

# P2P 网络中一种基于闲话的更新传播方法<sup>\*</sup>

周翠红<sup>a</sup>, 贾丽媛<sup>a</sup>, 左伟明<sup>a</sup>, 秦姣华<sup>b</sup>

(湖南城市学院 a. 计算机科学系; b. 数学与计算科学系, 湖南 益阳 413000)

**摘要:** 针对基于泛洪的更新传播方法存在开销高、可靠性和可伸缩性差等问题, 提出了一种基于闲话的更新传播方法, 其思想是每个副本节点将更新消息随机转发给一组邻居副本节点, 副本节点向其邻居副本节点请求更新副本。模拟实验结果表明, 与基于泛洪的更新传播方法相比, 基于闲话的更新传播方法在网络通信开销上减少了 38%, 在节点覆盖率和查询失效率上快速收敛, 具有良好的抗搅动能力。

**关键词:** 对等网络; 一致性维护; 更新传播; 闲话

**中图分类号:** TP393      **文献标志码:** A      **文章编号:** 1001-3695(2010)03-1110-04

doi:10.3969/j.issn.1001-3695.2010.03.084

## Gossip-based update propagation scheme in peer-to-peer networks

ZHOU Cui-hong<sup>a</sup>, JIA Li-yuan<sup>a</sup>, ZUO Wei-ming<sup>a</sup>, QIN Jiao-hua<sup>b</sup>

(a. Dept. of Computer Science, b. Dept. of Mathematics & Computational Science, Hunan City University, Yiyang Hunan 413000, China)

**Abstract:** The flooding-based update propagation scheme may induce high overhead of communication, and be not resilient to churn, and not scale well. This paper proposed a novel gossip-based push-pull update propagation scheme, where each replica peer pushed an update notification message to a set of randomly chosen neighbor replica peers, and each replica peer pulled the updated replica from its neighbor replica peers. The simulation results show that the gossip-based update propagation scheme can reduce the communication overhead by 38% in comparison of the flooding-based scheme, and can be rapid convergence in terms of the number of updated nodes and the query stale ratio can be rapid convergence, with a good resilience to churn.

**Key words:** peer-to-peer(P2P) network; consistency maintenance; update propagation; gossip

## 0 引言

P2P 网络是一种自组织的分布式系统, 其节点以分散方式提供和接收服务, 即每个节点既是服务提供者, 又是服务消费者<sup>[1]</sup>。P2P 网络已成为最重要的 Internet 应用之一。测量研究表明<sup>[2]</sup>, P2P 网络占用 43% 的网络带宽, 而 WWW 业务仅占用 14% 的网络带宽。P2P 网络可分为结构化和非结构化。在结构化 P2P 网络中, 网络拓扑结构和文件放置是严格控制和管理, 如 Chord<sup>[3]</sup>、CAN<sup>[4]</sup>、Pastry<sup>[5]</sup> 和 Tapestry<sup>[6]</sup>; 结构化 P2P 网络采用基于 DHT(distributed hash table) 的查找方法<sup>[7]</sup>。在非结构化 P2P 网络中, 网络拓扑结构和文件布置未严格控制和管理, 如 Gnutella<sup>[8]</sup> 和 KaZaA<sup>[9]</sup>; 非结构化 P2P 网络采用基于泛洪(flooding)或随机漫步(random walks)的查找方法<sup>[7]</sup>。由于结构化 P2P 网络存在网络拓扑构造和维护开销大等问题, 非结构化 P2P 采用简单的查找方法, 可适应实际动态网络, 已成为最成功的实际 P2P 网络。

为了提高 P2P 网络的可伸缩性和性能, 研究者通常采用复制和缓存方法, 将共享文件的副本或元数据存储或缓存在大量 P2P 节点上, 用于保证 P2P 网络负载均衡、减少服务响应时间和提高资源可用性。已有的研究<sup>[10,11]</sup> 主要关注搜索、复制

和缓存方法, 很少考虑共享文件与副本之间的一致性维护。一方面, 当前的 P2P 网络假定共享文件是静态的或者是很少变化的; 另一方面, 许多实际 P2P 应用, 如信任管理、在线拍卖、远程协作和目录服务等, 要求共享文件是动态的或者是频繁变化的。因此, 维护共享文件与副本之间的一致性保障 P2P 网络的正确性和可用性的关键。

实质上, P2P 网络中共享文件与副本一致性维护是一个组播问题, 即一个源节点(共享文件)将更新消息转发给一组接收节点(副本), 从而保证源节点与接收节点之间状态信息同步。P2P 网络是一种自组织的分散式系统, 即无中央控制和全局信息, 所以, 网络中一种好的副本更新传播方法必须满足: a) 快速和可靠传播, 即当 P2P 节点发生失效或搅动(churn)时, 更新消息可在有限跳步内传播所有在线节点; b) 可伸缩性, 即当 P2P 网络中副本节点数增多时, 更新传播方法的性能下降不明显; c) 成本效益更新开销, 即更新传播方法的网络通信开销低, 避免底层网络基础设施负载过重。

P2P 网络中已有的更新传播方法存在开销高、可靠性和可伸缩性差等问题。基于泛洪的推拉(push-pull)更新传播方法<sup>[3,12]</sup> 是: 在推阶段, 每个副本节点将更新消息转发给其所有邻居副本节点; 在拉阶段, 副本节点向其所有邻居副本节点请

收稿日期: 2009-06-16; 修回日期: 2009-07-17      基金项目: 湖南省自然科学基金资助项目(09JJ4033); 湖南省教育厅科研基金资助项目(09C209)

作者简介: 周翠红(1974-), 女, 湖南益阳人, 讲师, 硕士, 主要研究方向为网络安全、对等计算(zch2616598@163.com); 贾丽媛(1972-), 女, 湖南益阳人, 副教授, 硕士, 主要研究方向为数据挖掘、计算机网络; 左伟明(1962-), 女, 湖南宁乡人, 副教授, 主要研究方向为人工智能、计算机网络; 秦姣华(1973-), 女, 湖南益阳人, 副教授, 博士研究生, 主要研究方向为网络安全、信息安全。

求更新消息。虽然可保证快速和可靠地传播更新消息,基于泛洪的更新传播方法存在网络通信开销高和可伸缩性差等问题,即更新消息数以指数级别迅猛增加,严重消耗网络带宽等资源。为了降低更新传播开销,基于分发组播树的更新传播方法<sup>[13,14]</sup>是在P2P覆盖网络上构建一个以源节点为根节点且遍历所有副本节点的应用层组播树,且源节点将更新消息沿该组播树转发给所有副本节点。由于P2P网络是一种自组织的复杂系统,基于分发组播树的更新传播方法存在组播树构建和维护开销高、可靠性差等问题,即难以适应大量P2P节点失效和搅动。

为了解决上述问题,本文提出了非结构化P2P网络中一种低开销和高可靠的更新传播方法,即基于闲话的推拉更新传播方法。其主要思想是:在推阶段,每个副本节点将更新消息随机转发给一组邻居副本节点;在拉阶段,副本节点随机向一组邻居节点请求更新消息。实质上,基于闲话的更新传播方法是一种可靠和可伸缩的分散式传染病算法<sup>[15,16]</sup>,可提供快速、可靠和高吞吐量的更新传播,并且以高概率保证更新消息传递到所有副本节点。与已有的基于闲话的组播方法<sup>[15,16]</sup>不同,本文探讨了P2P网络中更新消息接收概率、闲话传播扇出数、网络搅动和网络拓扑之间的关系,分析了闲话传播扇出数和网络搅动等对该更新传播方法性能的影响。

## 1 相关工作

已有的P2P网络研究主要集中于文件搜索、复制和缓存方法,而很少关注共享文件与副本之间的一致性管理。构建一个可读可写P2P网络的关键是设计一种快速、高效和可靠的更新传播方法,保证共享文件与副本之间状态信息同步。

Lan等人<sup>[12]</sup>提出了基于泛洪的主动推和基于自适应轮询的被动拉的更新传播方法,即共享文件的管理节点采用泛洪的方式广播该共享文件的无效通知给所有在线副本节点;副本节点采用轮询方式从管理节点获得该文件的最新版本。但是,该更新传播方法假定只有管理节点可以更新共享文件,一旦管理节点失效或者离开,则无法更新共享文件;此外,该方法的网络通信开销过大。

为了减少更新传播的网络通信开销,Datta等人<sup>[13]</sup>提出了一种基于流言散布的混合推拉更新传播方法,即在推阶段,每个副本节点将接收到的更新消息以一定概率转发给一组随机选取的副本节点;在拉阶段,副本节点向多个副本节点发送更新查询消息。由于要求每个副本节点必须拥有所有副本节点的位置信息,该更新传播方法不是一个可伸缩的方法,它难以适应大规模副本节点。

Wang等人<sup>[14]</sup>提出了使用副本链的更新传播方法,即为每个共享文件建立一个包含所有副本节点的逻辑链,且副本链上每个副本节点存储双向最邻近的 $k$ 个节点。在推阶段,每个副本节点将接收到的更新消息向在每个方向上最邻近的 $k$ 个副本节点转发,并在每个方向上选取跳步数最大的在线副本节点作为更新消息的转发节点;在拉阶段,副本节点从其邻近的在线副本节点上获得最新版本的文件。但是,当网络失效和搅动比较频繁时,这种方法无法保证消息传递的可靠性。由于为每个共享文件构建和维护一个副本链必须要有所有副本节点的

位置信息,该方法也是一种不可伸缩的更新传播方法。

Roussopoulos等人<sup>[17]</sup>提出了基于激励的可控制更新传播协议CUP,即在CUP中,每个副本节点采用激励策略,根据参与更新消息的转发是否有利可图来决定何时接收和转发更新消息。但是这种更新传播方法中,CUP研究元数据的一致性维护,而不是共享文件与副本之间的一致性维护。

Chen等人<sup>[15]</sup>提出了在结构化P2P网络上为每个关键值构建一棵副本分割树。该方法采用DHT记录所有复制对等节点的位置信息,沿这棵副本分割树传播更新消息;此外,该方法将所有副本对等节点的位置信息存储在多个节点上,可解决热点和节点失效问题。

李振宇等人<sup>[18]</sup>提出了在非结构化P2P网络中一种基于分割树的副本一致性维护方法。该方法利用Chord协议作为组管理协议,通过动态分割Chord标志符空间构建更新传播树,具有良好的可伸缩性和容错能力,并减少冗余消息数。

以上方法存在更新传播树构建和维护开销高且抗搅动力差,难以在实际P2P网络上应用。本文基于闲话的更新传播方法不需要构建和维护更新传播树,抗搅动力强且冗余消息较少,易于在实际P2P网络中实现和推广。

## 2 基于闲话的更新传播方法

本文提出了非结构化P2P网络中一种基于闲话的推拉更新传播方法,其目的是减少网络通信开销以及保证更新消息快速和可靠地传递。在本文中,P2P网络中共享文件与副本之间是弱一致性,而不是强一致性。

基于闲话的推拉更新传播方法包括推阶段和拉阶段,且更新传播方法的推阶段与拉阶段是互相重叠进行的。基于闲话的推拉更新传播方法的基本思想是:在推阶段,当接收到更新请求时,源节点将更新消息转发给其邻居副本节点;而在拉阶段,当加入系统或者等待超时,副本节点检查其副本文件是否是最新版本,并向邻居节点发送更新查询请求。

基于闲话的更新传播方法的推阶段算法如下:

```
/* push phase at replica peer p */
p receives an update message m = {f, L} from neighbor peer q;
if (check_version(f in p, f in m) = Old-Version) do
    update_file(f in p, f in m);
    N = {a set of neighbor replica peers of p};
    S = Pr * (N - L);
    m' = {f in m, L + S};
    for (each s in S) do
        delivery_message(m', s);
    endfor;
else
    discard m;
endif;
```

在基于闲话的更新传播方法的推阶段,源节点以概率 $P_r$ 将更新消息转发给邻居副本节点;当接收到更新消息时,每个副本节点以概率 $P_r$ 将更新消息转发给一组未转发过该消息的邻居副本节点;当接收到重复更新消息时,副本节点丢弃该更新消息。在基于闲话的更新传播方法的推阶段算法中,更新消息 $m$ 由文件 $f$ 和已更新节点列表 $L$ 组成。文件 $f$ 包括更新文件、版本、大小和创建时间等信息;节点列表 $L$ 是已更新节点子集,其目的是用于减少重复转发的更新消息。

基于闲话的更新传播方法的拉阶段算法如下:

```

/* pull phase at replica peer p */
p joins into network again or Wait-time expires
N = { a set of neighbor replica peers of p };
m = { f in m, N };
for( each n in N) do
    pull_request(m, n);
endfor;
/* pull phase at replica peer q */
q receives a pull request m = { f, L } from neighbor peer p;
if( check_version(f in q, f in m) = Old-Version) do
    update_file(f in q, f in m);
    discard m;
else
    m' = { f in q, { p } };
    pull_response(m', p);
endif;

```

在基于闲话的更新传播方法的拉阶段,当加入 P2P 网络或者等待更新消息超时,副本节点向一组随机邻居副本节点发送更新查询请求;当接收到更新查询请求,副本节点返回一个更新响应。在基于闲话的更新传播方法的拉阶段算法中,当加入系统或者等待超时,副本节点  $p$  向其邻居副本节点发送更新查询请求  $m$ 。其中  $m$  是由副本节点  $p$  的文件  $f$  和副本节点列表  $N$  所组成。

### 3 性能评价指标

P2P 更新传播方法的性能评价指标包括网络通信开销、更新传播速度、查询失效率和可靠性等。网络通信开销是指 P2P 网络中所有副本节点转发的更新消息总数;更新传播速度是指 P2P 网络中接收到更新消息的副本节点数占网络中总副本节点数的比例,更新传播速度也可以用更新消息转发给所有在线副本节点所需覆盖网跳步数来衡量;查询失效率是指在当访问共享文件及其副本时,在所有被查询的副本节点中,副本文件未更新的副本节点所占比例;可靠性是指在网络失效和搅动下,性能指标的变化。

P2P 网络中一种好的更新传播方法是网络通信开销小、更新传播速度快、查询失效率高,以及在网络失效和搅动下性能下降不明显。

### 4 模拟实验及其结果分析

由于 P2P 网络具有幂律和小世界特征<sup>[19,20]</sup>,本文采用 BRUTE<sup>[22]</sup> 产生 8 000 个副本节点的 P2P 网络拓扑图,其中副本节点的平均度数为 10。P2P 节点是频繁加入和离开网络,其到达时间服从泊松分布;本文采用 turn-on/turn-off 方式模拟节点搅动,即当一个节点离开 P2P 网络时,随机选取一个新节点加入 P2P 网络,保证副本节点总数是恒定的。

本文采用 C/C++ 设计和实现了一个基于事件驱动的 P2P 网络模拟器,评估基于泛洪和基于闲话的推拉更新传播方法的性能。在软件模拟中,副本节点搅动比率  $R_{churn}$  为 0.1~0.4,副本节点搅动频率  $f_{churn}$  为每跳步 1 次搅动,副本节点转发概率  $P_{fanout}$  为 0.6~0.9。

#### 4.1 网络通信开销

图 1 给出了在节点搅动比率  $R_{churn}$  为 0.1 且转发概率  $P_{fanout}$  为 0.6 下的总更新消息数。如图 1 所示,在推拉情况下,与基于泛洪的更新传播方法相比,基于闲话的更新传播方法减少了

38% 的总更新消息数;在推情况下,与基于泛洪的更新传播方法相比,基于闲话的更新传播方法减少了 44% 的总更新消息数。

图 2 给出了在转发概率  $P_{fanout}$  为 0.6~0.9 下的总更新消息数。如图 2 所示,在转发概率为 0.6 时,基于闲话的更新传播方法的总更新消息数减少了 38%,而在转发概率为 0.9 时,其总更新消息数仅减少了 11%。

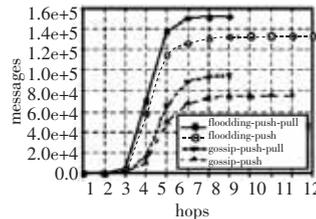


图 1 总更新消息数

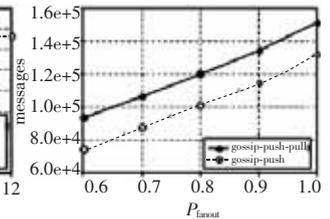


图 2 不同转发概率下的总更新消息数

图 3 给出了在节点搅动比率  $R_{churn}$  为 0.1~0.4 的总更新消息数。如图 3 所示,随着搅动比率的递增,在推拉情况下,基于闲话的更新传播方法的总更新消息数增长比较缓慢;而在推情况下,其总更新消息数却递减。这是因为在推拉阶段,当副本节点发生搅动且重新加入 P2P 网络时,它会向所有邻居副本节点发送更新查询请求;而在推阶段,发生搅动的副本节点不能接收和转发更新消息,而且当更多副本节点发生搅动时,更新消息迅速消亡,其他副本节点更难以接收到更新消息。

#### 4.2 更新传播速度

图 4 给出了在节点搅动比率  $R_{churn}$  为 0.1 且转发概率  $P_{fanout}$  为 0.6 下的节点覆盖率。如图 4 所示,在推拉情况下,基于泛洪和基于闲话的更新传播方法都只需 7 跳步就能达到 100% 的节点覆盖率;在推情况下,基于泛洪和基于闲话的推更新传播方法需要较长的跳步(分别为 12 跳步和 11 跳步)才能接近于 100% 的节点覆盖率(分别为 98.3% 和 96.8%)。图 4 表明,与基于泛洪的更新传播方法相比,基于闲话的更新传播方法在前 4 跳步内节点覆盖率的增长较慢,而在第 5 跳步和第 6 跳步内节点覆盖率的增长较快,并迅速接近于 100%。这是由于基于泛洪的更新传播方法产生大量的重复更新消息,这些更新消息被转发给大量已接收到更新消息的节点而导致节点覆盖率的增长缓慢;而基于闲话的更新传播方法减少了这些转发的重复更新消息,并保证更新消息转发给未接收到更新消息的副本节点。

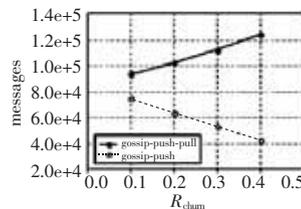


图 3 不同搅动比率下的总更新消息数

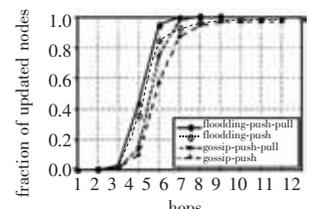


图 4 节点覆盖率

图 5 给出了在转发概率  $P_{fanout}$  为 0.6~0.9 下的节点覆盖率。如图 5(a) 和(b)所示,基于闲话的更新传播方法的节点覆盖率在第 6 跳步之后基本相同,即不同转发概率对节点覆盖率的影响不大。与基于泛洪的更新传播方法一样,基于闲话的更新传播方法能快速和高效地将更新消息转发给所有副本节点,且不同转发概率对其更新传播速度影响小。

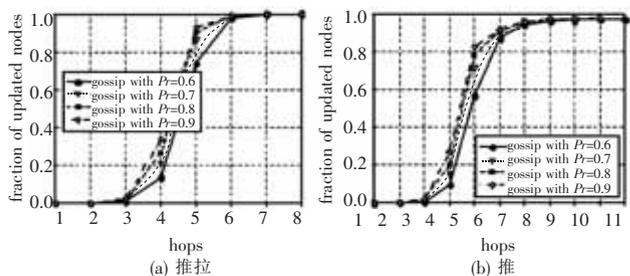


图 5 在不同转发概率下的节点覆盖率

图 6 给出了在节点搅动比率  $R_{churn}$  为 0.1 ~ 0.4 的节点覆盖率。如图 6(a) 所示,随着搅动比率增大,基于闲话的更新传播方法需要更多跳步来到达 100% 的节点覆盖率;图 6(b) 进一步显示,在推情况下,基于闲话的更新传播方法的节点覆盖率只能逐渐接近 100%,并且受搅动比率影响大。图 6 表明,基于闲话的更新传播方法的抗搅动能力好。

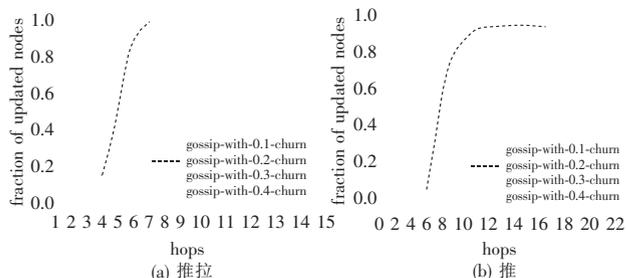


图 6 在不同搅动比率下的节点覆盖率

### 4.3 查询失效率

图 7 给出了在节点搅动比率  $R_{churn}$  为 0.1 且转发概率  $P_{fanout}$  为 0.6 下的查询失效率。如图 7 所示,与推情况相比,推拉情况下更新传播方法具有较低查询失效率,即更新传播方法能更快地传播消息。图 7 表明,在前 5 跳步内,基于闲话的更新传播方法的查询失效率稍微高于基于泛洪的更新传播方法,但是从第 6 跳步开始,这两种方法的查询失效率基本相同。

图 8 给出了在转发概率  $P_{fanout}$  为 0.6 ~ 0.9 下的查询失效率。图 8 也表明,在前 5 跳步内,基于闲话的更新传播方法的转发率越高,查询失效率越低,但是从第 6 跳步开始,不同转发概率的更新传播方法的查询失效率基本相同。因此,与基于泛洪的更新传播方法相比,基于闲话的更新传播方法的查询失效率在前 5 跳步内较高,且随着转发概率的增大而减小,但是从第 6 跳步开始,该方法的查询失效率基本相同,且转发概率影响不大。

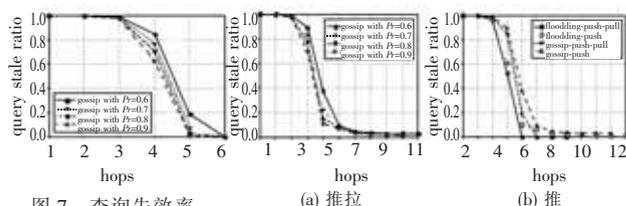


图 7 查询失效率

图 8 在不同转发概率下的查询失效率

图 9 给出了在节点搅动比率  $R_{churn}$  为 0.1 ~ 0.4 的查询失效率。如图 9(a) 所示,随着搅动比率增大,基于闲话的更新传播方法的查询失效率收敛速度快,如在第 6 跳步到达 0。如图 9(b) 所示,随着搅动比率增大,基于闲话的更新传播方法的查询失效率收敛速度慢,即跳步较长,且难以到达 0。因此,与推情况相比,推拉情况下的基于闲话的更新传播方法健壮性好,是一种可靠的更新传播方法。

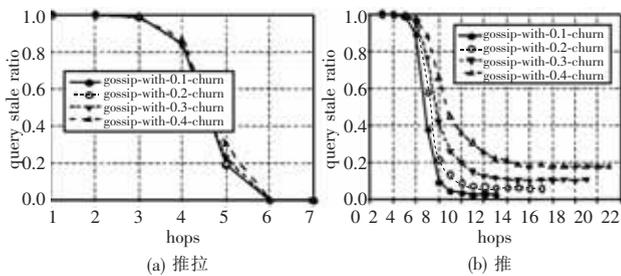


图 9 在不同搅动比率下的查询失效率

## 5 结束语

非结构化 P2P 网络中共享文件与副本文件之间一致性维护迫切要求一种快速、可伸缩和可靠的更新传播方法。本文提出了一种基于闲话的推拉更新传播方法,即每个副本节点将接收到的更新消息转发给随机选取的一组邻居副本节点;当重新加入 P2P 网络或者等待更新消息超时,副本节点向邻居副本节点发送更新查询请求消息。实验结果表明,本文方法的网络通信开销远小于已有的更新传播方法,且随着更新消息转发概率的递增而其网络通信开销呈线性增加。例如,在转发概率为 0.6 时,与基于泛洪的更新传播方法相比,本文方法在网络通信开销上减少了 38%;在转发概率为 0.9 时,网络通信开销减少了 11%。本文方法可高效和可靠地传播更新消息,其节点覆盖率达 100%,且查询效率快速收敛,不受更新消息转发概率大小的影响。当网络节点搅动比率增大时,本文方法在网络通信开销上虽然增加,但是具有良好的节点覆盖率和查询失效率,其抗搅动性强。因此,基于闲话的更新传播方法是一种快速、低开销和可靠的一致性维护方法。

### 参考文献:

- [1] MILOJICIC D, KALOGERAKI V, LUKOSE R, *et al.* Peer-to-peer computing [R]. [S. l.]: HP Laboratories, 2002.
- [2] SAROIU S, GUMMADI K P, DUNN R J, *et al.* An analysis of internet content delivery systems [C]//Proc of the 15th Symposium on Operating Systems Design and Implementation. Boston: [s. n.], 2002.
- [3] LV Q, CAO P, COHEN E, *et al.* Search and replication in unstructured peer-to-peer networks [C]//Proc of the 16th Annual ACM International Conference on Supercomputing. New York: ACM Press, 2002.
- [4] STOICA I, MORRIS R, KARGER D, *et al.* Chord: a scalable peer-to-peer lookup protocol for Internet applications [J]. *IEEE/ACM Trans on Networking*, 2003, 11 (1): 17-32.
- [5] RATNASAMY S, FRANCIS P, HANDLEY M, *et al.* A scalable content-addressable network [C]//Proc of ACM SIGCOMM. 2001.
- [6] ROWSTRON A, DRUSCHEL P. Pastry: scalable, distributed object location and routing for large-scale peer-to-peer systems [C]//Proc of IFIP/ACM International Conference on Distributed Systems Platforms. 2001.
- [7] ZHAO B Y, HUANG L, STRIBLING J, *et al.* Tapestry: a resilient global-scale overlay for service deployment [J]. *IEEE Journal on Selected Areas in Communications*, 2004, 22 (1): 41-53.
- [8] Gnutella [EB/OL]. [2008-10-19]. <http://www.gnutella.com/>.
- [9] KaZaA [EB/OL]. [2008-11-19]. <http://www.kazaa.com/>.
- [10] ANDROUTSELLIS-THEOTOKIS S, SPINELLIS D. A survey of peer-to-peer content distribution technologies [J]. *ACM Computing Surveys*, 2004, 26 (4): 335-371.

间越长的数据越先被发送,但由于它未考虑用户信道质量,某些高质量的信道被闲置,使得最终的数据正确接收率不高,从而增大了丢包率;而方案1很好地统筹了时延和信道信息之间的关系,在获得较大系统容量的同时,在用户数较大时仍然能够保持较低的平均丢包率。

图4比较了四种方案在不同的用户平均数据到达率下的平均时延。如图所示,四种方案下的平均时延均随着数据到达率的升高而增大。本文所提方案1随着数据到达率的升高,保持了较为稳定的平均时延;方案2之所以能取得较好的时延特性,是因为它以高丢包率和低系统容量为代价;方案3和4由于未考虑用户时延限制,当数据到达率大于17 kbps时,平均时延非常大,这将导致系统稳定性降低。

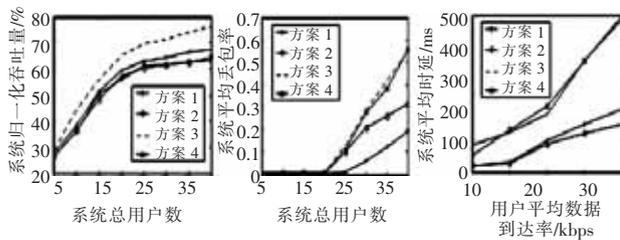


图2 不同方案下归一化系统吞吐量比较

图3 不同方案下平均丢包率比较

图4 不同方案下平均时延比较

由以上仿真实验可以看出,本文提出的方案1整体性能优于其他方案。

## 4 结束语

在现代通信中,随着许多高速无线实时业务的出现,通信系统对用户实时业务的传输时延提出了越来越高的要求。针对实时业务对传输时延的敏感性,本文在多用户OFDM下行链路中,提出了一种新的跨层自适应资源分配方案。该方案将用户时延信息作为一个重要的因素,与无线信道信息一并考虑来设置用户的调度优先级,避免了用户业务数据的等待时间过长,从而降低了平均时延。在对无线资源进行自适应分配时,采用了具有并行处理能力强、收敛速度快、易收敛到最优解等优点的HNN算法,提高了系统的整体性能。仿真结果显示,本文所提方案充分抵抗了用户实时业务对时延的敏感性,保证系统高容量的同时降低了平均丢包率和时延。这对于研究时

延敏感度高的实时通信,具有一定的理论和实际意义。

## 参考文献:

- BAKHTIARI E, KHALAJ B H. A new joint power and subcarrier allocation scheme for multiuser OFDM systems [C]//Proc of the 14th Personal, Indoor and Mobile Radio Communications. 2003: 1959-1963.
- YIN H, LIU H. An efficient multiuser loading algorithm for OFDM-based broadband wireless systems [C]//Proc of IEEE Global Telecommunications Conference. 2000: 103-107.
- WONG C, CHENG R, LETAIEF K, et al. Multiuser OFDM with adaptive subcarrier, bit, and power allocation [J]. IEEE Journal on Selected Areas in Communications, 1999, 17(10): 1747-1758.
- JANG J, LEE K B. Transmit power adaptation for multiuser OFDM system [J]. IEEE Journal on Selected Areas in Communications, 2003, 21(2): 171-178.
- KIVANC D, LI G, LIU H. Computationally efficient bandwidth allocation and power control for OFDMA [J]. IEEE Trans on Communications, 2005, 53(1): 107-116.
- JEONG S S, JEONG D G, JEON W S. Cross-layer design of packet scheduling and resource allocation in OFDMA wireless multimedia networks [C]//Proc of IEEE Vehicular Technology Conference. 2006: 309-313.
- KEMAL K, JOSEPH H K, MURALI K, et al. Cross-layer optimization for OFDMA-based wireless mesh backhaul networks [C]//Proc of Wireless Communication and Networking Conference. 2007: 276-281.
- DANIEL C, JOSE F M, DAVID G B, et al. A delay-centric dynamic resource allocation algorithm for wireless communication systems based on HNN [J]. IEEE Trans on Vehicular Technology, 2008, 57(6): 3653-3665.
- MA R N, CHU P, ZHANG S R. Stability condition for discrete delayed Hopfield neural networks [J]. IEEE Natural Computation, 2007, 1: 468-472.
- TAKAGI H. Queueing analysis; a foundation of performance evaluation vol. 3; discrete-time systems [M]. Amsterdam: Elsevier Science Publishers, 1993.
- LOVE D J, HEATH Jr R W. OFDM power loading using limited feedback [J]. IEEE Trans on Vehicular Technology, 2005, 54(5): 1773-1780.
- SHEN Z K, JEFFREY G A, BRIAN L E. Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints [J]. IEEE Trans on Communications, 2005, 4(6): 2726-2737.
- ROUSSOPOULOS M, BAKER M. CUP: controlled update propagation in peer-to-peer networks [C]//Proc of USENIX Annual Technical Conference. 2003.
- 李振宇, 谢高岗, 李忠诚. PATCOM: 基于分割树的无结构P2P系统一致性维护方法 [J]. 计算机学报, 2007, 30(9): 1500-1510.
- SAROUIS S, GUMMADI P K, GRIBBLE S D. A measurement study of peer-to-peer file sharing systems [C]//Proc of Multimedia Computing and Networking. 2002.
- GUMMADI K P, DUNN R J, SAROIU S, et al. Measurement, modeling, and analysis of a peer-to-peer file-sharing workload [C]//Proc of the 19th ACM Symposium on Operating Systems Principles. New York: ACM Press, 2003: 314-329.
- BRITE [EB/OL]. [2008-09-11]. <http://www.cs.bu.edu/brite/>.
- EUGSTER P T, GUERRAOU R. Probabilistic multicast [C]//Proc of International Conference on Dependable Systems and Networks. 2002.
- YU H, VAHDAT A. Consistent and automatic replica regeneration [J]. ACM Trans on Storage, 2004, 1(1): 3-37.

(上接第1113页)

- COHEN E, SHENKER S. Replication strategies in unstructured peer-to-peer networks [C]//Proc of ACM SIGCOMM. 2002.
- LAN J, LIU X, SHENOY P, et al. Consistency maintenance in peer-to-peer file sharing networks [C]//Proc of the 3rd IEEE Workshop on Internet Applications. 2003.
- DATTA A, HAUSWIRTH M, ABERER K. Updates in highly unreliable, replicated peer-to-peer systems [C]//Proc of the 23rd IEEE International Conference on Distributed Computing Systems. 2003.
- WANG Z, DAS S K, KUMAR M, et al. Update propagation through replica chain in decentralized and unstructured P2P systems [C]//Proc of the 4th International Conference on Peer-to-Peer Computing. 2004.
- CHEN X, REN S, WANG H, et al. SCOPE: scalable consistency maintenance in structured P2P systems [C]//Proc of IEEE INFOCOM'05. 2005.
- BIRMAN K P, HAYDEN M, OZKASAP O, et al. Bimodal multicast [J]. ACM Trans on Computer Systems, 1999, 17(2): 41-88.