

多 Torrent 下基于拍卖的种子节点激励算法^①李晓义^② 李治军 姜守旭

(哈尔滨工业大学计算机科学与技术学院 哈尔滨 150001)

摘 要 针对多 Torrent 文件系统种子节点缺乏有效激励的问题进行了研究,指出现实中多 Torrent 文件系统大量存在,这为种子节点激励提供了环境;争锋相对(TFT)机制其目的是用来最大化下载带宽,但没有考虑种子节点的重要作用,因此不能用于种子节点激励。进而提出了基于拍卖的种子节点激励算法(ABSI):种子节点对带宽进行拍卖,leecher 节点进行出价,种子节点根据 leecher 节点的出价进行带宽分配。同时分析了该算法对种子节点的激励效果以及在此情况下节点下载时间的变化。实验表明该算法可以很好地用于种子节点激励。

关键词 多 Torrent, 种子, 激励机制, 拍卖

0 引言

BitTorrent^[1]作为一种文件共享协议,其应用正变得越来越广泛^[2]。然而,BitTorrent 协议缺乏对种子(seeder)节点激励,这会导致长期文件可获得性问题,即许多 Torrent 经过一段时间后由于缺少部分文件块而使得节点下载不到完整的文件,缺少种子节点是导致这个问题的一个重要原因。种子节点拥有全部文件块而 leecher 节点只拥有部分文件块,没有种子节点的存在,系统中有些文件块不被任何节点所拥有,因此节点下载不到全部文件。文献[3]说明了这个问题的严重性:当系统处于无种子状态时,系统中的节点的下载速度迅速降到零,在此情况下用户只能选择离开系统。因此针对种子节点的激励非常重要,需要设计有效的激励措施来鼓励种子节点停留在系统中^[4]。对种子节点的激励可以分为单 Torrent 激励和多 Torrent 激励。在基于单 Torrent 的系统中,用户在一个 Torrent 中共享的数据并不能用在另一个 Torrent 中,leecher 节点下载完文件成为种子节点后就会选择离开,因此无法有效激励种子节点留在系统中。在多 Torrent 系统中,一个用户参与到多个 Torrent 可以有种子节点和 leecher 节点两种不同身份:可以在一个 Torrent 中作为种子节

点,而在另一个 Torrent 中作为 leecher 节点,这就为种子节点的激励提供了很好的环境。现实应用表明多 Torrent 应用大量存在:现实中 85% 的节点同时参与到多个 Torrent 中进行下载^[5];在在线游戏补丁更新和在线电影租赁及软件在线更新应用场景多 Torrent 会非常流行^[6]。这些应用的特点是参与用户众多,所需下载的文件都非常大,所有这些应用都适合采用多 Torrent 的系统发布文件。事实上,现有的一些公司已经开始使用这种方式来分发文件。

关于多 Torrent 及种子节点的激励机制的研究很少,而且都是基于争锋相对(tit for tat, TFT)机制及其相关改进,其目标都是最大化节点自身下载带宽。如 BitThief^[7]利用随机选择部分不需要传送带宽这一特点来尽可能多地挤占别的节点随机选择部分,BitTyrant^[8]依据别的节点向自己上传的带宽与自己向其上传的带宽的比由大到小选择来最大化自己的下载带宽,余一娇^[9]等对激励机制如何抑制 P2P 网络的搭便车行为进行了详细的分类讨论。他们的方法虽然可以使得节点的下载带宽尽可能增大,但在此情况下种子节点并不愿加入系统。对多 Torrent 的带宽分配的研究很少。Guo^[5]等人的研究虽然针对的是多 Torrent 的研究,但并没有给出带宽分配算法;Yang^[6]等人对 TFT 机制进行了修改,但没有考虑种子节点的重要性和系统中节点自治性

① 国家自然科学基金(60803148, 60973124),教育部高校博士点科研基金(20102302110036)和中央高校基本科研业务费专项资金(HIT-NSRIF.2010.047)资助项目。

② 男,1981年生,博士生;研究方向:P2P文件系统;联系人,E-mail:lixiaoyisuper@gmail.com
(收稿日期:2011-10-08)

(即每个节点可以理性地选择自己策略来达到最优化),因而没有说明 leecher 节点如何愿意向种子节点支付带宽而种子节点在什么情况下愿意加入系统做种,这些方法都不能直接用在多 Torrent 的种子节点激励。由于种子节点对系统性能的提升有重要作用,因此本文针对种子节点的激励进行了研究,提出了多 Torrent 下的基于拍卖的种子节点激励算法 (auction based seeder incentive algorithm, ABSI),实验分析已证明了其有效性。

1 种子节点激励问题描述

1.1 多 Torrent 整体结构

多 Torrent 文件系统由多个 Torrent 组成。参与一个文件,进行下载和上传的节点集合组成一个 Torrent。每个 Torrent 分别由种子节点和 leecher 节点组成,种子节点是拥有完整文件的节点,而 leecher 节点是只拥有部分文件块的节点。

因为现实中用户同时下载多个文件,所以用户同时参与多 Torrent 的现象大量存在。一个节点可以在一个 Torrent 中作为种子节点,而在另一个 Torrent 中作为 leecher 节点。在图 1 中,节点 i 作为 Torrent A 的种子节点,向节点 k, j 上传文件 A;而作为 leecher 节点,从节点 k, j 下载文件 B。如果节点 i 和节点 j 参与的 Torrent 有相同部分(在本例中为 Torrent A 和 Torrent B),那么节点 i 在 Torrent A 向节点 j 上传带宽,就可以在另一个 Torrent B 中从节点 j 处获得带宽,这就为种子节点共享带宽提供了一种激励:因为种子节点虽然不能在其充当种子节点的 Torrent 中获得带宽,但可以从其他 Torrent 中获得带宽,对种子节点进行了补偿。

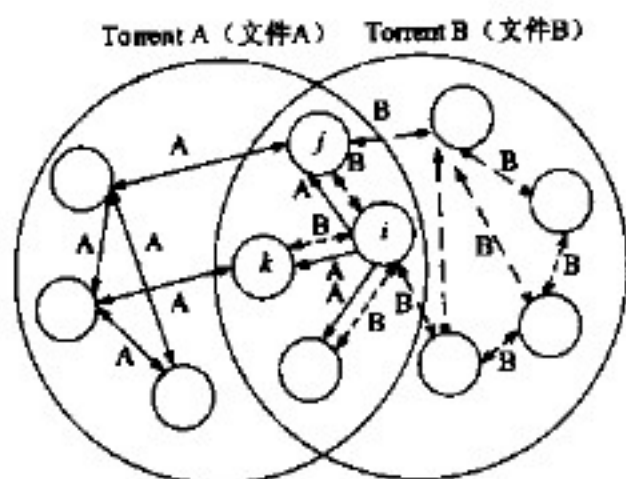


图 1 多 Torrent 系统结构

1.2 基于拍卖描述

由于种子节点的重要性,决定了种子节点做种

可以多获得一些带宽。但是种子节点并不知 leecher 节点愿意向其提供多少带宽,同时 leecher 节点也不知传送多少带宽种子节点才愿意加入系统,因此可以采用拍卖的方法进行。leecher 节点 i 进行出价,而种子节点 j 根据出价而选择如何传送带宽。这样可以认为 leecher 节点是竞拍者,种子节点是拍卖者,拍卖机制 M 决定了 leecher 节点如何出价以及种子节点如何进行带宽分配。下面是本文用到的一些定义:

定义 1 拍卖机制 $M(S, L, B, \pi, \mu)$, 其中 S 是拍卖者(种子)集合, L 是竞拍者(leecher)集合。对于节点 $i \in L, j \in S, B_{ij}$ 是节点 i 向节点 j 的出价,分配规则 $\pi(B)$ 是定义在出价 B 上的函数,表示种子节点如何根据 B 向 leecher 节点传送带宽,支付规则 $\mu(B)$, 表示拍卖成功的 leecher 节点需要向种子节点支付的带宽。

定义 2 对于种子节点 i , 设其要拍卖的带宽为 C (定值),其拍卖的绝对底价为 C' (由种子节点自己根据做种的成本等方面决定),在 t 时刻其拍卖的底价为 $\Delta(t)$ ($\Delta(t) > C'$), $\Delta(t)$ 仅种子节点 i 知道,而其他竞拍者不知。对于 leecher 节点 j , 其在时刻 t 的报价为 $y_j(t)$, $y_j(t)$ 小于等于节点 j 上传带宽的最大值。

定义 3 设时刻 t 所有 leecher 节点向种子节点 i 的报价为 $\{x_i(1, t), x_i(2, t), \dots, x_i(n, t)\}$, $x_i(t) = \max\{x_i(1, t), x_i(2, t), \dots, x_i(n, t)\}$, 如果 $x_i(t) > \Delta(t)$, 那么种子节点向出价最高 leecher 节点传送带宽,称为以价格 $x_i(t)$ 成交,种子节点选择加入系统,否则不加入系统。定义变量 $I(x_i(t), \Delta(t))$, 其值为 1 表示以 $x_i(t)$ 价格成交,否则表示不成交。假设对于种子节点 i , 从时刻 1 到时刻 N 总共进行了 N 次拍卖,在时刻 t , 如果拍卖成功,其可以获得的带宽为 $x_i(t)$, 否则其可以获得的带宽为 0。对于 leecher 节点 j , 如果其赢得拍卖,其可以从种子处获得的带宽为 C , 否则其从种子节点处的获得的带宽为 0。

定义 4 种子节点激励问题 (seeder promotion problem): 对于种子节点,有一个底价序列 $\{\Delta(1), \Delta(2), \dots, \Delta(N)\}$, 种子节点求最优序列 $\{\Delta^*(1), \Delta^*(2), \dots, \Delta^*(N)\}$, 使其可以获得的带宽最大,即

$$\operatorname{argmax}_{\Delta(t)} \sum_{t=1}^N (I(x_i(t), \Delta(t)) \times x_i(t)) \quad (1)$$

对于 leecher 节点 j 也有个出价序列 $\{y_j(1), y_j(2), \dots, y_j(N)\}$, 设在此出价序列下节点 j 可以收到的总

带宽为 $r(j)$, 而节点 j 求最优序列, 使得其收到的带宽最大, 总的出价最小, 即求最优出价序列使得

$$\max r(j) \text{ and } \min \sum y_j(t) \quad (2)$$

给定拍卖机制 M , 从系统来看主要关注种子节点是否愿意响应请求, 为此定义如下的种子响应率:

定义 5 种子响应率 $R_w(T) = q_w(T)/Q(T)$, 给定时间段 T 和拍卖机制 M , 设 $Q(T)$ 是系统中 leecher 节点向种子节点请求次数, 而 $q_w(T)$ 是在拍卖机制 M 下种子节点响应的次数。 $R_w(T)$ 取值范围为 $[0, 1]$, 当 $R_w(T)$ 为 0 时, 表示种子节点完全不愿意响应; $R_w(T)$ 为 1 时, 表示种子节点愿意响应所有请求。

2 种子节点激励问题求解

2.1 拍卖机制运行过程

种子节点的拍卖方式如下: 可以将整个时间轴分为很多时间片, 在每个时间片开始时, 种子节点运行一次拍卖, 从所有出价节点中选择出价最高的节点, 如果成交, 返回拍卖成功标志 (ACK), 在这个时间片内向出价最高者传送带宽; 否则返回拍卖不成功标志 (NACK), 在这个时间片内不传送带宽。其运行过程如图 2 所示。

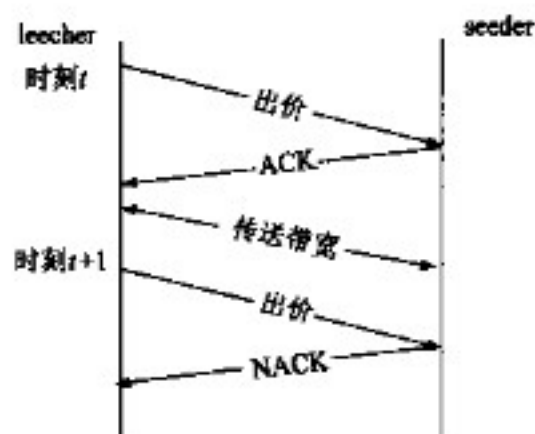


图 2 拍卖机制运行过程

2.2 基于拍卖的种子节点激励算法

由于 leecher 节点的出价及种子的底价相互影响, 而这些对于 leecher 节点和种子节点都无法预知, 因此 leecher 节点及种子都采用“贪心”策略。对于种子节点, 设 t 时刻的底价是 $\Delta(t)$, 如果成交, 那么在 $t+1$ 时刻就会增加为 $\Delta(t) + a$ (设增加 a), 希望通过成交价格来提高获得更多的收益; 反之在 $t+1$ 时刻将底价设置为 $\Delta(t) - a$, 这样来使得交易尽可能成功。对于 leecher j , 设时刻 t 的出价为 $y_j(t)$, 如果成交, 那么在 $t+1$ 时刻就会将出价减少

为 $y_j(t) - b$, 希望减少自己支付的带宽; 反之在 $t+1$ 时刻将出价设置为 $y_j(t) + b$, 这样来使得交易尽可能成功。参数 a 和参数 b 如何影响系统, 在后面的实验进行分析。该算法称为基于拍卖的种子节点激励算法 (ABSI) 其运行过程见图 3。

Seeder 拍卖算法

在 $t+1$ 时刻

1 if (t 时刻成交)

2 $\Delta(t+1) = \Delta(t) + a$;

3 向出价最高的 leecher 节点传送带宽;

4 从出价最高的 leecher 节点获得带宽;

|

5 else $\Delta(t+1) = \Delta(t) - a$

leecher 拍卖算法

在 $t+1$ 时刻

1 if (t 时刻成交)

|

2 $y_j(t+1) = y_j(t) - b$;

3 向种子节点传送带宽;

4 从种子节点获得带宽;

|

5 else $y_j(t+1) = y_j(t) + b$;

图 3 ABSI 运行过程

3 系统分析

由于 leecher 节点需要多给种子节点部分带宽来使得其愿意加入系统, 下面分析在此情况下 leecher 节点下载时间的变化。因为只有在存在种子的情况下才能实施以上激励, 因此分析了 Torrent 个数与存在种子的关系。

3.1 需要多给时的下载时间

在本节, 主要分析 leecher 节点需要多给种子节点带宽时的下载时间。应用文献 [10] 中的流体模型进行分析。考虑如下的场景: Torrent 1 缺少种子, 节点到来速率为 λ , 文件总大小为 m , 系统拥有的文件块大小为 a_1 , 缺少的文件块大小为 a_2 (需要从种子节点获得), $a_1 + a_2 = m$ 。当节点下载拥有的文件块时其处于的状态为状态 1, 当下载缺少的文件块时其处于的状态为状态 2, 设 t 时刻处于状态 1 和状态 2 的节点个数分别为 $x_1(t)$ 和 $x_2(t)$, 系统中总的节点个数 $x(t) = x_1(t) + x_2(t)$, 系统中所有节点的上传带宽都为 u_1 , 种子节点的上传带宽为 u_2 。这样节点在 Δt 时间内流入状态 1 的个数 λ , 流出状态 1

的个数为 $u_1 x_1(t)/a_1$, $dx_1(t)/dt$ 为 λ 与 $u_1 x_1(t)/a_1$ 的差。类似可知: 流入状态 2 的个数 $u_1 x_1(t)/a_1$, 流出状态 2 的速率为 $u_2 x_2(t)/a_2$, 其示意图如图 4 所示。表 1 是用到的一些符号。

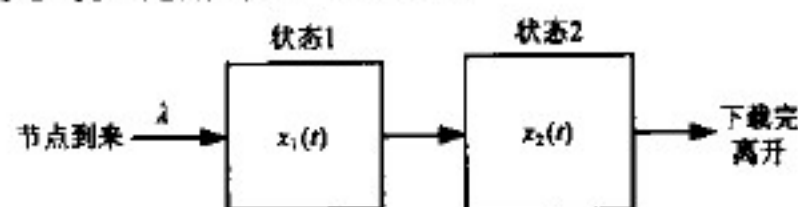


图 4 流体模型示意图

表 1 数学分析使用符号

符号	含义
λ	Torrent 1 中节点到来的速率, 认为服从泊松分布
$x_1(t)$	t 时刻处于状态 1 的节点个数
$x_2(t)$	t 时刻处于状态 2 的节点个数
u_1	leecher 节点上传带宽
u_2	seeder 节点上传带宽
a_1	leecher 拥有的文件块大小
a_2	leecher 缺少的文件块大小

根据文献[10]中流体模型, 得到如下关系:

$$\begin{cases} \frac{dx_1(t)}{dt} = \lambda - \frac{u_1 x_1(t)}{a_1} \\ \frac{dx_2(t)}{dt} = \frac{u_1 x_1(t)}{a_1} - \frac{u_2 x_2(t)}{a_2} \end{cases} \quad (3)$$

当系统处于稳定状态时, 即令式(3)中的 $dx_1(t)/dt$ 和 $dx_2(t)/dt$ 等于 0, 可以求得 x_1 和 x_2 。

由 $x_1 + x_2 = x$ 及 Little 定理 $x = \lambda T_D$, 定义 $\theta = u_1 - u_2$, 求得节点的下载时间 T_D 如下:

$$T_D = \frac{a_1}{u_1} + \frac{a_2}{u_2} = \frac{a_1}{u_1} + \frac{a_2}{u_1 - \theta} \quad (4)$$

由式(4)可知:(1)节点下载时间与节点到来速度 λ 无关, 这与 BitTorrent 有很好的扩展性有关;(2)种子节点上传速度对系统影响很大, 当 u_2 趋向于 0 时, 此时对应着种子节点不愿参与系统, 此时节点的下载时间趋向于无穷大;(3)当 θ 越大时, 此时 T_D 越大, 表明 leecher 节点需要用更多的带宽来换取种子节点的上传, 此时节点下载时间 T_D 增加;(4)搭便车 (free riding) 是 P2P 文件系统中大量存在的现象, 上面对 TFT 算法的修改并不会使 leecher 节点受到搭便车攻击, 因为只有当节点是种子节点时才会获得额外带宽。

3.2 系统中存在种子的概率

定理 1 随着 Torrent 个数增加, 系统中存在种子的概率增加; 当 Torrent 个数 $n \rightarrow \infty$ 时, 系统中存在

种子的概率趋于 1。

证明: 设有 Torrent 0, Torrent 1, ..., Torrent n 共 $n+1$ 个 Torrent, 其中 Torrent 0 缺少种子。对于 Torrent 1, 在 t_1 时刻有 Torrent 0 的种子加入系统要下载文件 1, 下载完文件在 t_2 时刻离开, 在 t_3 时刻有 Torrent 0 的种子加入系统, 在 t_4 时刻离开; 类似地, 对于 Torrent 2, 在 t_5 时刻有种子加入系统, 下载完文件在 t_6 时刻离开, 如图 5 所示。



图 5 种子节点存在概率的计算

设每个 Torrent 节点到来的速率为 λ_i , 对于 Torrent i ($i = 1, 2, \dots, n$), 其存在 Torrent 0 种子的概率为种子的停留时间 (种子下载自己所需的文件时留在系统的时间) 与种子到来间隔的比值, 设 $T_{i1}, T_{i2}, \dots, T_{im}$ 为 Torrent i 中种子节点停留时间, $I_{i1}, I_{i2}, \dots, I_{im}$ 为种子节点的到来间隔, 第 i 个 Torrent 存在 Torrent 0 种子的概率 P_i 为

$$\begin{aligned} P_i &= \frac{\lim_{m \rightarrow \infty} T_{i1} + T_{i2} + \dots + T_{im}}{\lim_{m \rightarrow \infty} I_{i1} + I_{i2} + \dots + I_{im}} \\ &= \frac{\lim_{m \rightarrow \infty} \frac{T_{i1} + T_{i2} + \dots + T_{im}}{m}}{\lim_{m \rightarrow \infty} \frac{I_{i1} + I_{i2} + \dots + I_{im}}{m}} = \bar{T}_i \lambda_i \end{aligned} \quad (5)$$

其中 $\bar{T}_i$ 是节点的平均停留时间。当有 n 个 Torrent 时, 系统中存在种子节点的概率 P 可用下式计算:

$$\begin{aligned} P &= 1 - (1 - P_1) \times (1 - P_2) \cdots \times (1 - P_n) \\ &= 1 - \prod_{i=1}^n (1 - \bar{T}_i \lambda_i) \end{aligned} \quad (6)$$

由式(6)可知, 当 n 非常大时, 即使每个 Torrent 中种子节点到来的速率很小, 停留的时间也很短, 那么系统中存在种子的概率也很大, 这样就能以很大的概率保证系统中种子节点的存在。

4 实验分析

为了考察激励机制的效果, 采用 netlogo^[11] 语言对多 Torrent 进行了建模仿真, 主要实现了如下功能:(1) 完成多个 Torrent 的仿真;(2) 节点下载完某个文件以后依据上面的算法选择是否做种, 当其下

载完所有文件后离开系统; (3) 对 BitTorrent 的协议进行了扩展, 加入了上面提到的协议。共有 4 个实验。实验 1 考察不同参数下的种子响应率, 实验 2 考察 Torrent 个数对存在种子概率的影响, 实验 3 观察下载时间随多给带宽的变化, 实验 4 将本文算法 ABSI 与文献[6]中的 CTFT 算法进行了对比。

实验 1: 种子响应率。实验设置: 共有两个 Torrent, 即 Torrent 1 和 Torrent 2。Torrent 1 的文件大小分别为 100MB, 200MB 和 300 MB, 节点到来服从泊松过程, 速率都为 1 个/s。当 Torrent 1 的节点缺少文件块时向 Torrent 2 的种子发送请求, Torrent 2 的

种子节点运行带宽拍卖算法。种子节点要拍卖的带宽为 100kB/s, 其拍卖底价为 120kB/s, 分为如下两种情况: (1) 等量线性增加: 种子节点和 leecher 采用的算法都是相同的线性增加的方法, 即设置的 a 和 b 都相同 ($a = b = 10\text{kB/s}$); (2) 不等量线性增加: 种子节点和 leecher 都线性增加, 但 $b = 2a$, $a = 10\text{kB/s}$, 在这两种情况下观察种子响应率, 其结果如图 6(a) 所示, 可以得出如下结果: (1) 基于拍卖的算法有较高的种子响应率; (2) 文件大小对种子响应率影响不大; (3) 不等量增加比线性增加种子节点的响应率要大。

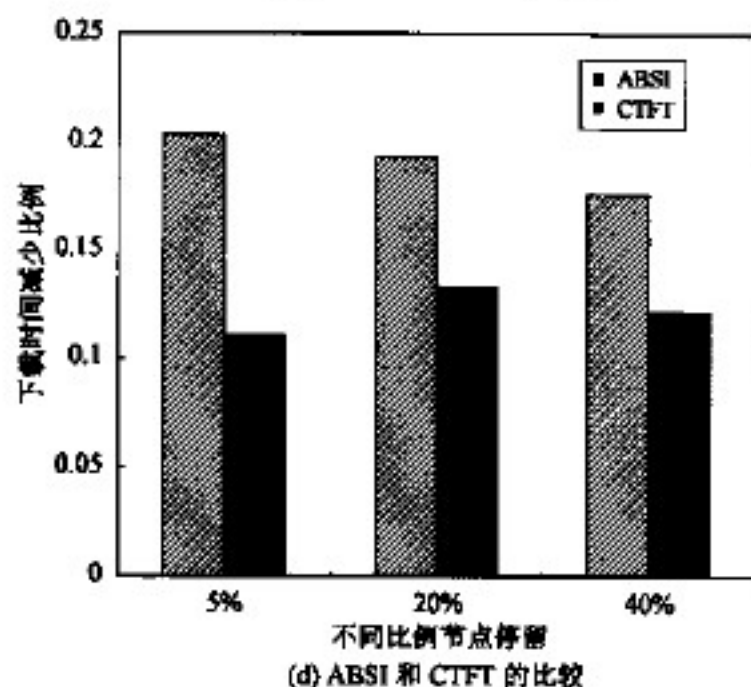
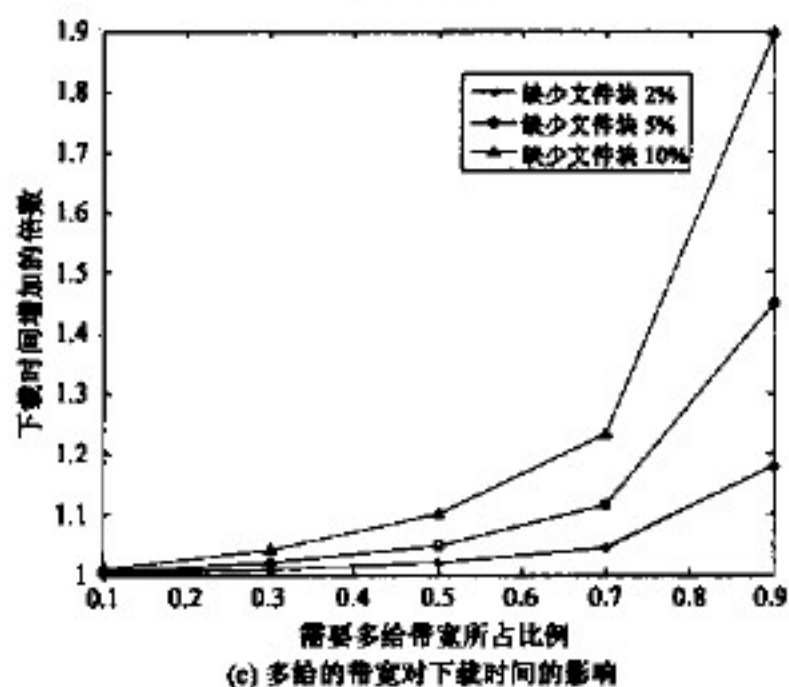
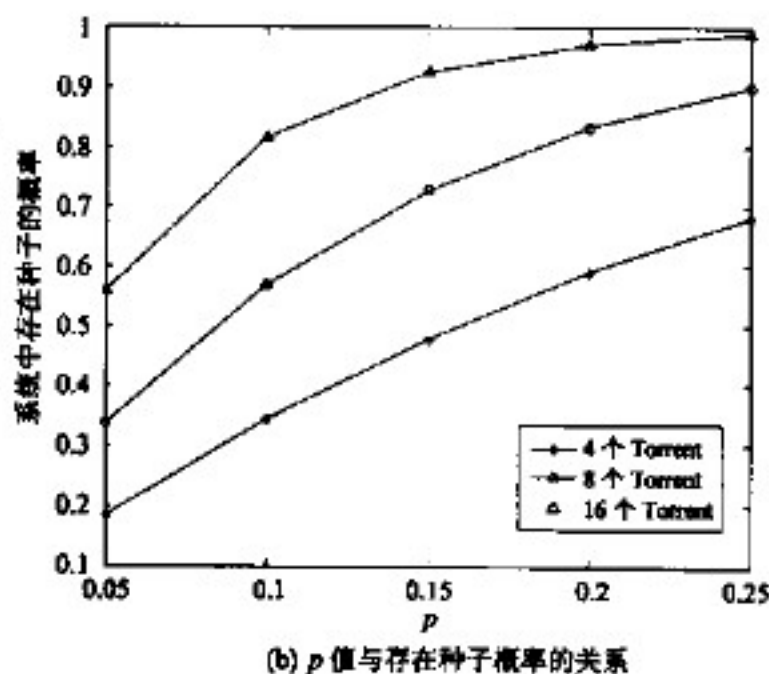
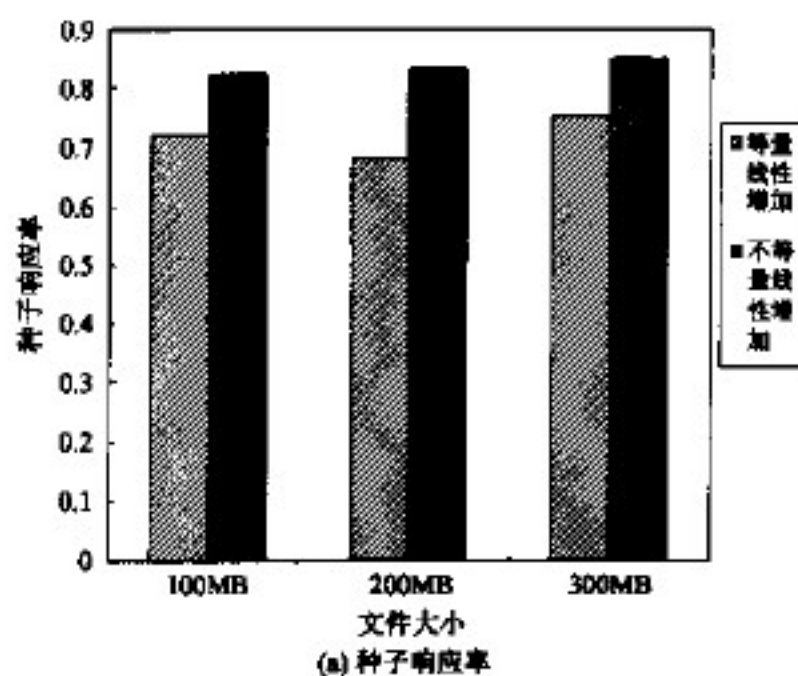


图 6 种子激励机制相关参数分析

实验 2: 存在种子的概率。为了使种子激励算法能够有效地工作, 需要系统中存在种子。本实验主要观察每个 Torrent 存在种子概率 p 值对系统中存在种子的概率的影响, 如图 6(b) 所示; 当 Torrent 个数为 4 时, 若 p 值大于 0.15, 存在种子的概率就接近 0.5; 当 Torrent 的个数为 8 时, 若 p 值大于 0.15, 存在种子的概率就大于 0.7; 当 Torrent 的个数为 16

时, 若 p 值大于 0.15, 存在种子的概率就大于 0.9。这样当 Torrent 个数很大时, 就能以很大的概率保证系统中种子节点的存在。

实验 3: 多给带宽对下载时间的影响。本实验主要观察多给的带宽对节点下载时间的影响, 其结果如图 6(c) 所示。可以看出, 当缺少的文件块所占的比例很少时, 节点下载时间的延长非常少; 当缺少

的文件块所占的比例为 5%, 即使多给的带宽为自身带宽的 70%, 节点下载时间仍不超过原来的 10%; 当缺少的文件块所占的比例为 10%, 多给的带宽为自身带宽的 70%, 节点下载时间仍不超过原来的 20%, 相比种子节点不加入系统时节点的等待时间, 这些延迟非常少, 因此种子节点的加入对于系统的下载时间有重要影响。

实验 4: ABSI 与 CTFT 的比较。本实验主要将本文算法与 CTFT 进行对比, 观察了不同比例种子节点停留对节点下载时间减少比例的影响, 其结果如图 6(d) 所示。可以看出, 节点停留时间很小时, ABSI 算法比 CTFT 算法提高的比例更明显。这主要是因为 ABSI 算法是基于拍卖的思想, 当种子节点越少时, 种子节点对于提高下载时间越重要, 而拍卖算法恰好表明了种子节点的重要性。同时 ABSI 算法是分布式的算法, 每次拍卖都在节点本地运行, 因此其算法开销并不会比 CTFT 算法高。

5 结 论

从以上分析可以得出如下结论: (1) TFT 策略并不适应于种子节点激励, 因此需要设计有效的激励算法来激励种子节点; (2) 通过基于拍卖的方法多给种子节点带宽, 可以激励种子节点参与系统。基于拍卖的方式相比于 TFT 方式而言, 除了考虑最大化下载带宽之外, 更多地考虑了种子节点的重要性, 因此能有效激励种子节点参与系统。

参考文献

- [1] Cohen B. Incentives Build Robustness in BitTorrent. In: Proceedings of the Workshop on Economics of Peer-to-

- Peer Systems 2003, Berkeley, USA, 2003. 251-260
- [2] Ipoque. <http://www.ipoque.com/resources/internet-studies/internet-study-2007>; Ipoque, 2007
- [3] Kaune S, Tyson G, Pussep K, et al. The Seeder Promotion Problem: Measurements, Analysis and Solution Space. In: Proceedings of the International Conference on Computer Communications and Networks, Zurich, Switzerland, 2010. 1-8
- [4] Levin D, LaCurtis K, Spring N, et al. BitTorrent is an Auction: Analyzing and Improving BitTorrent's Incentives. In: Proceedings of the ACM Special Interest Group on Data Communication, Seattle, USA, 2008. 243-254
- [5] Guo L, Chen S, Xiao Z, et al. Measurements, Analysis, and Modeling of Bittorrent-like Systems. In: Proceedings of the Internet Measurement Conference, Berkeley, USA, 2005. 35-48
- [6] Yang Y, Chow Alix LH, Golubchik L. Multi-torrent: a performance study. In: Proceedings of 16th Annual Meeting of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, Baltimore, USA, 2008. 1-8
- [7] Locher T, Moor P, Schmid S, et al. Free Riding in BitTorrent is Cheap. In: Proceedings of the 5th Workshop on Hot Topics in Networks (HotNets), Irvine, USA, 2006. 85-90
- [8] Piatek M, Isdal T, Anderson T, et al. Do incentives build robustness in bittorrent? In: Proceedings of the 4th Symposium on Networked Systems Design and Implementation, Massachusetts, USA, 2007. 1-14
- [9] 余一娇, 金海. 对等网络中的搭便车行为分析与抑制机制综述. 计算机学报, 2008, 31(1): 1-15
- [10] Qiu D Y, Srikanth R. Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks. In: Proceedings of the ACM Special Interest Group on Data Communication, Portland, USA, 2004. 367-378
- [11] Netlogo. <http://ccl.northwestern.edu/netlogo/>

An auction based seeder incentive algorithm for multi-torrent file systems

Li Xiaoyi, Li Zhijun, Jiang Shouxu

(School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001)

Abstract

This paper investigates the problem that the current multi-torrent file systems lack incentives for seeder nodes, and suggests that practical multi-torrent file systems are very common, thus a good environment for seeder incentive is created; The tit for tat (TFT) incentive mechanism aims to maximize the download speed without considering the importance of seeder, so it can not be used in seeder incentive. On the basis of this, it presents an auction based seeder incentive algorithm (ABSI); first the seeder runs an auction for its bandwidth, then the leechers send their bids to the seeder, and the seeder decides how to allocate the bandwidth. The effectiveness of the ABSI and the download time of the system are also analyzed. The experiments showed the benefits of the approach.

Key words: multi-torrent, seeder, incentive mechanism, auction