Relocation Adaptable Kite: A Scheme to Support Mobility in Named Data Networking

A DISSERTATION

submitted towards the partial fulfillment of the requirements for the award of the degree of

Master of Technology

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Monika

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247667 (INDIA)
May, 2016

CANDIDATE'S DECLARATION

I declare that the work presented in this dissertation with title "**Relocation Adaptable Kite: A mobility model for Named Data Networking**" towards the fulfilment of the requirement for the award of the degree of **Master of Technology** in **Computer Science & Engineering** submitted in the **Dept. of Computer Science & Engineering**, **Indian Institute of Technology, Roorkee**, India is an authentic record of my own work carried out during the period **from July 2014 to May 2015** under the supervision of **Dr. Manoj Mishra**, Professor, Dept. of CSE, IIT Roorkee.

The content of this dissertation has not been submitted by me for the award of any other degree of this or any other institute.

DATE : .... SIGNED: ....

PLACE: .... (MONIKA)

14535028

CERTIFICATE

This is to certify that the statement made by the candidate is correct to the best of my knowledge and belief.

DATE : .... SIGNED: ....

(DR. MANOJ MISHRA)
PROFESSOR
DEPT. OF CSE, IIT ROORKEE

Acknowledgements

FIRST and foremost, I would like to express my sincere gratitude towards my guides **Dr. Manoj Mishra**, Professor, Computer Science and Engineering, IIT Roorkee, for his ideal guidance throughout my entire graduate research period. I want to thank him for the advices, insightful discussions and constructive criticisms which certainly enhanced my knowledge and improved my skills. His constant encouragement, support and motivation have always been key sources of strength for me to overcome all the difficult and struggling phases.

I thank **Department of Computer Science and Engineering, IIT Roorkee** for providing lab and other resources for my graduate thesis work.

Finally, I thank all my friends and lab mates for being very good friends, for a lot many good discussions and for being along with me in the whole journey and making it so joyful.

Abstract

A Mobile node like smart phone can move at any speed 5 m/s walking speed or in a car at speed 30 m/s. Network and Applications should be intelligent enough to adopt data rate and Interest rate according to any speed and for any duration of active connection between mobile node and access point. The rate at which Interest packets are generated from mobile user in Named Data Networking is called Interest rate. High data rate makes user experience better and low Interest rate means lesser signaling overhead so that underlying network work efficiently specifically in mobile networks. There should be fair trade-off between data rate and Interest rate. In this thesis we propose a novel approach for providing mobility support for mobile producers in Named Data Networking, with significantly lower Interest rate than other available mobility support schemes. Dynamic retransmission timer and advance forwarding plane in Named Data Networking makes the proposed scheme suitable for any speed of mobile node. Dynamic retransmission timer increases the life-time of interest packet on each retransmission of Interest packet. More the count of retransmitted Interest packet lesser the retransmission rate of Interest packet(number of retransmitted Interest per second). And after a sufficient long period of active connection between mobile node and access point, the mobile node become reachable from the routing plane. After that there is no need of creating and maintaining traces. We evaluated our proposed algorithm for 60 different speeds of mobile node ranges from 5 to 300 m/s. A thorough analysis of results shows that proposed scheme performs better with reduced signaling overhead.

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE **Page**

Chapter 1

Introduction

In the Internet age, there is a strong need to connect to Internet anywhere and anytime. With evolving mobile technologies like smart phone from feature phone, people are always connected to Internet. On-line shopping , social networking, on-line gaming, on-line storing data like drop-box etc requires ubiquitous connectivity to Internet.

1.1 Motivation

According to the KCPB report: 2015 Internet trends[4],73% mobile user penetration has been recorded globally in 2014, it was 1% in 1995 [4]. 76% of total Internet users are reported to be smart-phone subscribers [4]. Providing ubiquitous connectivity to the mobile devices is a major issue in TCP/IP based networks. There has been tremendous amount of research on how to support mobility in TCP/IP model, but none of them completely resolves the problems associated with mobility because the TCP was designed to provide connectivity between two fixed nodes having specific destination addresses.

The other motivational factor is the evolution in data traffic of Internet. Internet data traffic is aided by mobile and video data by +21% year over year [4]. User generated and curated content is increasing year over year, for e.g. the rate of video views on Facebook per day is 4 Billion. It means that the content producers are shifting from stable servers to mobile users. For instance uploading videos, photos and audios, user produced news in the form of tweets via Twitter / Dataminr, live streaming of gaming etc. constitute main Internet traffic. The purpose of this thesis is to provide an algorithm that enables mobile user to handle mobility effectively (in various technologies 4G, Wi-fi, WiMax etc.) while frequently changing the Access Points as well as staying at same Access Point for long-time(longer relocation period).

1.2 Mobility Support in Named Data Networking

Named Data Networking (NDN) [5] is a future Internet architecture that promises to provide ubiquitous connectivity to mobile nodes. NDN is more suitable to mobile environment because of its two features explained below:

1. The data-centric nature: In NDN packets are identified by data names rather than their topological bound addresses (IP Address). Since self-identification of each data packet is carried out by its unique name, there is no requirement to connect to a particular location first then ask for data. A data can be obtained from any router in the network.

2. The stateful forwarding plane: Pending Interest Tables (PITs) maintains the state of each data packets in the forwarding plane which enables the backwards path forwarding of data. Data goes only where it is demanded, which makes both the location and identity of data consumers transparent to the routing plane and data producers.

There can be two roles of a mobile node in Internet plane:

1. Mobile consumer- Supported by stateful forwarding plane of NDN and in-network caching.

2. Mobile Producer- Lots of work is being done to provide ubiquitous connectivity to mobile producers in NDN, mapping based solutions, tracing based solutions and data collection based solutions. The proposed scheme is an enhanced version of a tracing based solution Kite.

Derived from stateful data forwarding of NDN, the key idea of Kite mobility model for NDN [2], is to completely exploit forwarding plane of NDN to monitor the moving nodes.

In particular, an Interest packet can be sent from an application to a routable anchor that can act as breadcrumbs by creating PIT entries which leads back to the mobile node on demand. So the mobile node (MN) can be reached via the anchor by Interests from correspondent nodes (CN). Kite is an alternative solution to mapping based schemes and it performs better than mapping based schemes when the frequency of relocation of mobile node is high. Kite is not a one fit solution to all mobility problems, it is not adaptable to long-term relocation of mobile node, so the proposed scheme Relocation Adaptable Kite (RA-Kite) provides solution to this problem.

The Relocation Adaptable Kite (RA-Kite) contributes following three new features to the mobility support of NDN:

1. Locator-free: The location of mobile node can be addressed by router-by- router PIT states (breadcrumbs method) which eliminates the need of explicit locator for mobile node. But the topological locations of mobile nodes are not detectable by the routers and correspondence nodes.

2. Scenario-aware: As application protocols directly generate and utilize traces, RA-Kite partially empowers the application protocol developers to devise their own designs of mobility support which can be tailored to their scenarios.Traces setup and maintenance can be carried out according to their needs, rather than only relying on some middle-wares or the network layer.

3. Enabling long-term relocation: Besides supporting short time stay at a location it also supports long-term stay. A mobile node can be reached along the traces if it stays less than a long-term relocation period threshold. This long-term relocation period threshold is dependent on user mobility. This threshold can be 10 secs if a mobile node is moving very fast and it can be 24 hours if mobile node is moving on slow. After staying for threshold period, FIB entry is created, then the node no longer needs to create traces along the path. Now, mobile node is globally reachable from the routing plane itself.

1.3 The Problem Statement

The work done in this thesis aims to provide a solution for long-term relocation of the mobile node which is not supported by Kite model. When a mobile node stays at same location for long time then the signaling overhead becomes more. Signaling overhead means number of packets required to tell the current location of mobile node.

This thesis provides unified solution for both micro and macro mobility and also provides support for long-term relocation of mobile node in NDN.This which includes the following activities:

1. Study the performance of Kite model at lower speeds. It is assumed that lower speed means longer relocation period at each node.
2. Proposing a scheme that extends workability of Kite, and supports mobility for long-term relocation of mobile node.
3. Simulation and analysis of proposed scheme and its comparison with Kite.

1.4 Contribution of The Thesis

This thesis contribution is two-fold:

1. Design and develop a unified solution to short-term and long-term relocation of a mobile node in NDN.
2. Providing starting point for further research on how mobility can be handled in NDN at Application level.

1.5 Organisation of The Report

This report includes 6 chapters

- Chapter 1 is the introduction, contains motivation problem statement and contribution of the thesis.
- Chapter 2 provides the related work, categorize the solutions based on how the mobile node is chased and how the data can be retrieved.
- Chapter 3 describes the Kite model in brief, drawbacks of Kite model, the reasons of drawbacks, and simulation results of Kite model.

- Chapter 4 describes the Relocation Adaptable Kite model (RA-Kite).
- Chapter 5 describes the experimental setup, simulation results of Kite model and comparison of Kite and RA- kite. It also include results and comparison of RA-Kite and MMS.
- Chapter 6 concludes the report and contains future scope of RA-Kite model.

CHAPTER 2

LITERATURE REVIEW

The consumer mobility is supported by in-network caching and self-identifying nature of content in NDN. There have been many proposed solution to provide producer mobility efficiently. One of the observation from those proposed schemes is: there should be an indirection mechanism that can tell the routing plane about the current location of mobile node, and then directs the NDN packets towards mobile node. Based on the indirection mechanism the proposed schemes can be categorized as: mapping based indirection mechanism, data rendezvous mechanism, tracing based indirection mechanism.

2.1 Mapping Based Solutions

The mapping based solutions are motivated from Mobile IP, a mobility model for TCP/IP. The mobile node produces data under a unique data name and it updates temporary location to stable mapping servers. According to role of mapping servers the solutions can be split in two categories: first mapping servers can only tell the current location of mobile node. The current location is called point-of-attachment, so mapping servers provide mapping between the data name to current point-of-attachment Figure 2.1. How this current point of attachment is carried in the interest packet is a different issue. Second mapping server can act as hybrid Figure 2.2, it can provide mapping between the data name to current point-of- attachment and it also tunnels the packet to current location of mobile node.[6, 7]

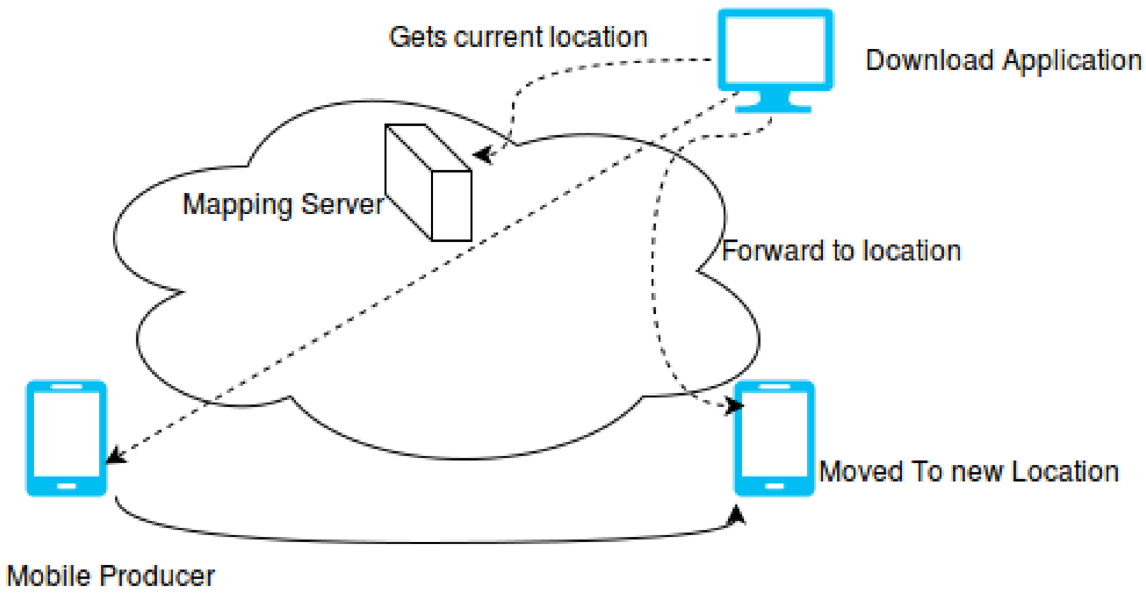

Figure 2.1: Mapping based scheme only mapping service like DNS[1]

In both the roles the state of location maintained at the mapping servers is hard-state. The hard-state makes it fault resistance and bottle neck. Also there is possibility of triangular routing same as in Mobile IP. There is another drawback of mapping based schemes, how to direct the packets to current location either by attaching the location information into the data name or providing the current location as hint in options of interest packet. If the location name is attached with the data name of packet then the packet becomes location dependent which can‚Äôt use in-network caching feature of NDN. If the point-of- attachment is provided as hint in the interest packet then the forwarding strategy needs to be changed[8].

2.2 Tracing Based Solution

Tracing based solution extends the adaptive and intelligent forwarding plane of NDN. As the nature of NDN per packet forwarding state is maintained in PIT at each router, these PIT entries serve the router-by- router reverse path to mobile node. Tracing needs the data produced by mobile node under a globally accessible name. The trace is setup by sending a traceable interest first, then the tracing interest traverses the path setup by previous interest as shown in Figure 2.3. Then mobile node replies to interest with the data packet. There is one overhead, the starting traceable interest to tell the current

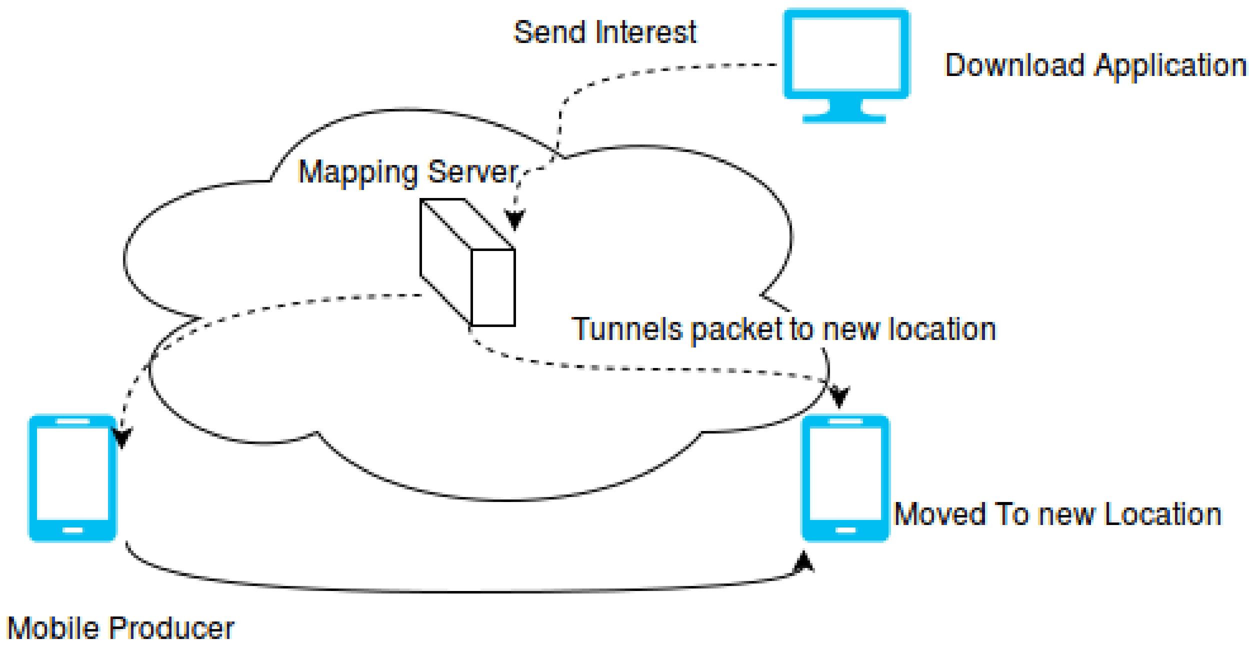

Figure 2.2: Mapping based scheme hybrid mapping server [1]

location of the mobile node. Now the trace is maintained by sending the traceable interest packet after certain amount of time before the entry expires in PIT.

Based on how traces are stored in router, there are two categories: first traces are stored as PIT entry in PIT table [4] a side-effect of stateful forwarding, second traces stored in temporary FIB, able (its new table maintained at each router). A new entry is added to temporary FIB whenever router receives and authenticates a traceable packet. The tracing packet is forwarded if there is a match in tFIB[9]. One of the tracing based solution is explained in Chapter 3.

The advantage of tracing based solution is there is no hard state maintained on any router, a soft-state that doesn‚Äôt exist after mobile node moves to new location because the PIT entry will expire after some time. And there is no tunneling required.

The disadvantage of tracing based solution is whenever a mobile node stays at a particular location, there is signaling overhead because mobile node keep sending traceable interest packet to prolong the PIT or temporary FIB entry on each router even when there is no active retrieval of the data.

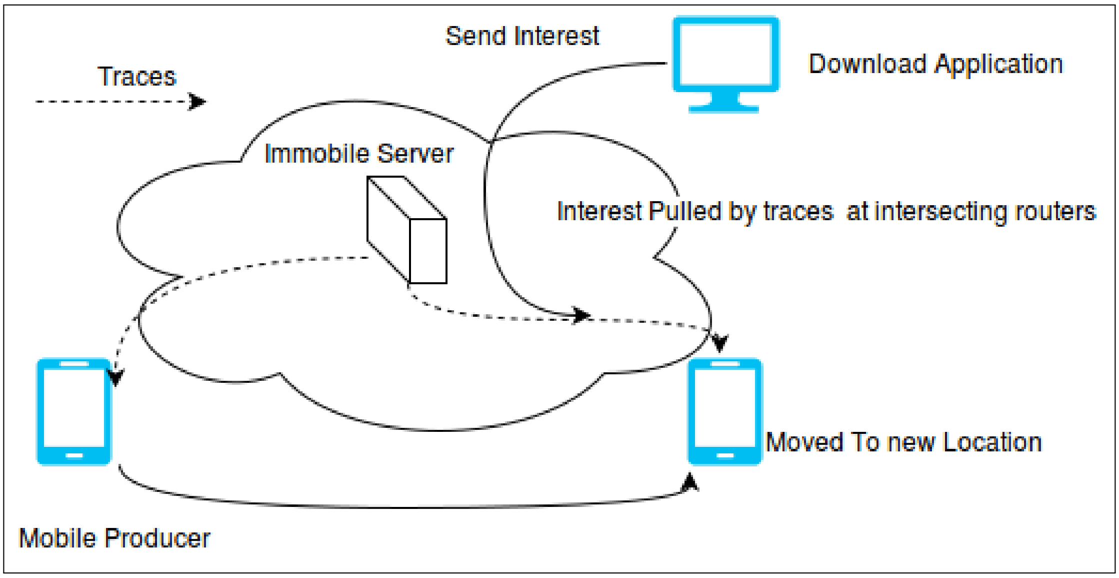

Figure 2.3: Tracing based scheme[2]

2.3 Data Rendezvous Based Solution

In NDN the data packet is easily detachable from the originator of the packet, this feature is leveraged by data rendezvous based solution. The other solution to chasing behind a mobile node is to move data produced by it to a stationary location, which is reached globally by routing plane. Here the distributed data repositories can be used to make content available easily at every location, because store space is no more an issue[3].

Based on the type of data rendezvous, there are two directions of research, first data depot that stores the data produced by mobile node and shares the name with the data, second data spot when data is location dependent for example data produced by vehicles about traffic at particular terminal point[3].

Data depot hosts the data and takes full credibility for providing the data, if data depot has data then interest packet is replied back with data packet and if data depot doesn‚Äôt have the data it gets the data from mobile node on demand. Data rendezvous scheme uses either mapping based scheme or tracing based scheme to chase the mobile node on demand when data is not present in data depot Figure 2.4.

The combination of data-depot and tracing based scheme can provide the best solution to mobility problem. As it requires least change in the current system. Most of mobility

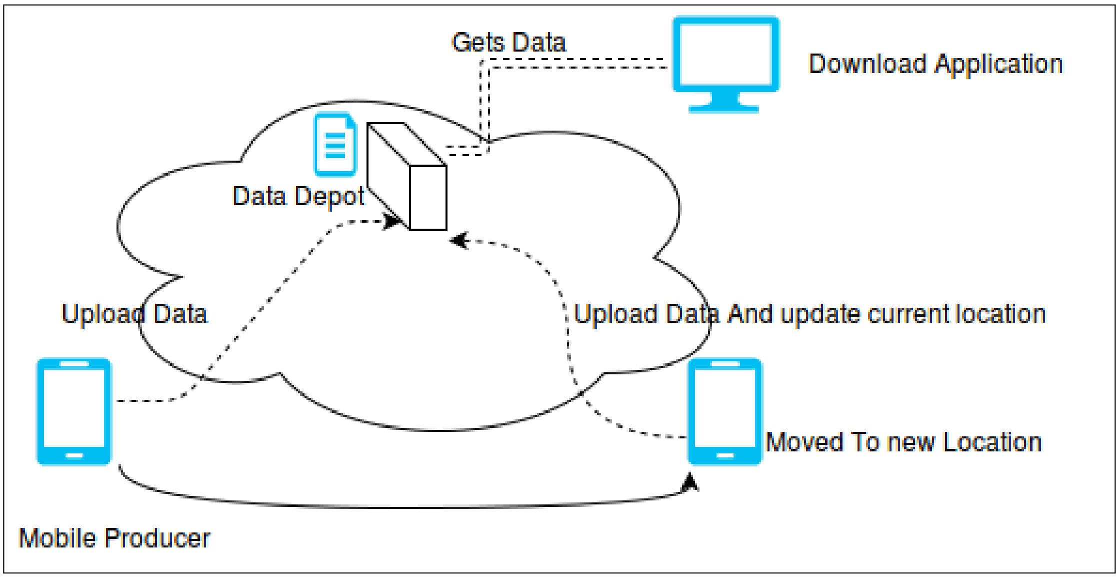

Figure 2.4: Data Rendezvous based scheme [3]

problems are solved in present scenario by providing distributed availability of content for example YouTube servers and drop-box etc.

CHAPTER 3

KITE MOBILITY MODEL FOR NDN

This thesis extends one of the tracing based solution- Kite model. The reasons to choose kite model are: Kite model is motivated by adaptive forwarding plane of NDN and traces are side-effect of forwarding plane. There is no need of explicitly mapping the data name to current location. There are registration overhead and hand-off overhead involved in updating current location on mapping servers in mapping based schemes. There is no such need in Kite model. The chapter explains kite framework; packet processing of interest and data packets; drawbacks of Kite model; reason of drawbacks and result comparison of kite and mapping based scheme as given in [2].

3.1 Kite Framework

Based on how correspondent node can connect to producer mobile node, there can be two variants of Kite [2], first Direct Kite when correspondent node is immobile and reachable by routing plane for example Facebook server, then mobile node can setup direct traces to correspondent node. Second Indirect Kite when correspondent node is also mobile, the trace is setup to immobile anchor which is hosting mobile services in routing plane for example Dropbox server, and mobile node wants to upload images in Dropbox. In Indirect kite, CN can connect to mobile by first sending interest to immobile anchor and then following the traces as shown in Figure 3.1.

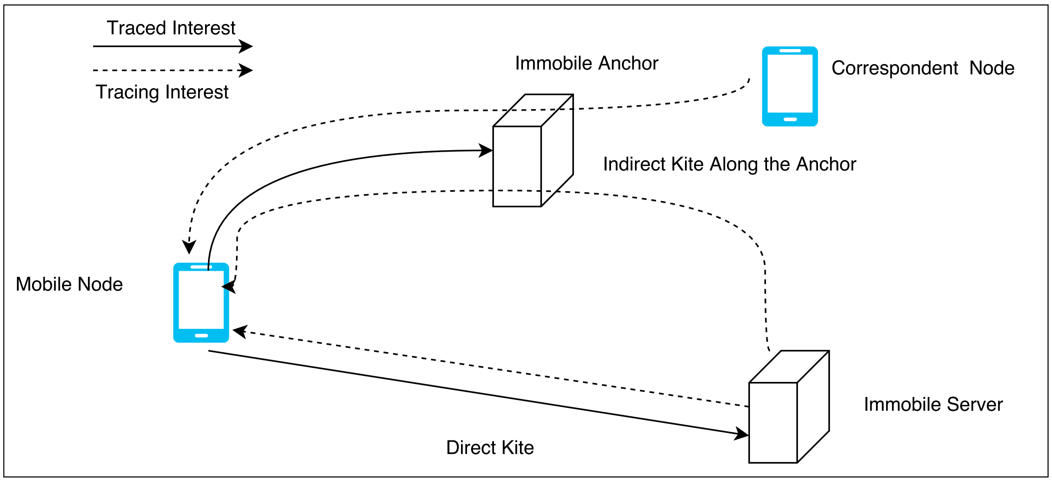

Figure 3.1: Kite Framework [2]

Terminologies used:

1. Trace source: It is the Mobile Node that needs to send traced packet so that it can be reached by other nodes in the plane.
2. Traced Interest: Tracing interest trace back to MN by the traces left by traced Interest
3. Tracing Interest: Interest traveling along the trace of previous Interest is called tracing interest.
4. Trace anchor: Which is reachable by announcing the trace prefix into the routing plane. That,Äôs why there is an assumption in Kite model that the underlying network should be stable so that an MN can reach trace anchor. It is application specific

3.2 Packet Processing

Figure 3.2 and Figure 3.3 explain the process of forwarding packet at each node in Kite model [2]. In NDN there are two types of packet 1) Interest packet that expresses the data name request in network, 2) Data packet that carries the data requested by Interest packet. In kite Interest packet can be of two types 1) Traced packet that generates the PIT entries in Trace Forwarding Table (TFT), 2) Tracing packet that follow the traces developed by Traced packet; each tracing packet has entry in Trace Name Table (TNT). TFT and TNT are two new tables are added within PIT table [2]. Figure 3.2 shows processing of traced packet. Figure 3.3 shows processing of tracing packet.

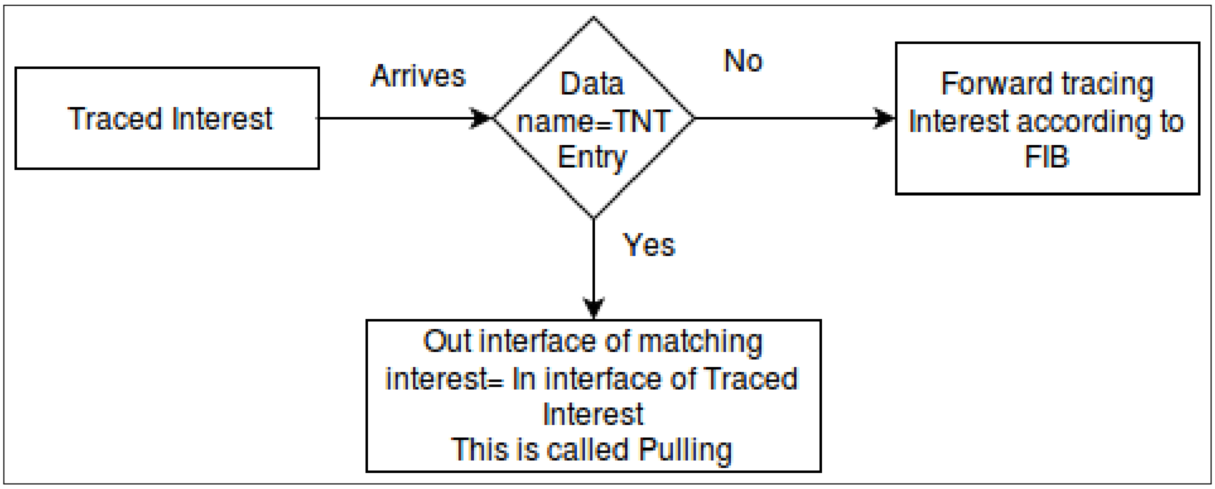

Figure 3.2: Packet processing of Traced Interest [2]

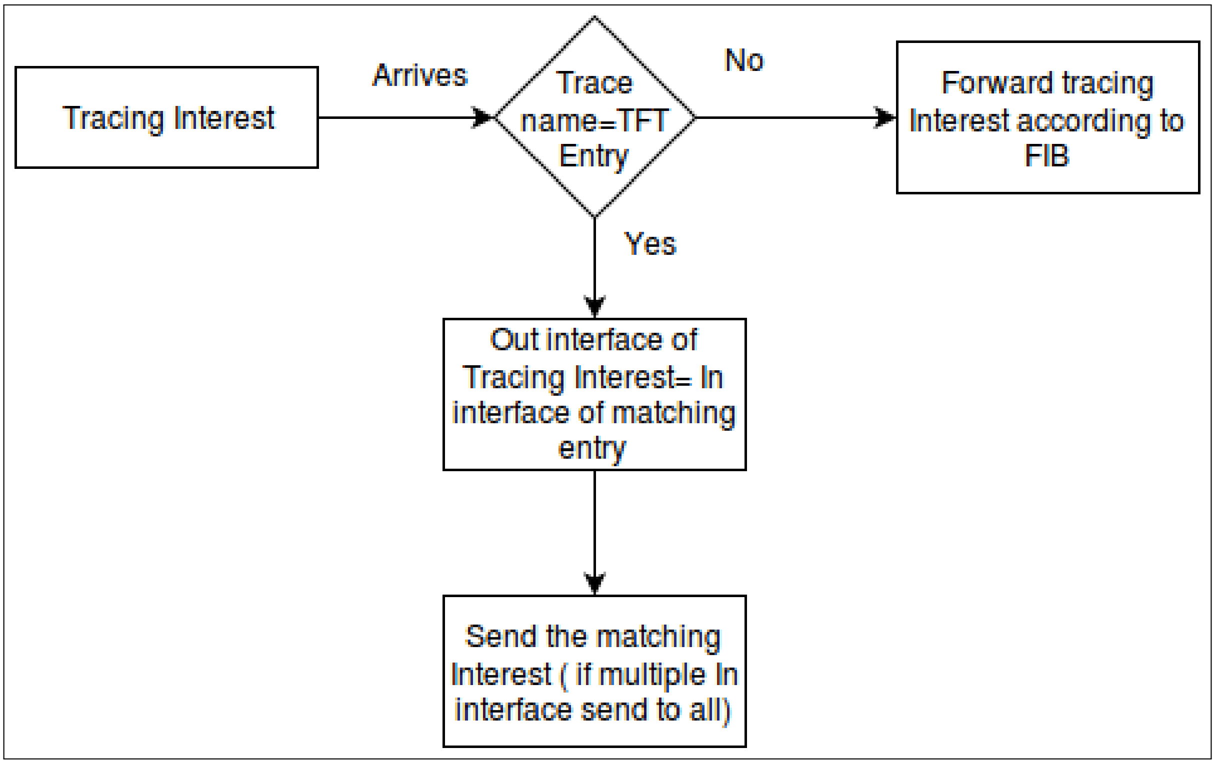

Figure 3.3: Packet processing of Tracing Interest [2]

3.3 Drawbacks Of Kite Model And Reasons For Drawbacks

1. A drawback of Kite model is that it is not suitable for mobile ad-hoc networking architecture, it assumes that the underlying routing plane is stable. The reason behind this is that to maintain trace and to setup trace the underlying network should be stable.

2. Kite is not suitable for long time relocation of mobile node [2], because of signaling overhead of Kite when the mobile node stays at a particular location for relatively longer duration. The tracing based chasing of mobile node is beneficial when mobile node is moving between different routers very fast. If a node is moving very fast in a region or within a router that means itś staying only for a short duration at same location. Here relocation is how long mobile node stays connected to a router, whenever it moves to new router it has relocated.

   Reasons why kite model is not adaptable for longer relocation of mobile node are:

   - Application is unaware of location change hence cant́ adopt according to how frequently location is being changing.
   - Re-transmission timer is not dynamic, if mobile node is staying for longer duration then re-transmission timer should be long enough so that the entries in PIT stays for longer duration. And if the relocation period is shorter than re-transmission timer should be short so that after moving to a new location the previous traces fade away fast.

   Results in [2] show that Kite over performed the mapping based solutions. The average uploading rate of of Kite is up-to 1.54 times of that of MMS.

3.4 Kite Performance At Lower Speed

The mobile node speed can serve as measure of relocation, at lower speed mobile node stay connected to same router for longer time, and at high speed mobile node changes router frequently hence stays connected to a router for lesser time. So we may assume lower speed as longer relocation period and relatively higher speed as shorter relocation period. It is possible that mobile node may move at high speed but stay connected to same router, but this has not been considered because in given scenario mobile node walks randomly and at high speed it changes router frequently. The simulation runs 50 times for various speeds lower than 50 m/s (5, 10, 15.. 30). The other simulation parameters are same- 4x4 grid and number of satisfied interest is considered as rate of upload. The graph in Figure 3.4 shows degradation in performance of Kite. Mapping based Mobility Support(MMS) performs better at lower speed of mobile node.

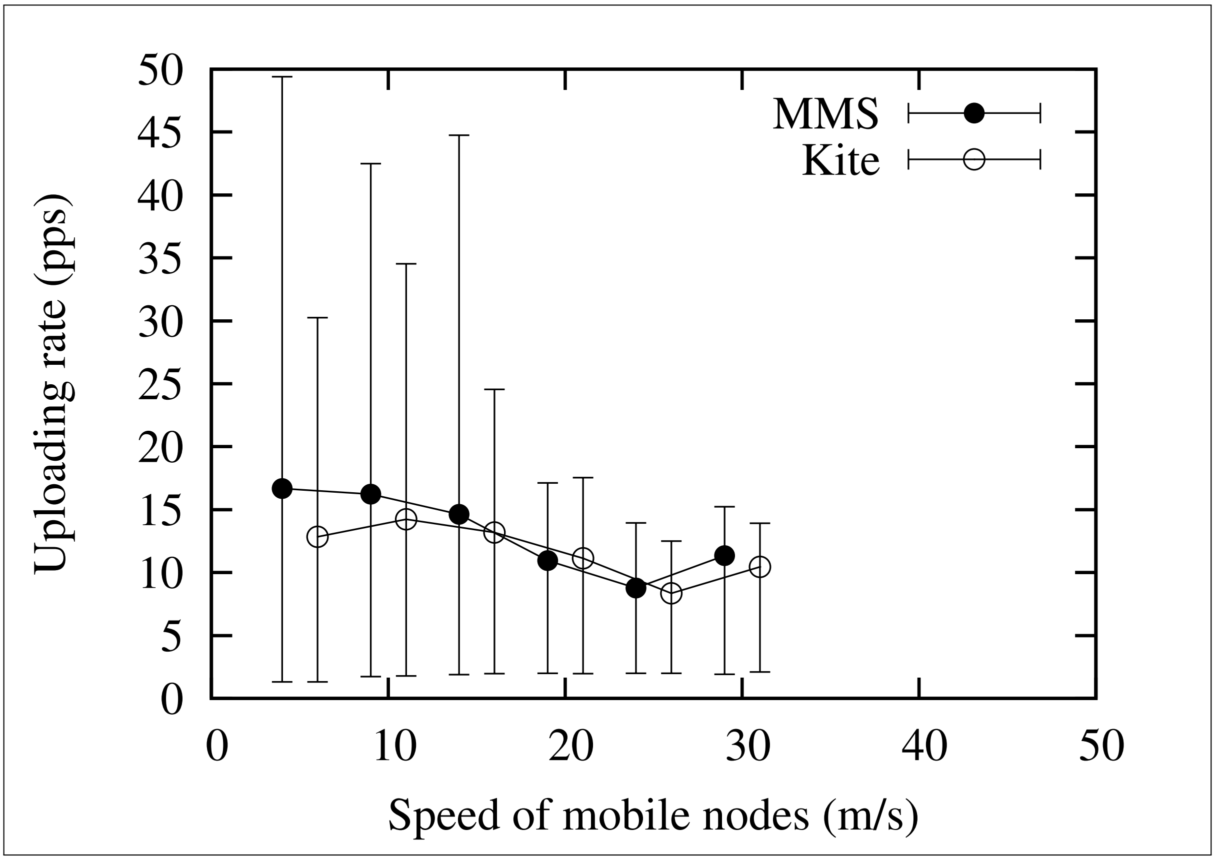

Figure 3.4: uploading rate at Kite at lower speed 5 to 30 m/s

CHAPTER 4

Relocation Adaptable Kite: Mobility model for NDN

As mentioned in chapter 3, Kite model is not suitable if mobile node stays at same location for long time then moves to another location and again stays there for long time.To make Kite relocation adaptable for longer durations, dynamic retransmission timer can be used to set life-time of packet. In this thesis retransmission time out (RTO) is equal to life time set in Interest packets, so both are used interchangeably. If life-time of Interest is long then packet entry stays in PIT for long time and there is no need to send interest packets frequently to prolong the trace. Also dynamic retransmission timer makes RA-Kite suitable for short stays of mobile node. If mobile node stays at same location for more than threshold time than FIB entry is created for the name prefix under which mobile node produces data. This threshold can be 12 hours, 24 hours or may be more. Now the mobile producer can be reached globally by routing plane.

In Internet, dynamic retransmission of unsatisfied packet is handled by TCP layer, but because NDN doesnÄôt have Transport layer, then either Transport layer functionality can be implemented by application layer or network layer. Due to lack of time we implemented only application layer algorithm. The forwarding strategy is almost same as [2] except, that is after some threshold time PIT entry corresponding to a mobile producer is moved to FIB table, otherwise the packet processing is same as explained in chapter 3 and given in [2]. In following description thresh_RTO and thresh_stay are same, and both are used interchangeably.

There are three algorithms that are implemented in RA-Kite:

ALGORITHM 1: on_association

Data: start-time, RTO

initialization;

start-time ← simulator.now()

RTO ← 2.0 seconds

Algorithm 1 is called when there is an association event on connection to new router. To record how long mobile node is connected to an access point, we should know the time of association to the AP and time of association to the AP. The time of association is stored in start time_variable. On disassociation stop_time is recorded, the stay is calculated as difference of stop_time and start_time . Also retransmission time out (RTO) is set to 2 seconds. Retransmission time out is the time interval, at this rate traced packet are sent.

ALGORITHM 2: on_deassocition

Data: stay1,stay2,thresh-RTO)

initialization;

stay2 ← stay1

stay1 ← simulator-now()-start-time

thresh-RTO ← 0.6*stay1+o.4*stay2

Algorithm 2 is called when mobile node is leaving the router(on de-association). The end time is saved and recent stay time is calculated by subtracting the start time. Then calculate the threshold stay based on these previously recorded stay variables. From these previous stay variables, it can be conferred that mobile node will at least stay at next location till this new threshold_stay time. So at next location we can safely increase the lifetime (RTO) of the packet multiplicative till it reaches this threshold stay, Once it reaches threshold_stay we increment life time (RTO) additively.

ALGORITHM 3: on_timeout

Data: RTO,long-relocation,thresh-RTO)

```
initialization;
temp-stay ← simulator-now()-start-time
if temp-stay > long-relocation then
  | FIB-Entry ← (prefix, metric,face)
else
  | if 2*RTO < thresh-RTO then
  |   | RTO ← 2*RTO
  | else
  |   | RTO ←RTO+1
  | end
  | send Interest()
end
```

Algorithm 3 is called when there is a time out traced packet. Hence we need to send next packet to prolong the PIT entry along the trace. the current stay time is calculated here, by subtracting current time and start_time. if this current_stay time is greater than long_relocation time then it means mobile node has stayed more than long_relocation time; and we can move the PIT entry of this node to FIB entry. After moving PIT entry to FIB entry, there is no need of sending more traced packets. But if current_stay time is less than long_relocation time, it means mobile node may or may not stay long. In that case keep sending traced packets to prolong the trace. But we can dynamically set life-time of these new packets by comparing the threshold_stay. Threshold_stay is calculated in algorithm 2, this threshold_stay is average staying period of mobile node based on it's previous stays. If life_time of new packet is set to this threshold stay, then we may need one packet to prolong trace by setting life-time of this packet to threshold stay. But in reality its not true, so we increment life_time of packet twice the previous lifetime till it reaches this threshold_stay. This multiplicative increment is called ratio R.

Initial values of each variable, long_relocation period is 10 seconds, the initial values of all variables are- stay1=0,stay2=0 and threshold_stay=0;the other values are set during simulation, when mobile node starts to move, it connected to a router(by coming in the transmission range of access point) let's suppose at simulator time 2.0. so start_time= 2.0 and retransmission time out(RTO) is set to 2.0; the retransmission time out is equal to lifetime of the interest packet, to ensure that packet stays in PIT till next traced packet is retransmitted. So the RTO is always equal to life-time of interest packet. after association to AP, first traced interest packet is sent. After 2.0 seconds retransmission timer goes off, next traced packet is to sent at 4.0 simulator time. And time_out procedure is called. It decides weather to send next packet or not and if yes then with what life_time value.

4.1 Setting The Parameters

4.1.1 Predicting Time Of Current Stay

To accurately predict the time of current stay, we need to decide how much information from the past history about previous stays should be considered. How many previous stays are sufficient to accurately predict next stay time? Using these stays we can predict the current stay, the weighted average of these stays gives the approximate value of current stay. And this approximated current stay can be used to set retransmission timer using this equation

$$thresh\_RTO = w1 * stay1 + w2 * stay2 + w3 * stay3 + w4 * stay4.....wN * stayN$$

Taking the ratio between each weight 1.5, the equations for different values of N are shown below.

For N=0 no stay record is maintained. For N=1 only one previous stay record is maintained, and the equation is

$$thresh\_RTO = stay1$$

For N=2 only two previous stay record is maintained, and the equation is

$$thresh\_RTO = 0.6 * stay1 + 0.4 * stay2$$

For N=3 three previous stay record is maintained, and the equation is

$$thresh\_RTO = 0.476 * stay1 + 0.315 * stay2 + 0.21 * stay3$$

For N=4 four previous stay record is maintained, and the equation is

$$thresh\_RTO = 0.415 * stay1 + 0.272 * stay2 + 0.184 * stay3 + 0.123 * stay4$$

The graph in Figure 4.1 shows the number of traced packets generated for different values of n (n = 0, 1, 2, 3, 4) and for different values of speed of mobile nodes.From the graph we conclude that best result is provided by recording two stays.

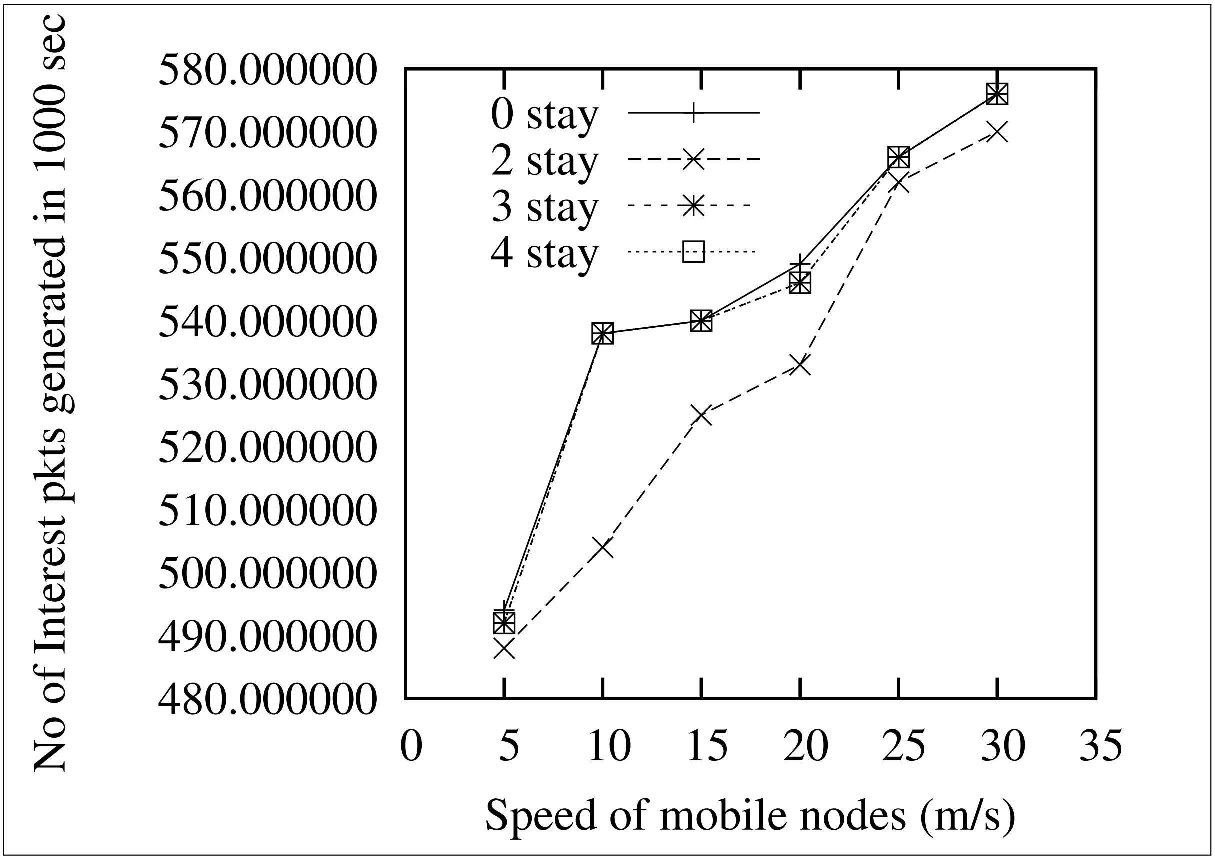

Figure 4.1: Number of previous stays

4.1.2 Setting Value Of Ratio Between Weights Assigned To Different Previous Record of Stays

To calculate the close approximate value for current stay we used the following polynomial equation

$$thresh\_RTO = w1 * stay1 + w2 * stay2 + w3 * stay3 + ...wn * stayn$$

This thres_RTO is estimated current stay which is used to set RTO value when mobile node retransmits the Interest packet. In this equation w1, w2......wn are weights in order w1>w2>w3>....>wn. This is because recent stays should be given more preference than older stays in calculating the current stay. The ratio between two weights is termed as R. Now we have to examine which ratio performs better to get more accurate next staying time of mobile node.

We experimented with ratio 1.25, 1.5 and 1.75. The graph in Figure 4.2 shows the signaling overhead for different values of R, where R is same as r. The best result is produced by ratio value 1.5.

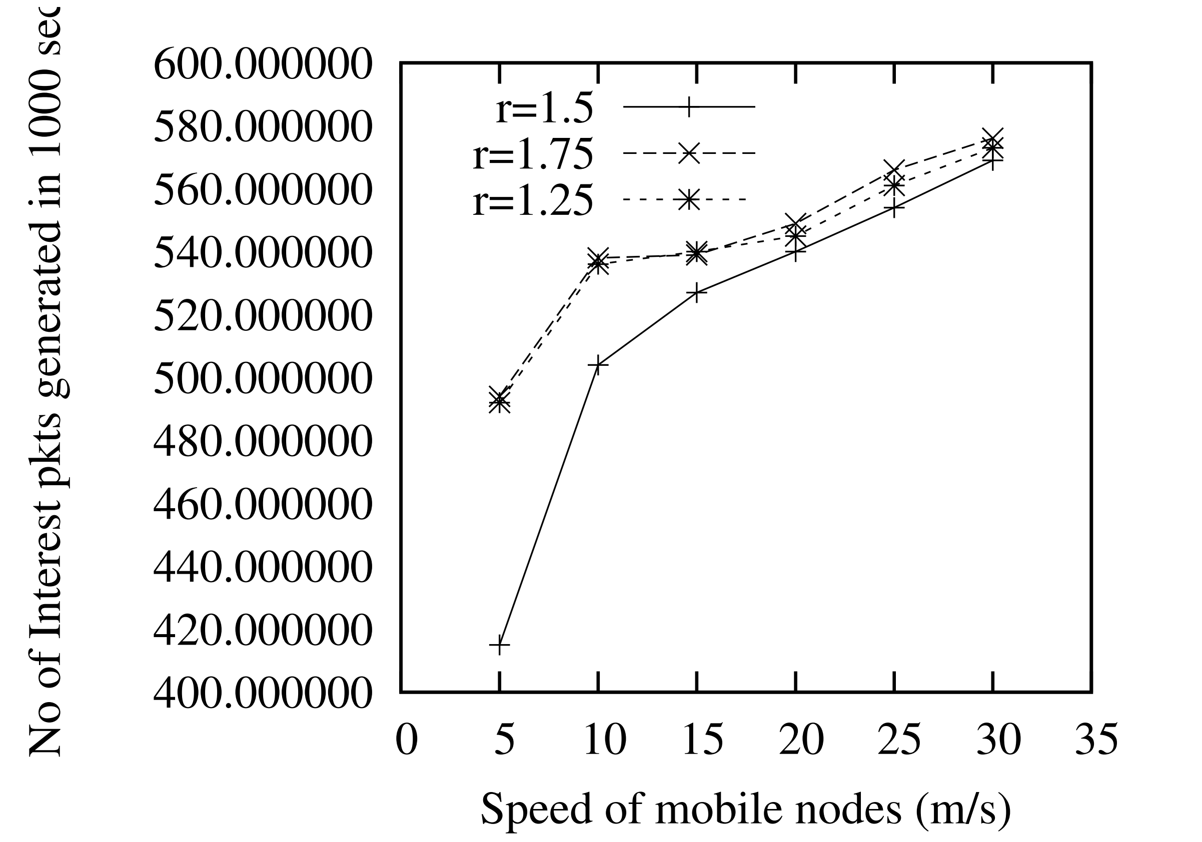

Figure 4.2: signaling overhead for different values of R, r is as same as R

4.1.3 Setting Multiplicative Increment In RTO

In algorithm 3 when we send an interest packet on timeout, we set the life-time of packet dynamically. We increase lifetime of packet two times when current thresh_stay is more than twice of current value of RTO.

if *2*RTO < thresh-RTO* **then**
   RTO ← 2*RTO

else
   RTO ←RTO+1

end

The value 2 is called multiplicative increment in RTO. The purpose of having this small check to decrease the signaling overhead. It only means that we increment value of RTO multiplicatively when it is safe to do so, otherwise RTO is incremented additively. We experimented over two values of this multiplicative increment 2 and 1.5. The Figure 4.3 shows graph with different values of r and n. Here r is the multiplicative increment and n is again the number of previous stays records, just to be in safer side that we are incrementing right value to RTO, taking right value of number of previous stays and assigning correct weights to different stays. The graph in Figure 4.3 shows least signaling overhead when number of stays are 2, multiplicative increment is 2 and the weights ratio is 1.5.

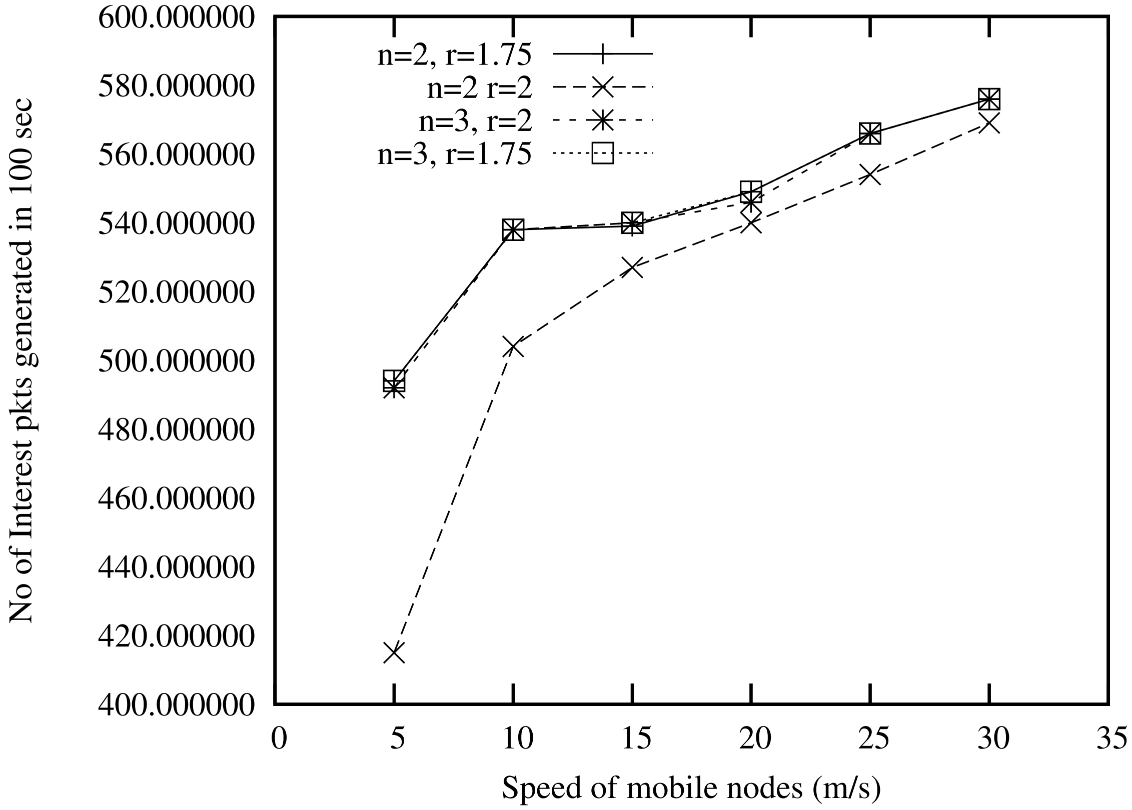

Figure 4.3: performance as a function of multiplicative increment in RTO

4.2 Scalable RA-Kite

RA-Kite is scalable, mobile nodes staying for longer time has no scalability issue, because they are not moving frequently. Mobile nodes staying for lesser time will not create FIB entry and can be traced back via hop-by- hop PIT entries.

4.3 When Mobile Node Moves After FIB Entry Has Been Made

When a mobile node moves from the place where FIB entry of that particular mobile node prefix has been made, then packet will arrive at old location as well as along the new trace path set by mobile node at new location. The upload rate will not be affected, but there will be unnecessary traffic overhead in the network. This situation is handled by intelligent forwarding strategy. when an Interest packet is send to an interface, if the data packet is not replied back (because mobile node has moved), then that particular interface is classified as RED. Here RED interface means, the interface is down and this interface may get alive in near future. When interface is down then a NACK is sent to up-router from which the Interest packet arrived. This way the trace will fade away in some time and mobile node can keep moving without any interruption in data upload rate and maintaining the signal overhead in limit.

CHAPTER 5

RESULTS AND SIMULATION OF RA-KITE

The simulations in this section are set up in a field of 400 X 400 m^2 with an infrastructural network composed of 16 fixed routers, which are allocated in, and connected via point-to-point links as, a 4 X 4 grid in 300 X 300 m^2. Every router and MN has a Wifi device with default setting.

Mobile node is producer that sends traced interest and sends back data packets on the request of Interest. The server is installed on grid node (0, 0), which is correspondence node (CN). The server application is credit-based flow control. On receipt of traced Interest sever sends credits number of Interest packets. In this scenario credit is set to 2. On arrival of each traced interest two Interest packets are sent. And on arrival of each data packet one Interest packet is sent. Retransmission timer on server application is set to 2 seconds. When an Interest is not satisfied in 2 secs, on time out Interest packet is sent again. The router in between the path (grid routers) do not resend unsatisfied packet.

The mobile node walks in random 2d motion, it changes its direction after every 150 m. The default retransmission timer is set to 2 secs, after each stay during simulation this retransmission timer is changed dynamically.The threshold stay is assumed to be 10 secs. As the grid is 4x4 in 400 x 400 m^2 area. Each router is separated by 100 meters, if a mobile node moves with 50 m/s, then mobile node changes router in 2 secs. So it is safely assumed that if a mobile node stays at a location for more than 10 secs it means it has stayed for long enough to make its entry in FIB table.

Two performance parameters were used to compare the performance of RA-Kite with Kite; signaling overhead and number of packets satisfied per second (Upload Rate). Signaling overhead is number packets generated per second to signal the current location of mobile node. Packets to tell the current location are called traced Interest packets that are generated by mobile node. So counting traced interest packet will give the signaling overhead. Uploading rate is number of Interest packets that has been satisfied by data packets in one second. So counting the number of satisfied Interests at server will give uploading rate.

Each simulation runs for 100 seconds, and each simulation is run for 50 times to calculate average traced packet or satisfied Interest. We calculate total number of interest packets satisfied during this simulation time. let's suppose total number of interest packets satisfied in one simulation are 30 packets. So the uploading rate will be 30 packets in 20 seconds, which calculated as 1.5 packets per second(pps).

5.1 Comparison RA-Kite And Kite

5.1.1 Signaling Overhead

Figure 5.1 shows the graph of signaling overhead using Kite and using RA-Kite. In kite the number of traced packets generated at the rate of one traced packet in two seconds, so the maximum rate of sending packets would be half packet per second (pps).

Using Kite, Average Interest generation rate is 0.49 pps (very close to max rate 0.5 pps). This average is taken over 60 different speeds from 5 m/s to 300 m/s. For each speed its 0.49 pps because retransmission of Interest packet is static. But in RA-Kite, for lower speed the over-head is low as compare to higher speed. Hence using RA-Kite signaling overhead is 0.3174 pps for same setup, over 60 different speeds from 5 m/s to 300 m/s.Overhead reduced to 35.22% of that of Kite. From the graph in Figure 5.1 overhead increases with speed. After speed 150 m/s there is only small change in overhead. The gain is more at lower speed.

The graph in Figure 5.1 also shows that Kite is not adaptable according to speed. Signaling overhead is almost same for every speed. But in RA-Kite when speed is low less signal overhead has achieved. In the real scenario more speed means more number of traced packet should be sent to tell the current location, because location is changing very fast. Hence for RA-Kite signaling overhead increases with speed but stays less than that of Kite for every speed.

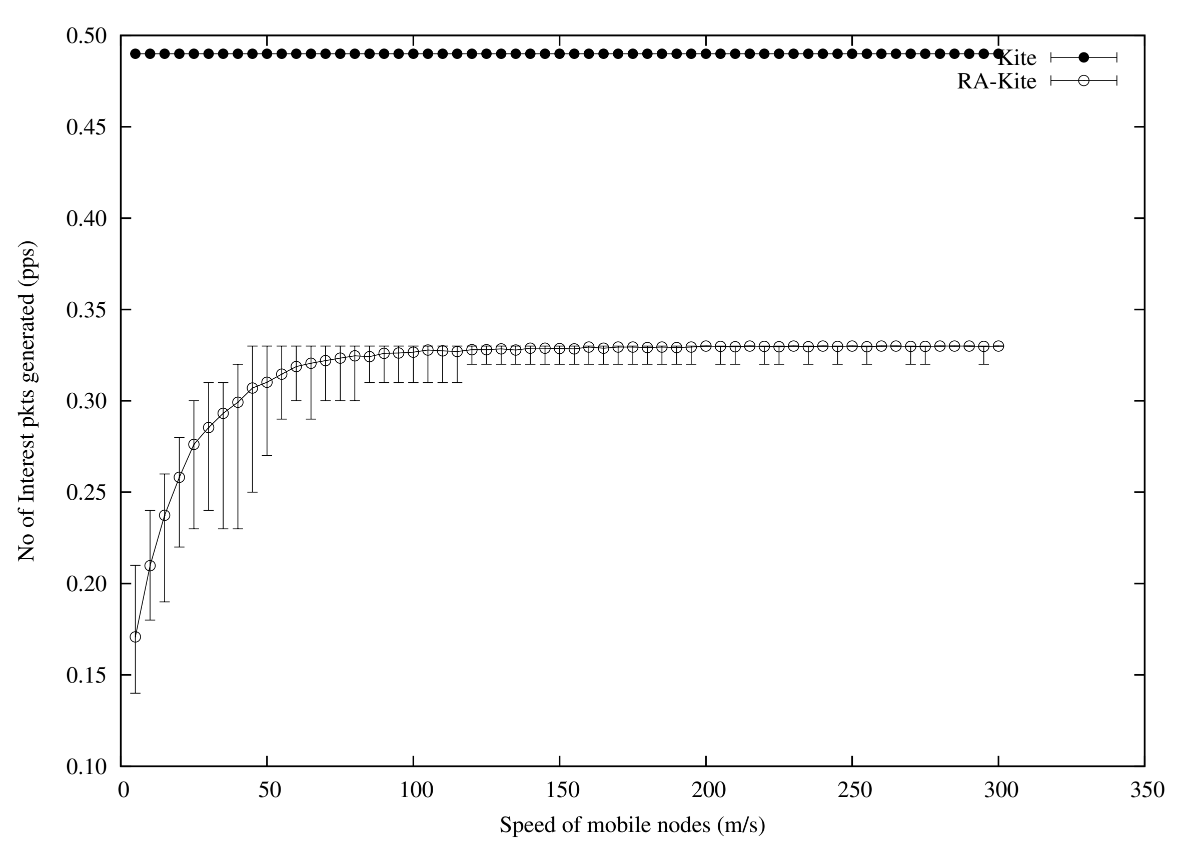

Figure 5.1: Signaling Overhead using RA-Kite and KIte

5.1.2 Upload Rate

Figure 5.2 shows graph of upload rate using Kite and using RA-Kite. Kite shows same behaviour as explained in [2]. As the credit has been changed from 20 to 2, so the graph uploading rate has become 15 packets per second to 1.5 packets per second for speed 50 m/s in Figure 5.2. For the same scenario RA-Kite performs better than Kite, for lower speed the effect is more.

Upload speed decreases as speed of mobile node increases, because when mobile node moves faster more number of packets get dropped and unsatisfied. Kite average upload rate over 60 different speeds (5 to 300 m/s) of mobile node is 1.2196 packets per second. Whereas using RA-Kite average upload rate over 60 different speeds (5 to 300 m/s) of mobile node is 1.8569 packets per second. RA-Kite has over performed Kite by 1.5225 times. When lower speed (5 to 30 m/s) of MN is considered, average uploading rate of Kite is 1.8246 packets per second, and average uploading rate of RA-Kite is 5.078 packets per second. The performance of RA-Kite is better at lower speed. Upload rate is increased and signaling over head is reduced.

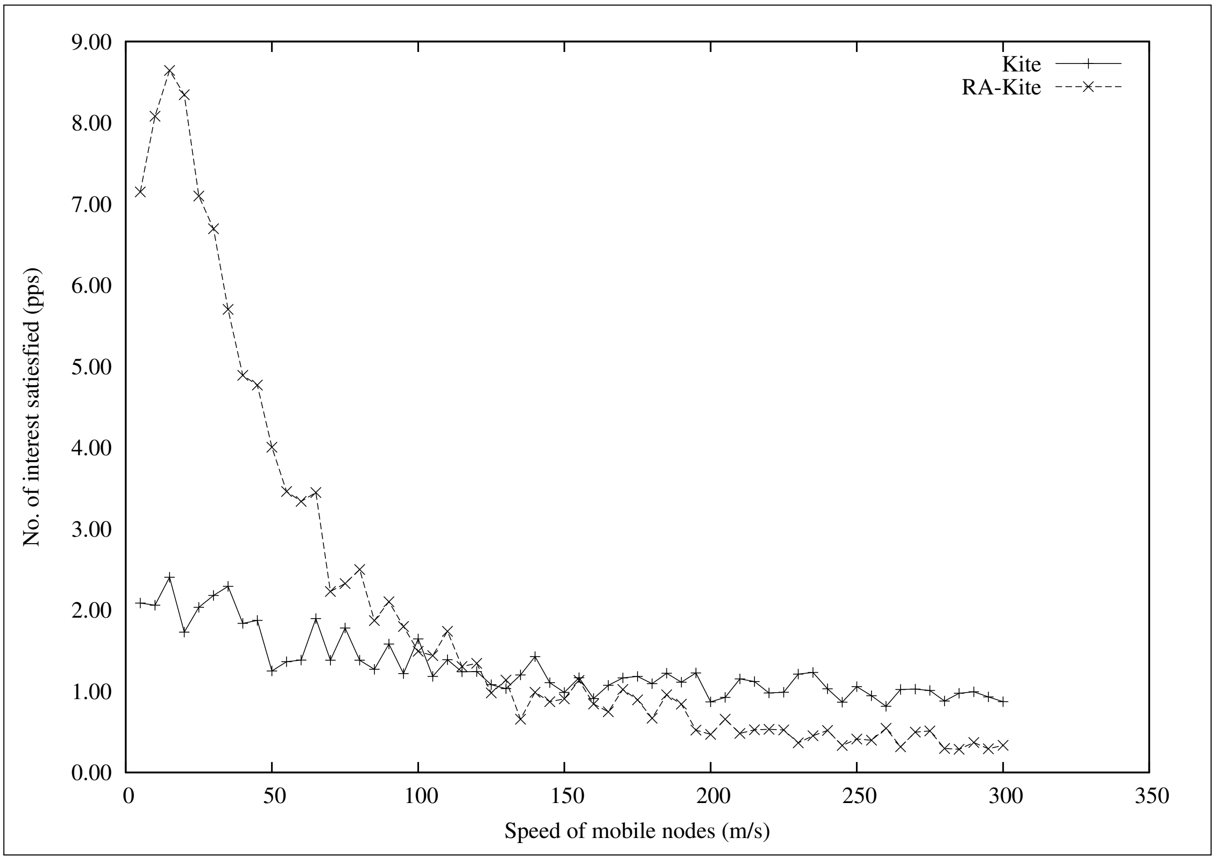

Figure 5.2: Upload rate Using RA-Kite and Using Kite Speed from 5 to 300 m/s

5.2 Comparison RA-Kite And MMS

5.2.1 Signaling Overhead

MMS is mapping based Mobility support scheme discussed in Chapter 2. Signal overhead using MMS and using Kite are same, because MMS has been implemented by disabling pulling mechanism in Kite. This means the mobile node will only notify the server about its current location. The Interest packets to notify the new location are sent at interval of 2 seconds just like Kite. So number of traced packet generated in MMS in 20 seconds will be at max 10, so the interest generation rate is half packet per second, using MMS average of 0.49 packets per second. But in RA-Kite, for lower speed the over-head is low as compare to higher speed. Hence using RA-Kite signaling overhead is 0.3174 pps for same setup, over 60 different speeds from 5 m/s to 300 m/s. Overhead reduced to 35.22% of that of MMS. The graph of signaling overhead using Kite and using MMS will be same as in Figure 5.1.

5.2.2 Upload Rate

Figure 5.3 shows upload rate using RA-Kite and using MMS. Simulation runs 50 times, for 100 seconds, for 60 different speeds (5 to 300 m/s). Upload rate of MMS is 1.1354 packets per second. Upload rate using RA-Kite using same parameters is 1.82996 packets per second. RA-Kite performs 1.612 times better than MMS.

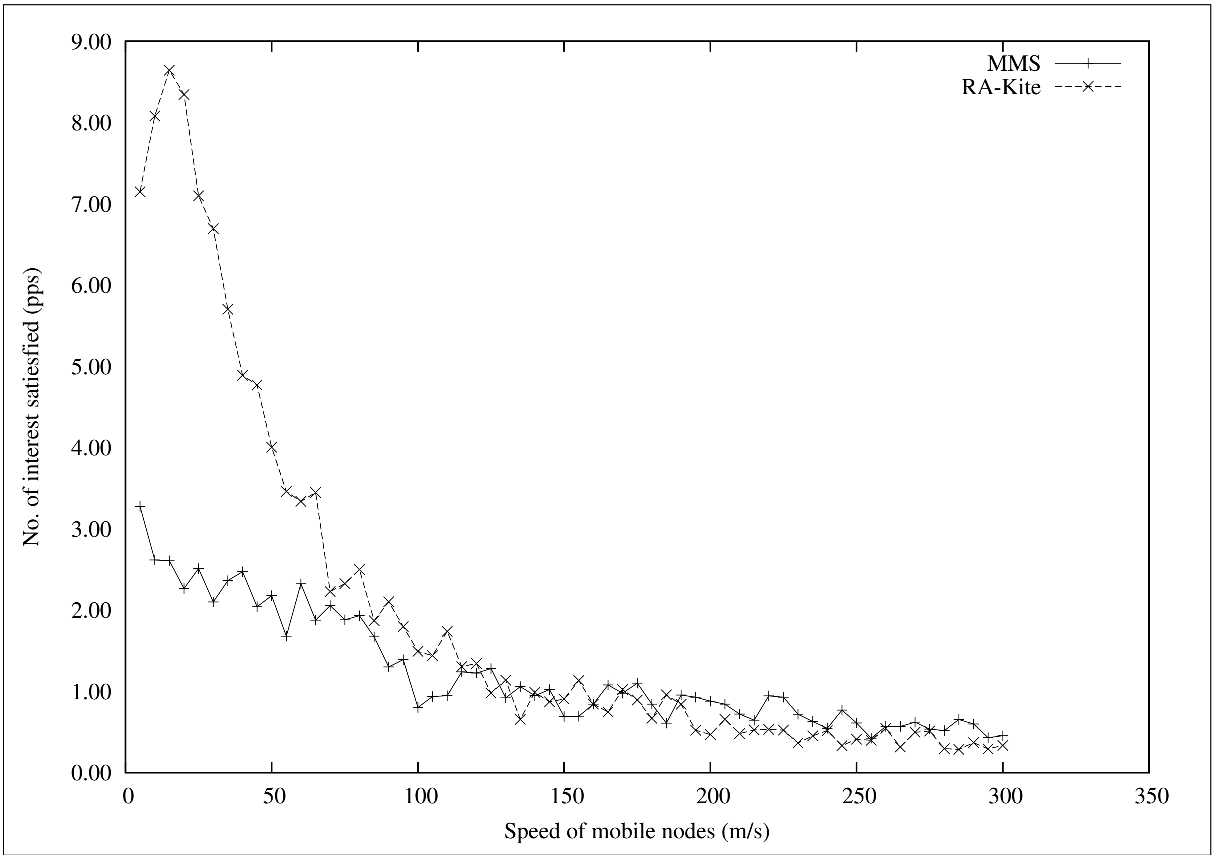

Figure 5.3: Upload Rate vs speed (50 to 300 m/s)

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

The dynamic re-transmission timer makes RA-Kite more suitable than other available solutions in both categories mapping based as well as in tracing based. RA-Kite performs better in realistic scenario when speed of mobile node is taken 5 to 100 m/s; 5 meter per second is walking speed of a person and 30 m/s(108 km/h) speed of a car or train, up to 100 m/s. This thesis aims at reducing signaling overhead and we achieved reducing overhead by up-to 35.22%. RA-Kite partially empowers the application protocol developers to devise their own designs of mobility support which can be tailored to their scenarios.Traces setup and maintenance can be carried out according to their needs, rather than only relying on some middle-wares or the network layer. RA-Kite is a unified solution to long-term and short-term relocation of mobile node.

There are some assumptions taken like underlying network should be infrastructure network to setup the trace path. If a middle router in the trace route moves to new location than the trace path is broken and again flooding will be used to send data packets at each available interface. In future we can make Kite RA-Kite suitable for adhoc networks as well. The other research direction can be providing mobility support in network layer. Network layer can observe the traced packet from the same source, this observation can give a router in trace path the idea that mobile node has stayed sufficiently long. Because the network layer can not change the packet in transmission it can only observe the traffic and derive that mobile node has stayed at a location.

BIBLIOGRAPHY

[1] Zhang, Y., Afanasyev, A., Burke, J. and Zhang, L.,), “A Survey of Mobility Support in Named Data Networking.,” 2016.

[2] Y. Zhang, H. Zhang, and L. Zhang, “Kite: A mobility support scheme for ndn,” in *Proceedings of the 1st international conference on Information-centric networking*, pp. 179–180, ACM, 2014.

[3] V. Jacobson, R. L. Braynard, T. Diebert, P. Mahadevan, M. Mosko, N. H. Briggs, S. Barber, M. F. Plass, I. Solis, E. Uzun, *et al.*, “Custodian-based information sharing,” *Communications Magazine, IEEE*, vol. 50, no. 7, pp. 38–43, 2012.

[4] Kleiner Perkins Caufieldamp; Byers (KPCB), “2015 Internet Trends,,” 2015.

[5] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, “Networking named content,” in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pp. 1–12, ACM, 2009.

[6] A. Azgin, R. Ravindran, and G. Wang, “A scalable mobility-centric architecture for named data networking,” *arXiv preprint arXiv:1406.7049*, 2014.

[7] F. Hermans, E. Ngai, and P. Gunningberg, “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*, pp. 13–18, ACM, 2012.

[8] D. Li and M. C. Cuah, “Scom: A scalable content centric network architecture with mobility support,” in *Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference on*, pp. 25–32, IEEE, 2013.

[9] J. Augé, G. Carofiglio, G. Grassi, L. Muscariello, G. Pau, and X. Zeng, "Anchorless producer mobility in icn," in *Proceedings of the 2nd International Conference on Information-Centric Networking*, pp. 189–190, ACM, 2015.