

## 基于 Gossip 的自适应成员关系管理协议

张治斌<sup>1</sup>, 冯文峰<sup>1</sup>, 黄永峰<sup>2</sup>

(1. 河南理工大学 计算机科学与技术学院, 河南 焦作 454002; 2. 清华大学 电子工程系, 北京 100084)

(fengwf@tsinghua.edu.cn)

**摘 要:**提出了面向动态异质环境的 P2P 成员关系管理协议。该协议能根据节点能力大小动态调整节点连接个数,从而使得节点连接数分布和节点能力度分布相匹配,有利于提高 P2P 网络的资源利用率和负载均衡。协议基本操作包括:节点加入、节点退出、节点失效恢复、节点能力度汇聚和节点关系更新。实验结果表明,和不考虑节点能力度的相关协议相比,与节点能力度动态适应的节点成员关系管理协议具有更高的资源利用率。

**关键词:**对等网; 分布式算法; Gossip 协议; 覆盖网络构建; 动态异质性

**中图分类号:** TP393 **文献标志码:** A

## Gossip-based adaptive membership management protocol

ZHANG Zhi-bin<sup>1</sup>, FENG Wen-feng<sup>1</sup>, HUANG Yong-feng<sup>2</sup>

(1. College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo Henan 454002, China;

2. Department of Electronic Engineering, Tsinghua University, Beijing 100084, China)

**Abstract:** A gossip-based adaptive membership protocol which is oriented to dynamic heterogeneous P2P was put forward. This protocol could dynamically adjust node degree according to node capability, and thus the node degree could be matched with the node capability, and then increasing the resource utilization and load balance. The basic operations of the protocol include: node joining, node exit, node failure restore, and node capability aggregation. The experimental results show that the proposed protocol which adapted to the node capability has higher resource utilization than the not-adapted.

**Key words:** Peer-to-Peer (P2P); distributed algorithms; Gossip-based protocol; overlay network establish; dynamic heterogeneous

## 0 引言

Gossip 是闲聊、咕哝、低声窃窃私语的意思,是社会网络中的一种信息传播方式。所谓 Gossip 协议<sup>[1]</sup>就是受这种信息传播方式的启发而设计的一种计算机间的通信协议。现代分布式系统经常在动态不可靠的网络上通过 Gossip 协议提供高可靠和可扩展性好的通信服务。因为闲聊这种信息传播方式和生物病毒的传播方式很相似,所以 Gossip 协议经常也被称为传染病协议。

Gossip 通信的概念可以用我国的老话“一传十,十传百”来类比。这种病毒式的信息传播方式的信息扩散速度是呈指数增长的,只需要  $n$  步,就可以将信息扩散到  $p^n$  个个体,其中  $p$  是每一步的传播能力。在我国的老话“一传十,十传百”中  $p = 10$ 。

计算机通过随机选择节点来实现 Gossip 协议:按照某个频率,每个计算机随机地选择别的节点,然后向它分享信息和数据。Gossip 协议具有可靠性和可扩展性等优点,但是也有带宽消耗大等缺点。一个 Gossip 协议通常满足以下条件:协议的核心是定时的、独立的节点间的互动;互动中交换的信息大小是受限的;每一次互动,双方的状态要改变;不能假定节点间的通信是可靠的,即允许丢包;节点间互动的频率相对于消息传递的延迟要低得多,以降低协议成本;随机节点选择中,节点可能从所有节点中选择,也可能从邻居节点中选择。

Gossip 协议的典型应用是实现路由,节点通过 Gossip 协议在节点间传播路由所需消息,从而建立路由网络,然后基于

该路由网络传递数据。如果带宽允许的话,一个基于 Gossip 的系统能够支持和实现任何的分布式服务。但是,由于 Gossip 协议对带宽的高消耗,所以,它在典型的情况下,只能以一种慵懒、分散、对等的方式来运行。

尽管可能会造成带宽冗余,Gossip 协议还是被普遍认为是在大规模动态非可靠系统中实现可靠传播的主要方式。目前,几乎所有大规模部署的 P2P 流媒体系统(PPLive、PPStream、UUSee、PPMate、SopCast 等)都基于 Gossip 协议来构建媒体播放网络拓扑和定位媒体数据。

基于 Gossip 的 P2P 流媒体网络拓扑构建机制中广泛应用的是 SCAMP(Scalable Membership Protocol)。它是一个可扩展的 P2P 成员关系管理协议;它是完全去中心化的,每个节点维护系统的部分视图;它是自重构的,系统大小改变时,每个成员的视图大小随之改变,由于系统变动造成的某些节点隔离,可以通过恢复机制自动恢复。它包括基本的 P2P 成员管理机制:成员加入、成员退出、成员隔离恢复以及成员关系更新等。

以上介绍的 Gossip 协议中的节点选择机制是均匀随机的:节点 A 决定进行一步扩散时,从整个网络中均匀随机的选择若干节点进行扩散。但节点有时需要选择具有某种特征的节点集,例如:和本地节点距离近的、带宽大的节点等。通过这种有偏的节点选择,能够更快更高效地完成协议目标。此时需要引入有偏 Gossip 协议。

## 1 有偏 Gossip 协议

在传统的 Gossip 协议中,所有的节点拥有相同个数的

收稿日期:2009-05-04;修回日期:2009-07-08。 基金项目:国家自然科学基金资助项目(60703053)。

作者简介:张治斌(1953-),男,河南洛阳人,副教授,主要研究方向:计算机网络通信; 冯文峰(1975-),男,河南济源人,讲师,博士,主要研究方向:分布式系统; 黄永峰(1968-),男,湖北赤壁人,副教授,博士,主要研究方向:计算机网络通信。

Gossip 目标、相同的传播间隔、传播相同数量的 Gossip 消息。但是这种无偏均匀的 Gossip 传播忽略了大规模分布式系统中的一个重要特性:动态异质性,即系统中节点的能力存在极大的差异,并且节点能力随时间动态变化。

节点能力度指节点在不同的应用背景中,根据不同能力指标(服务功能、服务意愿、拥有资源和所处地位等)而拥有的能力大小。节点的连接强度指节点和其他节点的连接权重之和,权重反映了节点之间的连接强度。节点的入连接强度表示了节点被其他节点服务时的情况,节点的出连接强度表现了节点为其他节点服务时的情况;节点的能力度和节点的出连接强度匹配,体现了“我为人人”的原则;节点的能力度和节点的入连接强度匹配,体现了“人人为我”的原则。

所谓节点异质性是指各参与节点在能力度方面的差异性。节点异质性是一个抽象概念,针对不同的应用环境,造成节点异质性的差异因素不同。例如:对于文件共享应用,差异性包括各参与节点拥有的文件资源的差异;对于流媒体应用,差异性包括各参与节点拥有的带宽资源的差异等。

所谓异质性的动态变化是指在平等网络生成构建和演化过程中,由于网络节点的加入、退出或失效,以及节点间交互行为的变化而使得参与节点的异质性也在动态变化。例如:由于节点的加入和退出行为,造成节点持续时间的时变特征;由于底层网络的拥塞行为,造成节点间连接的带宽、延迟和丢包等的动态变化;节点当前的主机负载的变化等。

和文献[2-5]一样,我们认识到需要考虑节点之间的差异,并且利用这种差异性来达到更高效的 Gossip 传播过程。要实现这种面向动态异质性的 Gossip 传播机制,主要面临两个挑战:

1) 节点能力度是随时间动态变化的,需要动态地跟踪和反映这种变化。

2) Gossip 协议的可靠性严重依赖于目标节点选择时的均匀随机性,根据节点能在力度有偏的选择目标节点破坏了均匀性,从而会影响 Gossip 传播的可靠性。

对 Gossip 传播协议的数学建模和经验评估<sup>[6]</sup>都支持这样一个结论:只要所有节点的目标节点个数(*fanout*)的均值大于等于  $\ln(n)$ , 就可以保证 Gossip 传播的可靠性,其中  $n$  是所有节点的个数。目标节点数 *fanout* 是一个关键参数,通过调节 *fanout*, 可以调节不同节点对信息传播的贡献度。一个 *fanout* 较大的节点会发送和接收更多的信息,反之,一个 *fanout* 较小的节点会发送和接收较少的信息。这样,根据节点的能力度差异,以  $\ln(n)$  为基准点调整节点的 *fanout* 参数,就可以进行适应于节点能力度差异的 Gossip 传播。这是有偏 Gossip 协议的基本设计思想。

尽管 Gossip 协议的可靠性可以由 *fanout* 参数的均值大于等于  $\ln(n)$  保证,传统的 Gossip 传播协议并没有根据节点能力度的分布来调整 *fanout* 参数的分布。有偏 Gossip 协议在保证 *fanout* 参数的均值大于等于  $\ln(n)$  的基础上,使得 *fanout* 参数的分布符合节点能力度的分布。例如,以节点上传带宽作为节点能力度的衡量指标。

一种原始的方法是每个节点定时测量自己的上传带宽,然后根据带宽的增加或减少成比例地增加或减少自己的 *fanout*。这种方法很容易导致系统可靠性的严重下降,如果所有节点的上传带宽在逐步降低,最后会导致 *fanout* 参数严重低于  $\ln(n)$ , 从而导致系统可靠性的降低。

一个有偏 Gossip 协议需要满足:1) 每个节点对 Gossip 传播的贡献和它的能力度相匹配,体现为节点 *fanout* 参数的分

布和节点能力度的分布相匹配;2) 为了保证系统可靠性,需要保持节点 *fanout* 参数的均值大于等于  $\ln(n)$ ;3) 要能够根据节点能力度的动态改变适应性地改变 *fanout* 参数。

在一个有偏 Gossip 协议中,每个节点必须知道它自己的能力度在所有节点中所处的位置。为了达到这个目标,需要一个节点能力度汇聚机制,以使每个节点能获得对所有节点能力的认识。节点能力汇聚也是基于 Gossip 机制,具体步骤如下:每个节点定时测量自己的能力度,并定时将自己的能力度和它所收到的节点能力度发送给它的邻居节点。收到节点能力度信息的节点,将这些信息合并到自己的本地节点能力度信息表中,从而获得对所有节点能力度的估计值。

有偏 Gossip 协议还需要一个 *fanout* 调节机制。节点使用它获得的对所有节点能力度的估计值和自己的能力度来计算自己的 *fanout* 参数。有偏 Gossip 协议要能够动态调节节点的 *fanout* 参数。

## 2 有偏的 P2P 成员关系管理协议

大规模动态对等网络系统的重要基础之一是节点成员管理,节点成员管理完成传输所需的网络拓扑构建和数据定位等功能。基于 Gossip 协议的节点成员管理因为具有自组织、可靠性高、可扩展性强等优点成为大规模动态对等网络系统中节点成员管理的主要方式,典型的协议包括 Epidemics<sup>[6]</sup>、PRM<sup>[7]</sup> 和 SCAMP<sup>[8]</sup> 等。

但是,如第 1 章所述,目前的节点成员管理协议并没有考虑节点的动态异质性。为此,本文设计了一种基于 Gossip 的有偏的 P2P 成员关系管理协议(Biased Membership Protocol, BMP)。

BMP 协议包括三个部分:1) 基于 Gossip 的成员关系管理协议,这部分主要借鉴 SCAMP 协议的相关思想;2) 基于 Gossip 的节点能力度汇聚协议,这部分主要借鉴文献[9]的相关思想;3) 根据节点能力度分布动态调整节点 *fanout* 参数的机制。

和 SCAMP 协议一样,在 BMP 协议中,每个节点  $k$  维护关于成员的两个列表:1) PartialView,存储节点  $k$  所知道的所有成员节点,代表节点  $k$  对整个系统的认知;2) InView,存储知道节点  $k$  的所有节点,代表系统对节点  $k$  的认知。PartialView 表的大小称为 *fanout*,代表了节点  $k$  的连接强度;节点  $k$  的能力度称为 *capability*,它是对节点能力的度量。节点  $k$  的 *fanout* 应该和 *capability* 相匹配。

### 2.1 基于 Gossip 的节点成员管理协议。

#### 2.1.1 入口机制

系统设置一个或几个引导(Bootstrap)节点,新节点通过引导节点获得它的联系(Contact)节点,联系节点必须从目前系统节点中随机选择,联系节点的随机性对节点 *fanout* 参数的分布有着重要的影响。当新节点获得联系节点后,这些联系节点就成为新节点的 PartialView 表的初始表项。新节点在获得初始 PartialView 表之后,向 PartialView 中的节点发送加入请求。

算法 1 新加入请求。

描述:新节点准备加入某个对等网络时调用;

输入:若干引导节点 Bootstraps。

//初始化新节点的 PartialView 表为空

PartialView =  $\emptyset$ ;

//查询引导节点获得联系节点 Contacts

for all nodes  $n \in$  Bootstraps do

if Contacts = getContact( $n$ ) is success then

```

PartialView = PartialView + Contacts;
break;
end if
end do
//向联系节点发送加入请求
for all nodes  $n \in$  PartialView do
    Send( $n$ , JoinRequest( $ID_{local}$ ,  $c$ , NEW));
end do

```

算法 1 中,加入请求信息 JoinRequest 包括三个部分: $ID_{local}$  是新节点的标志符; $c$  是一个重要的参数,用于控制  $fanout$  参数的大小;NEW 表示这是一个新加入请求。

### 2.1.2 加入机制

节点收到新加入请求时,它将该请求转发给 PartialView 表中的所有节点,并将加入请求另外转发给从 PartialView 表随机选择的  $c$  个节点。节点收到转发的加入请求时,以概率  $p = 1/(1 + fanout)$  将请求加入的节点加入 PartialView 表中,并向请求节点发送消息告诉对方将自己加入对方的 InView 表。如果请求节点没有加入 PartialView 表,则从 PartialView 表中随机选取一个节点,并将加入请求转发给它。

算法 2 新加入请求处理。

描述:联系节点收到新节点  $s$  的加入请求时调用;

输入:加入请求 JoinRequest( $s$ ,  $c$ , NEW)。

//将加入请求转发给 PartialView 中的所有节点

```
for all nodes  $n \in$  PartialView do
```

```
    Send( $n$ , JoinRequest( $s$ ,  $c$ , FORWARD));
```

```
end for
```

//将  $c$  个加入请求转发给 PartialView 中随机选择的  $c$  个节点

```
for ( $i = 0$ ;  $i < c$ ;  $i++$ ) do
```

```
    Choose randomly  $n \in$  PartialView
```

```
    Send( $n$ , JoinRequest( $s$ ,  $c$ , FORWARD));
```

```
end for
```

算法 3 转发加入请求处理。

描述:节点收到转发的加入请求时调用;

输入:JoinRequest( $s$ ,  $c$ , FORWARD)。

//以概率  $p = 1/(1 + fanout)$  将请求节点加为本节点的成员关系 with probability  $p = 1/(1 + \text{sizeof PartialView})$

```
PartialView = PartialView + { $s$ };
```

//并告诉请求节点,将本节点加入对方的 InView 表

```
Send( $s$ ,  $ID_{local}$ );
```

//否则,转发请求给随机选择的一个成员节点

```
If  $s \notin$  PartialView then
```

```
    Choose randomly  $n \in$  PartialView
```

```
    send( $n$ , JoinRequest( $s$ ,  $c$ , FORWARD));
```

```
end if
```

### 2.1.3 退出机制

如果节点  $k$  准备从对等网络中退出,假定它的 PartialView = { $i_1, i_2, \dots, i_l$ }, InView = { $j_1, j_2, \dots, j_m$ },首先节点  $k$  通知 InView 表中的节点  $j_1, j_2, \dots, j_{m-c-1}$ ,将它们 PartialView 表中的关于节点  $k$  的表项更改为  $i_1, i_2, \dots, i_{m-c-1}$ ,然后节点  $k$  通知 Inview 表中的节点  $j_{m-c}, \dots, j_m$ ,将它们 PartialView 表中关于节点  $k$  的表项删除。

假定  $M_n$  是节点数为  $n$  时对等网络中总的连接个数,那么  $M_n = \sum_n fanout$ 。上述机制<sup>[8]</sup>保证了  $E[M_n] = (c+1)n \ln(n)$ 。

$$E(fanout) = E(M_n)/n = (c+1) \ln(n) \quad (1)$$

从式(1)知道, $c$  是一个很重要的参数,它决定了  $fanout$  参数的均值的大小。我们根据节点能力度分布来动态调节  $fanout$  参数也是通过对参数  $c$  的调节实现的。

## 2.2 基于 Gossip 的节点能力度汇聚协议

假定在时间步  $t$ ,每个节点  $i$  维护节点能力度和  $sum_{t,i}$ ,它的初始值  $sum_{0,i} = fanout_i$ 。在时间步 0,节点将  $sum_{t,i}$  发送给自己。在后续的时间步中,每个节点  $i$  都执行算法 4。

算法 4 节点能力度汇聚。

描述:节点  $i$  在每个时间步  $t$  所执行的算法。

步骤:

1) 设  $\{sum_1, sum_2, \dots, sum_r\}$  是第  $t-1$  个时间步节点  $i$  收到的  $r$  个能力度信息, $r$  是节点 InView 表的大小。

$$2) sum_{t,i} = \sum_{1 \leq i \leq r} sum_i$$

3) 将  $sum_{t,i}/(fanout + 1)$  发送给 PartialView 表中的  $fanout$  个节点和本节点。

4)  $sum_{t,i}$  即是能力度和的估计值

文献[10]在最多  $O(\ln(n) + \ln(1/\varepsilon) + \ln(1/\sigma))$  个时间步之后,算法 4 所获得的对能力度和的估计值  $sum_{t,i}$  的估计误差小于  $\varepsilon$  的概率大于  $1 - \sigma$ 。

## 2.3 更新机制

为了实现节点能力度和节点连接度数的匹配,需要一个根据节点能力度的变化更新节点连接度数的机制。这种更新机制通过引入节点生存期概念实现,所谓节点生存期是指节点从申请加入 P2P 网络开始的一段有效期,超过生存期的节点要从 P2P 网络中删除,并且被重新加入。

每个新加入节点有一个生存期,每个节点从它的 PartialView 表中删除过期的节点,每个节点在过期以后要从自己的 PartialView 表中任选一个节点作为引导节点,并重新加入 P2P 网络。

节点通过能力度汇聚机制获得自己的能力度在网络节点能力度分布中的位置,如果节点发现自己的能力度位置出现了较大的改变,即开始重新加入 P2P 网络,只是在重新加入的时候以与能力度匹配的  $c$  参数加入,从而获得新的  $fanout$  大小,即获得了新的节点连接度数。

## 3 模拟实验与结果分析

实验环境配置:采用文献[10]中开发的基于事件机制的开源的 P2P 模拟软件,并在其上实现 BMP 协议和 SCAMP 协议的所有算法。设定定时更新时间步大小为 1 s,参数  $c$  的期望值大小为 5。开始 50 s 之内,有  $n = 5000$  个节点按照随机时间加入系统,然后为了等待更新机制的完成,我们等待 50 s,获得每个节点的度数  $d$ 。

节点能力度分布:根据实际网络情况确定节点的能力度。目前,互联网终端节点主要存在两种接入方式:ADSL/cable modem 方式和以太网局域网方式。ADSL/cable modem 方式的下载带宽在 512 Kbps ~ 4 Mbps 之间,上传带宽在 256 Kbps ~ 1 Mbps 之间;以太网局域网方式的带宽在 3 ~ 8 Mbps 之间。根据经验,设定 ADSL/cable modem 方式的节点占总节点数的 80%,以太网局域网方式的节点占总节点数的 20%。以 512 Kbps 为单位进行规范化,则每个节点的能力度以 80% 概率在 [1,8] 均匀随机选择,以 20% 概率在 [6,16] 均匀随机选择。我们编写程序按照上述的随机性生成  $n = 5000$  个节点的能力度,其节点能力度分布如图 1 所示。

实验的主要目的是比较 BMP 和 SCAMP 两种成员关系管理协议中,节点度数对节点能力度的匹配程度。首先获得 SCAMP 协议中节点度数的分布如图 2 所示。其中,节点度数

的均值为 48.9, 和期望值  $c \cdot \ln(n) = 5 \times \ln(5000) = 45$  比较接近, 也验证了 SCAMP 协议节点度期望值为  $c \cdot \ln(n)$  的结论。从图 2 也可以看出, SCAMP 协议的节点度数分布与节点能力度分布相差甚远, 这是因为 SCAMP 协议并没有考虑节点能力的差异性的因素。在图 3 中, 我们以节点能力度为  $X$  轴, 以具有某节点能力度的所有节点的节点度数的均值为  $Y$  轴, 分别给出了 SCAMP 和 BMP 两种协议中, 节点能力度和节点度数的关系图。从图 3 中可以看出, SCAMP 协议中, 节点能力度和节点度数基本是相互独立的。而 BMP 协议中, 节点度数和节点能力度之间存在线性关系, 这体现了节点度数和节点能力相匹配的原则。

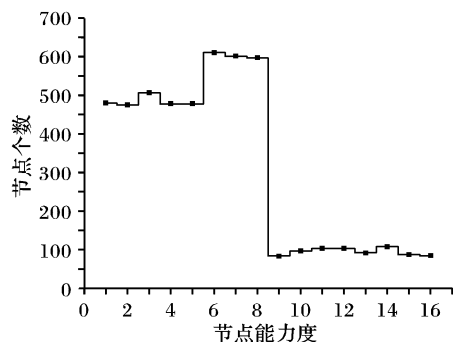


图1 节点能力度分布

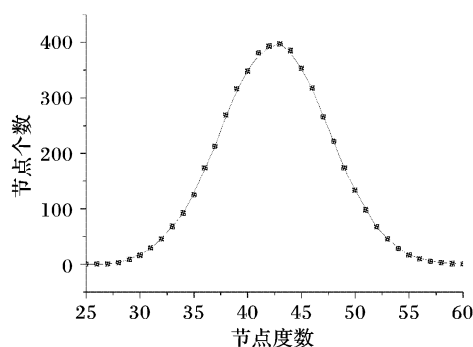


图2 SCAMP协议的节点度数分布

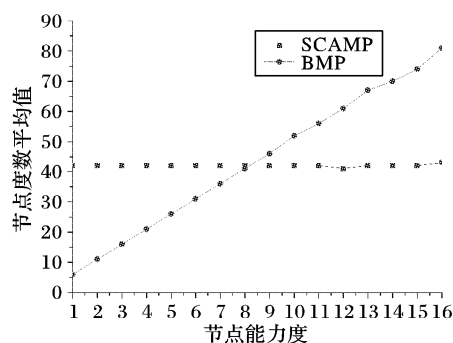


图3 节点度数和节点能力度关系

BMP 协议的节点度数分布如图 4 所示。节点度数的均值是 47.4, 和期望值 45 比较接近, 这也验证了 BMP 协议节点度期望值为  $c \cdot \ln(n)$  的结论。从而可以看出 BMP 协议的节点度数分布和节点能力度分布比较相似。图 4 中, 大部分节点的度数低于 48, 和节点能力度大部分低于 8 是相符的。图 3、4 都说明 BMP 协议实现了节点度数和节点能力的较好匹配, 从而为动态异质环境中的对等网络提供了基础。

#### 4 结语

本文提出了与 P2P 网络的动态异质性相适应的 P2P 网络拓扑构建协议, 它能根据节点能力度的分布动态调整节点

度数的分布, 从而反映节点能力和节点连接强度相适应的原则。模拟实验的结果证明了这种有偏 Gossip 协议的有效性。以后的研究方向是设计与媒体数据内容相适应的有偏 Gossip 协议, 以实现 P2P 网络拓扑构建, 例如: 基于节点之间流媒体播放偏移之间的同步关系, 构建具有有向无环特征的流媒体网络拓扑。

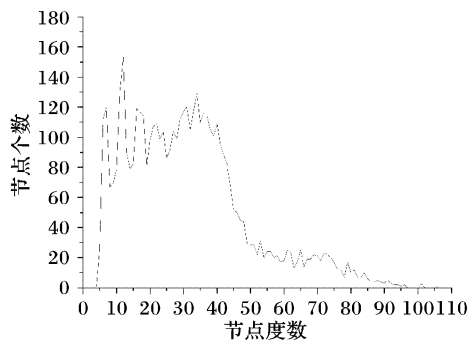


图4 BMP协议的节点度数分布

#### 参考文献:

- [1] ALLAVENA A, DEMERS A, HOPCROFT J E. Correctness of a gossip-based membership protocol [C]// Proceedings of 24th ACM Symposium on the Principle of Distributed Computing. New York: ACM Press, 2005: 292-301.
- [2] KYASANUR P, CHOUDHURY R R, GUPTA I. Smart gossip: An adaptive gossip-based broadcasting service for sensor networks [C]// MASS: 2006 IEEE International Conference on Mobile Ad-Hoc and Sensor Systems. Washington, DC: IEEE Computer Society, 2006: 91-100.
- [3] VENKATARAMAN V, YOSHIDA K, FRANCIS P. Chunkspread: Heterogeneous unstructured tree-based peer-to-peer multicast [C]// ICNP: Proceedings of 14th IEEE International Conference on Network Protocols. Washington, DC: IEEE Computer Society, 2006: 2-11.
- [4] VISHNUMURTHY V, FRANCIS P. On heterogeneous overlay construction and random node selection in unstructured P2P networks [C]// INFOCOM 2006: 25th IEEE International Conference on Computer Communications. Washington, DC: IEEE Press, 2006: 1-12.
- [5] BISHOP M, RAO S, SRIPANIDULCHAI K. Considering priority in overlay multicast protocols under heterogeneous environments [C]// INFOCOM 2006: 25th IEEE International Conference on Computer Communications. Washington, DC: IEEE Press, 2006: 1-13.
- [6] EUGSTER P T, GUERRAQUI R, KERMARREC A-M, et al. Epidemic information dissemination in distributed systems [J]. IEEE Computer, 2004, 37(5): 60-67.
- [7] BANERJEE S, LEE S, BHATTACHARJEE B, et al. Resilient multicast using overlays [J]. ACM SIGMETRICS Performance Evaluation Review, 2003, 31(1): 102-113.
- [8] GANESH A J, KERMARREC A-M, MASSOULIE L. Peer-to-peer membership management for gossip-based protocols [J]. IEEE Transactions on Computers, 2003, 52(2): 139-149.
- [9] KEMPE D, DOBRA A, GEHRKE J. Gossip-based computation of aggregate information [C]// FOCS: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science. Washington, DC: IEEE Computer Society, 2003: 482-491.
- [10] ZHANG MENG, CHENG CHUN-XIAO, XIONG YONG-QIANG, et al. Optimizing the throughput of data-driven based streaming in heterogeneous overlay network [C]// MMM'07: 13th International Multimedia Modeling Conference. Berlin: Springer-Verlag, 2007: 475-484.