

## 智慧协同网络服务内容在传输路径上的缓存分配策略

冯博昊<sup>1</sup>, 周华春<sup>1</sup>, 张宏科<sup>1</sup>, 张明川<sup>1,2</sup>

(1. 北京交通大学电子信息工程学院, 北京 100044; 2. 河南科技大学信息工程学院, 河南 洛阳 471023)

**摘 要:** 针对智慧协同网络提出一种服务内容在传输路径上的缓存分配策略。该策略根据服务内容的流行度部署其在传输路径上的缓存位置, 以求充分、高效地发挥网络缓存作用, 进而提升网络的总体性能。所提分配策略分别在 5 层树型拓扑和由 279 个节点组成的真实网络拓扑中进行了性能测试。结果显示, 该策略在所测的性能参数中表现出色, 就平均服务获取距离而言, 较命名数据网络(NDN, named data networking)所使用的 LCE(leave copy everywhere)策略, 其性能提高 20% 以上。

**关键词:** 缓存分配策略; 智慧协同网络; 智慧标识网络; 以信息为中心网络; 未来网络体系架构  
**中图分类号:** TN915 **文献标识码:** A

## Cache allocation policy of service contents along delivery paths for the smart collaborative network

FENG Bo-hao<sup>1</sup>, ZHOU Hua-chun<sup>1</sup>, ZHANG Hong-ke<sup>1</sup>, ZHANG Ming-chuan<sup>1,2</sup>

(1. Institute of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China;

2. Information Engineering College, Henan University of Science and Technology, Luoyang 471023, China)

**Abstract:** A cache allocation policy (CA) of service contents along delivery paths was proposed for the smart collaborative network. CA allocates on-path caches to service contents based on their popularity, expecting to fully and efficiently utilize the cache resource along the delivery path and further improving the network performance. The performance of proposed policy CA was evaluated under a 5-layer tree topology and a 279-node real network topology. The results show that CA outperforms others in terms of tested performance indexes. Compared with LCE(leave copy everywhere) used by named data networking (NDN), CA reduces the distance that users fetch service contents over 20%.

**Key words:** cache allocation policy, smart collaborative network, smart identifier network, information centric networking, future network architecture

### 1 引言

随着网络技术的飞速发展和不断创新, 互联网早已成为现代人们工作和生活必不可少的工具之一。然而, 随着互联网用户不断激增、互联网业务爆炸式增长等, 现有互联网的原始设计理念早已无法满足当今人们对网络的多元化要求, 诸多难以解决的问题暴露出来。因此, 如何设计能够满足未来发展需求的下一代互联网体系与机制已成为当今

学术领域研究的热点问题之一。

造成现有互联网诸多严重弊端的本质原因在于其“三重绑定”特性, 即“身份与位置绑定”、“资源与位置绑定”以及“控制与转发绑定”<sup>[1]</sup>。智慧协同网络(也称智慧标识网络)的提出旨在同时解决上述“三重绑定”的问题, 实现能够满足未来需求的下一代互联网体系结构。本文的工作集中于智慧协同网络的“资源与位置分离”机制上<sup>[2]</sup>。

现有互联网中, 服务资源的名称与其所处的网

收稿日期: 2015-05-11; 修回日期: 2015-08-27

基金项目: 国家基础研究发展计划(“973”计划)基金资助项目(No.2013CB329101); 国家高技术研究发展计划(“863”计划)基金资助项目(No.2015AA015702); 国家自然科学基金资助项目(No.61271202, No.U1404611)

**Foundation Items:** The National Basic Research Program of China (973 Program) (No.2013CB329101), The National High Technology of China (863 Program) (No. 2015AA015702), The National Natural Science Foundation of China (No.61271202, No.U1404611)

络位置耦合,并且数据采用通道传输,中间节点无法获悉所传数据的实际内容,不利于数据的高效分发。

为实现“资源与位置的分离”,智慧协同网络以独立于位置标识的服务标识(SID, service identifier)对服务内容进行命名,并且在路由器中引入了缓存功能。这样,服务资源可存储在网络中的任意位置,用户不必非要到遥远的服务器获取服务内容。服务内容的独立命名以及缓存的使用一方面有利于用户的就近获取服务资源,另一方面,网络中的重复流量也大幅降低,网络整体性能得到提升。该思路也与当今以信息为中心网络(ICN, information centric network)的概念不谋而合。学术界围绕 ICN 概念启动了大量的研究项目,例如 NDN 项目<sup>[3]</sup>、PURSUIT 项目<sup>[4]</sup>和 NetInF 项目<sup>[5]</sup>等。虽然这些项目在 ICN 的实施细节上存在差异,但其核心思想,即在对服务直接命名、利用路由器缓存服务内容上保持一致。

本文提出一种服务内容在传输路径上的缓存分配策略(CA, cache allocation policy)。其基本思想是:在网络中引入服务内容流行度感知节点,它通过周期性地收集用户请求来对服务内容进行流行度的排序,之后,将感知结果通告给用户所处的接入路由器。接入路由器随后根据被请求内容的流行度排序指定该服务内容在传输路径中被缓存的位置,以求合理利用传输路径上的缓存资源,充分、高效地发挥网络缓存的作用,提升网络的总体性能。

## 2 相关工作

毫无疑问,缓存对于“资源与位置分离”网络架构的性能改善是至关重要的。由于缓存路由器面向网络中所有服务内容进行存储,各服务内容特点不一,如何合理规划和利用缓存资源、将流行服务内容存储在离用户更近的地方一直是诸如网页缓存、以信息为中心网络等缓存网络研究的热点议题。缓存的分配问题可分为 2 类,第一类问题是如何在网络中选择合适的缓存位置并设置相应的缓存大小以存储服务内容<sup>[6-8]</sup>。第二类问题则是在给定网络缓存位置的情况下,研究存储哪些服务内容以求网络性能最优。本文专注于第二类问题,更准确地,是如何在传输路径上缓存资源有限的情况下,合理地为用户选择最佳的缓存位置以达到

降低用户获取时延、网络重复流量的目标。该问题也与缓存准入策略,即缓存节点决定哪些服务内容可以进入缓存的议题十分相似,本文将一并讨论。

LCD(leave copy down)<sup>[9]</sup>和 MCD(move copy down)<sup>[9]</sup>是网页缓存下的 2 种服务内容在传输路径上的准入策略。LCD 和 MCD 每次只将服务内容缓存在服务器或命中节点下游直连的节点上。LCD 与 MCD 的区别在于 MCD 将服务内容下移后,该服务内容会被命中节点删除。LCD、MCD 根据需求下移服务内容,服务内容每被请求一次,其缓存位置下移一跳,请求的越多,服务内容下移的距离就越远、其离用户也就更近。这样,流行内容由于拥有更多的请求数量将被缓存在网络的边缘。可见, LCD 和 MCD 可粗略地反映服务内容的流行度。

文献[9]还与 LCE(leave copy everywhere)和 Prob( $p$ )2 种策略进行比较。LCE 也是 NDN(named data networking)所使用的缓存准入策略,它可视为一种极端的策略,即服务内容存储在传输路径上的所有节点上。其缺点在于沿途路由器都会存储重复的服务内容、造成缓存资源的浪费。此外,被缓存的服务内容极易被新的服务内容所替换,使缓存中的服务内容极不稳定。多数服务内容在被二次请求前就被删除,缓存无法发挥出预期的效果。Prob( $p$ )则是沿途节点按照一个固定概率  $p$  存储服务内容,同样也无法体现出内容的流行度。

文献[10]提出一种服务内容在传输路径上的缓存准入策略 Betw。该策略根据介数中心性(BC, betweenness centrality)来缓存服务内容。节点的 BC 是指该节点处在拓扑中任意其他两点的最短路径上的次数,它反映出该节点在拓扑中的重要程度。Betw 的优势在于服务内容被存储在拓扑中最重要的节点上,利于共享。但 BC 是拓扑特性,与内容的流行度没有关系,无法反映出内容的流行度。此外,高 BC 节点的缓存替换速率将极高,造成被缓存内容的不稳定。

ProbCache、ProbCache<sup>+[11, 12]</sup>根据传输路径上剩余缓存容量、离用户的距离计算出缓存概率,它需要在请求报文和服务内容报文中分别携带距离用户和服务器的跳数信息。该策略可有效地减少进入缓存的服务内容的数量,但服务内容的流行度这一重要因素没有被直接考虑进来。

文献[13, 14]利用散列函数和求余的方式对服务内容的命名空间进行划分。其优点在于不同的内

容被指定到不同位置,保证了服务内容的稳定性。其缺点在于,服务标识本身无法体现出流行度,如果不添加较为复杂的流量平衡算法,缓存节点易出现负载失衡问题。

此外,利用最优化或线性规划等算法是解决缓存分配问题的常见方式,文献[15,16]通过计算缓存服务内容的收益来决定是否存储在节点中。但由于其计算粒度为一次请求,复杂度极高,只能作为最优结果进行参考而不适宜被实际系统进行实时测算。

### 3 服务内容在传输路径上的缓存分配策略

本节首先介绍智慧协同网络用于实现“资源与位置分离”机制的基本模型;其次,在分析现有服务请求模型的基础上,给出本文所提缓存分配策略的设计原则;最后,详细说明所提缓存分配策略的具体机制。

#### 3.1 智慧协同网络的基本模型

图1给出了智慧协同网络用于实现“资源与位置分离”机制的基本模型。智慧协同网络选择解析的方法进行服务内容与位置的映射,映射关系被保存在服务标识查询系统中。当用户发送请求报文至接入路由器后,接入路由器代替用户向服务标识查询系统询问SID所对应的位置信息。服务标识查询系统查询后返回相应的位置信息。接入路由器收到后,在请求报文中填写SID所对应的目的位置信息并根据路由表转发请求报文。沿途路由器收到请求报文后,查询SID所对应的服务内容是否被缓存在

本地,是则直接返回服务内容;否则继续根据路由表转发。当请求报文到达服务器后,服务器根据请求报文的目的地位置信息返回服务内容,传输路径上的路由器按需进行存储。

#### 3.2 服务请求模型分析

互联网是面向用户需求的网络,用户的服务请求模型对网络性能优化起到决定性的作用。大量文献使用Zipf分布作为服务请求的模型,即服务内容的流行度排序与请求概率呈反比。文献[17]给出了常见网络服务的 $a$ 的变化范围。

Zipf分布的数学表达式如式(1)所示,其中, $P(k,a,N)$ 代表排序为 $k$ 的服务内容的请求概率, $N$ 代表内容总体个数, $a$ 为函数的调节参数。

$$P(k,a,N) = \frac{1}{k^a \sum_{n=1}^N \frac{1}{n^a}} \quad (1)$$

假设 $N=10^6$ , $a$ 变化范围为0~2。图2给出了Zipf分布排序在前0.01%~10%的服务内容的累计请求概率。从图2可以看到,随着 $a$ 的增大,服务请求愈加集中,排序靠前的服务内容获得更多的请求概率,累计概率的增长率大幅增加后逐渐递减,但递减时,累计概率已有较高的值。例如,当 $a=1.5$ 时,排序在前0.01%的服务内容已占有92.43%的请求概率。考虑到 $a$ 通常大于0.6,并且 $a$ 越大,排序靠前的服务内容的累计概率越大,因此,服务内容应按照流行度排序依次存储在从用户到服务器的传输路径上。

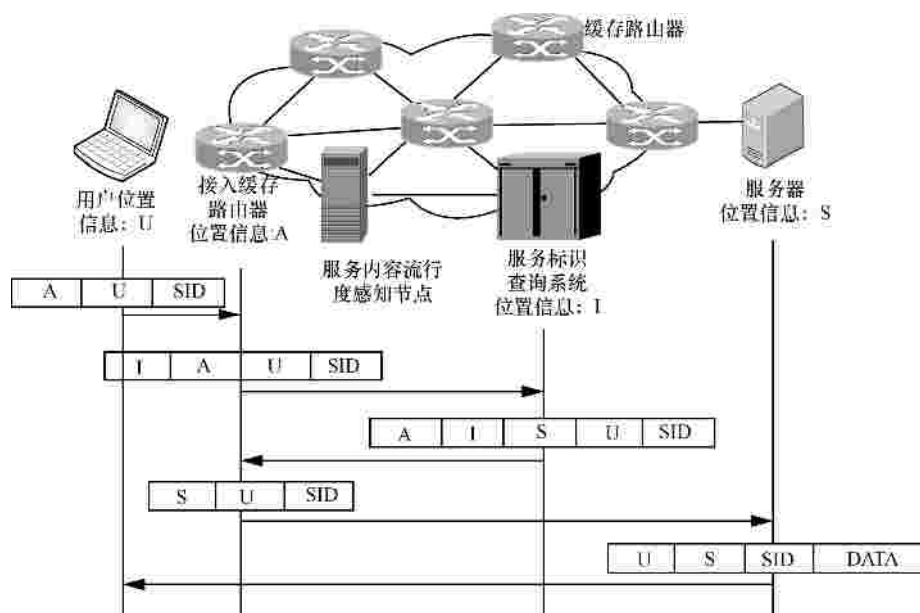


图1 智慧协同网络用于实现“资源与位置的分离”的基本模型

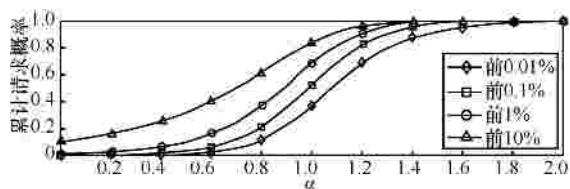


图2 Zipf 分布的累计概率

此外, 由于路由器的容量要远小于服务内容的总量, 过多的服务内容进入缓存势必会引起较高的缓存替换速率, 导致缓存的服务内容不稳定。鉴于每条传输路径的缓存资源有限且存在其他交叉路径竞争缓存资源的现象, 假设用户到服务器的传输路径由该用户独享, 且能容纳  $c$  个服务内容, 那么排序在  $c$  后的服务内容都不应该进入缓存, 以免增加路由器的缓存替代速率, 从而降低缓存性能。

因此, 服务内容应按照请求排序依次存储在从用户到服务器的传输路径上, 并且非传输路径所能容纳的服务内容不能进入缓存。这也是本文所提出的缓存分配策略的设计原则。

### 3.3 具体机制

#### 3.3.1 服务内容流行度的感知

为实现服务内容流行度的测算, CA 中引入了“服务内容流行度感知节点”。如图 1 所示, 它部署在网络边缘、靠近用户接入路由器的位置, 负责某一区域服务内容流行度的测算。服务内容流行度感知节点统一通告一个任播地址, 接入路由器将信令报文发往该任播地址, 便可与最近的服务内容流行度感知节点进行交互。为了感知服务内容的流行度, 接入路由器每收到一次服务请求, 都将该请求中的 SID 提取出来发往服务内容流行度感知节点, 便于后者对服务内容的流行度进行评估。网络初始时, 接入路由器还需将自己的位置信息注册到服务内容流行度感知节点, 以便后者主动推送关于服务内容流行度排序的信令报文。此外, 接入路由器还需向服务内容流行度感知节点通告其所有传输路径中最大的路径缓存容量, 即传输路径上所有缓存节点缓存容量之和, 以便后者决定向其推送排序列表的长度。

服务内容流行度感知节点周期性的测算服务内容的流行度, 假设感知周期为  $t$ 。服务内容的流行度通过其请求数量来体现。服务内容流行度感知节点为每个被请求的服务内容维护如表 1 所示的流行度表项。该表项由 3 部分组成, “服务标识”用于标识一个服务内容, “本周请求数”记录在当前周期内该服务内容的请求数量, “流行度”则为上一周期测算

的服务内容流行度的值。当前周期结束后, 服务内容流行度感知节点重新计算服务内容的流行度。

表 1 服务内容流行度感知节点维护的流行度

服务标识	本周请求数	流行度
SID1	$Count_1$	$Value_1$
SID2	$Count_2$	$Value_2$
Ⅲ	Ⅲ	Ⅲ

服务内容流行度感知节点使用指数加权移动平均值(EWMA, exponentially weighted moving average)<sup>[18]</sup>的方式计算流行内容的流行度。EWMA 的表达式如式(2)所示。 $Value$  的初始值为服务内容第一个周期的请求数量。 $\gamma$  是过去测量值的权重系数,  $\gamma$  越小, 过去测量值所占比重越大。

服务内容流行度感知节点根据新计算出来的流行度对服务内容进行排序并将该结果推送给向其注册的接入路由器。同时, 计算出来的流行度将覆盖流行度表项中原有的流行度。

$$Value_i = \begin{cases} Count_i, & i = 1 \\ \gamma Count_i + (1 - \gamma) Value_{i-1}, & i \neq 1 \end{cases} \quad (2)$$

其中,  $0 < \gamma < 1, i \in N$ 。

接入路由器收到服务内容流行度感知节点推送的包含内容流行度排序的信令报文后, 创建或更新其维护的排序表项, 如表 2 所示, 其只需要记录 SID 和流行度排序, 并维护一个计时器。考虑到信令报文的传输时间, 计时器的计时长略长于  $t$ , 超时后, 接入路由器删除相应表项。

表 2 接入路由器维护的排序

服务标识	排序
SID1	Rank1
SID2	Rank2
Ⅲ	Ⅲ

#### 3.3.2 传输路径上路由器的相关操作

用户发送的请求报文如图 3 所示。目的位置信息、源位置信息分别为目的节点和源节点的通信地址。SID 为用户请求的服务标识, Rank 为该 SID 的流行度排序, 初始值为 MAX, MAX 为一个很大的整数, 远远大于路径缓存容量。CacheCapacity 用于记录传输路径上的缓存容量, 以确定服务内容的存储位置, 其初始值为 0。Node-ID 用于记录缓存该服务内容的节点标识, 初始值为 NULL。回复的数

据报文格式报头与请求报文格式类似，区别在于前者不含 Rank 和 CacheCapacity 域。

目的位置信息	
源位置信息	
SID	
Rank	CacheCapacity
Node-ID	

图3 请求报文格式

路由器的处理流程如图4所示。当接入路由器收到请求报文后，首先根据SID查询排序表项，如果存在对应表项，则在请求报文中填写对应表项中的Rank值。之后，按照普通路由器处理。普通路由器首先将自身的缓存容量与CacheCapacity相加，之后与Rank比较。如果新的CacheCapacity小于Rank，则说明该服务内容应该存储在后续传输节点上。如果新的CacheCapacity大于或等于Rank，则说明该路由器需要存储该服务内容。此时，该路由器将自己的ID号写入请求报文的Node-ID中并将Rank改写为MAX，这样，后续路由器将不会存储该服务内容。之后，普通路由器将查询本地缓存列表，如果存有该SID的服务内容，普通路由器复制请求报文中Node-ID域至返回的数据报文头部对应处并返回服务内容。否则，路由器根据路由表发送请求报文。当服务内容回传时，沿途路由器查看返回报文中的Node-ID是否为自己的节点标识，若是，

则缓存该服务内容，否，则直接转发。

图5给出了一个简单的用户发送请求报文至服务器的例子，以说明路由器对请求报文操作的过程。假设每台路由器可以容纳5个服务内容，被请求内容排序为12，节点1、2、3均没有存储相应的服务内容。节点1为用户的接入路由器，它首先查询本地排序列表，发现SID的排序为12，并将12填写至请求报文中的Rank域；之后改写CacheCapacity为5。节点1将请求发往下一跳节点2。节点2首先改写CacheCapacity为10，发现小于Rank域的12，继续发往下一跳。节点3首先改写CacheCapacity为15，发现大于Rank域的12，改写Node-ID域为3并将Rank域设为MAX。当服务内容回传时，只有节点3进行存储。

CA的提出旨在将流行的服务内容存储在传输路径有限的缓存资源上，并且，越流行的内容越靠近用户。不同于NDN所使用的LCE策略，在CA中，服务内容在请求时已经被指定缓存位置，路由器的缓存替代速率大幅降低，非流行服务内容替代流行服务内容的现象也得到有效的解决，缓存内容的稳定性得以保证，其被共享的几率也大幅增加。然而，CA付出的代价是每个接入网的服务内容流行度感知节点需要为每个SID记录请求数量。一方面，服务请求报文转发到服务内容流行度感知节点会造成链路带宽的开销；另一方面，对于服务内容流行度感知节点需要维护

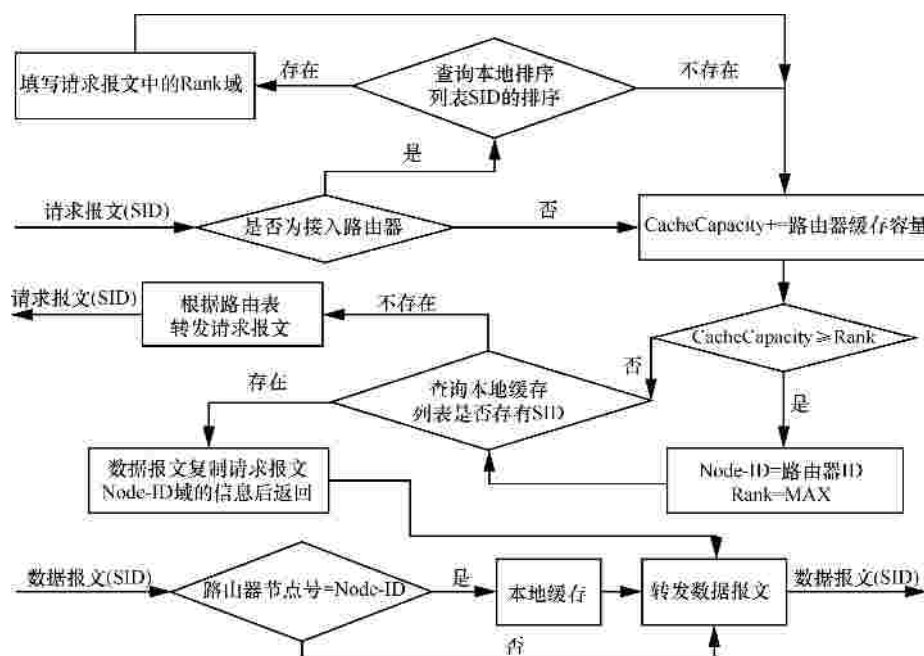


图4 路由器处理流程

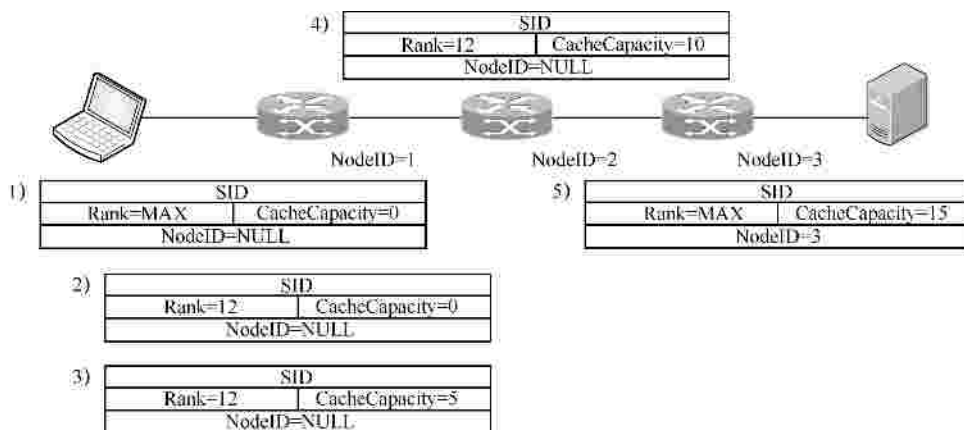


图 5 路由器对请求报文处理过程示例

很长的 SID 列表。

对于链路开销的问题, 由于一个区域内的请求分布大致相同, 接入网的服务内容流行度感知节点可随机选取某个或某几个用户作为探测服务内容流行度的样本, 这样可有效降低链路的开销。对于服务内容流行度感知节点自身处理开销的问题, 本文所提流行度测试方法仅仅是初步方案, 由于传输路径上无法存储所有服务内容, 服务内容流行度感知节点也没必要存储所有 SID 的请求数量。该问题的实质其实与缓存替代策略所解决的问题完全一样, 即在一个有限长度的列表中进行表项优先级的排序。由于服务内容流行度感知节点只需要知道服务内容的排序而不是具体请求数量, 可使用基于频率的替代策略进行流行度的排序, 如 P-LFU(period least frequently used)、LFU-DA(LFU-dynamic aging)<sup>[19]</sup>等。

#### 4 性能测试与分析

为了验证所提缓存分配策略 CA 的性能, 本文利用基于 NS3<sup>[20]</sup>的 ndnSIM<sup>[21]</sup>实现了缓存分配策略 CA。此外, 本文还实现了 LCD、MCD、Prob( $p$ )、ProbCache (PRC)、ProbCache<sup>+</sup> (PRC+)以及 Betw 用于性能比较。所有策略均使用(LRU, least recently used)作为缓存替代策略。此外, LCE+LRU 也是 NDN 所使用的默认策略, ndnSIM 已经提供。本文分别在 5 层的树型拓扑和 279 个节点的真实网络拓扑中对所有策略进行性能测试。树型拓扑常用于接入网, 但其无法表现出真实网络的无标度特性。故本文从 Rocketfuel's AT&T topology<sup>[22]</sup>中选择编号为 1221 的真实网络拓扑。2 种拓扑如图 6 所示。

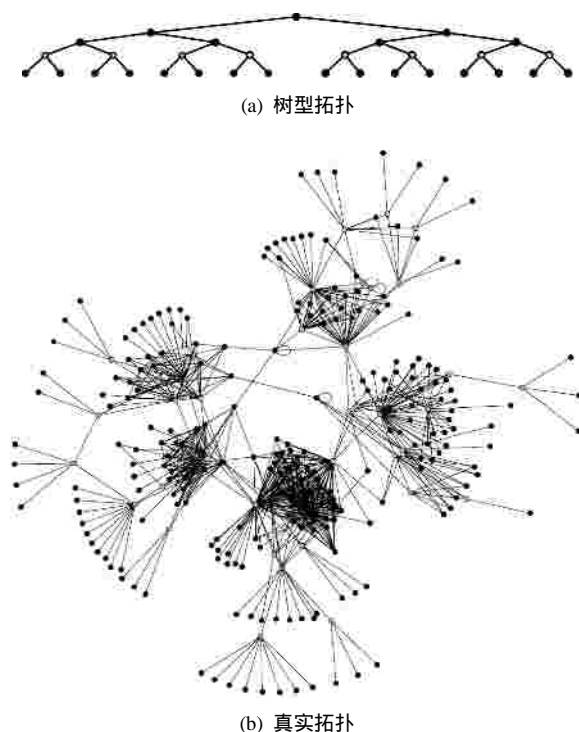


图 6 真实网络拓扑

服务内容的请求服从 Zipf 分布, 其中  $a=1.0$ 。服务内容的数量为  $2.5 \times 10^4$ , 它们的数据报文大小完全相同。用户请求报文的到达过程符合泊松过程, 每个用户的请求到达速率为 100 Hz。仿真持续时间为 1 000 s。缓存初始为空, 其容量为 25 个服务内容。仿真相关参数的总结如表 3 所示。

在 5 层树型拓扑中, 一个服务器接入在树根处, 16 个用户模拟器分别接入到 16 个叶节点上, 故用户到服务器的距离为 6 跳 (用户到接入路由器、服务器到第一跳路由器均计为 1 跳, 真实拓扑也以此方式计算跳数)。树型拓扑中节点的带宽设置的较

大，所以拥塞问题不会对最终的结果造成多大的影响。真实拓扑共有 731 条链路，其中，接入节点有 169 个，网关节点 45 个，骨干网节点 65 个。本文并未改变该拓扑的原有参数。50 个服务器和 100 个用户模拟器将随机接入到不同的接入节点中。 $2.5 \times 10^4$  个内容将随机分布到这 100 个服务器中，而每个服务内容只被某一个服务器提供。另外，在 2 种拓扑中，每个网关节点（灰色）都连接一个服务内容流行度感知节点。树型拓扑的网关节点为第 5 层节点。

表 3 仿真参数	
参数	设定值
请求分布	Zipf(1.0)
服务内容数量	$2.5 \times 10^4$
拓扑类型	树型拓扑(31 个节点)、真实拓扑(279 个节点)
缓存大小	25
单个用户请求频率	100 Hz
仿真时间	1 000 s
所测策略	CA、LCD、MCD、Prob(0.3)、ProbCache (PRC)、ProbCache <sup>+</sup> (PRC+)、Betw 以及 LCE
缓存替代策略	LRU
性能指标	服务获取距离、缓存删除操作总量、请求报文数量

测试的性能指标包括服务获取距离、缓存删除操作总量以及请求报文数量。这些指标又分为实时指标和平均指标 2 类。实时指标为每 5 s 内的平均值、平均指标为 1 000 s 内的总平均值。

服务获取距离定义为请求报文到达服务器或存有服务内容副本的节点所经过的跳数。平均值的计算方式为规定时间内所有请求报文的跳数之和与总请求数的商。它反映出缓存的共享率，距离越短，效率越高，用户的服务获取时延就越低；节点的缓存删除操作总量定义为网络中所有节点每秒删除缓存内容数量之和，它反映出缓存内容的稳定性。请求报文数量记录规定时间内网络中所有节点发送的请求报文的数量，由于一个请求报文回复一个数据报文，它可以粗略地反映出网络中的流量状况。

需要说明的是，在测算服务获取距离时，由于其他策略并未考虑服务解析的问题，为了公平起见，缓存分配策略 CA 在计算服务获取距离时不包含解析过程所产生的跳数，即假设接入路由器已知服务器的位置信息。另外，缓存分配策略 CA 的测算周期默认为 10 s， $g$  默认为 0.85。另外，考虑到测试参数实时的简洁性，性能表现中等的 MCD、

PRC 以及 Prob(0.3)将不在图中呈现，其平均值将在对应列表中体现。

4.1 缓存分配策略 CA 自身参数对其的影响

本文首先测试缓存分配策略 CA 自身参数对其的影响。1 000 s 内的总平均服务获取距离随测算周期以及  $g$  大小变化的情况如图 7 所示。可以看到，在 2 种拓扑下，平均服务距离均未发生明显变化。因此，测算周期以及  $g$  并非是影响缓存分配策略 CA 性能的主要因素。另外，缓存分配策略 CA 在 2 种拓扑下平均服务获取距离的差异源于 2 种拓扑服务器数量、用户模拟器数量等设定以及拓扑本身特性的不同。

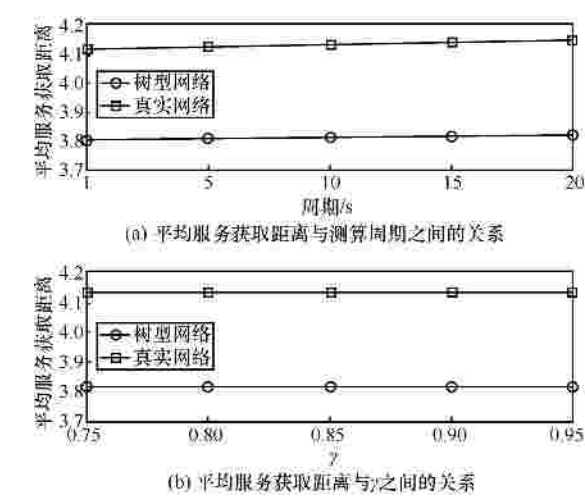


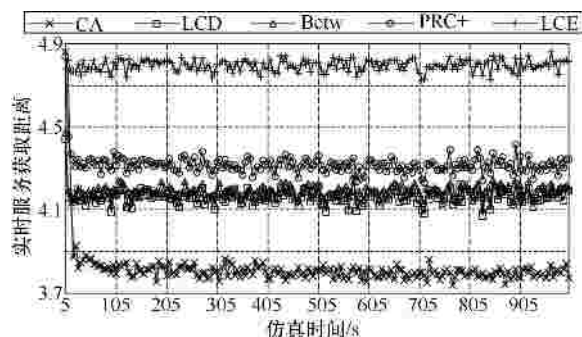
图 7 平均服务获取距离

4.2 各策略服务获取距离的比较

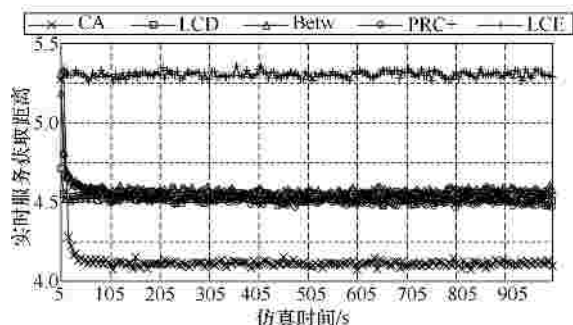
图 8 分别给出了在树型拓扑和真实网络拓扑下各策略的实时服务获取距离。各策略 1 000 s 内的总平均服务获取距离以及相对于 LCE 的减小率如表 4 所示。可以看出，2 种拓扑下性能最好的是 CA，在树型拓扑下平均服务获取距离为 3.81 跳，在真实拓扑下为 4.13 跳，相对于 LCE，减小率大于 20%。其次是 LCD 相对于 LCE 的减小率为 14% 左右。

LCE 性能较差的原因在于其不限制服务内容进入缓存。由于传输路径上的缓存资源有限，其总容量要远远小于服务内容的请求数量。在这种情况下，当服务内容回传时，途径节点都会缓存相同的服务内容，造成缓存资源的极度浪费。此外，这种方式加速了缓存内容的替代速率，很多服务内容在被二次请求前就被删除，缓存没有发挥出很好的共享作用，致使用户的服务获取距离较高。





(a) 树型拓扑下各策略的实时服务获取距离



(b) 真实网络拓扑下各策略的实时服务获取距离

图 8 实时服务获取距离

Prob(0.3)是一种纯基于概率的策略,其性能优于 LCE 的原因在于它并不是让所有服务内容都进入缓存。这样,传输路径上缓存内容的多样性得到一定的保证,同时,缓存内容的替代速率也大幅降低,故其性能较之 LCE 有所提高。

ProbCache 和 ProCache<sup>+</sup>也是一种基于概率的策略,不同之处在于它们还考虑了服务内容回传时路径上所剩缓存容量以及离用户的距离等因素。由于其概率计算更具针对性,其性能较之 Prob(0.3)提高较多。

Betw 根据 BC 缓存服务内容,它利用 BC 确定服务内容在传输路径上的存储位置。但 Betw 的缺陷在于 BC 是反映网络拓扑的特性,它的性能与拓扑直接相关,而且高 BC 的节点缓存替代速率较高,不易实现服务内容的共享。

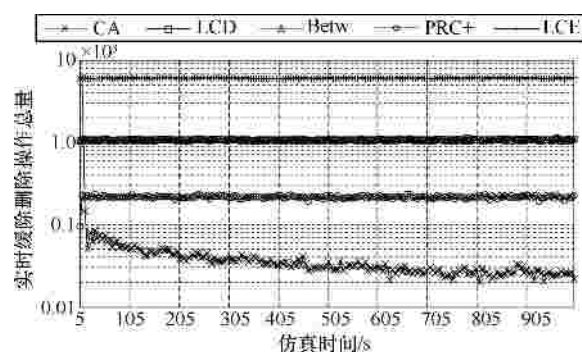
LCD 和 MCD 则是根据请求下移数据,它可以粗略的反映出服务内容的流行度。对于流行内容,由于其拥有较多的请求数量,其最终会被下移到离用户更近的地方。另外,MCD 性能逊于 LCD 的原因在于命中节点会删除被命中的服务内容,降低了服务内容被共享的概率。

CA 表现最为出色,原因在于其按照用户需求存储服务内容。如前所述,服务请求服从 Zipf 分布,服务内容的请求数量随流行度排序的升高呈指数递减。而 CA 按照服务内容流行度排序递增的方式

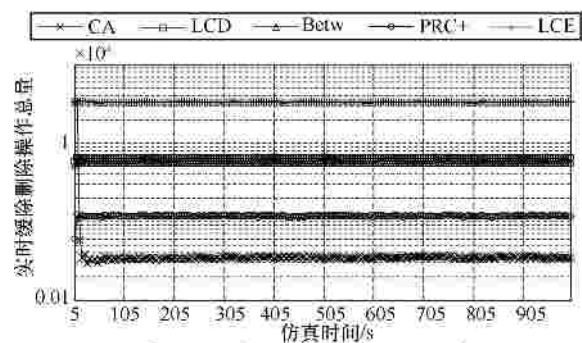
从用户方向依次在传输路径上存储服务内容,这样,更多的服务请求能够被就近满足。此外,由于传输路径上的缓存资源远不足以存储用户所请求的所有内容,排序靠后的服务内容将不会进入缓存,这大幅降低了缓存内容的替换速率,保证缓存内容的稳定性,便于服务内容的共享。

#### 4.3 各策略服务器缓存删除操作总量的比较

图 9 分别给出了在树型拓扑和真实网络拓扑下各策略的实时缓存删除操作总量。各策略 1 000 s 内的总平均缓存删除操作总量以及相对于 LCE 的减小率如表 4 所示。可以看出,2 种拓扑下性能最好的是 CA,在树型拓扑下缓存删除操作总量为每秒 95.98 次,在真实拓扑下为每秒 680.44 次,相对于 LCE,减小率在 98% 左右。这说明在缓存分配策略 CA 中,被缓存的服务内容相对稳定,利于服务内容的共享。其原因在于缓存分配策略 CA 指定了服务内容在传输路径的存储位置并阻止非流行服务内容进入缓存,有效降低了缓存内容的替代速率。



(a) 树型拓扑下各策略的实时缓存删除操作总量



(b) 真实网络拓扑下各策略的实时缓存删除操作总量

图 9 实时缓存删除操作总量

#### 4.4 各策略请求报文数量的比较

图 10 给出了在树型拓扑和真实网络拓扑下各策略的实时请求报文数量。各策略 1 000 s 内的总平均请求报文数量以及相对于 LCE 的减小率如表 4 所示。



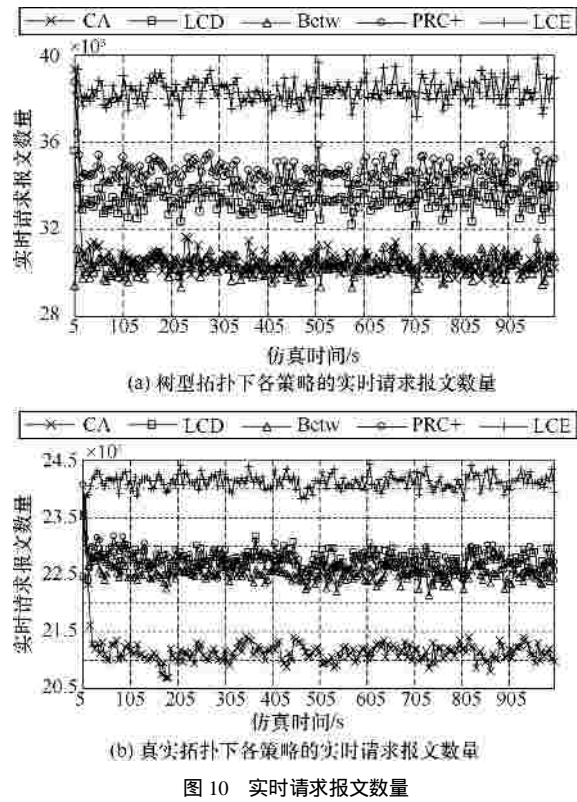


图 10 实时请求报文数量

缓存分配策略 CA 在 2 种拓扑下表现良好，相对于 LCE 的减小率分别为 20.34%和 12.27%。真实拓扑相对于 LCE 的减小率低于树型拓扑的原因在于真实网络是无标度网络并且用户传输路径相互重叠、相互影响，加之其节点数量、用户模拟器数量远远多于树型拓扑，服务请求在网络中的传输更具一般性。此外，虽然 Betw 的请求报文数量也很低，但由于其只考虑了拓扑特性而不是服务内容的流行度，其用户的服务获取距离并不是最短的。缓存分配策略 CA 则是根据服务内容的流行度存储，故在

请求报文数量降低的同时，用户的平均服务获取距离最小。

综上所述，缓存分配策略 CA 性能出色的原因在于其按照服务内容流行度排序递增的方式从用户方向依次在传输路径上存储服务内容。此外，由于传输路径上的缓存资源远不足以存储用户所请求的所有内容，排序靠后的服务内容将不会进入缓存，这大幅降低了缓存内容的替换速率，保证缓存内容的稳定性。测试结果显示，缓存分配策略 CA 在服务获取距离、缓存删除操作总量、请求报文数量 3 方面性能出色，这也证明了本文设计缓存分配策略 CA 的原则是有效的。

5 结束语

本文提出了一种服务内容在传输路径上的缓存分配策略，旨在合理利用传输路径上的缓存资源，充分、高效地发挥网络缓存的作用，进而提升网络的总体性能。本文分别在 5 层的树型拓扑和由 279 个节点组成的真实网络中对缓存分配策略 CA 进行性能测试。同时，本文还实现了 LCD、MCD、Betw、ProbCache、ProbCache<sup>+</sup>、Prob(0.3)的功能并与缓存分配策略 CA、NDN 所使用的策略 LCE 进行性能对比。结果显示，缓存分配策略 CA 在所测的性能参数中表现出色，就平均服务获取距离而言，较之 LCE，其性能提高 20% 以上。

参考文献：

[1] 张宏科, 罗洪斌. 智慧协同网络体系基础研究[J]. 电子学报, 2013, 41(7): 1249-1254.  
ZHANG H K, LUO H B. Fundamental research on theories of smart and

表 4 各性能指标平均值以及相对于 LCE 的减小率

策略	服务获取距离				缓存删除操作总量				请求报文数量			
	树型拓扑		真实拓扑		树型拓扑		真实拓扑		树型拓扑		真实拓扑	
	平均值	减小率	平均值	减小率	平均值	减小率	平均值	减小率	平均值	减小率	平均值	减小率
CA	3.81	20.47%	4.13	22.14%	95.98	98.42%	680.44	97.95%	30 548.28	20.34%	211 615.98	12.27%
LCD	4.16	13.20%	4.53	14.67%	1 076.90	82.22%	6 413.18	80.71%	33 345.95	13.04%	227 571.49	5.66%
MCD	4.26	11.25%	4.99	5.97%	1 075.05	82.25%	5 991.55	81.98%	34 092.95	11.09%	235 987.36	2.17%
Betw	4.19	12.67%	4.58	13.74%	1 076.08	82.24%	5 584.23	83.20%	30 290.57	21.01%	225 086.92	6.69%
PRC	4.42	7.93%	4.72	11.07%	761.88	87.42%	5 022.37	84.89%	35 363.64	7.78%	227 967.82	5.49%
PRC+	4.32	9.95%	4.52	14.79%	217.75	96.41%	1 162.75	96.50%	34 591.54	9.79%	227 308.99	5.76%
Prob(0.3)	4.56	4.92%	4.87	8.25%	1 709.41	71.78%	10 300.91	69.01%	36 507.35	4.80%	228 771.90	5.16%
LCE	4.80	0%	5.30	0%	6 058.11	0%	33 243.98	0%	38 346.72	0%	241 212.06	0%

- cooperative networks[J]. Acta Electronic Silica, 2013, 41(7): 1249-1254.
- [2] 苏伟, 陈佳, 周华春, 等. 智慧协同网络中的服务机理研究[J]. 电子学报, 2013, 41(7): 1255-1260.
- SU W, CHEN J, ZHOU H C, et al. Research on the service mechanisms in smart and cooperative networks[J]. Acta Electronic Silica, 2013, 41(7): 1255-1260.
- [3] ZHANG L, ESTRIN D, BURKE J, et al. Named data networking (NDN) project NDN-0001[J]. Acm Sigcomm Computer Communication Review, 2010, 44(3): 66-73.
- [4] TROSSEN D, PARISIS G, VISALA K, et al. Conceptual architecture: principles, patterns and subcomponents descriptions[R]. Technical Report, FP7-INFOS-ICT-257217\_PURSUIT\_D2.2, 2011.
- [5] AHLGREN B, D'AMBROSIO M, DANNEWITZ C, et al. Second NetInf architecture description[R]. Technical Report, FP7-ICT-2007-1-216041-4WARD / D-6.2, 2010.
- [6] ROSSI D, ROSSINI G. On sizing CCN content stores by exploiting topological information[C]//Computer Communications Workshops (INFOCOM WKSHPS), IEEE Conference. Orlando, c2012: 280-285.
- [7] LI Z, XIE G, WANG Y, et al. Optimal cache allocation for content-centric networking[C]//IEEE International Conference on Network Protocols (ICNP). Goettingen, c2013: 1-10.
- [8] XU Y, LI Y, LIN T, et al. A novel cache size optimization scheme based on manifold learning in content centric networking[J]. Journal of Network & Computer Applications, 2014, 37(1): 273-281.
- [9] LAOUTARIS N, SYNTILA S, STAVRAKAKIS I. Meta algorithms for hierarchical Web caches[C]// IEEE International Conference on Performance, Computing, and Communications. Phoenix, c2004: 445-452.
- [10] WEI K C, HE D, PSARAS I, et al. Cache "less for more" in information-centric networks (extended version)[J]. Computer Communications, 2013, 36(7): 758-770.
- [11] PAVLOU G, PSARAS I, WEI K C. Probabilistic in-network caching for information-centric networks[C]//The Second Edition of the ICN Workshop on Information-Centric Networking, ICN '12. New York, c2012: 55-60.
- [12] PSARAS I, WEI K C, PAVLOU G. In-network cache management and resource allocation for information-centric networks[J]. Parallel & Distributed Systems IEEE Transactions, 2014, 25(11): 2920-2931.
- [13] ROSENSWEIG E J, KUROSE J. Breadcrumbs: efficient, best-effort content location in cache networks[C]// IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). Rio de Janeiro, c2009: 2631-2635.
- [14] ZHU Y, CHEN M, NAKAO A. CONIC: content-oriented network with indexed caching[C]// IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). San Diego, c2010: 1-6.
- [15] SOURLAS V, GKATZIKIS L, FLEGKAS P, et al. Distributed cache management in information-centric networks[J]. IEEE Transactions on Network & Service Management, 2013, 10(3): 286-299.
- [16] KIM Y, YEOM I. Performance analysis of in-network caching for content-centric networking[J]. Computer Networks, 2013, 57(13): 2465-2482.
- [17] PENTIKOUSIS K, OHLMAN B, DAVIES E, et al. Information-centric networking: evaluation methodology[J/OL]. <http://www.ietf.org/proceedings/90/slides/slides-90-icnrg-8.pdf>.
- [18] EWMA[EB/OL]. <http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc431.htm>.
- [19] ARLITT M, CHERKASOVA L, DILLEY J, et al. Evaluating content management techniques for web proxy caches[R]. Technical Report, HPL-98-173, 1999.
- [20] NS-3[EB/OL]. <https://www.nsnam.org/>.
- [21] ndnSIM documentation[EB/OL]. <http://ndnSIM.net/>.
- [22] SPRING N, MAHAJAN R, WETHERALL D. Measuring ISP topologies with rocketfuel[J]. Networking IEEE/ACM Transactions, 2002, 32(4): 133-145.

#### 作者简介:



冯博昊 (1988-), 男, 北京人, 北京交通大学博士生, 主要研究方向为未来互联网体系架构、信息为中心网络、网络缓存、软件定义网络、时延容忍网络等。



周华春 (1965-), 男, 安徽贵池人, 北京交通大学教授、博士生导师, 主要研究方向为未来互联网体系架构、移动互联网、网络安全、空间网络等。



张宏科 (1957-), 男, 山西大同人, 北京交通大学教授、博士生导师, 主要研究方向为下一代互联网架构、协议理论与技术、移动互联网络路由、传感器网络技术。



张明川 (1977-), 男, 河南汝州人, 河南科技大学副教授、硕士生导师, 主要研究方向为下一代互联网架构、信息中心网络、路由协议、物联网技术等。