

基于内容流行度和节点重要度的 CCN 缓存策略

郑凯月, 潘沛生

(南京邮电大学 通信与信息工程学院, 江苏 南京 210003)

摘要: 内容中心网络 (content centric network, CCN) 强调以内容资源本身为中心, CCN 网络在路由节点对内容进行缓存, 使内容接近用户, 因而缓存策略研究成为 CCN 网络的关键技术之一。传统的主流缓存策略 Always (处处缓存)、LCD (leave copy down) 都会很大程度上造成节点空间的浪费和缓存内容的冗余。改进的 Betw (betweenness) 方案仅考虑了节点介数, 容易造成两个问题: 一是导致最流行的内容只缓存在高介数的节点, 无法到达最靠近用户的位置; 二是导致高介数的节点缓存更换频繁, 流行内容缓存时间短。为了解决这一问题, 提出一种内容流行度与节点重要度相结合的缓存机制, 实现优化的缓存协作。在 ndnSIM 平台进行仿真, 结果表明该方案提高了路由节点缓存命中率, 减小了获取内容的平均跳数。

关键词: 内容中心网络; 内容流行度; 节点重要度; 协作缓存; 流行度排名

中图分类号: TP393

文献标识码: A

文章编号: 1673-629X(2018)06-0073-04

doi: 10.3969/j.issn.1673-629X.2018.06.016

A Caching Scheme Based on Content Popularity and Betweenness in Content Centric Network

ZHENG Kai-yue, PAN Pei-sheng

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

Abstract: Content center network (CCN) emphasizes the content resource itself as the center. CCN network caches the content in the routing node in order to make the content close to the user, so the cache scheme research has become one of the key technologies of CCN network. The traditional mainstream cache strategy like Always (always cached), LCD (leave copy down) will be largely caused by the waste of node space and the redundancy of content. The improved scheme Betw only takes into account the betweenness, which can lead to two main problems: one is that the most popular content is only cached in the node with the larger betweenness, which can not reach the nearest user position; the other is that the Betw has the more frequent replacement, which will decrease the time of the content cache in node. For this, we propose a coporate caching scheme considering content popularity and betweenness to realize optimized cache cooperation. The simulation on the ndnSIM platform shows that the proposed scheme improves the hit rate of the routing node and reduces the average hop count for getting the content.

Key words: CCN; content popularity; betweenness; coporate cache; popularity ranking

0 引言

随着互联网技术的发展, 以 IP 为基础的 TCP/IP 网络体系架构逐渐暴露出诸多问题, 比如可扩展性较差、安全可控性低、移动性支持不足、服务质量难以保证等。互联网用户行为也发生变化, 转变为关注数据的内容而不是关注服务器和主机 IP 地址, 因此如何更快、更准确、更高效地获取数据成为下一代网络研究的热点问题。

内容中心网络 (CCN) 是一个以内容为中心, 将信

息对象作为架构的网络体系。CCN 网络每一个节点都有一定的缓存功能, 利用网络内置缓存提高传输效率。缓存的研究主要有两个方面: 一是缓存放置策略, 二是缓存替换策略。缓存放置策略用于决定某一节点处是否缓存该内容, 缓存替换策略用于解决当某一节点缓存已满的情况下如何实现新到达内容的缓存问题。传统的缓存放置策略有: LCE^[1] (leave copy everywhere), 即 Always 缓存策略; LCD (leave copy down); Prob (copy with probability); Betw^[2] (cache based on

收稿日期: 2017-06-23

修回日期: 2017-10-25

网络出版时间: 2018-02-24

基金项目: 江苏省高校自然科学基金基础研究计划重大项目 (08KJD510001)

作者简介: 郑凯月 (1992-), 女, 硕士研究生, 研究方向为内容中心网络; 潘沛生, 博士, 教授, 研究方向为协作通信。

网络出版地址: <http://kns.cnki.net/kcms/detail/61.1450.TP.20180224.1519.058.html>

betweenness)。

(1) LCE 策略要求内容在分发路径的所有节点都缓存,这样会导致网络中缓存节点空间的浪费和缓存内容的冗余。

(2) LCD 策略要求只在命中节点下一跳放置缓存,这样需要多次请求同一内容才会将该内容复制到靠近客户端的地方,同样会产生大量的内容冗余备份。

(3) Prob 策略每个沿途节点都以概率 p 缓存内容项,而以概率 $1-p$ 不缓存内容项, p 的值可以依据缓存情况进行调整。该算法可以认为是 Always 缓存策略的一般化,当 $p=1$ 时,即退化为 Always 缓存策略^[3-5]。

(4) Betw 策略在请求内容返回时选择感兴趣请求路径上最重要的节点进行缓存,其他节点不再缓存。Betw 策略在很多不同的网络拓扑结构中,在节点命中率和获取内容的平均跳数方面都获得了较好的性能表现。但是,重要节点的缓存空间有限,节点越重要,到达的请求就越多,需要缓存的内容就更多,这时缓存替换策略便会频繁启用^[6]。在重要缓存节点缓存满了的情况下,缓存替换策略将会把之前缓存的内容剔除掉,即便是新缓存的具有很高流行度的内容也会被快速替换掉,致使后续请求无法充分利用前期缓存的内容。而且重要节点并不是最靠近用户的节点,这样会导致最流行的内容无法到达最靠近用户的节点,使获取内容的平均跳数增大,即用户获取内容的时延增大^[7]。

基于以上方案存在的各种问题,文中提出在内容流行度进行排名的基础上考虑节点的中心度^[8],将流行度量化为请求频率,比如对内容 a 的流行度量化为请求频率 $q(a)$,对系统中命名的内容项都基于全局流行度进行排名。节点的中心度反映节点在网络中的重要性^[9],中心度等于路由器节点的度,即与该路由器相关联的链路的条数。该缓存机制在请求内容沿原路径返回时,将内容缓存在具有最大节点中心度的节点上,缓存满了后在该节点内生成内容流行度排名表,然后将新到达内容的流行度分别与节点内最大最小流行度进行比较,然后决定是否在该节点缓存新来的内容。

1 基于内容流行度和节点中心度的缓存机制

1.1 设计思想

该方案主要是针对 Betw 方案做出的改进。将节点中心度与内容流行度相结合,在节点中心度最高的节点放置缓存内容,当缓存满了之后,对节点内的内容进行流行度排名,在重要节点内生成一张流行度排名列表 popularity precedence table (PPT),新到达的内容的流行度分别与具有最大、最小流行度的内容进行比较,将大于最大流行度的内容放在此重要节点的下一级节点,将小于最小流行度的内容放在此重要节点的

上一级节点,将位于最大最小流行度之间的内容放在此节点内,剔除掉最小流行度的内容。这样一来,将内容实现了分布式的缓存,减缓了重要节点的缓存替换率和负荷,又能让最流行的内容逐渐靠近用户,减少内容冗余。

在图 1 所示的拓扑图中, $t=0$ 时刻,所有的缓存节点均为空。当用户 A 向内容中心 (SEVER) 请求内容 a 时, SEVER 向用户 A 返回内容 a 所经过的路径为 $V_1 \rightarrow V_2 \rightarrow V_4 \rightarrow V_5$, 由于在该路径上节点 V_2 的中心度最大,将内容 a 缓存在 V_2 , 随后当 A、B、C、D 中某几个用户请求相同内容时,即可在节点 V_2 命中。但是网络内部内容繁多, V_2 节点缓存容量有限。于是当 V_2 节点缓存满了之后,需要对 V_2 节点内的内容的流行度等级进行排名。当后续一段时间内新内容到达时,将新内容的流行度等分别与节点内最大最小流行度进行比较: 若大于最大流行度内容,则将新内容缓存在 V_2 的下一级节点即 V_3 、 V_4 节点,这样流行度很大的内容将更靠近用户; 若新到达的内容小于最小流行度内容,则将新到达的内容缓存在 V_1 节点; 若新到达内容的流行度处于最大流行度与最小流行度之间,则将最小流行度内容剔除,替换成新内容,再重新对节点内的内容流行度进行排名。这种缓存策略,不仅能将近来一段时间内的不再流行的内容替换掉、增大缓存中内容的存活时间,又能将近来最流行的内容缓存在靠近用户的网络边缘,减少内容冗余。

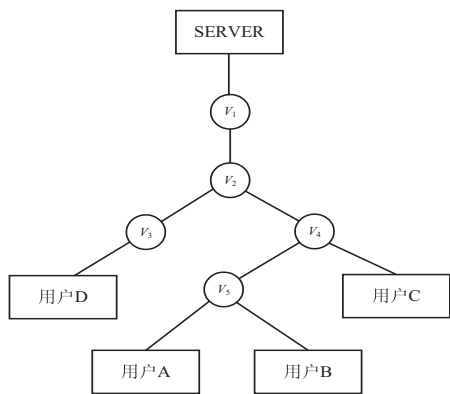


图 1 缓存决策实例

1.2 内容流行度和节点中心度的计算

1.2.1 内容流行度

内容流行度^[10]代表用户对内容的喜爱程度,通过在兴趣包和数据包中携带流行度值标签的形式实现内容流行度的记录。当前的内容流行度与历史内容的流行度和衰减系数有关,在过去一段统计时间内包含 M 个请求内容,在这一段周期内统计出各个请求内容的访问频次^[11]。内容流行度值是对内容 a 在请求周期内的请求次数估计值。式 1 为在第 n 个时间周期内的内容 a 的估计值。

$$P_a[n] = \alpha \times P_a[n - 1] + (1 - \alpha) \times F_a(n - 1)$$

(1)

其中, $P_a[n]$ 表示第 n 个周期内的内容 a 的流行度; $P_a[n - 1]$ 表示第 $n - 1$ 个周期内的内容 a 的流行度; α 表示衰减系数; $F_a(n - 1)$ 表示第 $n - 1$ 个周期内的内容 a 的访问频次。

1.2.2 节点中心度

将节点中心度量化为节点介数,即 Betw 介数缓存方法提出的节点介数计算方法,节点介数表征节点的重要程度,也就是与该节点相关联的链路条数^[12]。

在 CCN 网络中,有多个内容分发路径要经过同一个路径节点,那么这个节点在网络中的重要度就比较高,即中心度高。文献[1]给出了节点介数的定义:

若 $G(V,E)$ ^[1] 是一个具有 n 个节点的无向图向量, n 个节点为 $V = \{v_1, v_2, \dots, v_n\}$, 用 $C_{B-Sp}(v)$ 表示节点介数,表达式为:

$$C_{B-Sp}(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

(2)

其中, σ_{st} 表示两个顶点 s 与 t 之间的最短路径数; $\sigma_{st}(v)$ 表示经过顶点 v 的最短路径数。

在 Betw 算法中,使用的路由转发算法是最短路径算法,所以文中只统计节点间最短路径的数目。

对于一些移动的网络、自组织网络等,这些网络拓扑有一定不确定性,在这些网络中很难获得节点的信息,因此计算节点介数就很难实现^[13]。文献[2]提出一种方法:即每个节点基于它的自我中心网络而非整个网络来计算其近似介数,计算方法如下所示。

设 A 是自我中心网络 G 的 $N \times N$ 对称邻接矩阵:

$$A(i,j) = \begin{cases} 1, & \text{节点 } i \text{ 与 } j \text{ 之间有连接} \\ 0, & \text{其他} \end{cases}$$

(3)

自我中心网络节点的介数由矩阵 $A^2 [1 - A]_{i,j}$ 来确定, 1 是一个全 1 矩阵, $A^2 [1 - A]_{i,j}$ 中所有元素倒数之和即为该自我网络中心节点的中心度^[14]。

1.3 内容流行度排名表(PPT)

如表 1 所示,当重要节点的缓存满了以后,对节点内的内容流行度进行排名,具有最大流行度的内容排在最上面,新到达内容的流行度与表内的最大最小流行度进行比较,之后做出相应的判断。

表 1 内容流行度排名表

名字	第 $n - 1$ 个周期流行度	第 n 个周期流行度
内容 a	$P_a(n - 1)$	$P_a(n)$
内容 b	$P_b(n - 1)$	$P_b(n)$
内容 c	$P_c(n - 1)$	$P_c(n)$
内容 d	$P_d(n - 1)$	$P_d(n)$
内容 e	万方数据 $P_e(n - 1)$	$P_e(n)$

1.4 算法描述

以下内容对提出的 corporate 缓存策略作伪代码解释。

Pseudocode I 兴趣包到达 CCN 节点时算法:

Initialize

$(P_a(1) = P_a(2) = \dots = P_a(n) = P_a(0) = 0, F_a(0) = F_a(1) = F_a(2) = \dots = 0)$

For each (V_k from i to j)

Calculate $P_a[n] = \alpha \times P_a[n - 1] + (1 - \alpha) \times F_a(n - 1)$

The interest packet carry the popularity label $P_a[n]$

If data in cache | entry in PIT

The interest deliver the popularity label to data packet

Then send(data)

Else

Forward request to the next hop towards j

Pseudocode II 数据包到达节点时执行的缓存策略:

Calculate Betw(v_1, v_2, \dots, v_n)

$V_k = \max \{ \text{Betw}(v_1, v_2, \dots, v_n) \}$

For each (V_k from j to i)

If V_k not full

Cache data

Else

Ranking the content popularity $P[n]$ in V_k

If($P[n]_{\text{new}}$ (新到达内容流行度) $> P_{\text{max}}$)

Cache V_k 下(V_k 的下一级节点即靠近用户端)

If($P[n]_{\text{new}}$ (新到达内容流行度) $> P_{\text{min}}$)

Delete P_{min} content Cache $P[n]_{\text{new}}$ content in V_k

Else if($P_{\text{max}} > P[n]_{\text{new}}$ (新到达内容流行度) $> P_{\text{min}}$)

Remove P_{min} content to V_k

Cache $P[n]_{\text{new}}$ content in V_k

2 仿真分析

文中将所提方案 (corporate) 缓存策略与 Always、LCD、Betw 缓存放置策略进行比较,分别结合 LRU 缓存替换策略,通过 ndnSIM 仿真器,实现 CCN 仿真模型,得到仿真数据,然后将数据导入 Matlab 软件中进行处理,得到仿真结果图,最后对缓存结果进行评估。

图 2 和图 3 是在不同 CS 缓存容量下 (CS 单位为块 chunk) 四种缓存策略的性能图。图 2 是网络源端命中率,内容源端命中率越低,表示中间节点和边缘节点发挥作用越大,缓存性能越好。可以看出,四种缓存策略在 CS 逐渐变大的情况下,都会出现源端命中率逐渐递减的结果,与其他三种策略相比,corporate 策略源端命中率更低,性能更好。图 3 是获取请求内容所需要经过的平均跳数,可见 corporate 策略性能最好,在 CS 比较小时由于该策略较其他策略的优越性,会出现快速递减趋势,随后递减的趋势趋于稳定。

图 4 和图 5 是在不同的 Zipf 分布参数^[8]下比较四种缓存策略的性能。可见, corporate 策略较其他三种缓存策略,无论是在缓存内容命中率还是在获取内容需要经过的平均跳数上都有非常明显的性能提升效果,验证了所提方案的优越性。

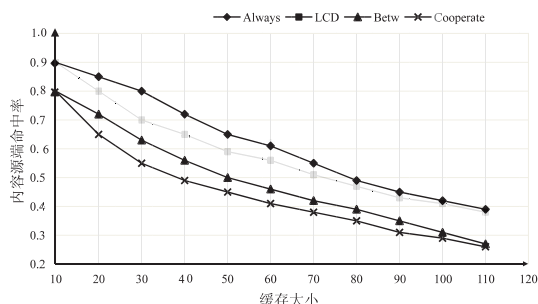


图 2 网络源端命中率

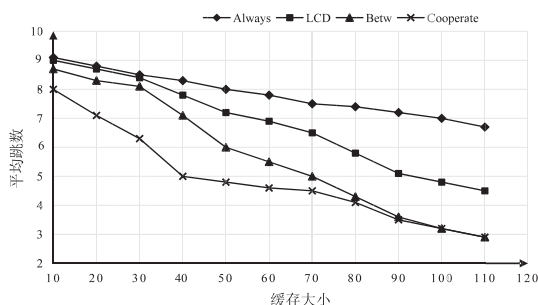


图 3 网络拓扑中获取内容平均跳数

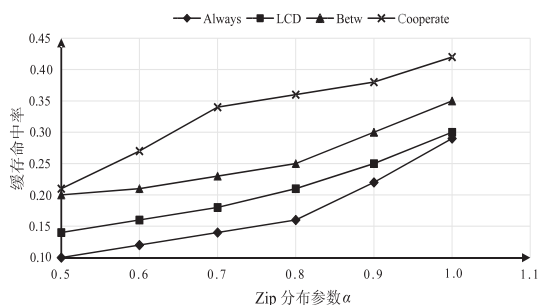


图 4 网络中缓存内容的命中率

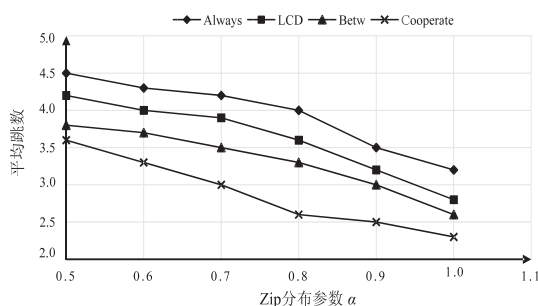


图 5 网络拓扑中获取内容的平均跳数

3 结束语

为了解决之前在 CCN 网络中已存在的各种缓存策略的问题,将缓存满了的中心节点中的内容流行度进行排名比较,然后将新来内容的流行度分别与最大、最小流行度进行比较,最后做出相应判决,将内容缓存在最合适的节点上。该方案既解决了 Betw 缓存策略重

要节点缓存替换频繁的问题,又使流行内容更加靠近用户;不但可以减少内容的冗余,各处节点也能充分利用,内容流行度越高越靠近用户,从而大大提高了缓存性能。下一步将深入研究内容流行度的排名和预测^[13],从而制定更为合理的缓存策略,以更好地提升缓存性能。

参考文献:

- [1] 崔现东,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.
- [2] 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.
- [3] 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.
- [4] TARNOI S, SUKSOMBOON K, KUMWILAISAK W, et al. Performance of probabilistic caching and cache replacement policies for content-centric networks [C]//39th annual IEEE conference on local computer network. [s. l.]: IEEE, 2014.
- [5] 林闯,贾子骁,孟坤.自适应的未来网络体系架构[J].计算机学报,2012,35(6):1077-1093.
- [6] 王国卿,黄韬,刘江,等.一种基于逗留时间的新型内容中心网络缓存策略[J].计算机学报,2015,38(3):472-482.
- [7] 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.
- [8] 曾宇晶,靳明双,罗洪斌.基于内容分块流行度分级的信息中心网络缓存策略[J].电子学报,2016,44(2):358-364.
- [9] 徐昌彪,王华.CCN中基于内容流行度和节点重要度的缓存设计[J].电子应用,2017,43(3):100-103.
- [10] 曲桦,王伟萍,赵季红.内容中心网络中一种改进型缓存机制[J].计算机工程,2015,41(3):41-46.
- [11] CUI Yufei, ZHAO Min, WU Muqing. A centralized control caching strategy based on popularity and betweenness centrality in CCN [C]//IEEE conference on international symposium on wireless communication systems. Poznan, Poland: IEEE, 2016: 286-291.
- [12] CHAI W K, HE Diliang, IOANNIS P, et al. Cache "less for more" in information-centric networks (extended version) [J]. Computer Communications, 2013, 36(7): 758-770.
- [13] 董美姣.基于内容流行度预测的内容中心网络缓存技术研究[D].北京:北京邮电大学,2015.
- [14] GUAN Jianfeng, HE Yunhang, WEI Quan, et al. A classification-based wisdom caching scheme for content centric networking [C]//IEEE conference on computer communications workshops. San Francisco, CA, USA: IEEE, 2016.