nodeID$ to the next hop $nodeID$. In the following, with the notation $RT[\text{'}nodeA\text{'}] = \text{'}nodeB\text{'}$ we denote that the next hop towards '$nodeA$' is '$nodeB$'. If it is permitted by the underlay routing protocol, a $nodeID$ can be mapped to multiple next hop $nodeID$s.
- **Forwarding Information Base** (FIB): FIB data structures map an item's $name$ to a list of producers' $nodeID$s. Hence, with the notation $FIB[\text{'}itemA\text{'}] = [\text{'}nodeA\text{'}, \text{'}nodeB\text{'}]$ we denote that '$itemA$' is hosted by nodes '$nodeA$' and '$nodeB$'. Similarly, $FIB[\text{'}itemA\text{'}].append(\text{'}nodeC\text{'})$ is used for indicating that '$nodeC$' is added to the list of nodes hosting '$itemA$'.
- **Pending Interest Table** (PIT): PIT data structures are not modified, thus mapping a requested item's $name$ to the $nodeID$ of the next hop on the path from the producer to the consumers. Therefore, $PIT[\text{'}itemA\text{'}] = [\text{'}nodeB\text{'}]$ indicates that the next hop towards the consumer that requested '$itemA$' is '$nodeB$'.[5] Similar to NDN, Interest packets for the same $name$ can be aggregated, hence a PIT entry may contain multiple $nodeID$s: in that case content items are multicast.
- **Content Storage** (CS): CS is a data structure used for storing content items. It maps a $name$ to its $data$, as well as to its $type$. We distinguish two types of content items, namely *permanent* and *ephemeral*. Permanent content items are added in the CS by a content owner application and remain there until the application decides to remove them. On the other hand, ephemeral content items are added in the CS "opportunistically", e.g., through a caching process and they have limited lifetime (defined for example by the caching strategy).

3.3. Protocols

Node mobility is handled by a protocol that implements link state discovery and basic forwarding functions, i.e., nodes are expected to be able to forward a network packet to adjacent nodes, as well as to the broadcast address. In our design, we deploy the Babel routing protocol in order to monitor node reachability, discover changes of network graph, and update routing state across the network. Operating at node-level, Babel *quickly populates the RT* data structure. Other protocols, that bind destination nodes with next-hop nodes, are also compatible with our design.

[5] PIT in NDN contains mappings from a $name$ to one or more $face$s. We replace $face$ with $nodeID$ in order to enable unicast in our architecture where machines have a single broadcast face (i.e., the wireless link). A different study to support unicast is found in [28].

Algorithm 1
Request a content item identified by $name$

```
nodeIDArray ← FIB[name]
if nodeIDArray == null then
    advertisement_request(name)
else
    nodeID ← select_from(nodeIDArray)
    nextHop ← RT[nodeID]
    send_interest(nextHop, name)
end if
```

Algorithm 2
Handle an Advertisement Request for $name$ that includes $nonce$, sent by $consumerID$

```
if nonce ∈ received_nonces then
    drop request
    return
end if
nodeIDArray ← FIB[name]
if nodeIDArray == null then
    advertisement_request(name)
else
    send_advert(name, nodeIDArray, consumerID)
end if
```

In order to implement content lookup, we introduce to the NDN-based substrate two new messages *Advertisement Request* and *Advertisement*: in order for a consumer to locate a content item he broadcasts an Advertisement Request; then a producer responds with a unicast Advertisement message that updates the FIB structure of on-path CRs. We choose to introduce these two new messages for two reasons: (i) we wanted to make sure that the Advertisement Request reaches a content producer (and it is not served for example by an in-network storage node), and (ii) we needed a special type of packet for updating FIB entries: doing that with Data packets would require to modifying the NDN's Data packet format. Then, content retrieval is implemented by having the consumer emitting Interests and the producer responding with Data packets, that are forwarded using the modified FIB and the PIT structures, respectively.

The procedures of emitting the Advertisement Request and the Advertisement are seen in Algorithms 1 and 2, respectively. Initially, a consumer, that wishes content $name$, searches for a producer in $FIB[name]$: If it is $null$, then the consumer issues an Advertisement Request otherwise it issues an Interest packet. An Advertisement Request for a $name$ is a message transmitted to the broadcast address. It contains the $nodeID$ of the requesting node, also named $consumerID$, and the desired $name$. When a CR receives an Advertisement Request, it examines if it has already received this request, and, if not, it looks for a local producer in $FIB[name]$: if no local producer is found, the CR forwards the Advertisement Request to the broadcast address; otherwise the CR sends an Advertisement to the $consumerID$ of the Advertisement Request. Advertisements are unicast packet transmissions thus avoiding the issue of broadcast storms. In order to mitigate the overhead of broacasting Advertisement requests, CRs store the last received packets in order to detect and drop duplicate transmissions (based on unique packet nonces) and random delay is added to the transmission of broadcast packets. The Advertisement contains the value of $FIB[name]$ and the $nodeID$ of the producer, also named $producerID$, hence all intermediate CRs update their FIBs accordingly.

The forwarding of Interests relies on the $name$ of the message, the FIB and the RT of the CR. According to Algorithm 3, upon receiving an Interest, the CR first checks the $CS[name]$; if this is not $null$, i.e., this node has the desired item in its CS, it responds with a Data packet. Otherwise, the CR searches for a producer in $FIB[name]$; if no

Algorithm 3
Handle an Interest packet for $name$ received from $prevHop$

$content \leftarrow CS[name]$
if $c \neq null$ **then**
  $forward(prevHop, content)$
else
  $nodeIDArray \leftarrow FIB[name]$
  if $nodeIDArray == null$ **then**
    $drop\ Interest$
  else
    $PIT[name].append(prevHop)$
    $nodeID \leftarrow select\_from(nodeIDArray)$
    $nextHop \leftarrow RT[nodeID]$
    $send\_interest(nextHop, name)$
  end if
end if

Algorithm 4
Handle a Data packet identified by $name$ that contains $data$)

$cachingStrategy(name, data)$
$nexthops \leftarrow PIT[name]$
$foreach\ hop \in nexthops$
  $forward(name, data, hop)$

producer is found, it drops the Interest packet, otherwise it selects a $producerID$ included in $FIB[name]$. FIB can store multiple producers for a single content, hence the selection process can implement different strategies, e.g., a destination $producerID$ can be selected randomly, or based on a resource optimization strategy [29]. Having selected the $producerID$ from the $FIB[name]$, the CR forwards the Interest to the $RT[producerID]$, i.e., to the next hop towards the $producerID$ as retrieved by the routing table, and updates the PIT accordingly. Hence, it is observed that an Interest packet can be satisfied by a node other than the $producerID$, which is one of the key functions of ICN architectures.

Data packets are forwarded to the node that requested them by following the reverse path of the Interest packet, using the PIT entries (Algorithm 4). Therefore, the same breadcrumb approach of NDN is used. Moreover, and similar to NDN, Interests can be merged creating multicast delivery trees and a node that forwards a Data packet may as well cache it in order to respond to a future Interest packet, thus supporting all the native functions of NDN.

3.4. Tunable parameters and application-specific strategies

Our solution can be fine-tuned to meet the requirements of multiple, diverse application scenarios. Application developers can adapt our approach by considering alternative design choices for the following.

- **Routing protocol selection** Our solution requires the existence of a routing table which indicates the next hop towards a destination, however it does not dictate how this table is populated. It is noted that the RT data structure does not prevent multi-path routing; hence routing solutions that map a destination to multiple next hops can also be considered.
- **Congestion control and transport protocols** Our design does not impose any particular congestion control or transport protocol, being compatible with advanced NDN-specific transport and congestion control protocols [30].
- **Caching strategies** Caching is an integral part of ICN architectures and various design choices can be considered for this function [31].
- **Interest forwarding strategies** As already discussed, a FIB entry may contain multiple $nodeID$s (e.g., in the case of multiple sources). Our implementation selects the most recent entry, however, other more advanced strategies can be considered [29].
- **Timer values** Timers configure many important functions, such as, Advertisement periods, the lifetime of PIT and FIB, retransmission timers. These timers affect significantly the performance, especially in the presence of heavy mobility.
- **Security solutions** In this paper, we do not consider any security threat. Of course this is not always a realistic assumption in real world scenarios. Our solution is compatible with a number of security solutions used in the context of ICN (e.g., in [27] we used Identity-Based Proxy Re-Encryption [32], in order to protect content confidentiality and integrity, as well as in order to enable content hosting by authorized nodes only). Furthermore, the Babel security extensions (such as Babel hashed message authentication code cryptographic authentication [33]) are compatible with our approach.

4. Evaluation

4.1. Evaluation setup

Our evaluation is conducted using the Mininet Wi-Fi emulator, a software that creates networks of virtual mobile nodes with wireless links. Mininet nodes execute the services of the host operating system, such as the TCP/IP networking stack, thus offering increased realism compared to simulators.

In order to deliver a fair comparison of the different solutions, we implement a custom software tool that supports full control over all designs.[6] Our tool constitutes a "cross-architecture" template for implementing different network stacks where the key-functions of each design are implemented as modules of the template, e.g., cache module, routing module, forwarding module, producer/receiver module.[7] Every design is materialized by combining different modules from a shared pool of modules, e.g., use different forwarding and routing modules but the same application modules as well as the same overlay module; all designs perform over a UDP network overlay, where the $nodeId$ of a node is its IPv4 address. The reuse of "mutual" modules of these designs avoids the performance biases due to implementation specifics of each design's prototype implementation, such as code design and optimization, different programming languages, user or kernel operation etc. The results of our experiments validate the known performance trends of the evaluated designs, hence we argue that the means of our evaluation are reliable.

4.2. Solutions

We compare our design against: (i) *IP-Babel*: an IP solution that uses Babel and a reactive content discovery service, (ii) *proNDN*: a proactive NDN-based solution and (iii) *E-CHANET*: a reactive NDN-based solution. We choose to compare our hybrid solution against IP-Babel, in order to isolate and, in turn, highlight the performance gains offered by ICN content delivery functions. We also compare our design with a proactive and a reactive design for NDN MANETs in order to verify that our hybrid design reduces the control plane overhead, thus enhancing scalability.

[6] The source code of our implementation will be provided with the publication of the paper.

[7] We implement the features of the designs that we discuss in this paper instead of the complete configuration toolset, e.g., the implemented FIB entry includes a content name and a list of faces but does not include the *Origin, Cost, Child_Inherit, Capture* fields that are defined in the NDN documentation.

4.3. Metrics

In the evaluation, we study the following metrics:

- Successful content **deliveries**: This metric shows the efficiency of a solution in delivering content. It counts the total number of Data packets that were successfully delivered to all consumers. We present it as the percentage of the highest score measured across all designs.
- Network **traffic**: This metric shows the traffic footprint of the evaluated solution in terms of bandwidth resources. It counts the cumulative number of packets received by all network nodes during an experiment. We present it as the percentage of the highest score measured across all designs.
- **Cached data**: This metric indicates the performance of in-network caching. It counts the total number of chunk requests that were served by all on-path caches. We present it as the percentage of the successful data deliveries.

We also introduce a derived metric, which clarifies the performance comparison of the selected solutions:

- **Bandwidth efficiency**: This metric quantifies the ability of a solution to handle mobility while remaining inexpensive. It corresponds to the ratio of successful content deliveries to the overall network traffic; an 0.1 value shows that 10 packets were pushed in the network in order to deliver 1 Data packet. This metric can take values in the range of $[0 \dots 1)$, hence when the ratio is near 1, it is an evidence of increased effectiveness. On the other hand, when the ratio is near 0, this metric suggests scalability; the network links are saturated with control plane traffic and, eventually, collapse due to congestion.

4.4. Software configuration

Unless stated otherwise, each experiment is set up as follows. We consider a content catalog consisting of 50 items each of which is composed of 30 chunks.[8] Content items are (permanently) stored in producer nodes and may be (temporally) cached by an intermediate node. All ICN-based nodes are equipped with fixed-size LRU caches that can hold 10% of the overall content catalog. Clients request content items chunk-by-chunk with a rate of 4 chunks/s. Finally, in terms of content popularity, consumer requests follow the Zipf distribution (with $\alpha = 1$) where the sequence of transfers is random following the uniform distribution.

In the beginning of the experiment, all consumers start sending chunk requests in parallel. The experiment stops after a pre-determined time period, which is equivalent to the minimum time that would be needed to deliver all content items in an ideal network; this time is calculated by dividing the maximum number of chunks a consumer can request by the transfer rate of requests.

The range of wireless links is set to 75 m and the trajectory of mobile nodes is either user-defined (with 5 m/s velocity) or random based on the "RandomDirection" model of Mininet Wi-Fi (with 5 m/s maximum velocity). The interval of Babel "hello" messages and of proNDN Advertisements is 4 s, which is the default value of the Babel version specified in [6]. To enhance the integrity of our results, all runs are tested over the exactly same network conditions, including random node mobility. The results present a 95% confidence interval with a 4% margin of error. In the following, we present our results for two simulated scenarios, i.e., one with mobile consumers only, and another with mobile consumers and producers.

[8] A chunk in ICN refers to an autonomous fragment of information, which, in most cases, fits within a network packet.

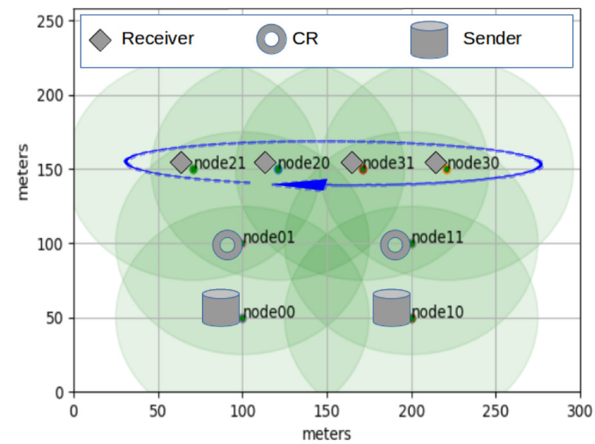

Fig. 1. Topology used in experiment of Section 4.5. Mobile content consumers move continuously, with equal velocity between points (50,150) and (250,150); producers and CR are stationary.

Table 1
Evaluation results for a network topology consisting of mobile consumers and stationary nodes. Deliveries are presented as a percentage of the whole content catalog. Network traffic is presented as a percentage of NDN (which is the solution with the maximum traffic).

	IP-Babel	proNDN	E-CHANET	Hybrid (Cache 0%)
Deliveries (%)	34	78	54	46 (38)
Traffic (%)	6	100	17	7 (13)
Cached data (%)	–	40	37	53 (–)
Deliveries ratio (10^{-2})	4	1	4	8 (6)

4.5. Scenario 1: Mobile consumers, stationary producers

In this scenario we use the topology illustrated in Fig. 1. This topology includes two static producer nodes (node00 and node10) each hosting the entire content catalog and four mobile consumer nodes that move with identical velocity and direction between points (50, 150) and (250, 150). Given the wireless links range, the producers are reachable through two static intermediate nodes (node01 and node11) that can act as caching points. While consumers move, they either receive content from other producers (as it happens with proNDN and E-CHANET) or from the same producer but over a different path (as in the case of IP-Babel and Hybrid).

The measured metrics are displayed in Table 1, where the columns represent the different solutions and the rows represent different metrics. We observe that the our solution provides the best utilization factor since the relation between control plane and data plane is better that any other solution. The worst score in the same metric is seen in the proNDN solution. In order to interpret the utilization results, we study the primitive metrics. First, the highest number of deliveries are presented in proNDN and E-CHANET, implying that the NDN content-based mobility mechanisms outperform the IP node-based techniques in adapting to mobility. Additionally, although our solution and IP-Babel share the same node-mobility handling mechanism, our solution delivers 35% more messages by exploiting the native functions of ICN. In particular, we ran our solution with caching disabled only to discover that its performance is degraded by 24% but remains 11% higher than IP-Babel, thus highlighting the individual gains of caching and multisource, respectively. Finally, we see that proNDN and E-CHANET create 14.6 and 2.5 times the traffic produced by our solution, respectively. In proNDN, the cost for managing node- and content-mobility is intense since producers flood the entire content catalog of 50 items every 4 s. Notice that, the grouping of Advertisements will not

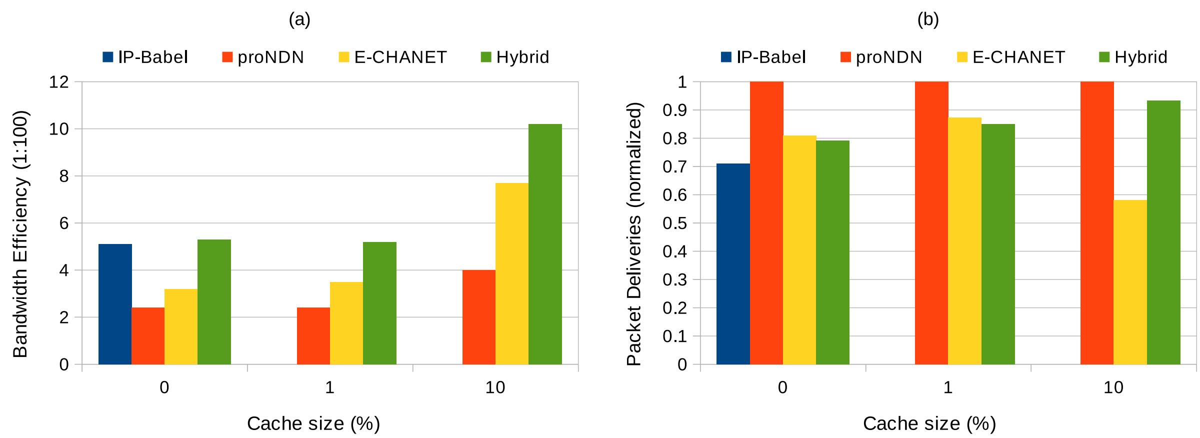

Fig. 2. Performance of different designs for three cache size ratios: 0, 1 and 10%. Subfigure (a) plots the bandwidth efficiency and subfigure (b) plots the number of overall packet deliveries normalized to the performance of the design with the most deliveries.

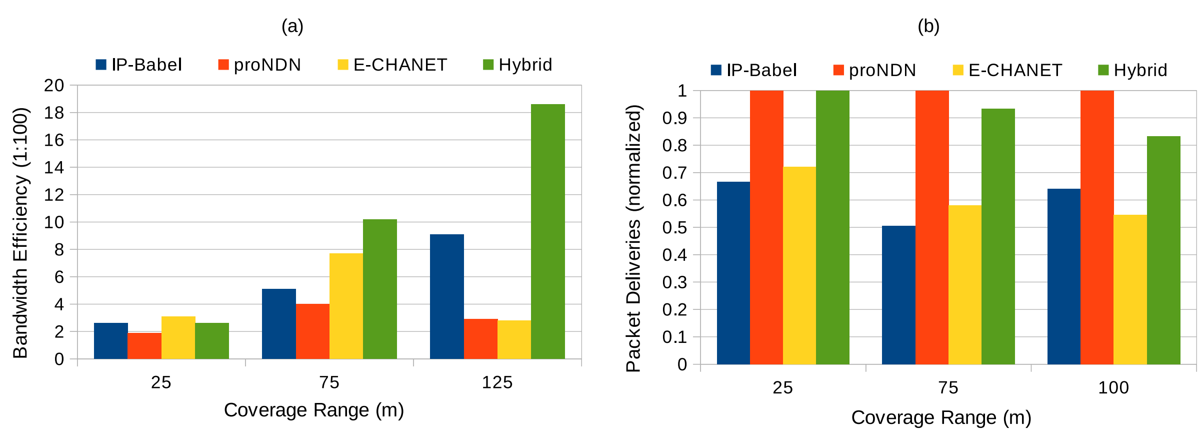

Fig. 3. Performance of different designs for three different coverage ranges: 25, 75 and 125 m. Subfigure (a) plots the bandwidth efficiency and subfigure (b) plots the number of overall packet deliveries normalized to the performance of the design with the most deliveries.

help since Advertisements are named NDN items that require a unique identifier in the header of the network packet. In addition, advertising a designated prefix instead of flat content items is not advisable because it is incompatible with multisource. In case of E-CHANET, we observe an increased traffic footprint due to the inefficiency of controlled flooding to exterminate packet duplicates during Interest flooding. The exploited controlled flooding technique, which is driven by hop counters, misbehaves when the paths to the producers have equal hops. Consequently, our approach outperforms the other solutions, maintaining (practicably) low control plane overhead while enhancing content delivery through in-network caching, multisource and multicast.

4.6. Scenario 2: Mobile consumers and producers

We now investigate performance in a more challenging scenario, where the network consists exclusively of mobile nodes that receive and send content simultaneously. In this scenario, we deploy 10 nodes that move randomly in a 300 × 300 m area. Each node produces 5 individual content items and consumes numerous according to the previously mentioned popularity distribution. This scenario sheds light on the ICN gains during network partitioning, a rather challenging issue of MANETs of particular importance in delay-tolerant networking scenarios.

4.6.1. Cache capacity

We first measure our metrics as a function of the cache capacity of network nodes by experimenting with three different sizes: 0, 1 and 10% of total content catalog. When cache capacity is set to 0%, the cache is disabled, whereas when the size is 10%, we can assume that LRU caches will store one tenth of the most popular network content. We present the bandwidth efficiency and the number of deliveries in Fig. 2. As expected, we find that the performance of all designs is enhanced as cache capacity increases, since the cached replies reduce the forwarding path and consequently the network traffic. Additionally, caching constitutes a form of multisource that addresses the network partitioning effect. Again, our hybrid design exhibits the highest bandwidth efficiency for all cache sizes. Nevertheless, it is also seen that the gains of caching are evident for large cache sizes and that the usefulness of small LRU caches is limited. In order to enhance redundancy elimination through in-network caching other strategies, that outperform LRU, must be considered [34].

4.6.2. Network connectivity

We now investigate our metrics as a function of the network connectivity degree by experimenting with three different link coverage ranges: 25, 75 and 125 m. When the range of wireless links is 25 m, network partitioning is prevalent and nodes are often isolated, however this situation is acceptable to delay-tolerant applications. When range is 125 m, the network graph is coherent. We present the bandwidth efficiency in Fig. 3(a), while the number of deliveries offered by each design is illustrated in Fig. 3(b). We see that the performance of the proactive node-based designs (IP-Babel and Hybrid) increases as the connectivity increases, thus showing that a stable network graph offers more deliveries while keeping control plane overhead low. On the contrary, the content-based designs perform best when connectivity is average, exhibiting mild network partitioning. The bandwidth efficiency decreases in highly partitioned networks, where deliveries are

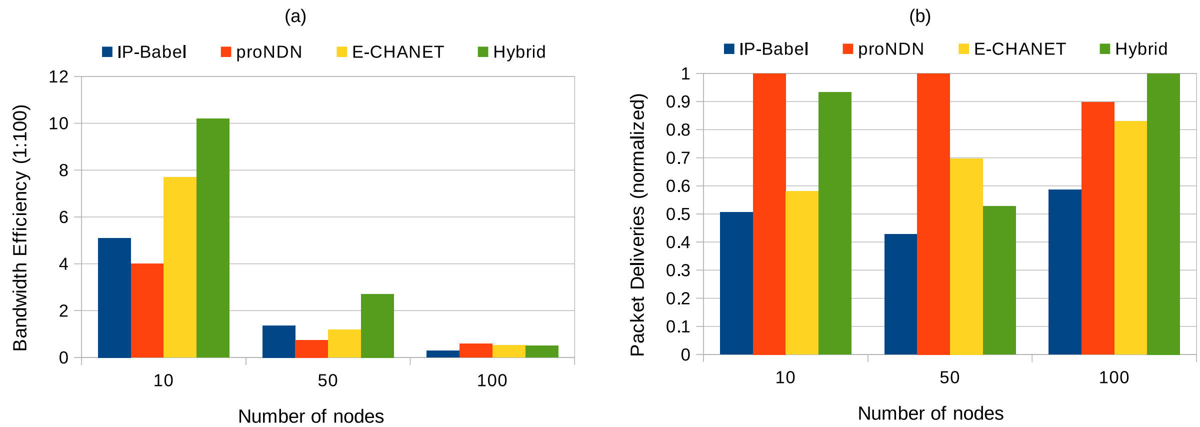

Fig. 4. Performance of different designs for three different node populations: 10, 50 and 100. Subfigure (a) plots the bandwidth efficiency and subfigure (b) plots the number of overall packet deliveries normalized to the performance of the design with the most deliveries.

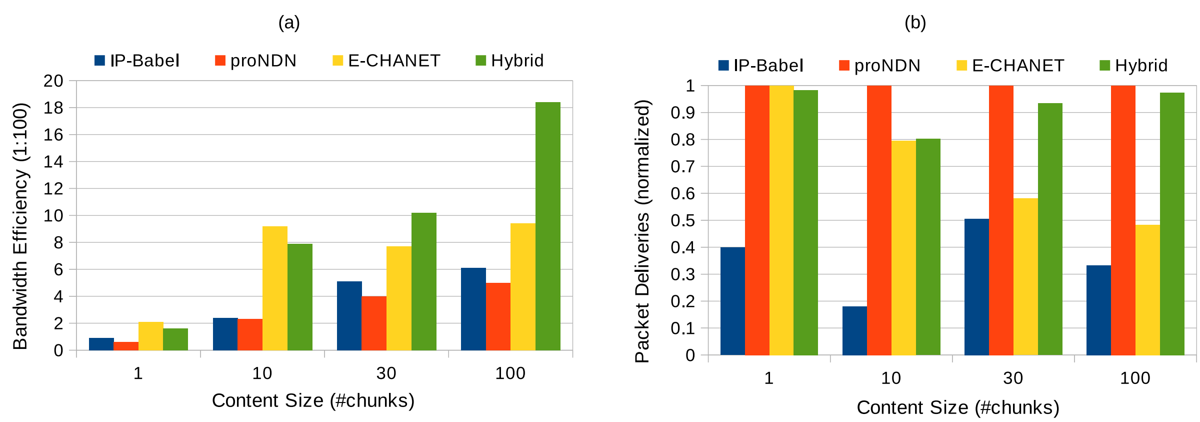

Fig. 5. Performance of different designs for four different content sizes: 1, 10, 30 and 100 chunks. Subfigure (a) plots the bandwidth efficiency and subfigure (b) plots the number of overall packet deliveries normalized to the performance of the design with the most deliveries.

sparse compared to content-based flooding, and in coherent networks, where the content-level flooding reaches the entire network, thus dominating the number deliveries. Our solution is the most efficient in case of 75 and 125 m range and maximizes deliveries when the link coverage is 25 m, when the E-CHANET is the most efficient.

4.6.3. Number of nodes

We examine our metrics as functions of the number of nodes in our network, experimenting with three different population sizes: 10, 50 and 100. Given fixed total area of network, the larger number of nodes is expected to cause longer data dissemination paths and increased control plane overhead. In order to enhance the comparability of the setups, we deploy different content catalog (and cache) sizes that are proportional to the node population, namely 50, 250 and 500 content items for 10, 50 and 100 nodes respectively. We present the bandwidth efficiency and the number of deliveries in Fig. 4. We observe that the performance of all designs is highly affected by the number of nodes, since the bandwidth efficiency for all designs falls dramatically as this number increases. We consider this to be a result of the increased control plane traffic that rises by approximately 400% when 100 nodes are deployed (compared to when 10 nodes are deployed). Thereafter, our results verify that the need for an efficient routing solution is magnified in the presence of populous networks. While all designs perform similarly in case of 100 nodes, in the other cases our hybrid algorithm delivers significant performance gains.

4.6.4. Transfer size

Finally, we explore our metrics as functions of the number of chunks per content item by experimenting with four different content sizes: 1, 10, 30 and 100 chunks per object. Given fixed chunk requests, larger files require longer transfers, hence mobility events are more likely to challenge service continuity. We present the bandwidth efficiency and the number of deliveries in Fig. 5. We observe that the performance of proactive designs increases as the transfer duration increases, showing that in larger files the cost for content discovery and mobility management is relatively low. In contrast, reactive E-CHANET presents a performance spike when content items consist of approximately 10 chunks, where it outperforms the other designs, but performs relatively poorly in other cases. The results reveal a fundamental difference of proactive and reactive solutions, which can be analyzed in three distinct phases with regard to content size.

First, when content consists of a single chunk, E-CHANET is similar to an enhanced flooding protocol. The first chunk request is flooded in order to discover the producer's location, which is stored locally for future downloads. Thereafter, flooding takes place only during the first time a content is downloaded and when the producer is unreachable, thus offering the highest bandwidth efficiency in this setup. In the second phase, content consists of roughly 10 chunks that, according to the 4 chunks/s request rate, lead to a 2-3 s transfer duration. In this timeframe, network changes are not very likely, thus service continuity is seldom affected and E-CHANET performs again better that the others. In phase three, where transfers are long enough to be disrupted, the performance of E-CHANET is heavily affected by multiple topology shifts, thus presenting significant performance degradation. The data plane shrinks due to late detections of mobility events and E-CHANET delivers on average 50% less data than proNDN. Our hybrid design performs slightly worse than E-CHANET for 1 and 10 chunks, where it achieves 24% and 20$ less efficient, but outperforms E-CHANET in case of 30 and 100 chunks, where it increases the bandwidth efficiency by 32% and 97%, respectively.

5. Conclusions

In this paper we presented a novel IP-NDN hybrid design for MANETs. Our solution improves content delivery while maintaining low control traffic overhead. We enhanced the efficiency of the NDN architecture by exploiting the BABEL routing protocol and introduced an information-centric design that does not require any modification to the IP substrate and co-exists with legacy nodes, thus delivering an incrementally deployable NDN solution. Through experimentation, we explored the ratio of delivered data to induced network traffic, verifying the trade-off between performance and scalability and that there is not a one-size-fits-all solution. Nonetheless, our hybrid solution outperforms pure IP and NDN-based solutions in the vast majority of cases investigated with our experiments. The only cases were our solution was the second most efficient design (although delivering an equal amount of data with the first) were: (a) when the transfer is too short to be affected by a topology change and (b) when network partitioning is prevalent. These operating regimes are the subject of future work. It became evident that IP MANETs have a lot to gain from a content-centric networking approach, while NDN MANETs can substantially enhance their scalability with an addition of a node-based routing protocol.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: We are members of the Mobile Multimedia Laboratory in Athens University of Economics and Business, Dept. of Informatics, School of Information Sciences and Technology.

CRediT authorship contribution statement

Yiannis Thomas: Conceptualization, Methodology, Software, Writing - original draft. **Nikos Fotiou:** Conceptualization, Methodology, Writing - original draft. **Stavros Toumpis:** Conceptualization, Writing - review & editing. **George C. Polyzos:** Conceptualization, Supervision, Writing - review & editing.

Acknowledgment

The research work was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.), Greece under the "First Call for H.F.R.I. Research Projects to support Faculty members and Researchers and the procurement of high-cost research equipment grant" (Project Number: 352, Project Title: Wireless Mobile Delay-Tolerant Network Analysis and Experimentation). The procurement of equipment was supported by RC-AUEB, Greece (contract number EP-3164-01/00-01)

References

[1] D. Trossen, M. Sarela, K. Sollins, Arguments for an information-centric internetworking architecture, ACM SIGCOMM Comput. Commun. Rev. 40 (2010).
[2] X. Liu, Z. Li, P. Yang, Y. Dong, Information-centric mobile ad hoc networks and content routing: A survey, Elsevier Ad Hoc Netw. 58 (2017).
[3] G. Xylomenos, C.N. Ververidis, V.A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K.V. Katsaros, G.C. Polyzos, A survey of information-centric networking research, IEEE Commun. Surv. Tutor. 16 (2014).
[4] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, J. Wilcox, Information-centric networking: seeing the forest for the trees, in: Proceedings of the 10th ACM Workshop on Hot Topics in Networks, ACM, 2011, p. 1.
[5] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, k. claffy, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang, Named data networking, ACM SIGCOMM Comput. Commun. Rev. 44 (2014).
[6] J. Chroboczek, The Babel Routing Protocol, Vol. 6126, RFC, 2011.
[7] J. Macker, Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations, RFC, 2501.
[8] L. Gupta, R. Jain, G. Vaszkun, Survey of important issues in UAV communication networks, IEEE Commun. Surv. Tutor. 18 (2015).
[9] C.N. Ververidis, G.C. Polyzos, Service discovery for mobile ad hoc networks: a survey of issues and techniques, IEEE Commun. Surv. Tutor. 10 (2008).
[10] B. Wissingh, et al., Information-Centric Networking (ICN): CCNx and NDN Terminology, RFC-draft, 2019.
[11] A.M. Hoque, S.O. Amin, A. Alyyan, B. Zhang, L. Zhang, L. Wang, NLSR: named-data link state routing protocol, in: Proceedings of the 3rd ACM SIGCOMM Workshop on Information-Centric Networking, 2013, pp. 15–20.
[12] V. Lehman, A. Gawande, B. Zhang, L. Zhang, R. Aldecoa, D. Krioukov, L. Wang, An experimental investigation of hyperbolic routing with a smart forwarding plane in NDN, in: 2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS), IEEE, 2016, pp. 1–10.
[13] C. Tsilopoulos, G. Xylomenos, Y. Thomas, Reducing forwarding state in content-centric networks with semi-stateless forwarding, in: IEEE INFOCOM 2014-IEEE Conference on Computer Communications, IEEE, 2014, pp. 2067–2075.
[14] M. Amadeo, A. Molinaro, G. Ruggeri, E-CHANET: Routing, forwarding and transport in information-centric multihop wireless networks, Elsevier Comput. Commun. 36 (2013).
[15] P. Mendes, R.C. Sofia, J. Soares, V. Tsaoussidis, S. Diamantopoulos, C.-A. Sarros, Information-centric routing for opportunistic wireless networks, in: Proceedings of the 5th ACM Conference on Information-Centric Networking, ACM, 2018, pp. 194–195.
[16] M. Meddeb, A. Dhraief, A. Belghith, T. Monteil, K. Drira, S. Gannouni, AFIRM: Adaptive forwarding based link recovery for mobility support in NDN/IoT networks, Future Gener. Comput. Syst. 87 (2018) 351–363.
[17] B. Nour, K. Sharif, F. Li, S. Biswas, H. Moungla, M. Guizani, Y. Wang, A survey of internet of things communication using ICN: A use case perspective, Comput. Commun. (2019).
[18] H. Khelifi, S. Luo, B. Nour, A. Sellami, H. Moungla, F. Naït-Abdesselam, An optimized proactive caching scheme based on mobility prediction for vehicular networks, in: 2018 IEEE Global Communications Conference (GLOBECOM), IEEE, 2018, pp. 1–6.
[19] H. Khelifi, S. Luo, B. Nour, H. Moungla, Y. Faheem, R. Hussain, A. Ksentini, Named data networking in vehicular ad hoc networks: State-of-the-art and challenges, IEEE Commun. Surv. Tutor. (2019).
[20] Z. Rezaeifar, J. Wang, H. Oh, S.-B. Lee, J. Hur, A reliable adaptive forwarding approach in named data networking, Future Gener. Comput. Syst. 96 (2019) 538–551.
[21] D. Saxena, V. Raychoudhury, N-FIB: Scalable, memory efficient name-based forwarding, J. Netw. Comput. Appl. 76 (2016) 101–109.
[22] S. Muralidharan, A. Roy, N. Saxena, MDP-IoT: MDP based interest forwarding for heterogeneous traffic in IoT-NDN environment, Future Gener. Comput. Syst. 79 (2018) 892–908.
[23] G. Xylomenos, Y. Thomas, X. Vasilakos, M. Georgiades, A. Phinikarides, I. Doumanis, S. Porter, D. Trossen, S. Robitzsch, M.J. Reed, et al., IP over ICN goes live, in: 2018 European Conference on Networks and Communications (EuCNC), IEEE, 2018, pp. 319–323.
[24] G. Carofiglio, L. Muscariello, J. Augé, M. Papalini, M. Sardara, A. Compagno, Enabling ICN in the internet protocol: Analysis and evaluation of the hybrid-ICN architecture, in: Proceedings of the 6th ACM Conference on Information-Centric Networking, ACM, 2019, pp. 55–66.
[25] A. Detti, D. Tassetto, N.B. Melazzi, F. Fedi, Exploiting content centric networking to develop topic-based, publish–subscribe manet systems, Ad Hoc Netw. 24 (2015) 115–133.
[26] Y. Thomas, C. Tsilopoulos, G. Xylomenos, G.C. Polyzos, Accelerating file downloads in publish–subscribe internetworking with multisource and multipath transfers, in: WTC 2014; World Telecommunications Congress 2014, VDE, 2014, pp. 1–6.
[27] E. Aliaj, G. Dimaki, P. Getsopoulos, Y. Thomas, N. Fotiou, S. Toumpis, I. Koutsopoulos, V. Siris, G.C. Polyzos, A platform for wireless maritime networking experimentation, in: Proc. of the Global Information Infrastructure and Networking Symposium (GIIS), 2018.
[28] C. Anastasiades, J. Weber, T. Braun, Dynamic unicast: Information-centric multi-hop routing for mobile ad-hoc networks, Comput. Netw. 107 (2016) 208–219.
[29] C. Yi, A. Afanasyev, I. Moiseenko, L. Wang, B. Zhang, L. Zhang, A case for stateful forwarding plane, Elsevier Comput. Commun. 36 (2013).
[30] Y. Ren, J. Li, S. Shi, L. Li, G. Wang, B. Zhang, Congestion control in named data networking–a survey, Elsevier Comput. Commun. 86 (2016).
[31] M. Zhang, H. Luo, H. Zhang, A survey of caching mechanisms in information-centric networking, IEEE Commun. Surv. Tutor. 17 (2015).
[32] G. Ateniese, K. Fu, M. Green, S. Hohenberger, Improved proxy re-encryption schemes with applications to secure distributed storage, ACM Trans. Inf. Syst. Secur. 9 (2006) 1–30.
[33] D. Ovsienko, Babel Hashed Message Authentication Code (HMAC) Cryptographic Authentication, Vol. 7948, RFC, 2014.
[34] I. Abdullahi, S. Arif, S. Hassan, Survey on caching approaches in information centric networking, Elsevier J. Netw. Comput. Appl. 56 (2015).