UNIVERSITÄT BERN

Bloom filter-based routing in NDNs, March 21, 2016

Ali Marandi
Communication and Distributed Systems
Institute of Computer Science
University of Bern, Switzerland
marandi@inf.unibe.ch

Outline

- NDN
- Routing in NDN
- Bloom Filter (BF)
- My BF-based based routing methodology for NDN
- A little, but important, related work

NDN (1)

- Today's internet architecture (IP-based)
  - *"Packets named only communication endpoints"* , not contents…
- Today, internet is rather a *distribution network,* i.e. not anymore only a *communication network*
- **NDN main idea**: kind of generalization so that *"packets can name objects other than communication endpoints"*
- *"The design also builds in security primitives (via **signatures** on all named data)"*
- In fact producers sign the data to gain receivers' *trust*

NDN (2)

- NDN data retrieval
  - *Consumers* diffuse *interest* messages towards *producer*
  - Once *content* is discovered, it gets back from the reverse path, and will be cached (congestion control)
  - Loop freedom is ensured for interests using *nonces*
- NDN data structures
  - PIT, CS, FIB
- PITs also maintain *route traces,* and *nonce* information
- Unlike IP, FIB could have a list of faces for each prefix
- More than one face for a prefix ? *strategy* chooses the face(s) over which interest will be sent

UNIVERSITÄT BERN

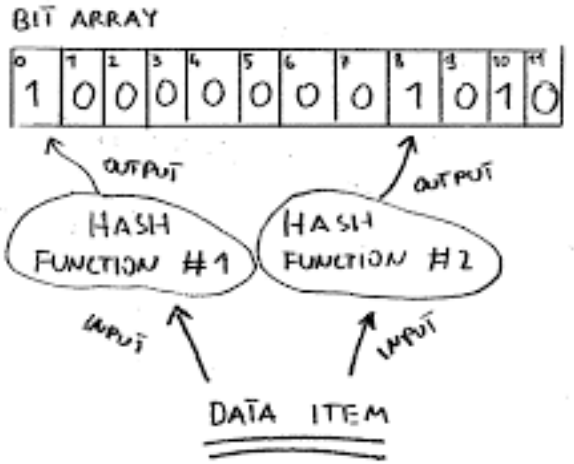
Routing in NDN

- An NDN routing protocol performs different tasks
  - Locating the content
  - Filling the FIBs
  - Interface ranking
  - Adaptation to topology changes
    - Resiliency to link *failure*
    - Ability to adapt to link *recovery*
  - Adaption to prefix changes

- **Important question:** Knowledge from topology ? To which extent ?

BF

- **Usage**: representing large sets in a compact way
- What is BF ?
  - A bit array initialized by zero
  - certain hash functions
- The *insert* and *contains* operators are performed by hash functions
- False negative is impossible
- False positive is the only price to pay !



- **Benefit in networking**: reduced overhead of transferring or storing informaition

Theoretical tradeoffs

- m: number of bits
- n: number of inserted elements
- *k:* number of hash functions

$$k = \frac{m.\ln 2}{n}$$

- p: false positive probability

$$m = \frac{-n \ln p}{(\ln 2)^2}$$

- *union* and *intersection* of BFs with the same size and set of hash functions can be implemented with bitwise OR and AND operations, respectively.
- Union is lossless, but for intersection this property is not garaunteed

Introduction to my methodology

- Main incentives
  - *Content advertisement*
  - *Interest accumulation*
  - We can design a BF-based approach to *routing in NDN*
- Basic idea
  - Content advertisement from upstream
  - Interest accumulation from downstream
  - Benefiting from *union* when *multipath*
  - Benefiting from *intersection* for FIB population

Problem formalization

- Two sets for node i
  - D_i set of demand interests, the set of all interests that can be serviced by node i
  - C_i set of content offers, the list of all contents that can be accessed through the node i
- $D_i \cap C_i$ will contain the set of interests that can go through node i
- My aim is to propose a mechanism that will generate for each node the relevant sets and will calculate the intersection.

Mechanism

- Assumptions:
  - Fixed size BFs, i.e., a given number of bits and a given and fixed number of hash functions,
  - They each can store up to n values with a false positive probability p
- BF^I for accumulated interests
- BF^C for content advertisements
- $D_i = \text{union}\ (BF^I)$
- $C_i = \text{union}\ (BF^C)$
- Finding $D_i \cap C_i$ is straightforward
- If two filters match, it means that node i can be on a path connecting the Interest to the Content
- We store such matches in a *match record*

Link failures /recoveries

- A link is considered failed when it does not reply with a *contentAdvert* or *accumInterest* for a while
- Routing updates from past will be no longer valid after failure detection, and must be removed
- Upon recovery, the node will again send *contentAdvert* and *accumInterest*; we can then proceed with the new and the correct routing information

UNIVERSITÄT BERN

Security

- Routing updates must be carried in an NDN data packet
- Routers can verify the signature of each routing message to ensure that it was generated by the claimed origin router and was not tampered during dissemination

Related work on routing in NDN

- There are plenty…But the main ones:
- NLSR, I discuss concisely
- Bloom filter-based routing: beyond the scope of this presentation

UNIVERSITÄT BERN

My approach vs NLSR

- NLSR first builds the topology at every node, then it can perform interface ranking. My approach, however, acts without topology awareness.
- Each node in NLSR has to maintain a lot of information
  - Topology information (each node needs to know topology precisely)
  - Dijkstra output for all of nodes (for each node multiple shortest path next hops)
  - Information on all of prefixes available anywhere in the net
- NLSR populates the FIB for all of prefixes available anywhere, but there are lots of names in the network…(scalability problem)

My approach vs NLSR (2)

- In NDN, a routing protocol should be able to both diffuse information about permanently stored named data and discover temporary copies that may be cached outside the ordinary paths towards original content providers.
- NLSR explicitly targets a design which uses only *Interest/Data* packets to populate NDN node FIBs with routes towards permanent content copies.
- On the other hand, other works focus on how to efficiently forward content requests towards temporary cached copies by exploiting solely local information.
- My approach does both, thanks to contentAdvert

Thank you !
Questions ?