



Exploiting Knowledge for Better Mobility Support in the Future Internet

Zhongda Xia¹ · Yu Zhang^{1,2} · Binxing Fang^{1,2}

Accepted: 27 September 2021 / Published online: 4 February 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

The ever-increasing mobile traffic calls for efficient mobility support at a global scale. Supporting “seamless” communication with network entities whose network location constantly changes, i.e., the mobility support problem, is extremely challenging for IP networking due to its host-centric communication model. Information-centric networking (ICN), as exemplified by the Named Data Networking (NDN) network architecture, offers new opportunities for mobility support. We identify a common design space for providing IP and NDN mobility support where solutions track the changing network locations of mobile network endpoints, and find that available design choices have been exhausted in this design space, leaving no room for substantial performance improvement. Recognizing this limitation, this paper proposes two novel knowledge-driven mobility support approaches to comprehensively improve mobility support performance. Such approaches exploit knowledge such as network topology and movement trajectory to tweak the network for better mobility support performance. A cross-architectural quantitative evaluation framework covering two communication scenarios and 5 quantifiable metrics is proposed to evaluate mobility support performance. Experiment results show that the knowledge-driven approaches significantly improve mobility support performance, demonstrating the potential of the knowledge-driven vision for providing better mobility support.

Keywords Mobility support · Named data networking · Knowledge-driven networking · Quantitative evaluation framework

1 Introduction

Mobility support refers to the ability to carry out “seamless” communication when network entities (e.g., subnets, hosts, applications or data) constantly change their network locations. As more and more mobile devices connect to the Internet, popular applications such as video conferencing and augmented-reality (AR), as well as emerging mobile networking scenarios such as smart cities [2–4] and

intelligent traffic system (ITS) [2, 5, 6] call for efficient and scalable mobility support.

IP, the de facto network architecture of the Internet, adopts the host-centric communication paradigm, and completely lacks native support for mobility. IP mobility support solutions must maintain the persistence and reachability of IP addresses to prevent the host-to-host connection from breaking. Named Data Networking (NDN) [7] is a proposed future Internet architecture following the data-centric communication paradigm. NDN is more friendly to mobility by natively supporting the mobility of network endpoints requesting data (consumers). Nevertheless, NDN still requires additional measures to support the mobility of network endpoints providing data (producers).

This paper examines the mobility support solutions for IP and NDN, and identifies a common design space where solutions track the network location of a mobile network entity. We further find out that such solutions can be generalized as combinations of four basic cross-architectural mobility support approaches: proxy-based, resolution-based, routing-based, and trace-based. These four approaches represent different ways to balance between

✉ Yu Zhang
yuzhang@hit.edu.cn

Zhongda Xia
xiazhongda@hit.edu.cn

Binxing Fang
bxfang@hit.edu.cn

¹ Harbin Institute of Technology, Harbin, Heilongjiang, China

² Peng Cheng Laboratory, Shenzhen, Guangdong, China

user experience and network overhead, and corresponding solutions are subject to the same performance barriers. Recognizing this limitation, this paper proposes to expand the mobility support design space with *knowledge-driven (KD)* approaches. Such approaches exploit knowledge about the network, user, application, etc., to comprehensively improve mobility support performance over various combinations of network architectures and mobility support solutions. Two KD approaches, namely *Topology-driven Intermediate Placement (TIP)* and *Trajectory-driven Reachability Update (TRU)*, are proposed to optimize forwarding paths and reduce signaling overhead without inducing additional costs.

To evaluate mobility support performance across network architectures, and thus examine the efficacy of KD approaches, a cross-architectural quantitative evaluation framework is proposed. Under two communication scenarios, 5 quantifiable metrics reflecting user experience and network overhead are defined. The calculation formulae for these metrics are deduced by analyzing the key differences among approaches and architectures. By feeding network topology and movement track to the formulae, mobility support performance can be quantified via numerical simulations. Experiment results show that the proposed KD approaches are superior according to all 5 metrics, preliminarily proving these approaches effective, and further indicating the potential of the knowledge-driven vision at providing better mobility support in future networks.

This paper completes our previous work [8] presented in the 2nd EAI International Conference on Artificial Intelligence for Communications and Networks (EAI AICON 2020). In the previous work, we provided a rough design of KD approaches, and preliminary results on cross-architectural mobility support performance evaluation. This paper matures the design of KD approaches, including the proposal of a network architecture to support such approaches. This paper also provides extensive details on various aspects, and conducts extensive experiments under more topology and mobility settings.

2 Background

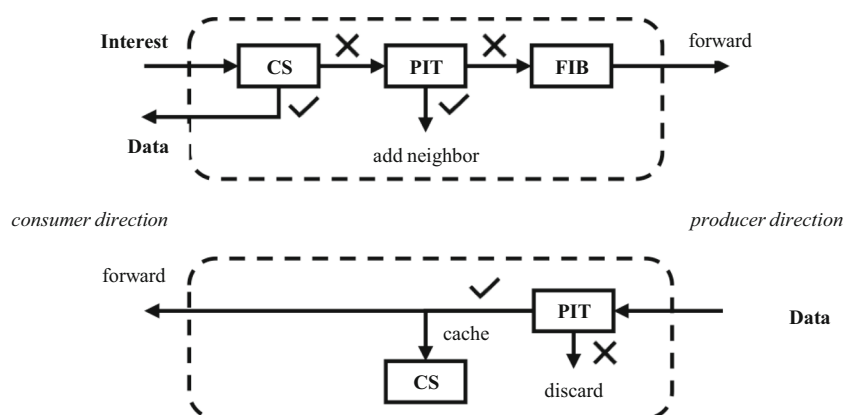
2.1 Mobility Support in IP and NDN

IP was born in an era where most, if not all, network endpoints are immobile. With the absence of mobility, network endpoints are assigned IP addresses according to their network locations to achieve scalable IP address routing. If a network endpoint changes its network location, it would by default be assigned a new IP address, and any ongoing communication (e.g., TCP connections) using the previous IP address would break. To provide mobility support in IP, a mobile network endpoint must be constantly reachable via the same IP address, such that upper layer activities are unaffected by mobility. Because IP routing alone cannot achieve the above goal in a scalable way, it is generally believed that IP provides no mobility support.

Proposed ICN architectures, however, consider modern needs of network applications, and generally provide built-in mobility support considering the ever-increasing traffic from/to mobile endpoints. For example, NDN natively supports the mobility of consumers (network endpoints that retrieve data) by adopting a pull-based communication model. As shown in Fig. 1, an Interest (request for named data) sent by a consumer leaves a “breadcrumb trail” in Pending Interest Table (PIT) for the corresponding Data (named data) to follow back. If a consumer changes its network location before Data comes back, the present “breadcrumb trail” would become broken because it can no longer lead Data to the consumer. No additional mechanisms are required to deal with such mobility, a consumer simply needs to retransmit the Interest after the relocation (the change of network location) to fetch Data back.

On the contrary, the mobility of producers (network endpoints that serve data) is not natively supported in NDN. Because NDN deals with data, not hosts, to support producer mobility essentially means that data need to be constantly available under the same names. Thus the solution space for NDN producer mobility is a superset

Fig. 1 NDN forwarding



of the solution space for IP mobility, in other words, guaranteeing reachability to producers is adequate but not absolutely necessary for supporting NDN producer mobility. As pointed out by Zhang et al. [9], because NDN communication decouples data from its producer, NDN producer mobility support may be provided by proactively moving or disseminating data to make data easily retrievable.

For our preliminary effort at applying the knowledge-driven vision to improving mobility support, we focus on the common design space for IP and NDN producer mobility support solutions, namely the *host-centric* design space, where solutions provide *host-centric mobility support* by tracking mobile endpoints’ changing network locations, thus guarantee successful packet (i.e., IP and Interest packets) delivery using persistent network layer names (i.e., IP address and NDN name/name prefix). Specifically, we attempt to identify basic mobility support approaches in the host-centric design space, analyze their inherent limitations, and then propose to overcome such limitations with knowledge-driven mechanisms.

2.2 Basic Host-Centric Mobility Support Approaches

By examining proposed solutions for IP and NDN producer mobility, we identify four basic host-centric mobility support approaches, namely proxy-based, resolution-based, routing-based and trace-based approach. These approaches represent distinct ways to provide host-centric mobility support, and almost all host-centric mobility support solutions can be generalized as one or a combination of such approaches.

Each approach may be roughly broken down into two processes, namely the *reachability information update* and *packet forwarding* process. Four network elements are involved in these processes:

- Mobile endpoint (*M*): the device/host/subnet that connects to the network, its mobility is reflected by the change of PoAs.
- Mobile identifier (*ID*): the network layer name associated with a mobile endpoint, and packets carrying and forwarded according to a mobile identifier are supposed to reach the corresponding mobile endpoint.
- Correspondent endpoint (*C*): the network endpoint that sends packets to a mobile endpoint.
- Rendezvous server (*R*): an immobile network endpoint

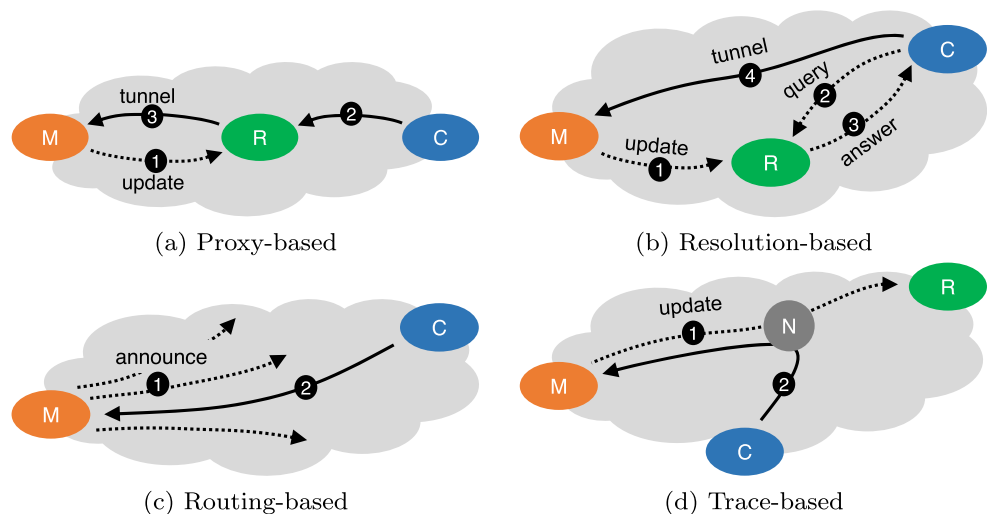
The reachability information update process maintains various forms of reachability information associated with a mobile identifier according to the corresponding mobile endpoint’s latest network location. The packet forwarding process delivers packets according to the maintained reachability information.

2.2.1 Proxy-Based Approach

In this approach, reachability information takes the shape of a globally reachable (i.e., routed) network layer name, namely a *locator*. After each relocation, a new locator is assigned to *M* at the current PoA. In the reachability information update process, *M* sends a signaling packet to *R* (Fig. 2a: step 1); this signaling packet must carry *M*’s mobile identifier *ID* and *M*’s current locator *LOC*; *R* receives this packet and then updates the reachability information associated with *ID* to *LOC*. In the packet forwarding process, *R* announces *ID* via routing beforehand as a bootstrapping procedure, so that packets destined to *M* always reach *R* first; *R* will then tunnel forward such packets to *M* using *M*’s latest locator (Fig. 2a: step 3).

The Mobile IP protocol [10] proposed by IETF is the most representative mobility support solution adopting the

Fig. 2 Host-centric mobility support approaches



proxy-based approach. In Mobile IP, the role of R is fulfilled by a Home Agent (HA), a mobile identifier is called the Home Address (HoA), and a locator is called the Care-of Address (CoA). Tunnel forwarding is performed by encapsulating the original packet from C in a new IP packet whose destination address is M 's current CoA. Mobile IP is also adapted on to NDN [11–16]. Such solutions work in a similar way to Mobile IP, except that IP packets become Interests, and IP addresses become NDN names/name prefixes. These solutions also propose new tunneling mechanisms for NDN, including prepending locator to Interest name, and Interest-in-Interest packet encapsulation /decapsulation schemes.

2.2.2 Resolution-Based Approach

This approach employs the same form of reachability information as the described proxy-based approach (Section 2.2.1). M is also assigned a new locator LOC after each location, and the reachability information update process is the same with the proxy-based approach. In the packet forwarding process, C will query R about M 's current locator, producing an exchange of packets between C and R (Fig. 2b: step 2 and 3). After learning M 's locator, C will directly tunnel packets to M .

Back to My Mac (BTMM) [17] by Apple, Inc., which has seen large scale commercial use covering millions of users, is a typical mobility support solution adopting the resolution-based approach. BTMM relies on the name resolution service provided by the Domain Name System (DNS) to maintain and provide the mappings between mobile identifiers and locators, and uses IPv6 Unique Local Address (ULA) as the network identifier of mobile endpoints. In NDN, SNAMP [18] is proposed to enhance NDN routing scalability, while also capable of providing producer mobility support via the resolution-based approach. The SNAMP design includes the specifications of a DNS-like resolution system for NDN, and the proposal to add a Forwarding Hint field to Interest packets. The Forwarding Hint field specifies an additional NDN name/name prefix that guides Interest forwarding, which provides an elegant way to perform tunnel forwarding in NDN.

2.2.3 Routing-Based Approach

In this approach, reachability information is distributed across the network. Each networking node maintains local routes for ID to determine the nexthop when forwarding packets to M . To update reachability information, M broadcasts signaling packets into the network after each relocation (Fig. 2c: step 1). As a result, routes on each node are updated to establish shortest forwarding paths for ID

from anywhere in the network. In the packet forwarding process, packets from C will follow the shortest path to M .

Note that it is generally impractical to produce optimal paths in large networks under frequent mobility, because propagating signaling packets throughout the network is rather expensive, and the time it takes for routes to converge is also non-trivial. To support mobility via this approach, proposed solutions need to balance between signaling overhead and path optimality. Connexion [19] by Boeing, Inc. takes care of the reachability of IP prefixes assigned to airplanes. Connexion directly uses the Border Gateway Protocol (BGP) to update routes. However, BGP is not designed to take care of frequent updates, and thus suffers from heavy signaling overhead, and long convergence time, as already noted above. To address this issue, WINMO [20] combines the routing-based and proxy-based approach to control signaling overhead at the cost of sub-optimal paths and a more complex protocol. Solutions have also been proposed for NDN [21, 22], which employ Interest packets for signaling purposes to implement light-weight and reactive routing protocols.

2.2.4 Trace-Based Approach

Like the routing-based approach, this approach also employs distributed local routes as the reachability information. However, only a very limited number of routes need to be updated upon each relocation. In the reachability information update process, M sends a signaling packet to R (Fig. 2d: step 1). This packet triggers each node on the shortest path between M and R to create or update the route for ID to point to M 's direction, constructing a single forwarding path for ID from R to M . In the packet forwarding process, R announces a network layer name in the bootstrapping steps to attract packets destined to M towards it,¹ when a packet meets the established forwarding path, either on its way to R (e.g., at node N as marked in Fig. 2d) or at R , the packet will then follow the forwarding path to M .

KITE [23], which is designed for NDN, is the latest representative solution adopting this approach. In the initial design of KITE, signaling Interests sent by M immediately established the forwarding path. In a later revision of the KITE design, the signaling Interest from M no longer sets up routes due to security concerns; instead, R will validate the signaling Interest and return Data if successful, and the returned Data will trigger route updates.

¹The network layer name to announce is determined by the forwarding mechanisms of a specific network architecture. For example, both IP and NDN use longest prefix matching when selecting routes, thus R should announce a prefix of ID .

2.3 Related Work

Gao et al. [24] attempts to determine the most fitting architectural approach to realize location-independent communication, which means communicating without caring about changing network locations. They performed empirical evaluation on the mobility support performance of three “puristic” mobility support approaches, which are similar to the proxy-based, resolution-based, and routing-based approach. The adopted quantitative methodology covers three quantifiable metrics: update cost, path stretch, and Forwarding Information Base (FIB) size (number of routes on each node).

Chaganti et al. [25] extends the work of Gao et al. [24]. They propose new, and more complex metrics that reflect realistic mobility performance. A discrete-event simulator is developed to collect the proposed metrics under parameterized mobility models. Based on the results, the paper analyzes how each approach trades-off a critical user experience metric against a combined cost.

Both work above tries to determine the most suitable basic approach to support location-independent communication, which is roughly equivalent to supporting mobility, aiming at providing a guiding principle for the design of a mobility-oriented network architecture. This paper aims to investigate how a new genre of network architecture driven by knowledge may further optimize mobility support performance under specific scenarios, when various underlying network architectures and mobility support approaches are adopted. Thus our evaluation framework considers specific existing network architectures and how their design choices affect mobility support performance, and also considers how different communication scenarios shape the way mobility support performance should be evaluated. Compared to [24], the proposed framework has richer expressive power due to the exploitation of expert knowledge. New metrics, approaches, and architectures can be supported by following the

reasoning workflow to distinguish critical design choices that have an impact on metric values. Compared to [25], maintaining a discrete-event simulator requires much more development efforts than using the proposed framework, which makes their method less suitable for studying novel and ever-evolving mobility support technology.

3 Knowledge-Driven Mobility Support

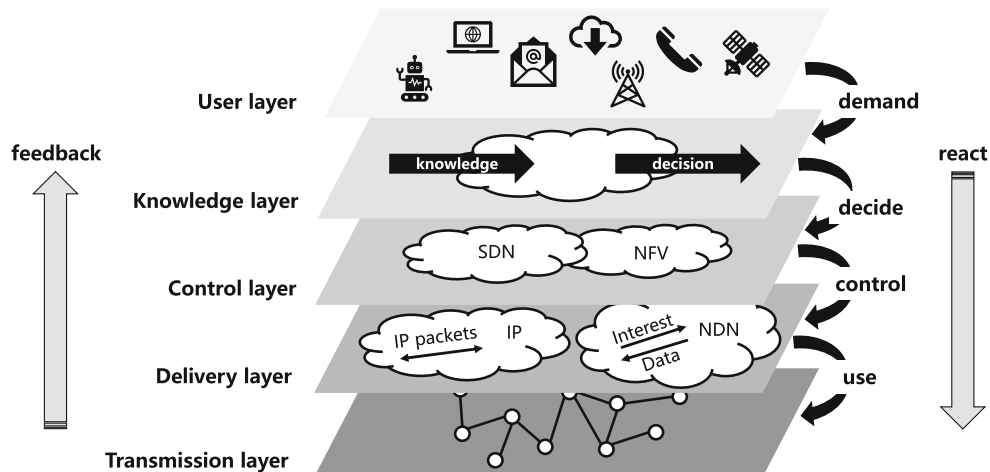
The four basic approaches represent different tradeoff decisions between user experience and network overhead, as will be later shown in Section 5. Optimizing the forwarding path to mobile endpoints would lead to either high signaling overhead (routing-based approach) or longer delays (resolution-based approach), vice versa, to control signaling overhead would lead to sub-optimal paths (proxy-based and trace-based approach). Inspired by the knowledge-driven vision, where knowledge about the network, application, user, etc. is exploited to optimize the network for stringent performance requirements, we propose to comprehensively improve mobility support performance with the help of knowledge-driven mechanisms.

3.1 A Knowledge-Driven Network Architecture

Practicing the knowledge-driven vision, knowledge-driven mechanisms need to collect knowledge, process knowledge, make decisions based on knowledge, and finally manage the network (including network services) to fulfill the goal of the decisions.

Considering such requirements, we propose a preliminary design of a *Knowledge-Driven Networking (KDN)* architecture, as shown in Fig. 3. KDN follows a layered structure consisting of four layers, stacking systems serving different purposes to achieve higher level goals. The applications layer hosts network applications that define various

Fig. 3 Illustration of the knowledge-driven network architecture



Quality-of-Service (QoS) and Quality-of-Experience (QoE) requirements. Such requirements are fed to the knowledge layer as the constraints under which optimal decisions considering network overhead are made. Such decisions are then fed to the control layer, where control-oriented network architectures/frameworks such as Software-Defined Networking (SDN) [26] and Network Function Virtualization (NFV) [27] interpret such decisions as series of management operations on the underlying networks. The delivery layer sits between the control layer and physical network to deliver packet as the control layer instructs. Delivery-oriented network architectures such as IP and NDN belong to this layer. Under this layered structure, delivery-oriented network architectures are tweaked by the actions resulting from higher level requirements, while observations about the network environment are fed as knowledge to the knowledge layer to finer-tune packet delivery behavior. Aside from offering a glimpse at how knowledge-driven mechanisms may be realized on top of existing networks, the KDN architecture also sketches out key technologies, including present and desired ones, for achieving the knowledge-driven vision.

Algorithm 1 Optimal location calculation process of TIP.

· **Input:** network topology $topo$, network locations of immobile endpoints $loc_{immobile}$, network locations of mobile endpoints loc_{mobile}

Output: optimal location of the considered intermediate (R) loc_R

```

1:  $betweennessValueMap \leftarrow$ 
   BetweennessCentrality( $topology=topo$ ,
    $sources=loc_{immobile}$ ,  $targets=loc_{mobile}$ )
2:  $loc_R \leftarrow \emptyset$ 
3:  $maxBetweenness \leftarrow \infty$ 
4: for  $node, value$  in  $betweennessValueMap$  do
5:   if  $loc_R == \emptyset$  or  $value > maxBetweenness$  then
6:      $loc_R \leftarrow node$ 
7:      $maxBetweenness \leftarrow value$ 
8: return  $loc_R$ 

```

3.2 Knowledge-Driven Mobility Support Approaches

3.2.1 Topology-Driven Intermediate Placement

Three of the four basic approaches employ an intermediate R . The location of this intermediate has significant impact on the mobility support performance. For example, for the proxy-based approach, if R is on the shortest path between C and M , the forwarding path would be optimal; if R is close to M , the signaling overhead would be small. However, network communications are highly dynamic, it is impossible to place the intermediate at an optimal

location beforehand. The Topology-driven Intermediate Placement (TIP) approach is proposed to dynamically change the intermediate's network location to optimize mobility support performance according to the changing distributions of relevant network endpoints.

TIP first figures out an optimal location for deploying the intermediate according to endpoints' latest locations as shown in Algorithm 1, and then migrates a deployed intermediate to this location. Thus TIP may enhance the performance of any host dimension approaches that rely on an intermediate in a transparent manner. With a network topology and network locations of relevant entities as input, TIP calculates the optimal location using the betweenness centrality algorithm. Specifically, TIP chooses the node with the highest betweenness centrality. Informally speaking, this node would be near most shortest paths between pairs of C s and M s. Note that in most cases the locations of mobile endpoints can only be predicted. The accuracy of such predictions varies according to the nature of movements. For example, the location of a train can be rather reliably determined given that a train travels at constant speed and follows predetermined routes, however, the locations of a pedestrian can only be roughly predicted. The prediction part concerns the collection and processing of knowledge, and detailed study is left for future work.

3.3 Trajectory-Driven Reachability Update

All four basic approaches rely on the propagation of signaling packets to update states in the network (routing-based and trace-based approach) and network services (proxy-based and resolution-based approach). The Trajectory-driven Reachability Update (TRU) approach provides a way to carry out reachability update while inducing little to none signaling overhead, which would have an optimization effect on all basic approaches. As shown in Fig. 4,

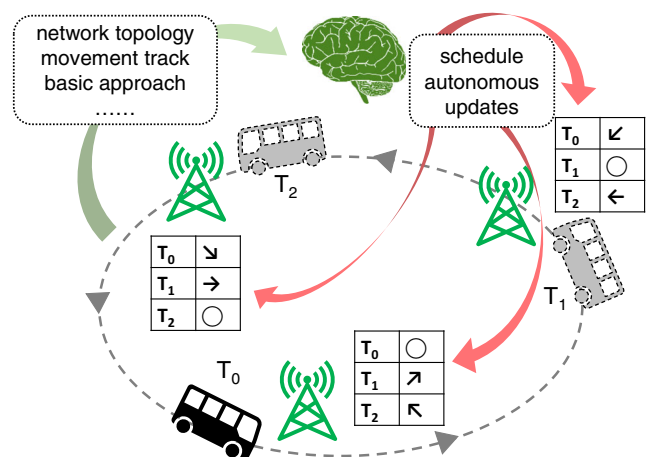


Fig. 4 Illustration of the TRU approach

based on knowledge about network topology and the movement track of a mobile endpoint, TRU allows each network node to perform reachability update in an autonomous way. For the proxy-based and resolution-based approach, R learns the timed sequence of locators for each mobile endpoint using its service, and updates locators at the specified time points. For the routing-based and trace-based approach, the required route changes upon each relocation is pre-calculated and uploaded to each node, then each node automatically updates routes at designated time.

4 A Quantitative Cross-architectural Mobility Support Performance Evaluation Framework

Given the definition of a quantifiable metric, description of a network architecture (architecture for short), and description of a mobility support approach (approach for short), the proposed quantitative evaluation framework outputs the formula for calculating the value of this metric from arbitrary network topologies and movement tracks.

To determine the formula for each metric, a reasoning workflow based on expert knowledge is adopted, as illustrated in Fig. 5. The definition of a metric is determined first in a way that is independent from architectures and approaches. The key differences and commonalities between different approaches and architectures are then extracted from their designs based on the impact on the metric in question. The differences between critical design choices are eventually reflected in the formulae for the considered approaches (if not affected by architectures) or combinations of architecture /approach, to produce different metric values under the same inputs. By following this thought process, the proposed framework may be easily expanded to support more architectures and approaches.

4.1 Inputs for Metric Calculation

Network Topology A network topology is represented by an undirected, weightless, and fully connected graph $G = (V, E)$. Each vertex in V represents a networking

node, and each edge in E represents a link between nodes. This representation is suitable for describing infrastructure network environment where networking nodes are connected with reliable point-to-point links.

Network endpoints are not included in the topology. When calculating the paths between pairs of network endpoints, only the links between networking nodes are considered. For example, the shortest path between two network endpoints would be the shortest path between their PoAs.

Movement Track The movement track of a mobile endpoint is a sequence of PoAs to which it connects to ordered by time, the length of this sequence would thus equal to the number of relocations in the considered period plus 1. In a topology $G = (V, E)$, the movement track of a mobile endpoint M covering n relocations is identified by $MT(G, M) = (M_0, M_1, M_2, \dots, M_n)$, where $M_i \in V (0 \leq i \leq n)$, and $M_i \neq M_{i+1} (0 \leq i \leq n - 1)$.

4.2 Metrics

The proposed framework covers a total number of 5 metrics that reflect the mobility support performance under two mobile communication scenarios. The formulae inherit the definitions of M , C , and R in Section 2.2 that represent the three types of endpoints. When paths are considered, endpoints stand for the starting, ending, or breaking point of *shortest paths* between their PoAs. For example, MS stands for the shortest path between the PoAs of M and S , MSC represents the path joined by two shortest paths MS and CS . The $l()$ function produces the length of a path, e.g., $l(MS)$ produces the length of MS . The $j()$ function stands for the cross point of two paths, e.g., $j(MS, CS)$ stands for the cross point of MS and CS .

4.2.1 Basic Scenario

This scenario depicts the most basic form of communication where C continuously sends packets to M . Four metrics are proposed to reflect the mobility support performance in

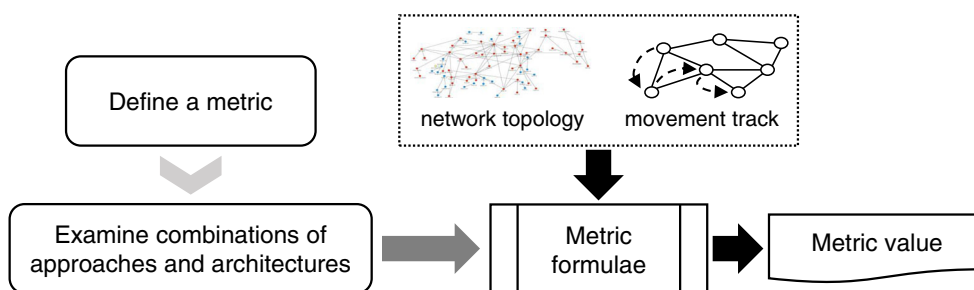


Fig. 5 The reasoning workflow for producing metric calculation formula

this scenario: *path stretch* and *handover delay* reflect user experience, while *signaling traffic* and *maintained states* measure network overhead. We proceed to explain how the critical differences between approaches and architectures shape the metric formulae, while a complete list of formulae is provided in Table 1.

Path Stretch (Stretch) This metric is the ratio of the length of the complete forwarding path from C to M to the length of the shortest possible path between C and M . Thus this metric reflects the optimality of the actual forwarding paths produced by an approach, and a smaller value indicates better performance.

Because there is no fundamental difference between IP and NDN in terms of the way packets (IP packets vs. Interest packets) are forwarded according to the carried network layer name (IP address vs. NDN name), calculation of this metric is architecture-independent. Regarding different

approaches, only the proxy-based and trace-based approach produce sub-optimal paths as a result of triangular routing. The trace-based approach produces shorter paths because packets may take shortcuts to M before reaching R if the established routes are met.

Neither knowledge-driven (KD) approaches affect the way this metric is calculated when enabled. TIP changes the PoA of R , not the way packets are forwarded, the value of its metric should be calculated in the same way using the new location of R . TRU does not affect packet forwarding at all, thus has no impact on this metric.

Handover Delay (Delay) This metric stands for the time units it takes for M to receive the first packet from C after a relocation. In NDN, the first packet is assumed to be a retransmission of a “lost” Interest already forwarded to M ’s last PoA. This metric thus reflects the smoothness of mobile communication, and a smaller value indicates

Table 1 Metric calculation formulae for the basic scenario

Metric	Approach	Architecture	
		IP	NDN
Stretch	Proxy		$l(CSM)/l(CM)$
	Resolution		1
	Routing		
	Trace		$l(Cj(CS, MS)M)/l(CM)$
Delay	Proxy		$\max(l(MS), l(CS)) + l(MS)$
	Proxy-TRU		$l(MS)$
	Resolution		$\max(l(MS), l(CS)) + l(CS) + l(CM)$
	Resolution-TRU		$l(CS) + l(CM)$
	Routing	$l(CM) \cdot 2$	$l(j(CM, CM')M) \cdot 2$
	Routing-TRU	$l(CM)$	$l(j(CM, CM')M)$
	Trace	$\max(l(XC), l(XM)) + l(XM)$	$\min(l(YM) \cdot 2, \max(l(XC), l(XM))) + l(XM)$
	Trace-TRU		$l(YM)$
Traffic	Proxy		$l(MS)$
	Proxy-TRU		0
	Resolution		$l(MS) + l(CS) \cdot 2$
	Resolution-TRU		$l(CS) \cdot 2$
	Routing		$ V - 1$
	Routing-TRU		0
	Trace		$l(MS)$
	Trace-TRU		0
State	Proxy		1
	Resolution		
	Routing		$ V $
	Trace		$l(MS)$

(M' stands for the last PoA of M ; $X = j(MS, CS)$, $Y = j(M'S, MS)$)

better performance because users would experience less disruptions. The formulae assume that retransmissions take place at the earliest possible time, which would be the moment that M connects to a PoA if relocations happen instantaneously.

Regarding the calculation of this metric, the critical difference between IP and NDN comes from NDN’s in-network retransmission capability. Specifically, an NDN node may retransmit a pending Interest (an Interest still waiting for Data) on its own if new route(s) to forwarding this Interest becomes available. The routing-based and trace-based approach are affected by this difference because they both update routes.

Considering KD approaches, TIP still only affects metric value by changing the location of R ; TRU eliminates the need to propagate signaling packets, thus changes the formula by removing the corresponding portion of delay.

Signaling Traffic (Traffic) This metric stands for the overall signaling traffic measured in the total number of hops traversed by signaling packets. Thus, signaling traffic reflects the network overhead of an approach in terms of traffic volume, and a smaller value indicates better performance, indicating better scalability.

This metric is architecture-independent, signaling packets are propagated in the same way regardless of network architectures. The formula for the routing-based approach considers a star topology, and assumes that routes are aggregated as much as possible.

For KD approaches, TRU eliminates signaling traffic for all non-KD approaches but the resolution-based approach, where C still needs to query R before sending packets.

Maintained States (State) This metric measures the total number of states that are maintained in the network and/or in R . This metric reflects the network overhead in terms of storage cost, and, similar to signaling traffic, a smaller value indicates better performance.

This metric is architecture-independent. The formula for the routing-based approach means that each node maintains one route at all time. KD approaches do not affect the formulae, because they do not change the type of reachability information, or how such information is stored.

4.3 Compound Scenario - Upload

The upload scenario describes a common application task where a certain amount of data is uploaded from M to C . A single metric upload time (Time) is proposed for this scenario to reflect the user experience. This metric measures the total amount of time required for finishing uploading all the data. The calculation of this metric depends on the

calculation of two basic scenario metrics: path stretch, and handover delay.

The following parameters are defined for this scenario ($i \geq 0$):

- UT : total amount of data to upload in unit size;
- r : upload rate ratio, a weighing factor reflecting network bandwidth;
- T_i : the period during which M stays immobile before i th relocation (stay period), i.e., the interval between $i - 1$ th and i th relocation, $T_0 = 0$;
- D_i : the amount of time it takes to complete the i th relocation (relocation period), $D_0 = 0$;

The formula is deducted as follows. Let the actual time M may upload data during each stay be $T_i - Delay_i$, and let U_i be the actual upload amount at the i th stay, then:

$$U_i = \frac{(T_i - Delay_i)r}{L(CM_i)Stretch_i} \text{ (if } T_i < Delay_i, \text{ then } U_i = 0) \quad (1)$$

Assume that after n relocations, $\sum_{i=1}^n U_i > UT$, then upload is finished during the last stay period, the elapsed time is:

$$GTime = \sum_{i=1}^{n-1} (D_i + T_i) \quad (2)$$

The time used to upload data during the last stay is thus:

$$UTime = T_n \frac{UT - \sum_{i=1}^n U_i}{U_n} \quad (3)$$

And further:

$$\begin{aligned} Time &= GTime + UTime \\ &= \sum_{i=1}^{n-1} (D_i + T_i) \\ &\quad + T_n \frac{UT - \sum_{i=1}^n (T_i - Delay_i)r / (L(CM_i)Stretch_i)}{(T_n - Delay_n)r / (L(CM_n)Stretch_n)} \end{aligned} \quad (4)$$

5 Evaluation

5.1 Experiment Settings

5.1.1 Network Topology

Experiments are conducted on 3 types of network topologies: real topology, balanced tree topology, and random graph topology:

- Real topology: topologies generated according to public information or measurements of the Internet.

- Balanced tree topology: topologies of the balanced tree shape that can be generated deterministically by setting the branching factor and height.
- Random graph topology: topologies with a certain number of nodes and randomly generated links among them.

A total number of 6 topologies of these 3 types of chosen as inputs: 2 real topologies of AS 1755 (RT-1755), and 6461 (RT-6461) from the Rocketfuel [1] dataset; 2 balanced tree topology with branching factor of 2, and height of 4 and 6 (BT-2-4 and BT-2-6); 2 random graph topologies whose number of nodes are 256 and 512, and link growth probability are 0.6 and 0.8, respectively (RG-256-0.6 and RG-512-0.8).

5.1.2 Movement Track

A movement pattern can be used to randomly generate movement tracks with specific statistical characteristics. The experiment uses movement tracks generated from the following three movement patterns:

- Completely random movement: the next PoA is always chosen uniformly at random from other nodes.
- Local movement: the next PoA is more likely a neighbor of the current PoA.
- Power law movement: the mobile endpoint frequently connects to a few PoAs, which resembles realistic cases where a person regularly visits several locations such as home, workplace, and restaurants.

A total number of 5 movement tracks are generated from the patterns above and used in experiments. One movement track (CRM) is generated with the completely random movement pattern; two movements tracks are generated using the local movement pattern, with the probability of relocating to a neighbor PoA (p) set to 0.3 (LM-0.3) and 0.7 (LM-0.7); two movement tracks are generated under the power law movement track, with α set to 2 (PM-2) and 3 (PM-3).

5.1.3 Randomness

A pseudo-random number generator is used to produce reproducible sequences of random numbers, based on which the initial positions of endpoints and the mobility tracks are generated. Experiments can be reproduced by using the same random seed. Under this randomness setting, repeated experiments are conducted under different random seeds, and the average value across these experiments is used.

5.2 Results

For each metric, the value for each evaluated approach and architecture combination is given under each set of network topology and movement track inputs. To identify different combinations, a combination is named by concatenating (in the following order) the shortened names of the basic approach (non-KD approach), any enabled KD approach(es), and finally architecture with “-”, e.g., “Trace-TIP-NDN”. If the result is the same for both IP and NDN, architecture name is ignored. If multiple basic approaches share the same value, the approach names are concatenated with “/”, e.g., “Proxy/Resolution-IP”.

5.2.1 Path Stretch

As shown in Table 2, the proxy-based approach has the highest value across all settings, reaching as high as 2.62 with the BT-2-4 topology and CRM movement track. The reason is that the proxy-based approach suffers the most from triangular path, all traffic must go through R first regardless of whether C and M are near each other or not. Although the trace-based approach also suffers from triangular path due to the use of an intermediate for packet forwarding, the average value is only 62% of that for the proxy-based approach. The advantage of the trace-based approach should be credited to updating routes, such that traffic does not necessarily need to go through R ; packets may take a shortcut to M if C and M are near each other to attenuate the triangular path effect. Regarding KD approaches, TIP is effective for optimizing both proxy-based and trace-based approach, resulting in 48% and 19% reduction respectively when enabled, almost leveling their performance.

5.2.2 Handover Delay

As shown in Fig. 6, the resolution-based approach produces the highest handover delay averaging 13.29 time units in both IP and NDN. This value is nearly 4 times of the lowest average value, which belongs to the routing-based approach in NDN. The extra delay is a result of the packet exchange before each transmission. Both routing-based and trace-based approach perform significantly better in NDN. The reason is that NDN’s stateful forwarding allows networking nodes to retransmit pending Interests (Interests in PIT) if new routes for forwarding such Interests become available. When routes are updated, the node on which a new route becomes available would retransmit the corresponding pending Interests immediately. Such in-network retransmissions may happen earlier than

Table 2 Results for path stretch

Combination	Mo. Track	Network topology						Avg.
		RT-1755	RT-6461	BT-2-4	BT-2-6	RG-256-0.6	RG-512-0.8	
Proxy	CRM	2.41	2.19	2.62	2.44	2.2	2.17	2.24
	LM-0.3	2.44	2.32	2.59	2.27	2.18	2.17	
	LM-0.7	2.47	2.32	2.36	2.15	2.19	2.17	
	PM-2	2.42	2.25	2.3	2.19	2.07	2.15	
	PM-3	2.41	2.25	2.29	2.17	2.03	2.17	
Proxy-TIP	CRM	1.13	1.04	1.1	1.06	1.7	2	1.44
	LM-0.3	1.06	1.05	1.09	1.01	1.82	1.97	
	LM-0.7	1.05	1.06	1.04	1	1.82	1.98	
	PM-2	1.11	1.06	1.09	1.04	1.58	1.71	
	PM-3	1.09	1.07	1.08	1.02	1.85	1.87	
Resolution/Routing	CRM				1			1
	LM-0.3							
	LM-0.7							
	PM-2							
	PM-3							
Trace	CRM	1.52	1.43	1	1	2.1	2.1	1.54
	LM-0.3	1.53	1.45	1	1	2.09	2.1	
	LM-0.7	1.55	1.45	1	1	2.1	2.12	
	PM-2	1.59	1.44	1	1	1.98	2.07	
	PM-3	1.59	1.44	1	1	1.95	2.09	
Trace-TIP	CRM	1.08	1.03	1	1	1.7	2	1.41
	LM-0.3	1.04	1.03	1	1	1.82	1.97	
	LM-0.7	1.04	1.03	1	1	1.82	1.97	
	PM-2	1.07	1.04	1	1	1.56	1.67	
	PM-3	1.05	1.03	1	1	1.79	1.82	

endpoint-initiated retransmissions, and the retransmitted packets would travel less distance to reach M . Another observation is that the routing-based and trace-based approach are highly influenced by mobility pattern in NDN. Without KD approaches, the biggest difference caused by mobility pattern is over 3 times (Routing/Trace-NDN in BT-2-6, CRM vs. LM-0.7), and with KD approaches, more than 4 times (Trace-TIP-NDN and Trace-TIP-TRU-NDN in BT-2-6, CRM vs. LM-0.7). The main reason is that when M moves locally, the retransmitted Interest very likely comes from a nearby node, and thus reaches M earlier.

For KD approaches, TIP and TRU both bring reductions, and the value further drops when TIP and TRU are both enabled. The shorter handover delay comes from optimized reachability information update process. TIP reduces the propagation time of signaling packets by moving R closer to M , which is a side effect of placing R near the shortest paths; TRU eliminates the need to propagate signaling packets, thus removes a significant portion of delay.

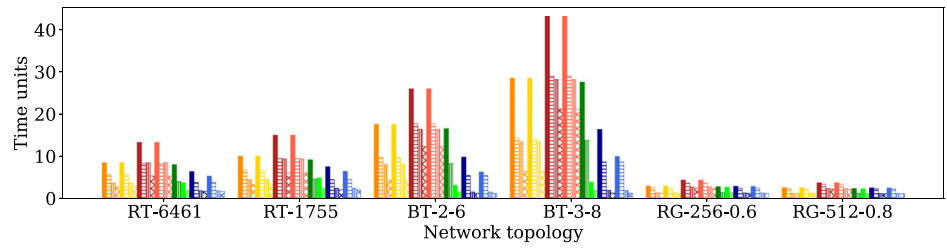
5.2.3 Signaling Traffic

As shown in Table 3, the routing-based approach generates the most amount of traffic in both IP and NDN. Flooding signaling packets across the entire network is extremely expensive. Regarding KD approaches, TIP has an optimization effect on the proxy-based, resolution-based, and trace-based approach due to shortened propagation paths of signaling packets; TRU produces zero signaling traffic except for when enabled over the resolution-based approach, since C still needs to query R .

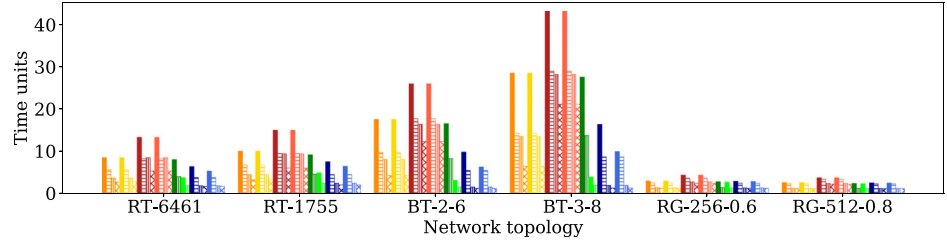
5.2.4 Maintained States

As shown in Table 4, the routing-based approach requires the network to maintain the most number of states. This is the necessary cost for globally optimizing forwarding paths. The trace-based approach generates only a moderate amount of states, but still manages to produce reasonably

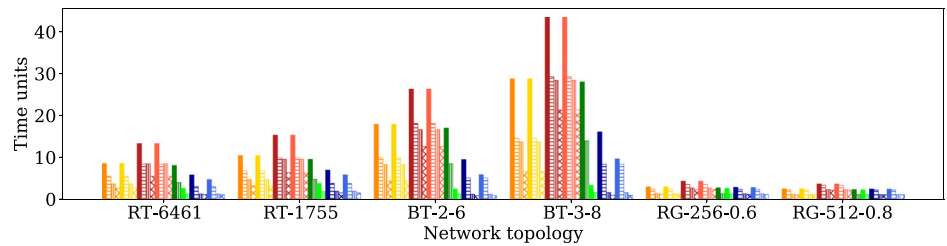
Fig. 6 Results for handover delay (with legend). Each bar represents an approach and architecture combination. Results for handover delay (with legend). The X axis is topology, the Y axis is the value of Delay, bars represent approach and architecture combinations



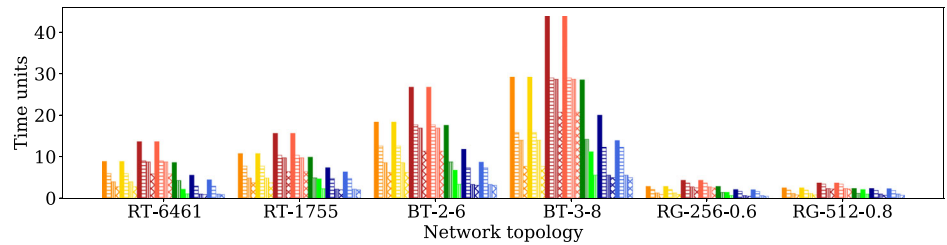
(a) Completely random mobility (CRM)



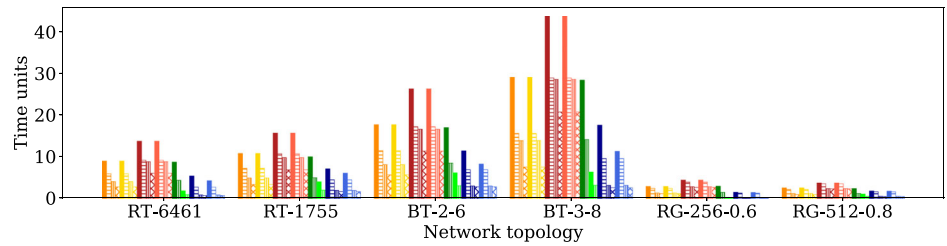
(b) Local movement, $p = 0.3$ (LM-0.3)



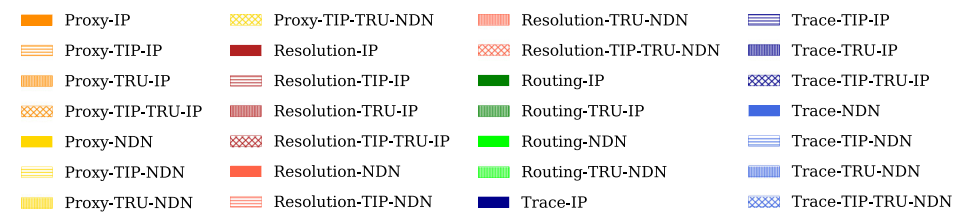
(c) Local movement, $p = 0.7$ (LM-0.7)



(d) Power law movement, $\alpha = 2$ (PM-2)



(e) Power law movement, $\alpha = 3$ (PM-3)



(f) Legend

Table 3 Results for signaling traffic

Combination	Mo. Track	Network topology						Avg.
		RT-1755	RT-6461	BT-2-4	BT-2-6	RG-256-0.6	RG-512-0.8	
Proxy/Trace	CRM	5.2	4.18	4.84	8.41	1.39	1.22	4.61
	LM-0.3	4.46	3.68	4.3	8.06	1.38	1.22	
	LM-0.7	4.74	3.74	4.48	8.33	1.4	1.2	
	PM-2	4.93	4.02	4.67	8.61	1.33	1.2	
	PM-3	4.95	4.07	4.65	8.13	1.33	1.2	
Routing	CRM	171	181	30	126	255	511	152.6
	LM-0.3	171	181	30	126	255	511	
	LM-0.7	171	181	30	126	255	511	
	PM-2	171	181	30	126	255	511	
	PM-3	171	181	30	126	255	511	
Proxy/Trace-TIP	CRM	4.17	3.26	3.98	6.44	1.11	1.18	3.23
	LM-0.3	3.27	2.76	2.76	4.27	1.24	1.16	
	LM-0.7	3.28	2.66	2.69	4.42	1.22	1.14	
	PM-2	3.8	2.84	3.39	6.26	0.96	0.85	
	PM-3	3.37	2.78	2.96	5.67	1.15	0.93	
Proxy/Routing/Trace-TRU	CRM				0			0
	LM-0.3							
	LM-0.7							
	PM-2							
	PM-3							
Resolution	CRM	14.84	13.12	14.88	24.69	4.11	3.6	14.14
	LM-0.3	14.1	12.62	14.34	24.34	4.1	3.6	
	LM-0.7	14.38	12.68	14.52	24.61	4.12	3.58	
	PM-2	14.57	12.96	14.71	24.89	4.05	3.58	
	PM-3	14.59	13.01	14.69	24.41	4.05	3.58	
Resolution-TIP	CRM	6.61	5.7	6.3	10.62	3.11	3.22	6.91
	LM-0.3	6.15	5.4	7.16	12.29	3.24	3.16	
	LM-0.7	6.48	5.6	7.69	12.62	3.22	3.2	
	PM-2	6.74	6.02	7.35	11.46	3.12	3.01	
	PM-3	7.11	6.14	7.78	11.43	3.51	3.19	
Resolution-TRU	CRM	9.64	8.94	10.04	16.28	2.72	2.38	9.52
	LM-0.3	9.64	8.94	10.04	16.28	2.72	2.38	
	LM-0.7	9.64	8.94	10.04	16.28	2.72	2.38	
	PM-2	9.64	8.94	10.04	16.28	2.72	2.38	
	PM-3	9.64	8.94	10.04	16.28	2.72	2.38	
Resolution-TIP-TRU	CRM	2.44	2.44	2.32	4.18	2	2.04	3.69
	LM-0.3	2.88	2.64	4.4	8.02	2	2	
	LM-0.7	3.2	2.94	5	8.2	2	2.06	
	PM-2	2.94	3.18	3.96	5.2	2.16	2.16	
	PM-3	3.74	3.36	4.82	5.76	2.36	2.26	

Table 4 Results for maintained states

Combination	Mo. Track	Network topology						Avg.
		RT-1755	RT-6461	BT-2-4	BT-2-6	RG-256-0.6	RG-512-0.8	
Proxy/Resolution	CRM	32.09	33.94	6.46	23.91	47.36	93.91	39.61
	LM-0.3							
	LM-0.7							
	PM-2							
	PM-3							
Proxy/Resolution-TIP	CRM	31.78	33.61	6.41	23.68	46.9	92.98	39.23
	LM-0.3							
	LM-0.7							
	PM-2							
	PM-3							
Routing	CRM	172	182	31	127	256	512	213.33
	LM-0.3							
	LM-0.7							
	PM-2							
	PM-3							
Trace	CRM	20.84	20.75	7.66	19.95	24.66	47.76	23.44
	LM-0.3	20.09	20.25	7.12	19.61	24.65	47.76	
	LM-0.7	20.38	20.3	7.3	19.88	24.67	47.74	
	PM-2	20.57	20.58	7.49	20.15	24.6	47.75	
	PM-3	20.58	20.63	7.47	19.68	24.6	47.75	
Trace-TIP	CRM	19.65	19.66	6.78	17.86	24.15	47.26	22.07
	LM-0.3	18.75	19.15	5.55	15.7	24.28	47.24	
	LM-0.7	18.76	19.05	5.48	15.85	24.26	47.22	
	PM-2	19.28	19.23	6.18	17.69	24	46.93	
	PM-3	18.85	19.18	5.76	17.1	24.19	47.01	

stretched forwarding paths. With TIP enabled, the results for the trace-based approach drops slightly due to the shortened distance between M and R .

5.2.5 Upload Time

The upload scenario is simulated with the following parameter settings:

- Total upload size: 1024;
- upload rate factor: 2;
- method for generating stay periods: choose uniformly between 16 and 48;
- method for generating relocation period: set to the length of the shortest path between the current and previous PoA.

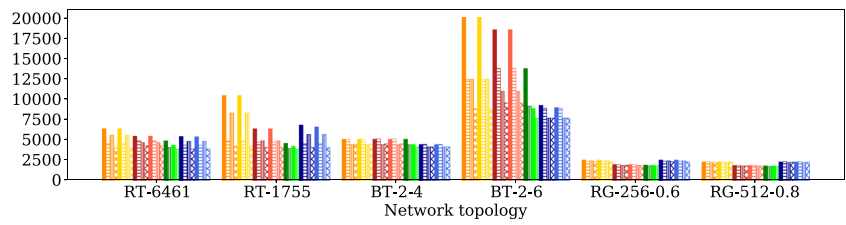
As shown in Fig. 7, mobility patterns have a significant impact on the results. The completely random movement generally producer the worst results (higher values). This

difference from the results in the basic scenario comes from the way relocation periods are determined. The farther M relocates, the longer the relocation period is, and the longer it would take to finish uploading. Comparing the performance of approach and architecture combinations, the routing-based approach in NDN performs the best in all network topology and movement track settings due to the combined (amplified) effect of the routing-based approach's path optimization capability and NDN's in-network retransmission feature, the reduced path stretch and handover delay means that more time can be used to upload data during each stay.

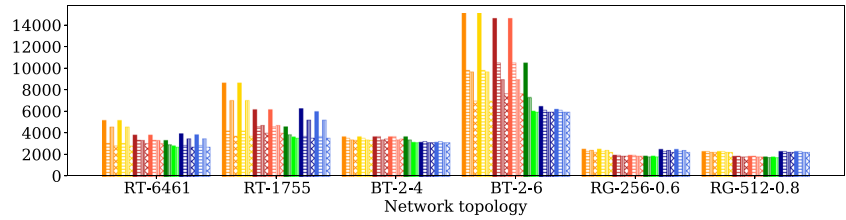
5.2.6 The Optimization Effect of KD Approaches

The effectiveness of KD approaches at optimizing mobility support performance is evaluated via the *Optimization Ratio (OR)*. *OR* is calculated for a metric regarding a basic approach and architecture combination (vanilla

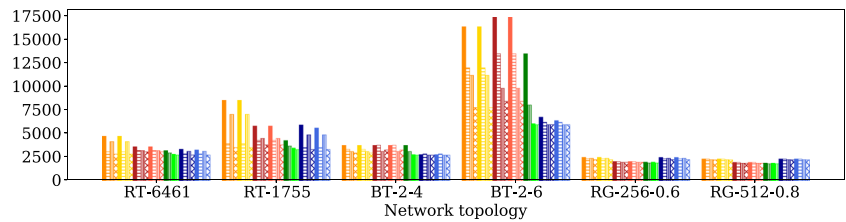
Fig. 7 Results for upload time (legend in Fig. 6f). Each bar represents an approach and architecture combination



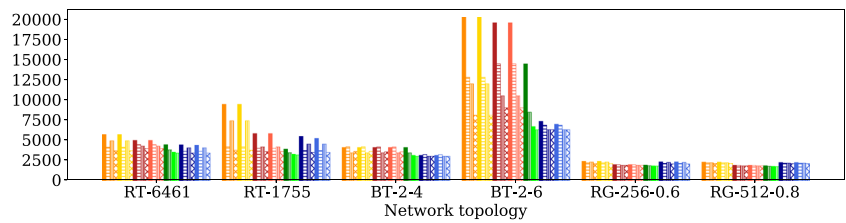
(a) Completely random mobility (CRM)



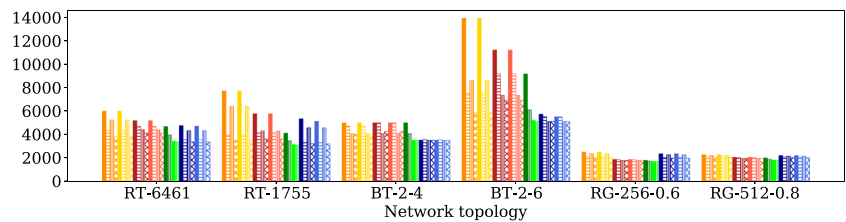
(b) Local movement, $p = 0.3$ (LM-0.3)



(c) Local movement, $p = 0.7$ (LM-0.7)



(d) Power law movement, $\alpha = 2$ (PM-2)



(e) Power law movement, $\alpha = 3$ (PM-3)

combination), and any enabled KD approach or KD approach combination as follows. First the average value v of the metric for the vanilla combination across all input settings is calculated; then for a combined KD approach or KD approach combination, another average value v_{KD} is calculated in the same manner. Finally:

$$OR = \begin{cases} \frac{v-v_{KD}}{v} & \text{if a smaller value of the metric indicates better performance} \\ \frac{v_{KD}-v}{v} & \text{if otherwise} \end{cases} \quad (5)$$

A positive OR indicates optimization of mobility support performance. And a higher positive value indicates stronger optimization effect. As shown in Fig. 8, the proposed KD approaches, either alone or combined, has an optimization effect on all 5 metrics covered by the evaluation framework. OR is almost always over or near 50%, reaching as high as 100%. This result is expected because KD approaches are supported by knowledge, a tool never before exploited for providing mobility support. However, such results also hint the existence of knowledge-related costs for the collection,

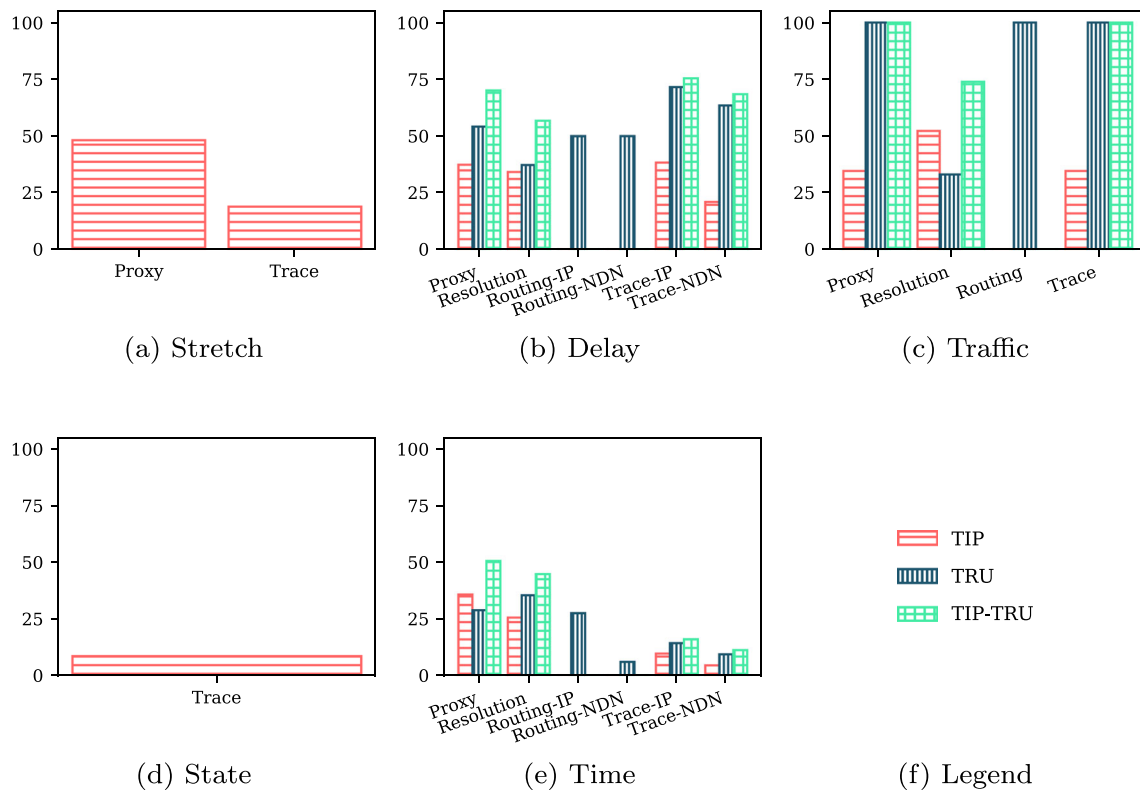


Fig. 8 Results for Optimization Ratio (with legend). The X axis is approach and architecture combinations, the Y axis is the value of OR as percentage, bars represent KD approaches

propagation, and processing of knowledge, and call for further investigation to better understand the limits of KD approaches.

6 Concluding Remarks

Rapidly increasing mobile traffic calls for scalable, low-latency, and seamless mobility support in the future Internet. This paper makes the observation that existing solutions in proposed network architectures have not been able to achieve fundamental improvements in terms of mobility support performance. The design choices in the explored design space limit such solutions to trading off between user experience and network overhead. Also, present mobility support performance evaluation methods generally lack support for newly proposed and evolving network architectures as well as mobility support solutions that come in a variety of design and implementation details. Regarding the former challenge, this paper sketches out two novel knowledge-driven mobility support approaches, as well as a layered architecture for supporting such novel approaches. As to the second challenge, a quantitative cross-architectural mobility support performance evaluation framework is proposed to both analyze existing solutions

and evaluate the optimization effect of knowledge-driven approaches. Experiment results show that knowledge-driven approaches significantly optimize all 5 proposed metrics, preliminarily demonstrating the advantage of exploiting knowledge to providing mobility support. Future research will focus on three directions: expand the framework to support quantitative analysis of knowledge-related costs; propose a roadmap for realizing the knowledge-driven vision under the proposed KDN architecture; and apply the knowledge-driven vision to new research topics, such as in-network and edge computing.

Acknowledgements This work is supported by Peng Cheng Laboratory Funds (No. PCL2021A02). An earlier version of this work has been presented as a conference paper [8] in EAI AICON 2020, the conference proceedings of which can be found at the following SpringerLink: <https://link.springer.com/book/10.1007%2F978-3-030-69066-3>.

Funding This work is supported by Peng Cheng Laboratory Funds (No. PCL2021A02).

Data Availability The Rocketfuel [1] network topologies used in the experiments are available at <https://research.cs.washington.edu/networking/rocketfuel>.

Code Availability The code for reproducing the results in this paper is available at <https://github.com/KDN-Mobility>.

References

- Spring N, Mahajan R, Wetherall D (2002) Measuring ISP topologies with Rocketfuel. *ACM SIGCOMM Comput Commun Rev* 32(4):133
- Qiu J, Du L, Zhang D, Su S, Tian Z (2019) Nei-TTE: intelligent traffic time estimation based on fine-grained time derivation of road segments for smart city. *IEEE Trans Ind Inform* 16(4): 2659
- Shafiq M, Tian Z, Bashir AK, Jolfaei A, Yu X (2020) Data mining and machine learning methods for sustainable smart cities traffic classification: A survey. *Sustain Cities Soc* 60:102177
- Shafiq M, Tian Z, Sun Y, Du X, Guizani M (2020) Selection of effective machine learning algorithm and Bot-IoT attacks traffic identification for internet of things in smart city. *Futur Gener Comput Syst* 107:433
- Tian Z, Gao X, Su S, Qiu J (2019) Vcash: a novel reputation framework for identifying denial of traffic service in internet of connected vehicles. *IEEE Internet Things J* 7(5):3901
- Tian Z, Gao X, Su S, Qiu J, Du X, Guizani M (2019) Evaluating reputation management schemes of internet of vehicles based on evolutionary game theory. *IEEE Trans Veh Technol* 68(6): 5971
- Zhang L, Afanasyev A, Burke J, Jacobson V, Claffy k, Crowley P, Papadopoulos C, Wang L, Zhang B (2014) Named data networking. *ACM SIGCOMM Comput Commun Rev* 44(3):66. <https://doi.org/10.1145/2656877.2656887>
- Xia Z, Zhang Y (2021) Towards knowledge-driven mobility support. In: *Artificial intelligence for communications and networks: Second EAI International Conference, AICON 2020, Virtual Event, December 19-20, 2020, Proceedings 2*. Springer International Publishing, pp 197–216
- Zhang Y, Afanasyev A, Burke J, Zhang L (2016) A survey of mobility support in named data networking. In: *2016 IEEE conference on computer communications workshops (INFOCOM WKSHPs)*, pp 83–88. <https://doi.org/10.1109/INFCOMW.2016.7562050>
- Perkins C, Bhagwat P (1994) A mobile networking system based on Internet protocol. *IEEE Personal Commun* 1(1):32. <https://doi.org/10.1109/98.911984>. Conference Name: IEEE Personal Communications
- Gao D, Rao Y, Foh CH, Zhang H, Vasilakos AV (2017) PMNDN: proxy based mobility support approach in mobile NDN environment. *IEEE Trans Netw Serv Manag* 14(1):191
- Azgin A, Ravindran R, Wang G (2014) A scalable mobility-centric architecture for named data networking. *arXiv:1406.7049[cs]*
- Lee J, Cho S, Kim D (2012) Device mobility management in content-centric networking. *IEEE Commun Mag* 50(12):28. <https://doi.org/10.1109/MCOM.2012.6384448>. Conference Name: IEEE Communications Magazine
- Hermans F, Ngai E, Gunningberg P (2012) Global source mobility in the content-centric networking architecture. In: *Proceedings of the 1st ACM workshop on emerging name-oriented mobile networking design - architecture, algorithms, and applications (Association for Computing Machinery), NoM '12*, pp 13–18. <https://doi.org/10.1145/2248361.2248366>
- Kim Dh, Kim Jh, Kim Ys, Yoon Hs, Yeom I (2012) Mobility support in content centric networks. In: *Proceedings of the second edition of the ICN workshop on Information-centric networking (Association for Computing Machinery), ICN '12*, pp 13–18. <https://doi.org/10.1145/2342488.2342492>
- Li D, CHuah MC (2013) SCOM: A scalable content centric network architecture with mobility support. In: *2013 IEEE 9th international conference on mobile ad-hoc and sensor networks*, pp 25–32. <https://doi.org/10.1109/MSN.2013.44>
- Zhu Z, Zhang L, Wakikawa R (2020) Understanding apple's back to my mac (BTMM) service. <https://tools.ietf.org/html/rfc6281>. Library Catalog: tools.ietf.org
- Afanasyev A, Yi C, Wang L, Zhang B, Zhang L (2015) SNAMP: Secure namespace mapping to scale NDN forwarding. In: *2015 IEEE conference on computer communications workshops (INFOCOM WKSHPs)*, pp 281–286. <https://doi.org/10.1109/INFCOMW.2015.7179398>. ISSN: null
- Rekhter Y, Li T (2020) A border gateway protocol 4 (BGP-4). <https://tools.ietf.org/html/rfc1771>. Library Catalog: tools.ietf.org
- Hu X, Li L, Mao ZM, Yang YR (2008) Wide-area IP network mobility. In: *IEEE INFOCOM 2008 - The 27th conference on computer communications*, pp 951–959. <https://doi.org/10.1109/INFOCOM.2008.148>. ISSN: 0743-166X
- Augé J, Carofiglio G, Grassi G, Muscariello L, Pau G, Zeng X (2018) MAP-Me: Managing anchor-less producer mobility in content-centric networks. *IEEE Trans Netw Serv Manag* 15(2):596. <https://doi.org/10.1109/TNSM.2018.2796720>. Conference Name: IEEE Transactions on Network and Service Management
- Meddeb M, Dhraief A, Belghith A, Monteil T, Drira K, Gannouni S (2018) AFIRM: Adaptive forwarding based link recovery for mobility support in NDN/IoT networks. *Futur Gener Comput Syst* 87:351
- Zhang Y, Xia Z, Mastorakis S, Zhang L (2018) KITE: producer mobility support in named data networking. In: *Proceedings of the 5th ACM conference on information-centric networking (Association for Computing Machinery), ICN '18*, pp 125–136. <https://doi.org/10.1145/3267955.3267959>
- Gao Z, Venkataramani A, Kurose J, Heimlicher S (2014) Towards a quantitative comparison of location-independent network architectures. In: *Proceedings of the 2014 ACM conference on SIGCOMM (Association for Computing Machinery), SIGCOMM '14*, pp 259–270. <https://doi.org/10.1145/2619239.2626333>
- Chaganti V, Kurose J, Venkataramani A (2008) A cross-architectural quantitative evaluation of mobility approaches. In: *IEEE INFOCOM 2008 - IEEE conference on computer communications*, pp 639–647. <https://doi.org/10.1109/INFOCOM.2008.8485893>
- Kim H, Feamster N (2013) Improving network management with software defined networking. *IEEE Commun Mag* 51(2):114. <https://doi.org/10.1109/MCOM.2013.6461195>. Conference Name: IEEE Communications Magazine
- Mijumbi R, Serrat J, Gorricho JL, Bouten N, De Turck F, Boutaba R (2016) Network function virtualization: state-of-the-art and research challenges. *IEEE Commun Surv Tutor* 18(1):236. <https://doi.org/10.1109/COMST.2015.2477041>. Conference Name: IEEE Communications Surveys Tutor

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.