

# 基于内容中心网的缓存算法研究

黄莹莹, 杨龙祥

(南京邮电大学 通信与信息工程学院, 江苏 南京 210003)

**摘要:**随着信息时代的高速发展,用户对于数据的请求日益增多。传统的以用户为中心的端到端 IP 通信已经无法满足用户需求。内容中心网络(content-centric network, CCN)作为一个新颖的网络构架被提出,它将数据的内容和位置分离开来,实现以内容为中心的网络模型,以有效地解决 IP 通信的弊端。在内容中心网中,每个组块都有自己唯一的名字,用于数据搜索。并且 CCN 中的路由器都有缓存功能,缓存使用频率高的组块,从而减少网络拥塞,提高网络性能。其中,缓存算法主要分为缓存决定策略和缓存取代策略,其对于整个系统性能的好坏至关重要。针对内容中心网缓存算法进行研究,介绍了 CCN 网络模型,并针对具体算法做出分析,概括总结了其优缺点。

**关键词:**未来网络;内容中心网络;缓存决定策略;缓存取代策略;路由选择

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2018)05-0187-06

doi:10.3969/j.issn.1673-629X.2018.05.042

## Survey of Caching Strategies in Content Centric Networks

HUANG Ying-ying, YANG Long-xiang

(School of Communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

**Abstract:** With the rapid development of information era, the demands of users for data are increasing. Traditional host-centric end-to-end communication cannot meet users' needs. Today's Internet is increasingly used for information dissemination rather than pair-wise communications between end hosts and pays more attention to the content itself rather than where it is physically located. Innovative content-centric network architecture is proposed to handle these problems. In CCN, a file is divided into several chunks and every chunk has its unique ID. Information dissemination and searching are all based on these IDs. In addition, routers in CCN all have buffer memory. Nodes can cache the contents passing through it for a while and deliver them to requesting users. With popular contents cached in nodes, network congestion can be avoided and Internet performance can be improved. In CCN, cache management is a crucial factor to improve performance. The cache management generally works at each router through two kinds of policies called cache decision policy and cache replacement policy. Research firstly introduces CCN node model and then summarizes several typical algorithms' merits and drawbacks.

**Key words:** future network; content-centric network; cache decision policy; cache replacement policy; routing selection

### 1 概述

随着网络技术的高速发展,用户对数据的需求日益增加,这种趋势会给网络带宽和数据存储空间带来极大的压力,会导致网络拥塞和服务器过载等现象。而且日益增加的用户数也使 IP 资源消耗殆尽,用户不得不在不同的时段共享同一 IP 地址,这使得身份认证机制难以实现,故而引发了网络的安全性问题。

并且,传统的 IP 通信是由数据源主导的用户之间端到端通信<sup>[1]</sup>。两个端口在传输数据之前,必须先创建一个路径,并维护该路径直至数据传输结束。如

图 1 所示,当用户 1 需要获取数据 A 时,首先需要和服务器建立连接,若服务器中有满足要求的数据则回传。当用户 2 和 3 也需要获取数据 A 时,它们必须重复执行相同的动作,并向服务器发出请求。这种源主导的端到端通信将导致严重的资源浪费。并且,如今高速发展网络更多的是被用来进行数据广播,而不是终端之间的两两通信。用户更在意的是想要获取的数据本身(what),而不是该数据的确切位置(where)。显然 IP 网络这个拓扑结构无法实现性能良好的数据分配,这种网络模型和用户需求的不匹配,会导致用户体验

收稿日期:2017-06-02

修回日期:2017-10-08

网络出版时间:2018-02-08

基金项目:国家自然科学基金(61372124);国家“973”重点基础研究发展计划项目(2013CB329104)

作者简介:黄莹莹(1993-),女,硕士,研究方向为移动通信与无线技术;杨龙祥,教授,博导,研究方向为移动无线通信系统和物联网。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20180207.1912.068.html>

差、网络带宽损耗、数据获取时延大等缺陷。

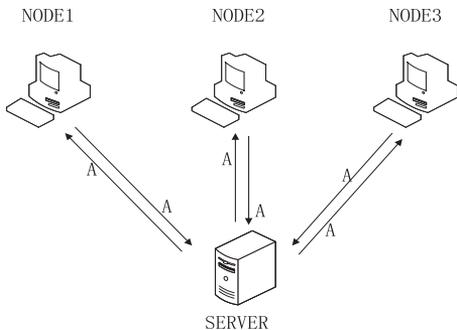


图 1 IP 结构的通信过程

针对上述问题,为了将源主导的端到端通信转变为接收端主导的信息搜索,提出了未来网络的概念。典型的未来网络缓存方案有面向数据的网络体系结构(data-oriented network architecture, DONA<sup>[2]</sup>)、内容中心网(content-centric network, CCN)、基于发布和订阅的互联网路由模式(publish-subscribe internet routing paradigm, PSIRP)等,其中内容中心网络受到业界的大量关注,若感兴趣,可以参照文献[3]。

文中主要对内容中心网的缓存算法进行了研究,分析了 CCN 体系结构与传统 IP 通信之间的差异,突出内容中心网的优点。介绍了 CCN 的最长前缀匹配查找方法、包结构以及请求转发机制。阐述了传统的缓存决定策略和缓存取代策略,在此基础上重点研究了几个典型的改进算法,分析其优缺点,并提出了未来的重点研究方向和相关工作。

## 2 CCN 工作机制

CCN 是以内容为中心的未来网络缓存方案,主要完成的工作是用户请求广播、数据搜索以及数据回传<sup>[4]</sup>。不同于以往的端到端 IP 通信,它实现了将数据本身和所在的端口分离,并且无需维护端到端之间的通信路径。它将侧重点放在数据本身而无需知道数据的具体位置。在 CCN 中,将为每个组块命名,且每个组块的名字都是唯一的,数据搜索基于组块的名字。在组块名字搜索的过程中,主要思想是字符串最长前缀匹配,典型的查找方法有基于 TCAM 的查找方法、基于字符串查找树的查找方法和基于散列技术的查找方法<sup>[5]</sup>。CCN 另一个重要特点是,路由器有本地缓存功能。在 IP 通信中,数据包被传输到下游用户端后,就会直接被丢弃。而 CCN 中的路由器会选择性缓存一些使用频率高的数据,当后续对这些数据的请求到达时,邻近路由器可以直接回传给用户,而不是每次都要从远处的服务器端获取。这种缓存机制可有效地减少网络拥塞,减少带宽的使用,提高网络吞吐量,降低用户获取数据的时延。

### 2.1 兴趣包和数据包

在 CCN 中有两种包格式,分别为兴趣包(interest packet)和数据包(data packet)<sup>[6]</sup>。在兴趣包和数据包的头部,都包含一个名为 Content Name 的存储空间,用来存储组块名字。若兴趣包中的 Content Name 是数据包中 Content Name 的前缀,则表明该数据包是请求获取的数据<sup>[7]</sup>。当用户请求获取某个数据块时,会创建一个包含该请求组块名字的兴趣包,然后在网络中广播该兴趣包。当网络中的任何路由器接收到该兴趣包时,会先提取出数据名字,由于路由器存在缓存功能,所以它会先查看自己的本地缓存,再决定是否将兴趣包转发给其他路由器。当服务器或某个路由器响应了该请求,则会将数据以数据包的形式回送给用户。

### 2.2 转发机制

在 CCN 转发机制中主要包含三种数据结构,分别是内容存储表(content store, CS)、待定请求表(pending interest table, PIT)和转发信息表(forwarding information base, FIB)。CS 用来缓存数据,从而实现最大化数据共享,减少网络拥塞。PIT 待定请求表,用来保存兴趣包转发轨迹。当 CCN 中搜索到满足要求的数据包时,该数据包会按照 PIT 中保存的路径轨迹回传给请求方。当路由器接收到兴趣包时,若该数据的请求已被转发过,则只需在 PIT 中添加对该数据的请求端口,当数据回传时,则会根据 PIT 中的请求端口一一回传该数据的副本。FIB 转发信息表,用来记录兴趣包可以被转发出去的一系列端口。

CCN 请求转发过程主要如下:当路由器接收到对某个数据的兴趣包时,首先会在兴趣包中提取出数据名字,利用名字的最长前缀匹配查找方法,在自己的缓存空间 CS 中搜索。若数据名字匹配成功,则将数据以数据包的形式回传给用户。若未成功,则会检查自己的 PIT,若 PIT 中存在对该数据的请求,则可直接更新 PIT 在其中添加请求端口,并丢弃该兴趣包,当数据包回传时会根据 PIT 中的请求端口,每个都回传一份副本。若 PIT 匹配也未成功,则会检查 FIB,如果存在一个匹配的 FIB 出口,则兴趣包会朝着这个方向传输给上游。再否则,匹配失败丢弃该兴趣包。

### 2.3 缓存机制

在 CCN 中,缓存管理机制对于改善网络系统性能极其重要。缓存管理主要涉及两个方面,缓存决定策略和缓存取代策略。缓存决定策略是指,当路由器接收到某个数据时决定是否缓存该数据。缓存取代策略是指,由于路由器的缓存空间有限,当缓存空间不足时,路由器决定选择删除缓存中的哪个数据,从而存储新到来的数据。

在缓存决定上,传统缓存方案有 leave copy every-

where (LCE)<sup>[8]</sup>、leave copy down (LCD)<sup>[9]</sup>、Prob-Cache<sup>[10]</sup>、hop-based probabilistic caching (HPC)<sup>[11]</sup>等,这些算法简单易实现,但多数没有考虑数据流行度,而且路由器间缺乏合作,缓存冗余,缓存空间资源浪费。

在缓存取代上,传统的缓存取代策略有最近最少使用替换算法(least recently used,LRU)<sup>[12]</sup>、使用频率最少替换算法(least frequently used,LFU)<sup>[13]</sup>。国内外研究数据表明,数据块流行度的高低,日益成为缓存取代策略需要考虑的重要因素。相继提出了一系列的基于内容流行度的缓存置换策略<sup>[14-15]</sup>,而传统的置换策略没有将流行度纳入研究重点,因此性能欠佳。

### 3 CCN 典型缓存算法

在CCN缓存方案中主要涉及两个问题,即数据缓存位置问题和数据搜索问题。

数据缓存位置问题主要是指缓存决定策略和缓存取代策略。早期的缓存决定和取代策略,简单易实现,但存在许多问题。路由器间缺乏合作,导致同一个数据在多个相邻路由器中缓存,缓存冗余浪费空间。并且缓存决定和取代时并没有考虑数据流行度,导致缓存使用频率不高的数据,影响系统性能。

在数据搜索方面,早期的路由转发只是朝着服务器的方向转发,这种搜索方式可能带来的问题是,所请求的数据在相邻路由器中,却将请求转发给了远处的服务器,获取数据的延时过大,影响系统性能。

因此,针对上述问题进行了改进,以下便是一些典型算法思想。

#### 3.1 基于内容流行度的路由器间合作缓存策略

由于文件在传输过程中被分成若干个组块,终端在请求某个文件时,则要对每个组块发送一个兴趣包。文献[16]基于该思想,提出一个上游节点可以建议下游节点缓存组块,并且缓存组块数目随着用户对该文件请求次数的增加而呈指数式增长。通过在组块中增加缓存建议标志,上游路由器在转发过程中可以设置该值来建议下游是否缓存该组块,从而实现各个路由器间的彼此合作,减少缓存冗余。

假设用户第一次请求某文件,原始服务器回传该文件,上游第一个路由器A会缓存该文件的第一个组块。当用户第二次请求该文件时,路由器A回传第一个组块给用户,其余组块仍有服务器回传。同时路由器A会建议其下游路由器B缓存第一个组块,其自身开始缓存第二第三个组块。以此类推,随着对该文件的请求次数增加,缓存的组块呈指数式增长。

在本算法中,提出了依据文件的请求频率来动态调整缓存组块数据的思想,缓存流行度高的文件,可以

有效减少网络拥塞,获得更高的缓存命中率。并且其实现了上下游路由器之间的缓存合作,能够有效地避免缓存冗余的问题。且研究表明,一个文件的前面组块访问频率会比后面的高<sup>[17]</sup>。本算法中将文件的前面组块缓存在靠近用户的路由器中,可以有效地减少缓存获取的时延,增强用户体验。

该算法的缺点在于,采用的是LRU缓存取代策略,没有依据组块的流行度来提出新的算法,可能会导致流行度低的文件取代流行度高的文件。

#### 3.2 基于 Consistent Hashing 的分布式协作缓存策略

CCN中,缓存和转发策略极其重要,传统缓存和转发策略虽然简单,但存在许多缺陷。文献[18]在此基础上进行改进,提出了Consistent Hashing思想及数据内容流行度预测算法,存储流行度高的数据。将路由器分组,各个组中的路由器彼此合作,实现数据请求有效转发。

在该算法中,如图2一个组中的各个路由器被分配若干个键值,拥有键的多少由该路由器的实际物理内存决定。即路由器的缓存空间越大,被分配的键越多。假设路由器1拥有键(2,6,7,12,13),路由器2拥有键(1,5,8,10),路由器3拥有键(0,4,9),路由器4拥有键(3,11)。这些键的范围从0至n,围成一个环。当数据请求到达时,首先使用散列函数求出兴趣包名字的散列值,然后针对求得的散列值来寻找对应的路由器。该路由器会根据提出的数据流行度预测公式和动态缓存门限公式,存储超过门限值的数据。

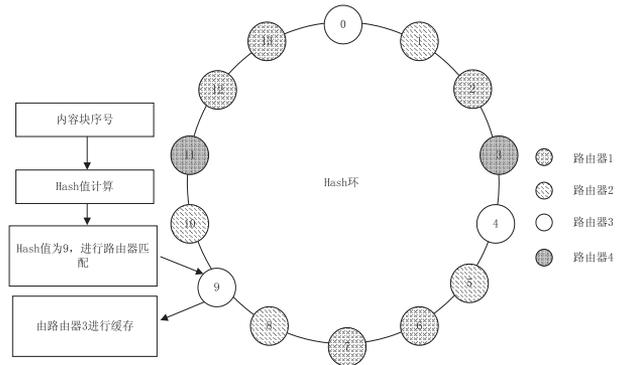


图2 基于 Consistent Hashing 的协同缓存策略

另外,在本算法中,主要提出了以下两条公式。式1是数据流行度的预测算法,该公式由两部分组成,利用数据的历史和当前请求频率来预测该数据的流行度。

$$LP_i(t+1) = \beta * \frac{F_i(t)}{\sum_{i=0}^j F_j(t)} + (1 - \beta) * LP_i(t - 1), \quad 0 < \beta < 1 \quad (1)$$

其中,  $F_i$  为数据  $i$  的访问频率;  $\beta$  为权重参数。式2是动态缓存门限公式。

$$\theta(t) = \frac{\sum_{i=0}^j LP_j(t+1)}{\sum_{i=0}^j C_j(t)} \quad (2)$$

其中,  $\sum_{i=0}^j LP_j(t+1)$  表示总流行度预测值;

$\sum_{i=0}^j C_j(t)$  表示  $t$  时刻数据包的总数。

由于数据包请求数量随着时间而变化,因此该公式计算出的流行度缓存门限能实现动态调节。摒弃传统采用的静态流行度门限,可以更佳有效地缓存使用频率高的数据。并且不同于传统的路由转发方式,当路由器收到数据请求时,通过散列环可以实现请求的有效转发,以及每个组中路由器的彼此协作,而不是传统的广播。

但是,该算法也存在一些缺点。在该算法中,网络中的路由器需要被分组,然后构成各自的散列环,在各自的散列环中,各个路由器又需要被分配不同的键值,可见整个流程所涉及的计算比较繁杂。在流行度预测公式和动态门限预测公式中,各个路由器需要周期性地计算通过它们的数据的流行度及各个时间段的缓存门限。都需要反复使用公式来进行数据更新,因此涉及的计算量也较多。综上所述,由于该算法在公式计

算、路由器分组以及散列环方面所做的工作比较多,因此复杂度高,实现起来有难度,而且运行时占用大量的资源,影响网络性能。

### 3.3 基于缓存容量感知的 CCN 缓存策略

该算法包含三个部分:缓存容量估计、选择缓存、缓存感知路由选择<sup>[19]</sup>。首先,通过式 3 估计每个路由器的缓存容量值 CCV。

$$CCV(i) = \frac{c}{L(i)} * \text{Cache Size}(i) \quad (3)$$

其中, Cache Size 是路由器的物理缓存大小;  $L$  是路由器的缓存负荷,即缓存消耗大小,是一段特定时间里的缓存消耗积累;  $c$  是补偿系数。

将最高的缓存容量值 CCV 记录在转发的兴趣包中,每经过一个路由器就更新 CCV 及当前路由器具有最高 CCV 的路由器的距离 NDV,从而找到那个具有最高的 CCV 路由器位置。图 3 描述了计算 CCV 值的过程以及如何将  $CCV_{\text{highest}}$  插入到转发的兴趣包中。例如节点 1,它的总缓存空间是 500,已缓存 200,  $c$  的默认值是 1,则节点 1 的 CCV 等于 2.5。其他节点的 CCV 以此类推计算。兴趣包经过各个节点,更新  $CCV_{\text{highest}}$  值,并记录各自的 NDV。

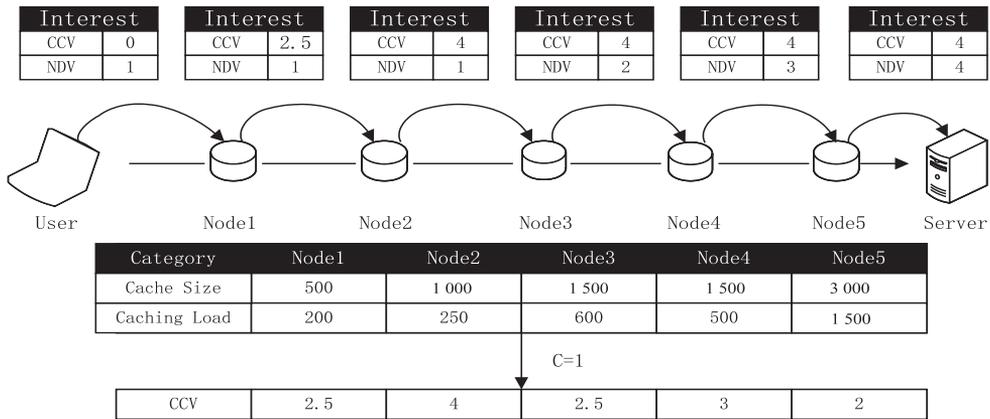


图 3 缓存空间估计示例

在选择性缓存中,返回路径上有最大缓存容量 CCV 的路由器一定要缓存该数据,若返回的数据流行度比较高,则利用 CCV 门限值公式,当中间路由器的缓存容量 CCV 大于这个门限值时,该路由器选择缓存数据,否则只是转发数据不存储。

在路由感知方面,路由器一定范围内创建临时 FIB,来指示到具有最大 CCV 的路由器。由于转发路径上至少有一个路由器会缓存该数据,如果路由器接收到该数据的请求,而它又没有缓存,则可以根据 FIB 出口转发。另外,根据统计的数据平均驻存时间为临时 FIB 表建立过期时间。

算法中,通过估计每个路由器的缓存容量,将选择

性缓存和路由选择联合起来,可以提高缓存效率,减少服务器的负荷。但是该算法的侧重点在缓存容量估计上,对于数据的真实流行度并没有提出具体的算法,对于缓存空间不足的情况也没有提出具体的取代策略。

### 3.4 基于流行度的动态调整缓存位置的缓存策略

在该算法中,提出了基于组块流行度来动态调整组块位置信息的思想<sup>[20]</sup>。每个组块中都包含一个流行度比较标志 PCF,若上游的路由器想将某个组块传至下游,首先会比较该组块的流行度和下游路由器中组块最小流行度,若大于则将 PCF 设置为 1,否则为 0。当 PCF=1 时,该上游路由器会将数据包中的缓存建议标志 CSF 设置为 1,以建议下游路由器缓存该组块。

这样算法就可以保证只会将流行度高的数据缓存在靠近用户的一端,降低传输时延。同时,由于传输路径上只有一个副本,这种合作缓存机制可以增加缓存多样性,避免缓存冗余的现象。

当缓存空间不足时,该算法通过比较组块的流行度,缓存流行度高的数据,并将被替换组块的 CSF 设置为 2,发送给上游路由器建议其缓存。如图 4 所示,上游路由器建议下游缓存流行度高的数据,下游路由器将替换的数据发送给上游路由器缓存。

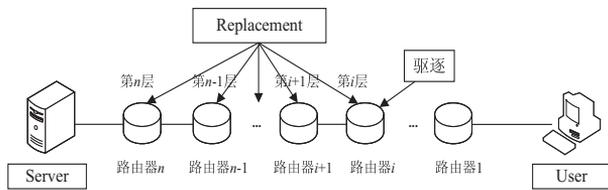


图4 动态调整缓存位置的工作过程

并且该算法为每个组块都建立了缓存历史轨迹,类似于 Breadcrumbs<sup>[21]</sup> 描述的方法,该轨迹中包含组块的 ID、组块来自哪个路由器以及传输至哪个路由器。当发生数据搜索时,不会直接将兴趣包传输给服务器,而是会先根据历史记录搜索,以防该数据就在附近路由器中。

不同于以前的算法,该算法将缓存的位置信息和数据搜索两者紧密联合起来,并结合了数据流行度,能够很好地实现将请求频率高的数据缓存在靠近用户的边缘路由器中,减少时延,改善用户体验,并且合作缓存减少冗余。其次,该算法是在文献[22]的基础上改进提出的,由于文献[22]是在分层的拓扑结构中设计的,所以应用到任意拓扑结构的 CCN 中会导致震荡和缓存频繁替代的现象。而该算法是针对 CCN 设计的,更加实用。

### 3.5 基于动态流行度门限的缓存机制

研究表明,对比数据请求分布、数据类型大小、缓存取代策略等因素,存储流行度高的数据对网络性能的影响更佳深刻。文献[23]着重研究的就是如何设置缓存的流行度门限值。

在文献[24]的算法中,引入了数据流行度的思想,为每个组块记录各自的请求次数。当某个组块的请求次数到达预先设置的门限值时,则被定义为是流行的数据,并被路由器缓存。该算法中仅存储流行的数据,在一定程度上可以节省资源,提高缓存命中率,但是会导致命中率的收敛速度慢,并且对于不同大小的缓存空间,其命中率性能不稳定。

本算法主要是在文献[24]基础上提出的,对上述提出的问题进行了改进。同样,也为每个组块创建了一个表,记录各自的 ID、流行度级别等。当数据到来时,若路由器缓存空间充足,则总会存储该数据。若缓

存空间不足,则会比较数据的请求次数是否超过了预先设定的流行度门限值,若超过,则采用 LRU 缓存替代策略,否则将新到来数据丢弃。对比于文献[24],本算法在缓存空间充足时无论新到来数据的流行度多少,都会存储该数据,当缓存空间不足时,则不流行的数据会被流行度高的数据取代。这样缓存处理方式能够获得更高的缓存命中率及更快的收敛速率。

上述的流行度门限都是固定的,但在实际生活中,一个数据的流行度、兴趣包到达的数量、文件的实际大小以及可用的路由器缓存总是随时间变化的,因此要设置动态门限。通过分析发现,流行度动态门限与文件大小和兴趣包数量成正比,与缓存空间成反比。综上所述,提出了公式 4:

$$P_{th} = \beta * \frac{n_{interest} F_p}{C_{size}} \quad (4)$$

其中,  $n_{interest}$  表示兴趣包数量;  $F_p$  表示文件大小;  $C_{size}$  表示缓存空间。

通过对文献[24]算法的改进及提出动态门限,可以提高缓存命中率、改善网络性能。但是,该算法计算量较大,且要实时跟踪一些参数,实现起来可能有点困难。并且该算法主要探讨的是门限值设定,对于路由器之间的合作缓存机制以及路由转发并没有进行深入探讨。

## 4 结束语

CCN 将由信息源主导的网络构架模式转变为由用户主导的方式,能够更好地实现数据搜索转发。CCN 中的缓存功能,对于更有效地使用带宽资源、改善用户体验至关重要。文中针对 CCN 的发展过程、网络模型以及一些典型的算法策略进行了详细阐述。

在 CCN 的设计方案中,主要包括缓存决定策略、缓存取代策略和路由转发协议。现有设计方案中,主要考虑的是数据流行度以及路由器间彼此协作。在此基础上可以引入其他因素,如数据往返时延、网络中能量损耗等。缓存性能的优化方法主要分为缓存大小规划、缓存空间共享机制、缓存决策策略、缓存替换算法、缓存对象可用性这 5 个方面<sup>[25]</sup>。

该课题之后的研究方向主要放在如何提出高效且易实现的算法方面。在缓存决定方面,制定出简单易实现的数据包流行度计算公式。另一研究重点是如何确定各路由器的数据缓存门限,仅缓存流行度值高于门限值的数据包,并尽量将其缓存在边缘路由器上,从而降低用户获取数据的时延。在缓存取代方面,主要是研究缓存替代的条件,通过比较新到来的数据包流行度值与已有缓存中最低流行度值,来决定是否要进行缓存替换。并且制定相应的策略,来处理被替换出

的数据包和流行度较低的新数据。并且各个路由器协同工作,从而有效减少缓存冗余,实现数据高效转发。

#### 参考文献:

- [1] AHLGREN B, DANNEWITZ C, IMBRENDA C, et al. A survey of information-centric networking[J]. IEEE Communications Magazine, 2012, 50(7): 26-36.
- [2] KOPONEN T, CHAWLA M, CHUN B G, et al. A data-oriented (and beyond) network architecture [J]. ACM SIGCOMM Computer Communication Review, 2007, 37(4): 181-192.
- [3] 夏春梅, 徐明伟. 信息中心网络研究综述[J]. 计算机科学与探索, 2013, 7(6): 481-493.
- [4] FANG Chao, YU F R, HUANG Tao, et al. A survey of energy-efficient caching in information-centric networking[J]. IEEE Communications Magazine, 2014, 52(11): 122-129.
- [5] 刘 斌, 汪 漪. 内容中心网络中名字查找技术的研究[J]. 电信科学, 2014, 30(9): 10-17.
- [6] JACOBSON V, SMETTERS D K, THORNTON J D, et al. Networking named content [J]. Communications of the ACM, 2012, 55(1): 117-124.
- [7] 闵二龙, 陈 震, 许宏峰, 等. 内容中心网络 CCN 研究进展探析[J]. 信息网络安全, 2012(2): 6-10.
- [8] JIANG Anxiao, BRUCK J. Optimal content placement for en-route web caching [C]//Proceedings of the second IEEE international symposium on network computing and applications. [s. l.]: IEEE, 2003: 9-16.
- [9] LAOUTARIS N, SYNTILA S, STAVRAKAKIS I. Meta algorithms for hierarchical Web caches [C]//IEEE international conference on performance, computing, and communications. Phoenix, AZ, USA; IEEE, 2004: 445-452.
- [10] PSARAS I, WEI K C, PAVLOU G. Probabilistic in-network caching for information-centric networks [C]//Proceedings of the second edition of the ICN workshop on information-centric networking. Helsinki, Finland; ACM, 2012: 55-60.
- [11] WANG Yu, XU Mingwei, FENG Zhen. Hop-based probabilistic caching for information-centric networks [C]//IEEE global communications conference. Atlanta, GA, USA; IEEE, 2013: 2102-2107.
- [12] MEGIDDO N, MODHA D S. Outperforming LRU with an adaptive replacement cache algorithm [J]. Computer, 2004, 37(4): 58-65.
- [13] PETEV P G, WINTERGERST M. Least frequently used eviction implementation; U. S. , US7552284 [P]. 2009-06-23.
- [14] 朱 轶, 糜正琨, 王文鼎. 一种基于内容流行度的内容中心网络缓存概率置换策略 [J]. 电子与信息学报, 2013, 35(6): 1305-1310.
- [15] 段 炼, 杨龙祥, 任美翠. 内容中心网络及其缓存策略研究 [J]. 计算机技术与发展, 2017, 27(3): 75-80.
- [16] CHO K, LEE M, PARK K, et al. WAVE: popularity-based and collaborative in-network caching for content-oriented networks [C]//Computer communications workshops. Orlando, FL, USA; IEEE, 2012: 316-321.
- [17] LIM S H, KO Y B, JUNG G H, et al. Inter-chunk popularity-based edge-first caching in content-centric networking [J]. IEEE Communications Letters, 2014, 18(8): 1331-1334.
- [18] THAR K, OO T Z, PHAM C, et al. Efficient forwarding and popularity based caching for content centric network [C]//International conference on information networking. Cambodia, Cambodia; IEEE, 2015: 330-335.
- [19] LEE S W, KIM D, KO Y B, et al. Cache capacity-aware CCN: selective caching and cache-aware routing [C]//Global communications conference. Atlanta, GA, USA; IEEE, 2014: 2114-2119.
- [20] XU Yuemei, MA Shuai, LI Yang, et al. P-CLS: a popularity-driven caching location and searching scheme in content centric networking [C]//Computing and communications conference. Nanjing, China; IEEE, 2016: 1-8.
- [21] ROSENSWEIG E J, KUROSE J. Breadcrumbs: efficient, best-effort content location in cache networks [C]//International conference on computer communications. [s. l.]: IEEE, 2009: 2631-2635.
- [22] LI Yang, LIN Tao, TANG Hui, et al. A chunk caching location and searching scheme in content centric networking [C]//IEEE international conference on communications. Ottawa, ON, Canada; IEEE, 2012: 2655-2659.
- [23] ONG M D, CHEN Min, TALEB T, et al. FGPC: fine-grained popularity-based caching design for content centric networking [C]//Proceedings of the 17th ACM international conference on modeling, analysis and simulation of wireless and mobile systems. Montreal, QC, Canada; ACM, 2014: 295-302.
- [24] BERNARDINI C, SILVERSTON T, FESTOR O. MPC: popularity-based caching strategy for content centric networks [C]//IEEE international conference on communications. [s. l.]: IEEE, 2013: 3619-3623.
- [25] 张国强, 李 杨, 林 涛, 等. 信息中心网络中的内置缓存技术研究 [J]. 软件学报, 2014, 25(1): 154-175.