

基于流行内容感知和跟踪的协同缓存策略

张果¹, 胡宇翔¹, 黄万伟¹, 汪斌强¹, 曹路佳²

(1. 国家数字交换系统工程技术研究中心, 河南 郑州 450002; 2. 解放军信息工程大学, 河南 郑州 450001)

摘要: 针对内容中心网络 (CCN, content centric networking) 中沿路径缓存策略存在的问题, 设计了一种最大介数节点和边缘节点协同的缓存策略。根据网络拓扑特性, 在最大介数节点感知流行内容, 边缘节点协同跟踪; 依据内容流行度和缓存空间, 指定内容在传输路径的存储位置。仿真实验表明, 与经典算法相比, 该策略有效提高了缓存命中率, 同时降低了平均路由跳数比, 改善了缓存系统的效率。

关键词: 内容中心网络; 沿路径协同缓存; 流行内容; 感知与跟踪

中图分类号: TP393

文献标识码: A

Coordinated caching scheme based on popular content awareness and tracking

ZHANG Guo¹, HU Yu-xiang¹, HUANG Wan-wei¹, WANG Bin-qiang¹, CAO Lu-jia²

(1. China National Digital Switching System Engineering & Technological R&D Center, Zhengzhou 450002, China;

2. PLA Information Engineering University, Zhengzhou 450001, China)

Abstract: To overcome the problems of the on-path caching schemes in the content centric networking, a coordinated caching scheme based on the node with the max betweenness value and edge node was designed. According to the topology characteristics, the popular content was identified at the node with the max betweenness value and tracked at the edge node. The on-path caching location was given by the popularity and the cache size. Simulation results show that, compared with the classical schemes, this scheme promotes the cache hit ratio and decreases the average hop ratio, thus enhancing the efficiency of the cache system.

Key words: content centric networking, on-path coordinated caching, popular content, awareness and tracking

1 引言

网络技术的不断发展和人们应用需求的不断变化, 使互联网承载的业务日益多样化。以流媒体^[1]为主体的网络应用, 在 IP 网络的通信模式下重复传输内容, 消耗了大量的带宽, 导致网络不堪重负。同时, IP 网络的地址耗尽、管控复杂等问题变得更加突出。针对上述问题, 国内外研究机构开展了一系列有益的探索, 其中, 改良式的 CDN (content

delivery network) 和 P2P (peer to peer) 技术在一定程度上解决了网络新应用产生的新问题, 但是无法从根本上解决 IP 网络先天设计缺陷带来的问题。

针对上述问题, 研究人员设计了以内容为网络通信主体的信息中心网络 (ICN, information-centric networking)^[2]结构。在通信过程中, ICN 只关注用户通信的具体内容, 不再关心内容的具体位置, 试图“一劳永逸”地解决当前网络中面临的所有问题。其中, 内容中心网络^[3,4]是一个具有代表性的 ICN

收稿日期: 2016-07-21; 修回日期: 2016-09-22

基金项目: 国家自然科学基金创新研究群体基金资助项目 (No.61521003); 国家重点基础研究发展计划 (“973”计划) 基金资助项目 (No.2012CB315901, No.2013CB329104); 国家自然科学基金资助项目 (No.61372121, No.61309019, No.61309020, No.61572519, No.61502530); 国家高技术研究发展计划 (“863”计划) 基金资助项目 (No.2015AA016102, No.2013AA013505)

Foundation Items: The Foundation for Innovative Research Groups of the National Natural Science Foundation of China (No.61521003), The National Basic Research Program of China (973 Program) (No.2012CB315901, No.2013CB329104), The National Natural Science Foundation of China (No.61372121, No.61309019, No.61309020, No.61572519, No.61502530), The National High Technology Research and Development Program of China (863 Program) (No.2015AA016102, No.2013AA013505)

体系结构，内容普遍缓存是其重要特征。

内容中心网络 (CCN, content centric networking) 最初设计的缓存策略 (CEE, cache everything everywhere) 要求内容存储在返回路径的所有节点上。这种不加选择的内容缓存策略产生大量的缓存冗余，导致节点缓存内容同质化；同时，CEE 缓存策略使内容在节点有限的缓存空间内频繁替换，极大地增加了节点处理消耗。

为提高缓存系统效率，研究者们开展了一系列研究，从传输的角度^[5]，可分为路径之外的缓存和沿路径缓存。近期，出现了混合缓存策略的应用研究^[6]，基于节点角色划分，实现核心区域缓存集中式协同和边界区域沿传输路径缓存的混合应用，来提高缓存系统效率。

路径之外的缓存需要大量的通信开销来协同节点间的状态信息，限制了网络的可扩展性。沿传输路径的缓存策略^[7-17]从网络拓扑特征、数学统计概率、网络的社团属性等多个角度开展研究，以此提高缓存系统效率。主要存在的问题一方面多是没有考虑内容流行度，或假设内容流行度为已知参数，在 CCN 内各个节点都需要流行度参与决策；另一方面，以概率方式或社团重要度缓存的策略，使内容在传输路径上缓存存在无序性。

针对沿路径缓存策略存在的不足，设计了一种基于传输路径上流行内容感知和边缘节点跟踪的协同缓存策略 (PCA-MBTE, coordinated caching scheme based on popular content awareness at the max betweenness node and tracking at the edge node)。PCA-MBTE 旨在通过部分节点间隐性协同识别和跟踪流行内容，使流行内容有序地缓存在传输路径上，从而提高缓存利用率。本文主要贡献和创新工作如下。

1) 分析了内容的流行度演化模型，基于拓扑信息，提出了在 CCN 的重要节点进行流行内容感知，并在边缘节点协同跟踪的内容流行度监测思想。

2) 设计了自适应的流行内容监测门限策略，使传输路径上的缓存空间得到充分利用。

3) 设计了最大介数节点缓存和基于边缘节点调度的协同缓存策略，使流行内容在传输路径上有序缓存。

2 相关研究

针对 CEE 策略带来的缓存冗余问题，Eum 等^[7]

提出了 RCOne 缓存策略，在内容回传的节点中随机选择一个节点进行缓存，这种随机化的缓存策略易于实现，但并未考虑内容的流行度。

Psaras 等^[8]设计了 ProbCache 缓存策略，在内容回传的路径上，基于加权概率在沿途节点缓存内容。该策略可快速地将内容复制到离用户较近的网络节点，提升共享路径节点的效用。由于没有考虑对象的内容流行度，增加了内容对象在边缘节点的缓存竞争。

Chai 等^[9]提出了 Betw 缓存策略，依据网络拓扑特性，内容缓存在路径上中心度最高的节点缓存，其他节点不再缓存。该策略减少了内容传输的平均跳数，然而，随着内容请求的增加，它会导致重要节点缓存替换频繁，缓存利用率下降。

崔现东等^[10]提出了一种基于节点介数和内容更替率的缓存策略，降低了重要节点的内容替换频率，但是该策略没有考虑内容流行度，对缓存系统性能提升有限。

Ren 等^[11]设计了基于节点缓存收益的分布式缓存策略，较好地改进了缓存性能，但是需要各个节点都掌握内容的流行度，而且执行缓存策略时，操作复杂度较高^[12]。

Xu 等^[13]通过在各个节点监测历史缓存内容的利用率，选取传输路径上内容利用率最大的节点进行缓存。该策略监测了部分内容的流行度，并且需要在各个节点部署监测机制。

罗熹等^[14]提出了基于内容迁移的缓存策略，基于节点的重要度，根据节点的替换率、链接可靠度等信息，当节点负载过大时，内容按照一定的机制向重要节点一跳范围内的邻居节点迁移，降低了重要节点的负载。

蔡君等^[15]提出了一种基于社团重要度的缓存策略，将内容缓存在所经社团的重要度最大的节点，社团内的节点根据节点重要度和内容流行度选取不同的概率进行内容替换，该策略也需要各个节点掌握当前节点的内容流行度。

吴海博等^[16]提出了一种启发式概率缓存方法，基于内容热度和缓存收益构造的缓存概率对经过节点的内容进行缓存。该策略增加了内容的多样性，但是不能保证内容在传输路径上有序缓存。

芮兰兰等^[17]提出了基于节点中心度和内容流行度匹配的缓存策略，采用内容全局流行特征分布特征，实现了缓存效率的提高。在 CCN 中，内容

缓存破坏内容流行度的全局分布特性，因此，这种策略在实际网络环境中效率改善有限。

3 PCA-MBTE 方案设计

内容在从服务器向用户分发的过程中，随着时间的推移和感兴趣用户的增多，逐渐成为流行内容。在这个过程中，零散的兴趣分组先是在中心节点汇聚，使内容首先在中心节点呈现流行特性或趋势，最后在边缘区域成为流行内容。针对内容传输过程中流行特征的变化，先在传输路径最大介数节点感知流行内容并缓存，然后在边缘节点跟踪其流行度，进而制定缓存策略，将内容缓存在合理的节点上。

3.1 符号定义

首先描述最大介数节点和边缘节点的定义。

最大介数节点是指在用户到内容服务器传输路径上的节点中，介数值 (betweenness value) 最大的节点。边缘节点是指和用户直接相连的节点，即一跳距离的节点。

为方便下文描述，定义参数如下。

h_1 : 兴趣分组经过的路由跳数，在边缘节点初始值为 0，到达一个节点计数值加 1。

h_D : 数据分组经过的路由跳数，当兴趣分组命中时，将初始值设为 $h_D=h_1$ ，到达一个节点计数值减 1。

C : 表示节点 V_i 的缓存空间，可容纳 C 个数据块 (chunk)。

V_i : 边缘节点到介数最大节点的集合 $\{V_1, V_2, V_3, \dots, V_i\}$, $|V|$ 表示节点数量。

U_i : 内容请求者。

P : 内容提供者。

d_{core} : 最大介数节点到边缘节点的距离。

$C_B(V_n)$: 表示节点 V_n 的介数值。

3.2 内容流行度演化模型

在初始状态下，CCN 各个节点缓存均为空。用

户发送兴趣分组，经过路由转发到达服务器获取请求内容，然后数据沿请求路径返回给用户。随着时间的变化，请求该内容的用户和兴趣分组数量持续增加，内容逐渐变得流行。从时间和空间的角度分析，在内容非流行逐渐变为流行内容的过程中，在某一时间段内，内容流行度在传输路径上的不同节点表现不尽相同。如图 1 所示，某一内容 C_x 在 CCN 内传输时内容流行度的演化。若用户 U_1 和 U_4 在时隙 T 内同时请求内容 C_x ，那么对应的内容流行度最大值在节点 V_4 ，用户 U_2 、 U_3 、 U_5 和 U_6 在不同的时隙请求内容 C_x ，使其内容流行度分别在节点 V_2 (对应传输路径 $V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4 \rightarrow V_5$) 和节点 V_6 (对应传输路径 $V_6 \rightarrow V_7 \rightarrow V_8 \rightarrow V_4 \rightarrow V_5$) 达到最大值。受 Chai 等^[9]提出的基于 Betw 的缓存算法启发 (以图 1 中传输路径 $V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4 \rightarrow V_5$ 为例)，如果在介数最大的节点 V_4 预先识别出可能的流行内容，并建议边缘节点 V_1 跟踪其流行度，然后根据流行内容的不同等级设计缓存策略，使这些内容在节点 V_1 、 V_2 和 V_3 上有序分布，则可达到提高缓存命中率 and 降低内容请求时延的目的。

3.3 流行内容感知和流行度跟踪协同

在 CCN 中，内容缓存提高命中率和减小访问时延的同时，破坏了内容流行度分布，如节点缓存某一内容后，请求该内容的兴趣分组不再被转发至上游节点，那么内容在上游节点的流行度统计信息则变得不完整。上述内容缓存破坏流行度的特性使内容在节点已经不具备全局流行特征，因此，以全局流行度作为内容缓存依据的相关策略在实际网络环境中难以得到有效的应用。在缺少大规模信息交互的情况下，CCN 节点仅能获知当前节点的内容请求强度，无法感知内容的全局流行特征。而由 3.2 节流行度演化模型分析可知，内容在从源服务器向用户移动的过程中，其流行度是不断变化的。Chai 等^[9]的仿真实验验证了在介数较大的节点缓存内容

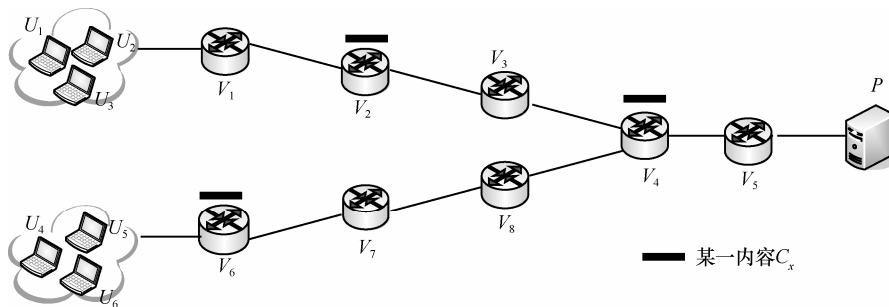


图 1 内容 C_x 流行度演化示意

可达到较好的缓存性能，这也表明内容在中心节点内容流行度较高，最后逐渐向边缘节点发散的特性。因此，考虑在最大介数节点感知流行内容，然后建议边缘节点跟踪该流行内容的流行度信息，再制定相应的缓存策略。

为方便下文讨论，定义如下。

定义 1 内容流行度。在任一节点，内容在一个确定时长内被请求的次数。那么对某一内容的流行度表示为

$$p(f) = \lambda_T \quad (1)$$

其中， T 为确定时长， λ_T 表示时间内内容被访问的次数。当内容 f 对应的流行度值 $p(f)$ 达到或超过某

$$P_f = \frac{\{p_f[i] + (1 - \beta)p_f[i - 1] + (1 - \beta)^2 p_f[i - 2] + \dots + (1 - \beta)^w p_f[i - w]\}}{\{1 + (1 - \beta) + (1 - \beta)^2 + \dots + (1 - \beta)^w\}} \quad (2)$$

其中， $p_f[i]$ 表示第 i 个窗口内内容 f 的即时流行度值， w 是观测时隙个数， $\beta (0 < \beta < 1)$ 为衰减因子，一般取 $\beta = \frac{2}{w+1}$ [18]。持续流行度是边缘节点评价内容流行程度的依据。

3.3.1 兴趣分组和数据分组格式扩展

兴趣分组和数据分组扩展结构如图 2 所示。扩展字段含义如下。

兴趣分组跳数是指用户发出兴趣分组，经过转发，到当前节点经过的路由跳数，兴趣分组到达边缘节点时，将边缘节点 ID 封装在该字段，跳数值为 0，每转发一次，计数值加 1。兴趣分组跳数用于计算路由距离。边缘节点 ID 用于标识兴趣分组来自于哪个边缘节点，保持边缘节点和最大介数节点的一致性。

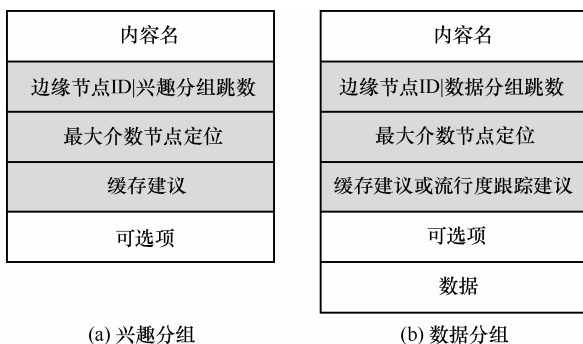


图 2 兴趣分组和数据分组扩展结构示意图

最大介数节点定位。网络各个节点的介数值可预先计算获知，兴趣分组在转发过程中，对比沿途节点的介数值，确定最大介数值节点，并根据到边

缘节点的距离（跳数）对节点位置进行定位。此处对应于数据分组中的最大介数节点定位，服务器收到兴趣分组后，从兴趣分组头解析出最大介数节点位置，然后封装在数据分组头部，用于识别最大介数节点。

缓存建议是指边缘节点经过计算，得出内容应该缓存的节点位置，用缓存节点到边缘节点的距离（路由跳数）来表示。需要指出的是，最大介数节点定位字段和缓存建议字段共用一个字段，在不同的条件下封装在不同的兴趣分组中。

定义 2 即时流行度。对定义 1 稍作改动，在大小为 WT 、步长为 T 的滑动窗口内，内容 f 在第 u 个时隙 T 内的流行度为 λ_u ，称 λ_u 为内容 f 在时隙 T 的即时流行度。即时流行度作为最大介数节点感知流行内容的度量指标。

定义 3 持续流行度。对于监测窗口内某一内容 f ，在大小为 WT 的滑动窗口内进行流行度统计，采用 EWMA (exponential weighted moving average) 模型来表示某时刻内容 f 的持续流行度。

数据分组跳数是指数据分组经过转发，到边缘节点的路由跳数，其初始值是在内容命中节点从兴趣分组解析获取，数据回传转发，每经过一个节点，其计数值减 1。另外，从兴趣分组内解析出的边缘节点 ID 和跳数值封装在一起。

缓存建议或流行度跟踪建议。缓存建议来自兴趣分组，解析出兴趣分组中的缓存建议然后封装在数据分组中，使数据在回传过程中能够缓存在指定位置。流行度跟踪建议是指内容在最大介数节点被确定为流行内容后，建议边缘节点跟踪的标识，在实际中，最大介数节点 ID 和流行度跟踪建议封装一起发送。

3.3.2 最大介数节点流行内容感知

1) 最大介数节点确定及处理流程

在用户到内容服务器传输路径上介数值最大的节点称为最大介数节点。当同一条路径上有 2 个（或 2 个以上）节点的介数值相同且最大时，选取距离用户最近的节点作为最大介数节点。节点介数

定义参照文献[9]。

$$BC(V_a) = \sum_{i \neq v \neq j \in V} \frac{\sigma_{ij}(V_a)}{\sigma_{ij}} \quad (3)$$

其中, σ_{ij} 表示连接节点 V_i 和节点 V_j 的所有最短路径数目, $\sigma_{ij}(V_a)$ 表示路径中经过节点 V_a 的路径数目。整个网络拓扑结构中各节点的介数值, 可通过预先计算获知。

当兴趣分组到达边缘节点时, 若在流行度跟踪表 (PTT, popularity tracking table) 命中, 表示该内容已被确定为流行内容, 则不需要再定位最大介数节点。若在跟踪表中未命中, 则需要确定最大介数节点。如算法 1 所示。

根据算法 1, 兴趣分组到达内容服务器时, 已经完成最大介数节点定位, 即得到了边缘节点到最大介数节点的距离。在数据回传过程中, 根据定位信息, 对最大介数节点进行标识, 以便于兴趣分组到达时能够分辨出是普通节点还是最大介数节点, 同时兴趣分组能够判别出是否到达属于自己的最大介数节点, 即转发该兴趣分组的边缘节点和最大介数节点一致, 保证二者属于同一传输路径。

算法 1 最大介数节点定位算法

初始化: content name(C_i), $C_B=0$

内容请求过程: 发送内容请求 (interest packet)

① for each V_n from U to P

② if 内容存储器 (CS, content store) 包含请求内容 C_i

③ then 返回数据;

④ else

⑤ 获取本地 $C_B(V_n)$;

⑥ if $C_B(V_n) > C_B$

⑦ then $C_B = C_B(V_n)$

⑧ $d_{core} = h_1$;

⑨ 按照前向转发库 (FIB) 表项执行下一跳路由转发。

传输路径最大介数节点确定后, 从服务器回传的数据经过最大介数节点时, 处理流程如下。

步骤 1 先查询流行内容感知表 (PCAT, popular content-aware table) 内是否包含该内容信息, 若没有相关信息, 表示该数据第一次被请求, 那么在最大介数节点的流行内容感知表内添加该内容的相关索引项, 以及对应的 d_{core} 值和边缘节点 ID, 执行步骤 3; 若包含该内容信息, 进入步骤 2。

步骤 2 查询结果和边缘节点 ID 匹配, 进一步

检测是否达到设定的阈值。若达到阈值, 则称为流行内容, 那么在回传数据分组头(如图 2 所示)封装建议边缘节点跟踪流行度标识和当前节点 (最大介数节点) ID, 以及对应的距离 d_{core} , 使边缘节点验证是否是相对应的最大介数节点发来的建议信息, 进而获取进一步的统计信息, 为后续缓存提供依据。若未达到阈值, 直接执行步骤 3。

步骤 3 转发数据。需要指出的是, 在最大介数节点的流行内容感知表内添加内容索引后, 后续兴趣分组到达最大介数节点, 在 PCAT 命中, 边缘节点 ID 匹配且 $h_1=d_{core}$, 则更新相对应的内容流行度计数和兴趣分组头的最大介数定位值为 d_{core} ; 否则, 不更新。一个确定时长结束, 删除访问次数小于某一特定值的内容表表项, 以此来防止流行内容感知表过大。

2) 流行度阈值 δ_{th} 和最大介数节点距离 d_{core} 的关系

Chai 等^[9]指出, 内容流行度符合 Zipf 函数分布, 即 $p(i) = \frac{\theta}{i^\alpha}, \theta = \left(\sum_{j=1}^{|F|} \frac{1}{j^\alpha} \right)^{-1}$, 其中, $p(i)$ 是第 i 个流行内容被访问的概率, α 是内容流行度变化因子, $|F|$ 是内容数量, 内容按照访问概率从高到低排序。

在监测时隙 T 内, 到达节点的兴趣分组总数为 N_{total} 时, 请求内容 m_y 的兴趣分组数目 $N_{total}p(y)$, 令 $\delta_{th} = n_{m_y}$, 则有

$$\delta_{th} = n_{m_y} = N_{total}p(y) = N_{total} \frac{\theta}{y^\alpha} \quad (4)$$

当最大介数节点到边缘节点距离为 d_{core} , 各节点配置的缓存空间大小都为 C 时, 传输路径上可缓存的内容数目为

$$y = d_{core} C \quad (5)$$

图 3 描述了当 $|F|$ 和 α 一定, N_{total} 取不同值时, 流行度阈值和最大介数节点位置 d_{core} 的关系。 d_{core} 增大, 阈值减小, 即在路径上可存储的内容数目就越多。在空间限制的情况下, 确定合理的阈值可提高缓存系统的效率, 式(4)和式(5)可为阈值的设定提供理论依据。

3.3.3 边缘节点流行度跟踪

边缘节点流行度跟踪涉及到以下 2 个方面。

1) 流行度跟踪表项的建立。边缘节点根据接收到的数据分组头内的跟踪流行度建议标识, 满足 $h_0=D$, 同时, 分组头内的边缘节点 ID 和当前节点

相同，那么该内容需要跟踪，则在边缘节点的流行度跟踪表 (PTT) 内添加该内容索引，同时添加对应的最大介数节点的位置信息(到边缘节点的距离(跳数)、相应的接口和对应的最大介数 ID，如表 1 所示。

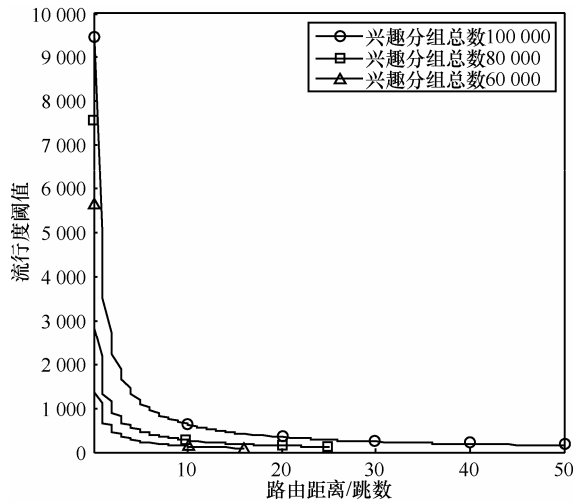


图 3 最大介数位置和阈值的关系

表 1 边缘节点 PTT 表结构

内容名索引	流行度	最大介数节点距离	边缘节点转发接口	最大介数节点 ID
ID ₁	50	10	FACE ₁	V ₁
ID ₂	60	9	FACE ₁	V ₁
ID ₃	45	12	FACE ₀	V ₂

2) 内容跟踪表项的更新机制。在流行度测量窗口内，当新的内容请求到达节点时，先在内容流行度跟踪表内进行查询，若命中，则表明内容需要跟踪，要更新对应的内容流行度统计值；未命中则不跟踪。当跟踪时隙结束时，根据式(2)得到内容流行度排名，为缓存策略提供依据。

3.4 最大介数节点和边缘节点沿路径协同缓存

缓存分为最大介数节点缓存和边缘节点调度的缓存，分别按照最大介数节点缓存(流行初期)和边缘节点调度缓存(流行后期)阐述如下。

3.4.1 最大介数节点缓存策略

如 3.3.2 节所述，传输路径最大介数节点确定后，在最大介数节点建立流行内容感知索引，统计后续到达该节点的相关兴趣分组，以获取内容的即时流行度信息。统计时隙结束时，节点根据内容流行度排名，缓存经过该节点的内容。缓存策略如下。

算法 2 缓存策略算法

初始化: content name (C_i)

内容请求: 发送内容请求 (interest packet)

- 1) for each V_n from U to P
- 2) if CS 包含请求内容 C_i
- 3) then
- 4) 返回数据;
- 5) else
- 6) if V_i 是最大介数节点且包含的索引
- 7) then
- 8) 更新 C_i 对应的统计值;
- 9) 按照 FIB 表项执行下一跳路由转发。

数据响应: 接收内容应答 (data packet)

- 10) for each V_n from P to U
- 11) if V_n 是最大介数节点
- 12) if 流行内容感知表包含 data packet

对应的内容索引

- 13) if data packet 对应的缓存标识为 true;

- 14) then
- 15) 缓存 data packet;

16) if data packet 对应的内容确定为流行内容

- 17) then
- 18) 封装流行度跟踪建议标识、

V_n 及 d_{core} 到 data packet;

- 19) else
- 20) 在流行度跟踪 (PTT) 表内建立对应的内容索引，边缘节点 ID 和最大介数值;

21) 按照未决兴趣表 (PIT) 项执行 data packet 下行转发。

3.4.2 边缘节点缓存调度策略

边缘节点缓存调度策略的核心思想: 边缘节点根据持续流行度排名，结合每个节点的缓存空间大小和在传输路径上的位置，制定缓存策略，如图 4 所示。边缘节点缓存调度策略是指在统计时隙结束后，边缘节点根据持续流行度排名，经过计算，指定内容在传输路径上的缓存位置。边缘节点指定的缓存节点应该在该传输路径的最大介数节点截止，最大介数节点的上游节点不作为该路径的缓存节点。这样设计使最大介数节点成为传输路径上的过滤识别节点，减少最大介数节点上游节点的缓存竞争和替换。需要指出的是，若边缘节点和最大介数节点之间的某一节点是其他路径的最大介点，在缓

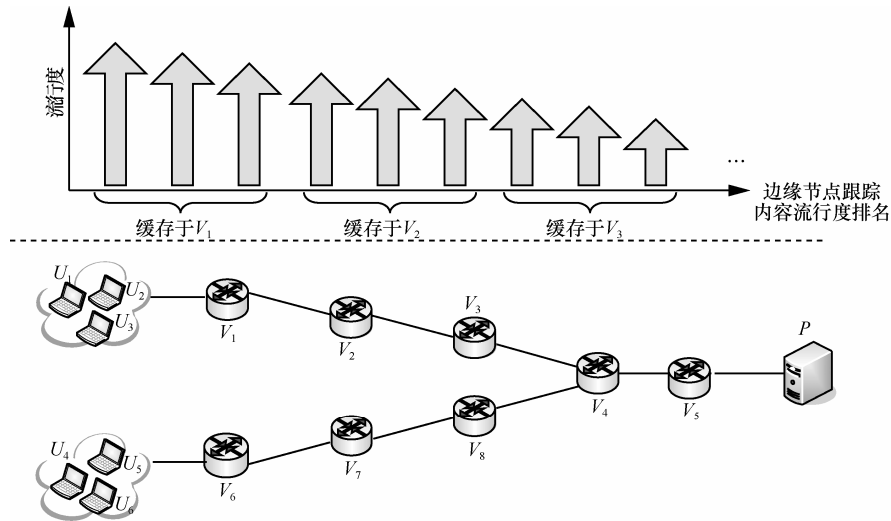


图 4 边缘节点缓存策略示意

存时跳过该节点，向上游节点推移一个缓存位置。

节点对应一个转发接口时，若内容 f 对应的流行度排名 P_i ，那么内容缓存节点到边缘节点的距离为

$$d_f = \left\lceil \frac{P_i}{C} \right\rceil \quad (6)$$

当边缘节点流行度跟踪对应多个接口，即流行内容对应着不同的传输路径时，需要分类进行缓存位置计算，举例如下。3 个转发接口 $FACE_0$ 、 $FACE_1$ 和 $FACE_2$ ，分别对应的流行内容集合为 S_0 、 S_1 和 S_2 。其中， a 、 b 、 c 分别是 S_0 、 S_1 和 S_2 的流行内容的排名最前 3 个内容，满足 $a+b+c=C$ ，那么生成的缓存节点到边缘节点的位置分别为

$$FACE_0: d_{f_face_0} = \begin{cases} 0, & P_i \leq a \\ 1 + \left\lceil \frac{P_i}{C} \right\rceil, & P_i > a \end{cases} \quad (7)$$

$$FACE_1: d_{f_face_1} = \begin{cases} 0, & P_i \leq b \\ 1 + \left\lceil \frac{P_i}{C} \right\rceil, & P_i > b \end{cases} \quad (8)$$

$$FACE_2: d_{f_face_2} = \begin{cases} 0, & P_i \leq c \\ 1 + \left\lceil \frac{P_i}{C} \right\rceil, & P_i > c \end{cases} \quad (9)$$

当 $d_f \geq d_{core}$ ，缓存建议标记为空，即不缓存。

在内容发起请求和数据传输过程中，根据 d_f 的值将内容缓存在指定位置，按照兴趣分组和数据分组的处理流程讨论如下。

1) 内容请求过程。用户发送兴趣分组请求内容 f ，到达边缘节点，在 CS (content store) 命中，则直接返回数据 (流行度更新部分不再阐述)。若在

CS 中未命中，则查询流行度跟踪表。

① 若在 PTT 命中，路由距离 h_1 开始计数，同时，更新相应的流行度统计值，并根据其流行度排名分 2 种情况处理：a) 需要将对应的 d_f 值添加在分组头部，向上游节点转发，到达节点后要识别节点是否为最大介数节点，当该节点是其他传输路径的最大介数节点时且在 CS 内未命中， $d_f = d_f + 1$ ，若此时 $d_f \geq d_{core}$ 则缓存建议标记为空，继续向上游节点转发，这种情况称之为边缘节点获得缓存建议；b) 未获得缓存建议，则携带对应的最大介数节点位置，向上游节点转发。

② 若未在 PTT 命中，则携带边缘节点介数值，路由距离 h_1 开始计数，向上游节点转发。

需要指出的是，兴趣分组经过最大介数节点或在最大介数节点命中，则更新对应的流行度统计值，和 3.3.2 节最大介数节点缓存策略处理方法相同，不再赘述。

2) 服务器或节点响应过程。当兴趣分组命中后，服务器或节点分 2 种情况解析兴趣分组：a) 有缓存建议，得到 h_1 和缓存建议位置，令 $h_D = h_1$ ，通过对比，若 $d_f < h_D$ ，则将 d_f 添加进数据分组头，同时令 $h_D = h_1$ ，返回途中将数据缓存在指定位置，若 $d_f \geq h_D$ ，则在返回数据时，缓存建议标识为空，即该内容不缓存；b) 无缓存建议，数据按照正常流程转发。

缓存替换采用 LRU 替换策略。

4 仿真实验环境与性能评价指标

4.1 仿真实验环境

在商用服务器 (2.70 GHz CPU, RAM 2.0 GB)

上搭建开源仿真平台^[19]。该平台实现了基本数据单元结构和路由转发流程，支持路由、转发和内容缓存算法扩展，同时，提供了经典的缓存算法代码。在实验中，按照 3.3.1 节对兴趣分组和数据分组头结构进行扩展，增加相应的字段。

网络拓扑设置。采用 GT-ITM 生成 50 个节点的随机网络拓扑结构。网络中内容对象总数 N 为 10^6 个 (chunk)，每个大小为 10 KB。为公平分析同类算法的性能，网络内各节点缓存容量相同。为分析缓存大小变化对算法性能的影响，缓存大小可变，链路带宽为 10 Mbit/s，时延为 2 ms。部署 2 个内容服务器，负责内容的存储和发布，各服务器随机存储 5×10^5 个 chunk。在网络边缘节点中随机选取 2 个节点与内容服务器直接相连。其余节点均作为用户接入节点。服务器部署完成后，各节点的介数值可使用工具^[20]一次性计算出来。

实验设计。用户发起兴趣分组，兴趣分组到达服从 $\lambda=100$ 个/秒的泊松过程^[9]，请求概率服从 Zipf 分布^[9]，即第 i 个内容的请求概率为：

$$p(i) = \frac{\theta}{i^\alpha}, \theta = \left(\sum_{j=1}^{|\mathcal{I}|} \frac{1}{j^\alpha} \right)^{-1}。仿真时间设为 500 s，统计$$

间隙 $T=2$ s，窗口长度 $W=3$ ，节点缓存初始状态为空。缓存空间和内容总数比值默认为 10^{-5} 。每个实验场景重复 3 次。针对不同的 d_{core} 路径长度，根据式(4)和式(5)预先进行阈值计算，供节点使用。在实验开始之前，兴趣分组随机分布在各个接入节点。缓存替换策略为 LRU。

在实验中，主要考虑 2 个因素对系统性能的影响：1) 节点缓存大小变化；2) 内容流行度变化。

4.2 性能评价指标

在实验中，将 PCA-MBTE 策略与 CCN 研究领域的经典算法 CEE^[3]、ProbCache^[8]和 Betw^[9]进行对比，性能评价指标包括以下几个方面。

缓存命中率 (CHR, cache hit ratio)：网络中节点缓存内容响应兴趣分组总数 (*interest*) 和兴趣分组总数 (*Interest*) 的比值。缓存内容的命中率越高，对应的源服务器响应率和负载就越小。

$$CHR = \frac{\sum_R interest}{\sum_R Interest} \quad (10)$$

平均路由跳数比 (AHR, average hop ratio)：兴

趣分组实际命中内容的节点经过的路由跳数 (*hop*) 与用户在源服务器获取内容所需的路由跳数 (*Hop*) 之比

$$AHR = \frac{\sum_R hop}{\sum_R Hop} \quad (11)$$

其中，式(10)和式(11)中， R 表示时间内发送的兴趣分组数量。

4.3 性能分析

4.3.1 缓存空间大小变化

图 5 和图 6 是 $\alpha=0.9$ ，缓存空间与总内容数比在 $[10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}]$ 变化时，4 种缓存策略的缓存命中率和路由跳数占比变化对比。

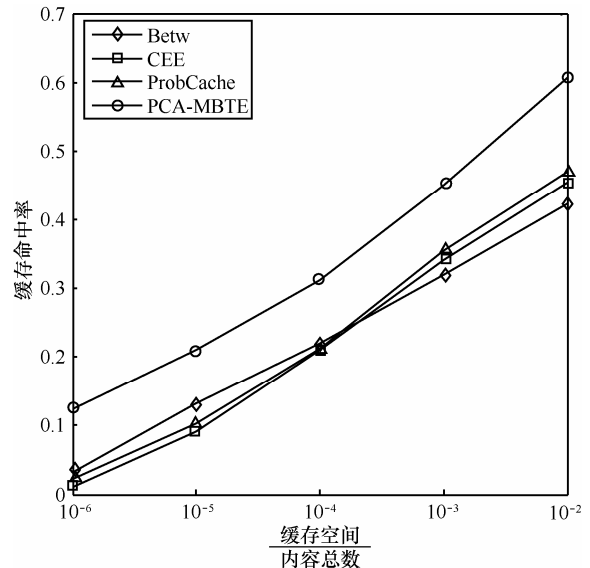


图 5 缓存命中率随内容缓存空间占比变化

由图 5 可以看出，随着缓存空间与内容总数比值的增大，4 种策略的缓存命中率均有不同程度的增加。ProbCache 和 CEE 的缓存命中率相差不大，而随着比值的增大，Betw 的性能略低于上述 2 种策略，这是因为随着空间的增大，上述 2 种策略能够容纳较多的内容。而 Betw 将内容存储在重要节点，导致重要节点内容替换频繁，同时，非重要节点缓存空间利用率不高，因此，整体性能与 ProbCache 和 CEE 相比有所下降。PCA-MBTE 策略在最大介数节点实时感知流行内容，并且和边缘节点协同，使内容在不同流行阶段缓存于传输路径的不同位置。如在流行初期，内容缓存在最大介数节点，随着流行度的增加，内容排列在边缘节点到最大介数节点的传输路径上，

能够满足用户对不同流行内容的需求。因此，在缓存空间较小时，也能表现出较好的性能。随着空间的增大，最大介数节点能够自动调整流行内容感知阈值，最大介数节点可缓存的内容增多，在边缘节点和最大介数节点间的空间也增大，从而能够容纳更多的流行内容，因此，缓存命中率迅速提升。

图 6 是平均路由跳数比的变化情况。随着缓存空间的增加，缓存系统容纳的内容增多，用户在节点即可获取所需内容，平均路由跳数比越来越小。和 CEE 相比，ProbCache 加快了内容向边缘节点缓存的移动速度，但是也未考虑内容流行度，导致替换频繁，因而整体性能和 CEE 相当，随着缓存空间的增大，二者性能有较大提升。而 Betw 策略使重要节点内容过载，缓存替换频繁，使本该在缓存节点获取的流行内容仍需到服务器获取，从而增加了路由跳数，因此，平均路由跳数占比减小速度较慢。PCA-MBTE 策略在缓存空间增大时，通过自动调整感知阈值，使更多的流行内容按照流行度排列在边缘节点和最大介数节点之间的缓存内，减少了用户获取内容的路由跳数，由图 6 可以看出，随着缓存空间与总内容数比值的增大，即空间的增加，平均路由跳数比快速降低。

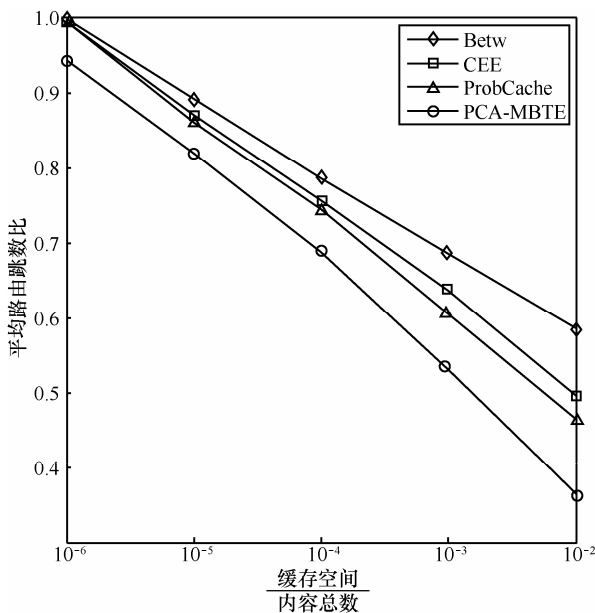


图 6 平均路由跳数比随缓存空间占比变化

4.3.2 内容流行度变化

图 7 和图 8 给出缓存空间与总内容数比值为 10⁻⁵ 时，4 种策略的缓存命中率和平均路由跳数

比随流行度因子 α 的变化情况。随着 α 的增大，网络内热点内容的流行度越来越高，即内容请求的集中度越来越高。从图 7 和图 8 可以看出，随着内容流行度的提高，4 种策略的缓存命中率均有所提高，平均路由跳数比也在减小，PCA-MBTE 策略明显优于其他 3 种策略。

图 7 中， $\alpha = 0.7$ 时，PCA-MBTE 策略的缓存命中率明显较高。此时，需要缓存的内容数目较多，ProbCache、CEE 和 Betw 因为节点空间限制，存储内容数量有限，并且对内容流行度变化不够敏感，因而性能提升有限。随着内容流行程度的增加，PCA-MBTE 的优势越来越明显。随着流行度因子变大，流行内容越来越集中，PCA-MBTE 策略能够将流行内容排列在合理的节点上，因而 α 越大，PCA-MBTE 策略的缓存命中率越大，缓存命中率增速越快。

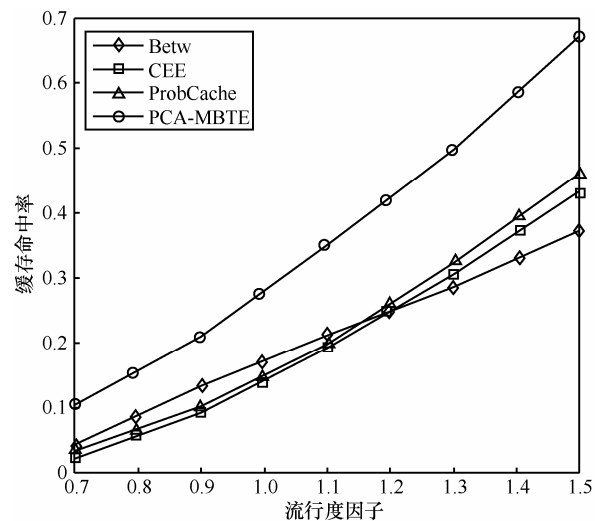


图 7 流行度变化时缓存命中率变化情况

图 8 中，4 种缓存策略的平均路由跳数比随着内容流行度的增加而减少。可以看出，Betw 策略的平均路由跳数比随着内容流行度的增加而降低，但是与 CEE 和 ProbCache 相比，由于只在重要节点缓存，缓存替换频繁，缓存整体利用率不高，从而导致性能提升有限。PCA-MBTE 策略考虑了内容流行度特性，在流行内容区分度不高的情况下 ($\alpha = 0.7$)，采用最大介数节点过滤和边缘节点跟踪的方式，能够给出相对合理的缓存位置建议，因而有较好的性能表现。最大介数节点可根据路由距离选择合适的流行内容感知阈值，总能较好地利用传输路径上的缓存空间。

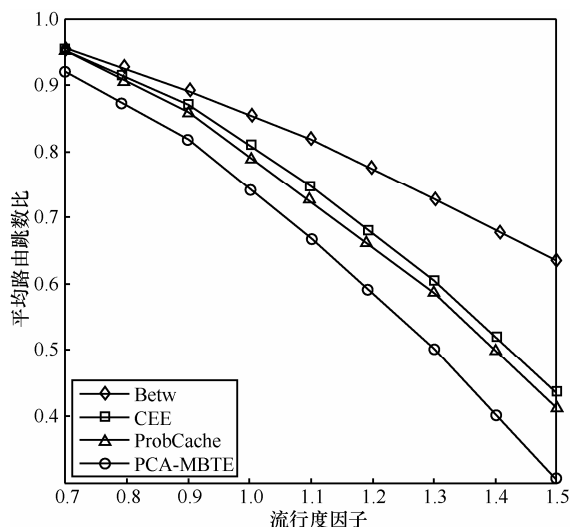


图 8 流行度变化时平均路由跳数比变化情况

4.3.3 处理开销

PCA-MBTE 策略的开销主要是部署监测算法的开销和兴趣分组中增加的字段带来的通信开销。

ZHANG 等^[21]分析了当前硬件处理工艺,设计了一种逐分组处理的支持 40 Gbit/s 处理速度流行度监测算法,并分析了算法的复杂度。如前文分析,当前的 CCN 缓存策略多数没有考虑内容流行度,或者假设内容流行度为已知量进行算法研究。PCA-MBTE 策略从应用部署的角度,不回避流行度监测算法的部署,以少量的监测点换取性能的有效提高。与 CEE、ProbCache 和 Betw 相比,增加了流行度监测消耗。

通信开销方面,以在分组头部扩展了有限的字段为代价,换取了内容流行度在传输路径上的分布信息,从而使缓存系统实现了对流行内容的有效调度,与取得的收益相比,通信开销可以忽略。

5 讨论

内容流行度是 CCN 中影响内容缓存效率的一个重要参数。然而,若在各个节点都部署流行度监测架构,会带来极大的资源浪费,在实际网络中也不太可行。从内容传输路径的角度,利用网络拓扑特征,在最大介数节点和边缘节点进行内容流行度监测,符合内容传输时流行度的时空变化特征,也满足了制定缓存策略时对流行度信息的需求。

以最大化消除系统冗余缓存为目的的缓存策略,并不一定能使缓存系统性能达到最优。不少研究^[22-25]表明,内容越流行,越靠近用户,缓存系统的效率就越高。非常流行的内容在网络内需要一定

的缓存冗余。PCA-MBTE 策略考虑了传输路径上的协同和差异缓存,并未消除不同路径上的缓存冗余。将内容按流行度有序排列在靠近用户的路由节点上,较好地满足了用户的内容请求。以传输路径最大介数节点作为缓存截止的标记,将流行内容缓存在靠用户一侧,也降低了网络内重要节点的负载。

PCA-MBTE 策略将流行度监测点部署在有限的节点上,并且根据传输路径的缓存空间容量来确定流行度感知阈值,使感知节点阈值设置具有自适应性,可以较好地利用传输路径上有限的缓存空间,能够在一定程度上提高缓存系统性能。但是,PCA-MBTE 没有考虑共用传输路径内缓存的竞争问题,缓存替换策略降低了这种竞争导致的损失,但并不能从根本上消除这种缺陷,这也是 PCA-MBTE 策略后续改进的一个方向。

6 结束语

内容普遍缓存是 CCN 的核心特征,及时、合理地将内容放置在指定的节点,是提升缓存效率的有效途径。考虑内容流行度在传输路径上的变化特点,设计了基于最大介数节点感知流行内容、边缘节点协同跟踪的缓存机制。在最大介数节点识别流行内容,同时,根据内容流行度进行缓存。边缘节点根据流行度跟踪信息,将内容有序地放置在边缘节点和最大介数节点间的缓存位置上。仿真显示,上述方法有效提高了缓存命中率,改善了缓存系统效率。后续研究工作考虑在最大介数节点和边缘节点的差异化缓存分配,以期更好地发挥算法的性能。

参考文献:

- [1] JANGAM A, RAVINDRAN R, CHAKRABORTI A, et al. Realtime multi-party video conferencing service over information centric network[C]//Workshop on Multimedia Streaming in Information Centric Networks (MUSIC) at ICME, 2015.
- [2] XYLOMENOS G, VERVERIDIS C, SIRIS V, et al. A survey of information-centric networking research[J]. IEEE Communications Surveys & Tutorials, 2014, 16(2): 1024-1029.
- [3] JACOBSON V, SMETTERS D K, THORNTON J D, et al. Networking named content[J]. Communications of the ACM, 2012, 55(1): 117-124.
- [4] ZHANG L, AFANASYEV A, BURKE J, et al. Named data networking[J]. ACM SIGCOMM Computer Communication Review, 2014, 44(3): 66-73.
- [5] ZHANG G Q, LI Y, LIN T. Caching in information centric networking: a survey[J]. Computer Networks the International Journal of Computer & Telecommunication Networking, 2013, 57(16): 3128-3141.
- [6] ZHANG G, WANG X, GAO Q, et al. A hybrid ICN cache coordination

- scheme based on role division between cache nodes[C]//IEEE Global Communications Conference, 2015.
- [7] EUM S, NAKAUCHI K, MURATA M, et al. CATT: potential based routing with content caching for ICN[C]//The 2nd ICN Workshop on Information-Centric Networking. 2012: 49-54.
- [8] PSARAS I, CHAI W K, PAVLOU G. Probabilistic in-network caching for information-centric networks[C]//ACM SIGCOMM Workshop on Information-Centric Networking. 2012: 55-60.
- [9] CHAI W K, HE D, PSARAS I, et al. Cache “less for more” in information-centric networks[C]//IFIP Networking. 2012: 27-40.
- [10] 崔晓东, 刘江, 黄韬, 等. 基于节点介数和替换率的内容中心网络网内缓存策略[J]. 电子与信息学报, 2014, 36(1): 1-7.
CUI X D, LIU J, HUANG T, et al. A novel in-network caching scheme based on betweenness and replacement rate in content centric networking[J]. Journal of Electronics & Information Technology, 2014, 36(1): 1-7.
- [11] REN J, QI W, WESTPHAL C, et al. Magic: a distributed max-gain in-network caching strategy in information-centric networks[C]//In IEEE INFOCOM Workshop NOM. 2014: 470-475.
- [12] CÉSAR B, THOMAS S, OLIVIER F. A comparison of caching strategies for content centric networking[C]//IEEE Global Communication Conference. 2015.
- [13] XU A F, TAN X D, TIAN Y. Design and evaluation of a utility-based caching mechanism for information-centric networks[C]//IEEE ICC'15. 2015:5535-5540.
- [14] 罗熹, 安莹, 王建新, 等. 内容中心网络中基于内容迁移的协作缓存机制[J]. 电子与信息学报, 2015, 37(11): 2790-2794.
LUO X, AN Y, WANG J X, et al. Cooperative caching mechanism with content migration in content-centric networking[J]. Journal of Electronics & Information Technology. 2015, 37(11): 2790-2794.
- [15] 蔡君, 余顺争, 刘外喜. 基于节点社团重要度的 ICN 缓存策略[J]. 通信学报, 2015, 36(6):173-182.
CAI J, YU S Z, LIU W X. Caching strategy based on node's importance to community in information-centric networks[J]. Journal on Communications, 2015, 36(6): 173-182.
- [16] 吴海博, 李俊, 智江. 基于概率的启发式 ICN 缓存内容放置方法[J]. 通信学报, 2016, 37(5):62-72.
WU H B, LI J, ZHI J. Probability-based heuristic content placement method for ICN caching[J]. Journal on Communications, 2015, 37(5): 62-72.
- [17] 芮兰兰, 彭昊, 黄豪球, 等. 基于内容流行度和节点中心度匹配的信息中心网络缓存策略[J]. 电子与信息学报, 2016, 38(2): 325-331.
RUI L L, PENG H, HUANG H Q, et al. Popularity and centrality based selective caching scheme for information-centric networks[J]. Journal of Electronics & Information Technology, 2016, 38(2): 325-331.
- [18] MONTGOMERY D, JOHNSON D, GARDINER J. Forecasting and time series analysis[M]. McGraw-Hill: New York, 1990.
- [19] CÉSAR B. Social CCNSim[EB/OL]. <https://github.com/mesarpe/socialcensim>.
- [20] DIMITRIS K. Social Network Visualizer[EB/OL]. <http://socnetv.sourceforge.net>.
- [21] ZHANG G, ZHANG J H, WANG B Q, et al. On-line popularity monitoring method based on Bloom filters and hash tables for differentiated traffic[J]. China Communications, 2016, 13(z1):72-86.
- [22] ROSSINI G, ROSSI D. Caching performance of content centric networks under multi-path routing (and more)[EB/OL]. <http://perso.telecom-paristech.fr/~drossi/paper/rossi11ccn-techrep1.pdf>. 2015-04-15.
- [23] 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.
- [24] FRICKER C, ROBERT P, ROBERTS J, et al. Impact of trafficmix on caching performance in a content-centric network[C]//IEEE INFOCOM NOMEN Workshops. 2012:310-315.
- [25] DABIRMOGHADDAM A, BARIJOUGH M M, GARCIA-LUNA-ACEVES J J. Understanding optimal caching and opportunistic caching at the edge of information-centric networks[C]//ACM International Conference on Information-Centric Networking. 2014: 47-56.

作者简介:



张果 (1985-), 男, 河南南阳人, 国家数字交换系统工程技术研究中心博士生, 主要研究方向为新型网络体系结构、内容中心网络。



胡宇翔 (1982-), 男, 河南周口人, 博士, 国家数字交换系统工程技术研究中心副教授, 主要研究方向为新型网络体系结构、网络安全。



黄万伟 (1979-), 男, 江苏盐城人, 博士, 国家数字交换系统工程技术研究中心讲师, 主要研究方向为网络信息安全。



汪斌强 (1963-), 男, 安徽安庆人, 博士, 国家数字交换系统工程技术研究中心教授、博士生导师, 主要研究方向为新型网络体系结构设计。

曹路佳 (1983-), 男, 河北抚宁人, 解放军信息工程大学助教, 主要研究方向为网络安全。