Computer Communications 91–92 (2016) 109–119

Contents lists available at ScienceDirect

Computer Communications

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

Hardware accelerator to speed up packet processing in NDN router

Weiwen Yu, Derek Pao*

Department of Electronic Engineering, City University of Hong Kong, Hong Kong

ARTICLE INFO

Article history:
Received 4 December 2015
Revised 17 June 2016
Accepted 18 June 2016
Available online 23 June 2016

Keywords:
Packet processing
Named data networking
Hardware lookup table

ABSTRACT

A hardware implementation of the *pending interest table* (PIT) for *named data networking* (NDN) is presented. One of the major challenges in this research is the per-packet update requirement in NDN packet processing. In general, the data structure of the lookup table is optimized in order to minimize the implementation cost and maximize the lookup performance. However, more computation steps are required to update the highly optimized data structure. Thus, the design of the hardware lookup table needs to tradeoff between the implementation cost, lookup performance and update cost. We employ an on-chip Bloom Filter and an off-chip linear-chained hash table in our design. The lookup operation for an interest/data packet and the associated update operation are integrated into one task. This can effectively reduce the overall processing time and the I/O communications with the software control unit. Our design also incorporates a name ID table (*nidT*) to store all distinct name IDs (*nid*) in the PIT. If the content name in an interest packet can be found in the *nidT*, then the router needs not look up the *forwarding information base* (FIB) to determine how to forward the interest packet. This can reduce the workload of the FIB significantly. For proof-of-concept, the proposed hardware architecture is implemented on a FPGA and the overall packet processing rate is about 56 to 60 million packets per second.

1. Introduction

The Internet is one of the greatest inventions of all time. It has enormous impacts on various aspects of human civilization, e.g. science and technology, business operations, social behavior, and government. The Internet is a host-based communication network. A host connected to the Internet is identified by an *Internet Protocol* (IP) address. In order to set up a communication channel, the end users must know the IP address of the other parties. Driven by advancements in both software and hardware technologies, the communication needs have gradually shifted from point-to-point message exchanges to content distributions, i.e. the dissemination of digital media such as music, picture, video, e-book, etc. The global demand for data is nearing 30 exabytes (30 billion gigabytes) per month in 2011. Also, it was estimated that about 500 exabytes of new on-line contents had been created in 2008 alone [1].

Under the current communication model of the Internet, a user has to find out where the contents are located in order to make a request to retrieve the required data. A user has to know "*where*" in addition to "*what*" he/she wants to retrieve. One can easily see a serious semantic gap between the existing host-based communication model and the content-centric communication needs.

Named Data Networking (NDN) [2,3] is a new network architecture proposed by the computer network community to better support the emerging communication needs. Packets under the NDN framework are identified by *names*, instead of the IP address/port number in the conventional *Internet Protocol*. Unlike the fixed-length IP address, a packet name can have arbitrary length. Packet forwarding in a NDN router involves more complex *name-lookup* operations.

Communications in NDN are initiated by receivers. A user wishing to retrieve a given content sends out an *interest* packet (a request) to a NDN router. The interest packet carries the name that identifies the desired data unit, e.g. a segment of a video file. When the NDN router receives an interest packet from an interface (called a *face* in the NDN terminology), the router will perform the following actions:

- The router will check its *cache store* (CS) to see if the requested data packet is already available. If a matching data packet can be found, then the data packet will be sent out along the face where the interest packet is received.
- If the requested data packet is not available in the CS, the router will search the *Pending Interest Table* (PIT) to see if the processing of the request for the same data packet is already in-progress. If a matching entry is found in the PIT, the given

* Corresponding author. Fax: +85234420562.
E-mail addresses: weiwenyu2-c@my.cityu.edu.hk (W. Yu), d.pao@cityu.edu.hk (D. Pao).

http://dx.doi.org/10.1016/j.comcom.2016.06.004
0140-3664/

face will be added to the list of faces associated with the PIT entry.
- If no matching entry is found in the PIT, a new entry for the packet name will be added to the PIT. The router will then look up the *Forwarding Information Base* (FIB) to determine the next-hop for the given *Interest* packet. The lookup operation in the FIB is a component-based *longest prefix match* (LPM).

A response to the interest packet (i.e. a *data* packet) will be generated when (i) the interest packet reaches the source of the requested data; or (ii) the requested data packet can be found in the CS of a NDN router on the path towards the data source. On receiving a data packet, the NDN router will perform the following actions:

- The router looks up its PIT. If a matching interest is found, the data packet will be sent along the list of faces associated with the pending interest, and the corresponding PIT entry will be removed. The data packet may be cached based on the cache management policy. As a result, the CS will also be updated.
- If the given name cannot be found in the PIT, the data packet is unsolicited and it will be discarded.

The functional requirements of packet processing in the NDN architecture have been defined, but the design and implementation of the NDN router is still a research problem pursued by the academia and industry. One of the major research issues is related to the implementation of the lookup tables to support NDN packet processing [4–18]. Most of the previous publications on the implementation of lookup tables in NDN router are software-based, and up to now there are only a couple of publications on hardware implementation. This development trend resembles the historical development of the IP address lookup problem. In the 1990s, researches on IP address lookup methods were mostly software-based [19–21]. Throughput of software-based methods is limited and cannot meet the throughput requirements of core router. Researches on IP address lookup methods gradually shifted to hardware-based approaches in the 2000s [22–26].

Packet processing in NDN involves 3 major lookup tables, namely the FIB, PIT and CS. The CS is an optional component. The network can function properly without the CS. However, its present may enhance the overall system performance. To the best of our knowledge, we have only seen 1 published paper that presents a hardware implementation of the FIB [18]. The method of [18] requires the router to have 2 switch fabrics. Another approach to speed up the FIB lookup is to exploit the massively parallel computation power of GPU [13–15]. Routers without the required special hardware (e.g. one more switch fabric or GPU) may only implement the FIB using conventional software technology. Previously published software implementations of the PIT and FIB can only achieve packet processing rate of a few million packets per second (MPPS). In this paper, we shall present a hardware accelerator that can improve the packet processing rate of the PIT up to 60 MPPS, and reduce the workload of FIB lookup significantly, e.g. by 10 times or more. Hence, a software-based FIB may offer sufficient throughput for the NDN router. Our method can be extended to include the CS table as well. A major issue that is being addressed in this study is the stateful processing requirement in NDN. The hardware accelerator provides a simple interface and aims to minimize the workload of the software. It functions like a TCAM (ternary content addressable memory) co-processor [27] in conventional TCP/IP router. The network processor submits lookup/update requests to the hardware accelerator, which carries out the request and returns the lookup results. The network processor will then carry out the required actions accordingly.

The organization of the remaining parts of this paper is as follows. Basic packet processing requirements in NDN is analyzed in Section 2. A brief review of related work is presented in Section 3. The architecture of the proposed hardware accelerator is presented in Section 4. Performance evaluations and simulations results are presented in Section 4.2. Section 5 is the conclusion and future work.

2. Packet processing challenges in NDN

A fundamental difference between the packet processing in NDN and conventional TCP/IP routers is that NDN requires stateful processing, whereas TCP/IP only requires stateless processing. In the conventional TCP/IP router, the processing (e.g. admission, queueing and forwarding) of an incoming packet is independent of the packet arrival history. The nominal update frequency to the IP routing table is about a few hundred times per second. Some recent hardware implementations of IP address lookup engine [26] and multi-field packet classifier [28] support packet processing rate at 300 plus MPPS. The update to lookup ratio in IP routing table is very low, e.g. a few updates per million lookup. The update to lookup ratio for packet classifier is even lower because classification rules are often defined manually by the network administrator. Hence, the designer can optimize the data structures of the hardware TCP/IP lookup tables to maximize the lookup performance. The system can afford to spend more time on each update operation. Typically, necessary modifications to the data structures for an update request are determined by the management software, and then corresponding memory-write commands are issued to the hardware to make the changes.

The processing of an incoming NDN packet depends on packets that have previously been received by the router. This has a great impact on the design of the hardware lookup tables. The stateful processing requirement of NDN leads to per-packet update to the PIT and the CS tables. For example, an insertion to the PIT is required for each interest packet, and a deletion from the PIT is required for each data packet. Suppose the most recently received data packets are saved in the cache store. Hence, a cache replacement is required for each data packet. This means that there will be 2 updates (1 insertion and 1 deletion) to the CS for each data packet. To cope with the per-packet update requirement, the hardware should be able to execute update and lookup operations to the PIT/CS tables with the same efficiency. As a result, the lookup table design techniques for TCP/IP packet processing may not be applied to NDN. In particular, update operation should not involve any precomputation by the software to determine how to modify the data structures.

Data and interest packets are named. A name is made up of multiple components represented in the type-length-value (TLV) format [29]. A content (e.g. a video) may be divided into multiple segments. Two segmenting conventions, namely the sequence-based segmentation and the byte-offset segmentation, have been proposed. For example, the name /com/youtube/funny-video-8/2/1234 contains 5 components, and it refers to a specific segment of a video called funny-video-8 published by youtube.com. The last two components specify the version number 2 and the segment number 1234 of the given video. Matching of names is component-based instead of character-based. For example, the name /abc/de does not match the name /ab/cde.

According to the NDN proposal a data packet satisfies an interest if the content name in the interest packet is a (component-based) prefix of the content name in the data packet. The objective of the NDN proposal is to allow more flexibility for traffic aggregation and discovery [30]. It is expected that vast majority of interest packets will carry full names. Exact match is used to look up an interest packet name in the PIT. To lookup a data packet name in the PIT, the router is required to find potential matching interest for every component-based prefix of the data name.