

边缘网络中去中心化的内容扩散系统设计

薛寒星 尤佳莉 王劲林

(中国科学院声学研究所国家网络新媒体工程技术研究中心 北京 100190)

(中国科学院大学 北京 100190)

摘要 网络边缘的终端设备构建的边缘覆盖网络是距离用户最近的网络。作为骨干网络的补充,边缘网络中的终端设备有高度的自治性,并且拥有大量空闲的资源。如何利用边缘网络的特性来缓解骨干网络在用户请求高峰期的负载压力具有重要的研究意义。为边缘网络设计一种去中心化的内容扩散系统,通过边缘设备之间的通信将热门内容的副本自适应地扩散到每个边缘设备的邻域内。通过仿真实验可知,去中心化的扩散算法可以实现快速的内容扩散。同时,边缘设备在扩散过程中会均衡邻居设备间的存储负载。内容扩散系统极大地改善了边缘网络中的用户访问效率。针对已完成扩散的内容的请求,用户以约98.1%的概率直接在边缘网络中的一跳邻域内获取高效的内容服务。

关键词 边缘网络 去中心化 内容扩散 覆盖率

中图分类号 TP3 文献标志码 A DOI:10.3969/j.issn.1000-386x.2020.03.022

DESIGN OF DECENTRALIZED CONTENT DIFFUSION SYSTEM IN EDGE NETWORK

Xue Hanxing You Jiali Wang Jinlin

(National Network New Media Engineering Research Center, Institute of Acoustics, Chinese Academy of Science, Beijing 100190, China)

(University of Chinese Academy of Science, Beijing 100190, China)

Abstract The edge coverage network is the closest network to the user, which is constructed by the smart terminals at the edge of the network. As a supplement to the backbone network, terminals devices in the edge network have a high degree of autonomy and a large amount of idle resources. During the peak period of users requests, how to use the characteristics of the edge network to alleviate the load pressure of the backbone network is of great significance. This paper designs a decentralized content diffusion system for the edge network. The replications of hot content could be adaptively diffused to the neighborhood of each edge device by the communication between the edge devices. Through simulation experiments, the decentralized algorithm could achieve fast content diffusion. Meanwhile, the edge terminal balanced the storage load between neighbor devices in the process of diffusion. The content diffusion system greatly improved the efficiency of users requests in the edge network. Aiming at the requests for content that has been diffused, the user directly obtains efficient content services in the one hip neighborhood in the edge network with a probability of about 98.1%.

Keywords Edge network Decentralized Content diffusion Coverage rate

0 引言

内容的分发和获取作为目前互联网中的主要应

用,占据了网络中的绝大部分流量。负责将内容传输到请求用户或网络边缘的内容扩散系统对内容服务的质量有很大的影响。同时,大量靠近用户的终端设备构建的边缘覆盖网络作为最靠近服务请求发起端的网

络,可以向用户提供低时延的快速服务响应,典型的场景包括智能路由器^[1]。不同于传统的网络层的路由器,智能路由器是距离用户最近的家用智能设备。它拥有更强的处理能力和存储能力,可以向本地用户和边缘网络中的邻居智能路由器提供低时延的内容服务。因此,在边缘网络中扩散部分热门请求内容的副本,使用户可以直接从本地或邻域边缘设备处获取内容服务^[2]。系统服务质量得到保障的同时骨干网络的负载压力也得到了缓解。因此,边缘网络中的内容扩散有重要的研究意义。因为边缘设备具有高度的自治性,边缘网络中的内容扩散需要一种去中心化的扩散方式。例如物联网、车联网、无线传感器网络等一些终端设备组成的边缘覆盖网络中的内容扩散均采用了去中心化的方式。

基于已有的对用户请求的分析可知,不同内容的请求数量服从长尾分布,即大部分的请求集中在流行的内容上^[3]。因此,针对热门的内容,本文为边缘网络设计了一种去中心化的内容扩散系统。通过模拟谣言传播来实现去中心化的扩散,即边缘设备向边缘网络中的邻居节点扩散内容的副本或消息。收到消息或副本的边缘设备在自身的邻域中进行进一步的扩散。直至内容的副本和消息扩散至边缘网络中的所有设备。为了合理地控制内容副本的冗余,系统根据内容的流行度定义了相应的邻域覆盖率要求。边缘设备根据邻居数量和内容的覆盖率要求在邻居节点中自适应地扩散一定数量的内容副本。仿真实验结果表明,本文设计的去中心化的扩散算法可以在边缘网络中实现快速的内容扩散。同时,扩散完成后的边缘网络可以极大地改善用户的访问效率。用户可以以极高的概率在一跳邻域内直接获取热门内容的服务。

1 相关工作

本文研究的去中心化的内容扩散系统的实质是将热门内容的副本自适应地预扩散到边缘设备构成的边缘网络中。因此,相关的研究包括内容的预取、复制和扩散策略。

现有的内容预取策略针对不同的应用场景进行了研究。在移动网络中, Ma 等^[4]提出通过预测用户的兴趣、移动性和吞吐量来选择合适的预取内容和缓存节点。在社交网络中, Li 等^[5]提出可以根据用户的兴趣来预取内容的部分前缀数据块,进而缓解用户的启动时延。在 CDN 和 P2P 混合的视频服务系统中, Zhang 等^[6]提出预测内容的稀缺程度和节点的能力,提前在能力充足的节点上部署未来稀缺的内容副本。上述的

预取策略的研究都是从个体用户的兴趣来选择更合适的内容进行预部署,并未考虑全局内容的流行情况和大量内容的扩散。不同于已有的内容预取策略的研究,本文从全局网络中的热门内容出发,通过去中心化的扩散实现热门内容的在整个边缘覆盖网中的预部署。

分布式服务系统经常会在覆盖网中部署部分内容的副本来改善服务的质量,例如部署热门内容的副本减少内容检索的路径长度和流量,部署稀有内容的副本来提高检索的成功率^[7]。无结构覆盖网络中内容的副本数量常见策略为全部相同或者正比于内容的流行度^[8]。DHT 覆盖网络中,节点可以通过哈希函数被划分到不同的区域,进而内容的副本被放置到不同的区域来提高检索的成功率^[9]。现有的内容复制策略的问题在于需要手动配置副本数量等参数,边缘设备不能根据邻域的网络和兴趣情况进行自适应的调节。本文提出了覆盖率的定义来反映内容副本在节点邻域的覆盖情况。边缘设备可以根据覆盖率需求和邻域情况自适应调整副本的数量。

内容扩散系统中的内容传输的实现方式包括“推”(Push)和“拉”(Pull)两种方式。Push 策略适用于热门内容的主动下发和推送, Pull 策略适用于用户被动拉取稀少的内容副本。本文设计的扩散系统通过 Push 的方式实现副本消息的扩散和缓存节点的选择,通过 Pull 策略实现进一步的内容传输。在内容传输过程的具体细节部分,现有扩散的研究是基于数据包或数据流的传输模型^[10],研究如何在更短的时间内将副本从单个种子节点扩散到指定的接受副本的节点集合^[11-12]。快速扩散的策略包括在扩散过程中的内容源节点优先选择带宽更高的接收节点或最稀缺的内容片段来传输,尽可能地减少完成扩散所需要的时间。在本文的分布式系统中,节点很难时刻获取到全局的内容稀缺程度或已缓存副本的节点信息。通过设计一种类似谣言传播的机制使得节点尽可能获取更多的局部信息来做出较好的传输决策。

2 系统设计

2.1 系统结构

整个内容服务系统主要由两部分组成:管理系统和边缘设备构成的边缘网络。图 1 所示为系统的功能结构。管理系统主要负责节点管理和内容管理。它的具体工作包括:通过推荐邻居列表来辅助设备加入边缘网络,采集并分析用户请求信息,根据分析结果选择

合适的扩散内容并制定相应的扩散目标,随机选择一个边缘设备作为种子节点来部署需要扩散的内容。边缘终端设备是整个内容服务系统的关键。边缘设备之间通过邻居维护构建了边缘覆盖网。在边缘网络中,边缘设备通过信息交互实现邻居间的内容扩散和内容服务。种子节点在收到内容副本后会主动在邻域内进行内容的自扩散。

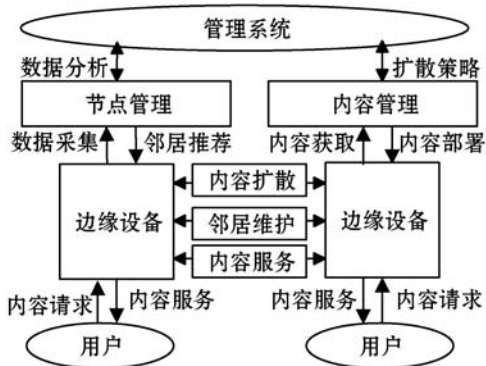


图1 系统功能结构

2.2 热门内容的覆盖率设计

本文的内容服务系统的设计目标是希望边缘设备可以直接在边缘网络中的邻域设备处获取到热门内容的服务。因此,邻域中内容副本的数量对系统的服务表现有重要的影响。本文首先定义了内容副本在邻域的覆盖率:

$$C_{i,j}(t) = \frac{x_{i,j}(t) + \sum_{y \in Nei_i} x_{y,j}(t)}{|Nei_i| + 1} \quad (1)$$

式中: $C_{i,j}(t)$ 表示节点*i*关于内容*j*在时刻*t*的覆盖率; Nei_i 表示节点*i*的邻居节点的集合; $x_{i,j}(t)$ 表示节点*i*在时刻*t*是否缓存了内容*j*的副本,若已缓存, $x_{i,j}(t) = 1$,否则 $x_{i,j}(t) = 0$ 。

覆盖率反映了在*t*时刻,内容副本在边缘设备的邻域的覆盖情况。覆盖率越大表示邻域中缓存该内容副本的边缘设备数量的比例越高。本文通过设定合理的覆盖率需求来指导边缘设备在邻域中的内容扩散。因为不同内容有不同的流行度,并且流行度会随着时间不断变化,管理系统需要周期性的分析内容的请求情况来为不同的内容设计合理的分发目标,即需求覆盖率。假设局部的边缘设备对不同内容的请求分布和全局的请求分布相近。系统的管理服务器将根据系统中一段时间内的内容访问数量来为每个内容计算相应的覆盖率需求:

$$C_{d(j)} = \frac{NR_j}{\sum_{m \in F} NR_m} \quad (2)$$

式中: $C_{d(j)}$ 表示内容*j*的需求覆盖率; NR_j 表示内容*j*的请求数量; F 表示系统中所有的内容集合。可以合理地假设内容在邻域的覆盖率大于需求覆盖率时,邻域中该内容的副本数量足以向邻域节点提供一跳的内容服务。

3 去中心化的扩散算法设计

3.1 设计思路

内容扩散系统的扩散目标是将热门内容的副本按照一定的覆盖率要求扩散至每个边缘设备的邻域内。因为边缘覆盖网络中的每个边缘设备有很高的自治性,本文设计了一种类似于谣言传播的去中心化的内容扩散算法来实现扩散目标。在扩散开始时,管理系统会随机选择一个边缘设备作为种子节点来部署需要扩散的内容。种子节点完成内容副本的缓存后会根据扩散算法开始内容的去中心化扩散。

扩散算法的具体思路是每个边缘设备向维护的邻域节点扩散内容的消息或副本。副本数量由节点的邻居数量和内容的覆盖率要求来决定。收到相应消息和副本的节点再使用同样的策略在自己的邻域中进行扩散。通过这种类似谣言传播的方式,热门内容的副本和消息会从种子节点逐步扩散至边缘网络中的所有边缘设备。值得注意的是,为了避免形成广播风暴,如果内容的覆盖率要求等相关信息没有改变,所有的边缘设备只会处理和转发一次该内容的消息。同时,为了充分利用所有边缘设备的资源,在扩散过程中边缘设备会优先选取空闲存储空间较大的邻居来缓存副本。如果所有邻居都没有空闲的存储空间,边缘设备会随机选取邻居来缓存副本。被选择的邻居执行相应的缓存替换策略来选择已缓存的副本进行替换。因为本系统中的覆盖率要求反映了内容在当前时刻的流行度,边缘设备可以选择拥有最小覆盖率要求的已缓存的副本进行替换。

3.2 具体实现

为了实现边缘设备间的去中心化内容扩散,系统将所有的消息分为四种类型:扩散消息(Diffusion Message)、缓存消息(Store Message)、请求消息(Request Message)、内容数据消息(Content Data Message)。边缘设备会根据收到的具体消息类型自适应地执行对应的扩散操作。以下为系统部分基本数据结构,包括边缘节点、扩散内容和不同类型消息的数据结构。其中,缓存消息和扩散消息的数据结构相同。

```

Class Node {
    ...
    Node neighborList[];
    FileState fileList[];
    Double storageSpace;
    ...
}
Class RequestMessage {
    Integer requestFileID;
}
Class FileState {
    ...
    Integer fileID;
    Double fileSize;
    Double requireCoverageRate;
    Integer processState;
    Boolean cacheState;
    NodeAddress contentSourceList[];
    ...
}
Class DiffusionMessage {
    Integer fileID;
    Double fileSize;
    Double requireCoverageRate;
    NodeAddress contentSourceList[];
}
Class ContentDataMessage {
    Integer fileID;
    Integer blockID;
}

```

简化的去中心化的内容扩散算法如算法 1 所示。其中,四种类型的消息具有不同的功能。扩散消息是整个扩散过程中的核心消息,包含了内容的 ID、大小、覆盖率需求和已缓存该内容副本的节点地址等基本信息。边缘设备收到扩散消息后会执行具体的扩散过程,包括选择邻居发送缓存消息直至邻域中该内容的覆盖率满足要求。如果没有合适的邻居可以缓存副本,边缘设备也会停止缓存消息的发送。边缘节点在邻域中的扩散过程结束后,它会向其余邻居继续广播该内容的扩散消息。缓存消息负责通知节点缓存内容的副本。收到缓存消息的边缘设备会明确是否存在足够的存储空间来缓存内容副本。若存在,设备会向已缓存的边缘设备发送请求信息。若不存在,设备执行缓存替换策略来确定是否可以缓存副本。边缘设备发送的扩散消息和缓存消息都必须携带本地已知的多个确认缓存该内容副本的边缘设备地址,使得需要缓存副本的设备可以获取多个内容源节点的地址进行内容请求。

算法 1 去中心化的内容扩散算法

Input: 节点 i 收到关于文件 j 的数据包

[主程序]:

Switch (数据包类型):

Case DiffusionMessage: DiffusionProcess(j)

Case StoreMessage: StoreProcess(j)

Case RequestMessage: SendContentData(j)

Case ContentDataMessage:

If completeCache(j): DiffusionProcess(j);

end Switch

DiffusionProcess (fileid j):

if AlreadyDiffuse(j) **and** $C_{d(j)}$ unchanged: **return**;

$C_{i,j}(t) = \text{ComputeCovRate}(j)$;

while $C_{i,j}(t) < C_{d(j)}$ **or** CandidateNeiList! = null **do**:

$x = \text{SelectNei}(\text{CandidateNeiList}, j)$; SendStoreMsg(x, j);

CandidateNeiList.RemoveNode(x);

if AgreeCache(x, j): $C_{i,j}(t) = \text{ComputeCovRate}(j)$;

end while

节点 i 向 Candidate NeiList 中的其余邻居节点广播 DiffusionMessage 消息;

end DiffusionProcess

StoreProcess (fileid j):

if FreeStorage(i) < FileSize(j): ReplaceCache(i, j);

SendAgreeCache(j); SendRequest(j);

end StoreProcess

请求消息和内容数据消息用来实现缓存副本的数据传输过程。确认缓存内容副本的设备会向已知确认缓存该内容副本的多个边缘设备发送请求消息。如果被请求的边缘设备已经完成了内容数据的接收并且拥有足够空闲的上传带宽,该边缘设备会与请求的边缘设备进行通信确认内容数据消息的发送。请求的边缘设备会从最快回复的内容源设备处接受内容数据。同时,在边缘设备完成某个内容的副本缓存后也会自动执行内容扩散过程。需要注意的是,因为每个边缘设备会缓存多个内容的副本,可能会同时收到来自不同设备关于不同内容的多个请求。此时边缘设备会采用轮询(Polling)策略选择传输的片段和接收的设备,即边缘设备会确认已经完成缓存的副本,从请求队列中针对每个内容选取一个请求设备并依次处理,周期性地重复上述的处理流程直至所有的请求都处理完成。

3.3 去中心化的内容扩散实例

图 2 展示了一个去中心化的内容扩散算法的简单实例,8 个边缘设备构建了一个边缘网络。假设文件 1 (F_1)的覆盖率需求为 0.3, N_1 作为 F_1 的种子节点开始进行相应的内容扩散。在第一个时刻, N_1 计算得到当前 F_1 的邻域覆盖率为 0.25,并不满足 0.3 的要求。 N_1 选择 N_2 来缓存 F_1 的副本,即发送缓存消息。通过请求消息和内容数据的传输, N_2 完成了副本的缓存。在第二个时刻, N_1 对 F_1 的邻域覆盖率已满足要求,继续向 N_3 和 N_4 发送扩散消息。因为 N_2 对 F_1 的覆盖

率(0.4)也满足要求,所以 N2 向 N5、N6 和 N7 发送扩散消息。因为 N6 的覆盖率不满足要求,N8 被选择来缓存 F1 的副本。N8 根据收到的缓存消息中携带的内容源的地址(N2 的地址),向 N2 发送请求消息。N2 发送内容数据完成传输。至此,所有节点关于 F1 的邻域覆盖率均满足了它的覆盖率需求(0.3)。整个去中心化的内容扩散过程结束。

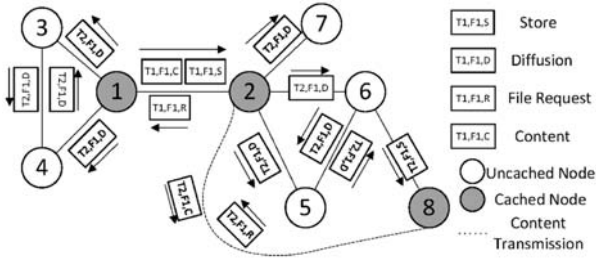


图 2 去中心化内容扩散的实例

4 仿真实验

4.1 实验设计

本文使用网络仿真器 Peersim 实现了整个内容分发系统在大规模网络下的仿真。假设每个边缘设备是同构的,即拥有相同的存储空间和带宽。采用两种不同的覆盖网络作为边缘网络来验证内容扩散系统的性能,包括 BA 网络和随机网络。因为边缘设备的能力有限,实验中将随机网络中的每个节点的邻居数量都粗略地限定在一定范围内来减轻边缘设备的负载。表 1 列出了具体的实验参数和对应的默认值。BA 网络的 K 表示新边缘设备加入网络中后连接的邻居数量。同时,根据上文中的文件的覆盖率需求的计算方式,假设网络中共有 10 000 个文件,可以根据已有的用户对文件的请求分布计算相应的覆盖率需求。图 3 为 100 个文件对应的覆盖率需求。

表 1 扩散参数默认取值

参数描述	默认值
网络规模	4 000
网络拓扑	随机网络(邻居数量[30,40]) BA 网络($K=30$)
文件数量	100
文件大小	200 MB
用户请求分布	Zipf 分布($\alpha=1.0$)
存储空间	8 GB
上行带宽	2 Mbit/s
下行带宽	20 Mbit/s

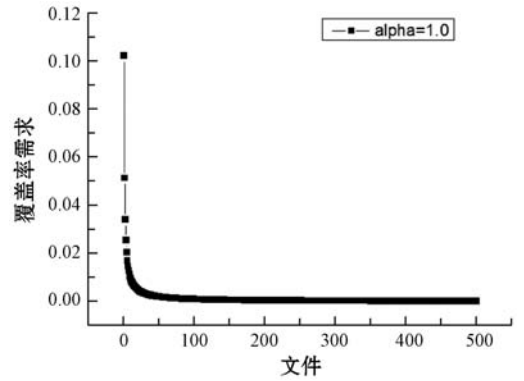


图 3 热门文件的覆盖率需求

4.2 仿真结果

因为本文设计的是一个全新的去中心化的内容扩散系统,没有较为相近的系统进行比较。故在实验评估部分,本文从扩散速率、副本数量和用户访问效率等三个方面对系统在不同网络拓扑下的性能进行了评估。

4.2.1 内容扩散速率分析

假设边缘设备完成一次内容副本传输所需的时间为一个时间单元。管理系统在每个时间单元向边缘覆盖网络注入一个新的文件,文件的覆盖率要求为图 4 中的覆盖率。图 4 展示了 1 个文件和 100 个文件在扩散的过程中,网络中的副本总量随着时间单元增加的变化情况。

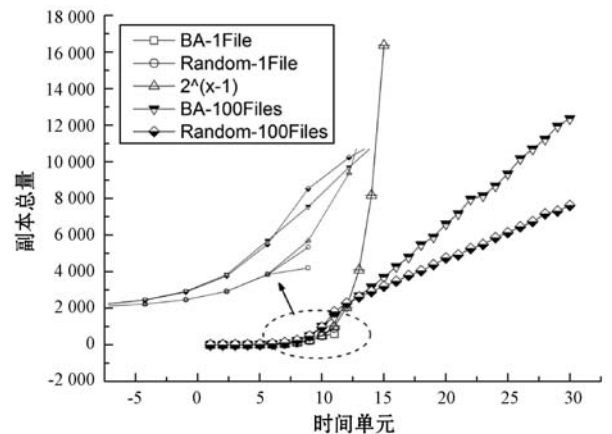


图 4 网络中的副本总量在不同时间单元的变化趋势

对图 4 中几条曲线重合度比较高的地方进行了局部放大。从放大的区域可知,在网络中只有 1 个文件在扩散时,文件的副本数量变化趋势与以 2 为底的幂指数曲线基本重合,实现了较为理想的快速扩散。在第 11 个时间单元,单个文件就完成了在网络中的内容扩散,即所有边缘设备的邻域都满足了该文件的覆盖率要求。当网络中有 100 个文件在扩散时,在第 11 个时间单元前,副本的总量都大于以 2 为底的幂指数,这是因为网络中每个时刻都会有一个新的文件加入网络中进行快速的扩散,即网络中同时存在多个文件在扩

散。在第12个时间单元后,网络中的副本数量从指数增长趋势变为近似线性的增长趋势。这是因为第12个时间单元后每个时间单元都会有文件已经完成扩散,即该文件的副本数量不会再增加。BA网络和随机网络的增长速率不同是因为同一文件在两种网络拓扑中完成扩散的副本数量不同。

4.2.2 扩散后的副本分析

在扩散完成后,不同内容的副本数量反映了副本的冗余情况。图5展示了扩散完成后不同文件在整个边缘网络中的副本数量。可以看出,最热门的5个内容在BA网络中的副本数量要明显小于它们在随机网络中的副本数量,这是因为BA网络中的边缘节点的邻居数量服从幂律分布,存在部分邻居数量很大的边缘设备。将内容的副本部署在这些边缘设备上将会增加更多边缘设备对该内容的覆盖率,相应的满足覆盖率需求的副本数量也会减少。但是其余大部分文件在BA网络中的副本数量却大于随机网络。这是因为目前的系统为了尽可能利用边缘设备资源采用了基于空闲存储资源的缓存节点选择策略,边缘设备会优先选择拥有更多空闲存储空间邻居来存储内容的副本。因此,在扩散了部分内容后,BA网络中的后续扩散可能会经常选择邻居数量较少但存储资源较多的节点来缓存副本。相应内容的副本数量则会增加很多。在本文的随机网络中,边缘设备的邻居数量被限制在一定范围内,边缘设备的邻居数量差异不大,所以不会存在与BA网络中类似的大幅波动。在随机网络中,不同内容的副本数量与内容的覆盖率要求的有相同的趋势。

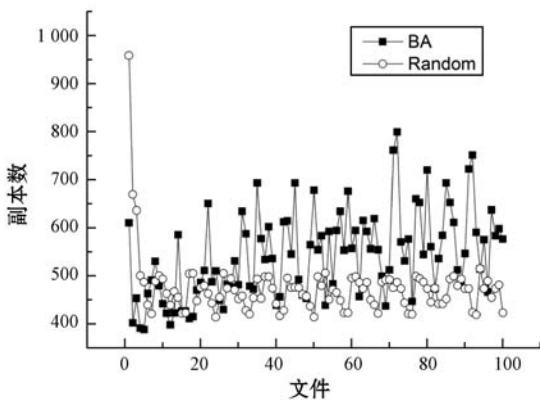


图5 扩散完成后不同文件的副本总量

表2统计了分发完成后所有文件的副本总量和边缘设备存储副本数量的标准差。副本总量反映了分发的带宽和存储资源的消耗,副本数量越大,边缘设备的带宽和存储资源的消耗越多。每个设备存储副本数量的标准差反映了边缘设备之间的存储负载差异,数值越小表示设备间的存储负载越均衡。根据表2中统计

的数据可知,相对于BA网络,内容扩散系统在随机网络中有更好的表现,使用更少的资源就可以满足所有设备对扩散内容的覆盖率要求。由于缓存节点的选择策略,BA网络中的内容扩散产生了较多的副本冗余。真实情况中,拥有较多邻居的边缘设备往往拥有更强的能力(存储,带宽资源)。因此,边缘设备异构的BA网络中内容扩散系统也会有很好的表现。根据存储副本数量的标准差可知,扩散系统在两种网络下都可以很好地均衡边缘设备的存储负载。

表2 文件的副本数量分析

指标	理论最优或最大值	BA网络	随机网络
所有内容的副本总量	160 000	54 443	47 634
边缘设备存储副本数量的标准差	0	0.237	0.084

4.2.3 用户访问效率分析

本文设计的内容扩散系统将热门内容的副本扩散至每个边缘设备的邻域内,希望用户可以在一跳邻居内直接获取热门内容的服务。在边缘网络中,用户首先在本地设备和边缘网络中的邻居设备检索请求的内容。在邻域内有请求内容的副本且该缓存设备有空闲上传带宽时,用户可以直接从边缘网络中获取低时延的内容服务。否则,用户向CDN服务器请求内容服务。假设每个边缘设备的上传带宽可以同时向一个邻居设备提供服务,边缘设备在收到多个邻居的请求时,将采用先到先服务(FCFS)的策略选择请求来提供服务。本文将边缘请求成功率定义为用户在边缘网络中成功获取服务的请求数量与总的用户请求数量的比值。边缘请求成功率的数值越高表示更多的用户请求可以直接在邻域内获取高效的服务,用户访问效率越高。

本节在扩散完成后模拟用户高峰期的请求情况,即所有用户同时发起内容的请求。通过边缘请求成功率来评估扩散完成后的用户访问效率。请求的内容范围为网络中的10 000个内容,不同内容的请求数量服从Zipf分布。实验在两种拓扑下分别模拟了10次高峰期的用户请求。表3统计了10次实验中的平均边缘请求成功率。表中的“理论最优”为边缘请求成功率的上限,即所有的用户请求中关于扩散完成的100个热门内容的请求数量所占的比例。

表3 扩散完成后的平均边缘请求成功率

指标	理论最优	BA网络	随机网络
平均边缘请求成功率	53.0%	52.0%	52.2%

由表3中数据可知,两种网络拓扑在内容扩散完

成后的边缘请求成功率约为 52.0%,非常接近理论最优 53.0%。结果证明基于覆盖率需求的内容扩散完成后,针对已经扩散过的热门内容的请求,用户在边缘网络中以约 98.1% 的概率直接在邻域内获取内容服务。用户的访问在边缘网络中得到了高效的服务。同时,约 52% 的内容流量从骨干网络卸载到边缘网络,骨干网络的负载压力也得到了极大的缓解。其中,边缘请求成功率略低于理论最优的数值。原因可能是极少部分局部区域对部分内容的请求频率略高于全局请求频率,导致依据全局请求情况设定的覆盖率需求不能完全满足局部的请求需求。

5 结 语

本文为边缘网络设计了一个去中心化的内容扩散系统。首先定义了一种邻域覆盖率来反映内容副本在邻域的覆盖情况。系统可以根据内容的请求数量设定合适的覆盖率需求,进而模拟谣言传播的方式将内容的副本在网络中进行扩散。实验证明,去中心化的系统可以实现快速的内容扩散。基于设备空闲存储资源的选择策略也均衡了边缘设备的存储负载。基于覆盖率的内容扩散完成后,用户在边缘网络可以拥有接近理论最优的边缘请求成功率。对于已完成扩散的内容的请求,用户有约 98.1% 的概率直接在邻域内获取高效的内容服务。目前的内容扩散系统中还存在管理系统来设定内容的覆盖率需求。在下一步,会进一步利用边缘节点的自治特性,取消管理系统,设计一个完全去中心化的内容扩散系统。边缘节点可以根据获取的局部信息自适应地调整内容的覆盖率需求,每个内容扩散完成后的副本数量也会尽可能正比于内容的流行度。

参 考 文 献

[1] Ma M, Wang Z, Su K, et al. Understanding content placement strategies in smarthrouter-based peer video CDN [C] // Proceedings of the 26th International Workshop on Network and Operating Systems Support for Digital Audio and Video, 2016:7.

[2] An D, Lin T. A proactive RAN caching scheme based on content popularity prediction [J]. Journal of Network New Media, 2016, 5(5):1-8.

[3] Choi J, Reaz A S, Mukherjee B. A survey of user behavior in VoD service and bandwidth-saving multicast streaming schemes [J]. IEEE Communications Surveys & Tutorials, 2012, 14(1):156-169.

[4] Ma G, Wang Z, Chen M, et al. APRank: Joint mobility and

preference-based mobile video prefetching [C] // 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017:7-12.

- [5] Li Z, Shen H, Wang H, et al. SocialTube: P2P-assisted video sharing in online social networks [J]. Proceedings—IEEE INFOCOM, 2012, 25(9):2886-2890.
- [6] Zhang Y X, Gao C L, Guo Y Z, et al. Proactive video push for optimizing bandwidth consumption in hybrid CDN-P2P VoD systems [C] // IEEE INFOCOM 2018—IEEE Conference on Computer Communications, 2018:9.
- [7] Gao G, Li R, Wen K, et al. Proactive replication for rare objects in unstructured peer-to-peer networks [J]. Journal of Network and Computer Applications, 2012, 35:85-96.
- [8] Spaho E, Barolli L, Xhafa F. Data replication strategies in P2P systems: a survey [C] // 2014 17th International Conference on Network-Based Information Systems (NBIS). IEEE, 2014:302-309.
- [9] Trifa Z, Khemakhem M. A novel replication technique to attenuate churn effects [J]. Peer-to-Peer Networking and Applications, 2016, 9(2):344-355.
- [10] Munding J, Weber R. Efficient file dissemination using peerto-peer technology [EB]. Private Communication, 2004.
- [11] Ku C F, Yang K H, Ho J M. An optimal scheduling for file dissemination under a full binary tree of trust relationship [C] // 2012 IEEE Global Communications Conference (GLOBECOM), 2012:751-757.
- [12] Zheng H, Yu H. Minimizing distribution time for one-to-many file distribution in P2P networks [C] // 2015 International Conference on Electrical and Information Technologies for Rail Transportation, 2016:597-607.

(上接第 100 页)

- [13] 余翔,王宏刚,段思睿. LTE-A 系统中基于 QoE 能效的无线资源分配算法 [J]. 计算机应用研究, 2019, 36(5):216-218,235.
- [14] 孟子潇. CoCache: 基于文件关联性的协同缓存策略 [J]. 计算机应用与软件, 2018, 35(7):52-56.
- [15] 王志君,梁利平等. 一种面向多媒体和通信应用的处理器指令集及架构实现 [J]. 湖南大学学报(自然科学版), 2014, 41(10):108-114.

(上接第 108 页)

- [16] Wahab O A, Otrok H, Mourad A. VANET QoS-OLSR: QoS-based clustering protocol for Vehicular Ad hoc Networks [J]. Computer Communications, 2013, 36(13):1422-1435.
- [17] Hornig R. An overview of the OMNeT++ simulation environment [C] // International Conference on Simulation Tools & Techniques for Communications. 2008:11-32.