

The Application of Network Coding in P2P Streammeida Technology

Ping LI

Dept. name of organization, Chengdu Neusoft Institute of Information, Chengdu, P.R.China

Email: address, li-p@neusoft.com

Abstract: As a new technology, network coding has very wide application in many fields. This paper proposes a P2P streaming media transfer algorithm using network coding technology, which depart the data layers and using the network coding technology to encode the data. Receiver port controls the transmission of data layers according the bandwidth situation and takes the strategy of playing while transmitting to improve the quality and the continuity of streaming media data. Test results obtained by simulation shows that when the seeds leave the system early and Peers arrive and depart frequently, it get a directly perceived of network coding which improve the downloads Peed and the robustness.

Keyword: P2P; Network coding; streammeida

网络编码在 P2P 流媒体技术的应用

李 平

成都东软信息技术职业学院, 成都, 中国, 611844

Email: address, li-p@neusoft.com

摘 要: 网络编码作为一种新的技术在已经应用于众多领域。本文提出了一种基于网络编码技术的 P2P 流媒体传输算法, 将数据分层并且利用网络编码技术压缩后进行传输。接收端根据带宽情况控制数据层的传输, 并且利用流媒体分层传输特点采取边播放边传输的策略, 提高了数据的播放质量和连续性。通过仿真结果分析表明, 在种子节点在线时间很短以及下载节点用户动态进出系统的情况下, 引进网络编码后, P2P 系统的性能有很大的改善, 提高了节点的下载速度以及增强了系统的鲁棒性。

关键词: P2P; 网络编码; 流媒体

1 引言

网络编码(NC)^[1,2,3]理论是 21 世纪初在信息论领域中的一个重要突破。传统的通信网络传送数据的方式是存储转发, 即除了数据的发送节点和接收节点以外的节点只负责路由, 而不对数据内容做任何处理, 中间节点扮演着转发器的角色。长期以来, 人们普遍认为在中间节点上对传输的数据进行加工不会有任何收益。然而 R Ahlswede 等人于 2000 年提出的网络编码理论彻底推翻了这种传统观点。网络编码技术可以达到多播路由传输的最大流界^[4], 提高了信息的传输效率, 目前已经成为网络数据传输的主流技术。

P2P 流媒体应用由于其高效的传输性和良好的健壮性受到广泛的关注。然而流媒体数据相比于普通数据来说具有稳定性要求, 如何保证流媒体数据的连续稳定传

输和高质量播放成为了目前的研究热点。将网络编码技术和 P2P 技术相结合可以提高下载速度和成功率, 而将其应用于流媒体技术, 可以提高 Peer 节点实际的下载速率以及对网络资源的利用率, 节省 CPU 占用率和内存空间, 提高 Peer 节点播放质量。

目前应用于 P2P 流媒体系统的编码技术的相关研究还没有发展成熟。文献^[5]给出了网络编码于一般网络中的具体实现模型; 文献^[6]给出了网络编码于无线网络中的具体实现模型。本文提出了一种基于网络编码技术的 P2P 流媒体传输模型。该算法利用对分层的数据包进行网络编码后传输, 在接收端根据带宽和解码后的数据信息来控制后续数据层的传输。同时为了减少播放的等待时间, 采取边播放边传输的策略, 有效的保证了较低的延迟以及流媒体数据的播放质量。仿真实验表明: 该算法在播放质量等性能上优于传统的非网络编码技术。

2 系统模型

在基于网络编码的 P2P 流媒体调度模型中^[7,8]，一个流媒体文件被分成多个数据层，并安装播放的顺序对它们从小到大编号。将整个数据层分成两部分，分别为优先层和后续层。假设流媒体文件 B 分成 m 个数据层，则得到数据层集合 $B=\{b_1,b_2,\dots,b_m\}$ ，则优先层 $M=\{b_1,b_2,\dots,b_n\}$ ，后续层 $F=\{b_{n+1},\dots,b_m\}$ 。其中参数 n 根据接收节点带宽以及播放速度等条件确定。每次编码随机在 $[1,m]$ 中选取一个权值 w，然后集合 $B=\{b_1,b_2,\dots,b_m\}$ 中随机选择 w 个数据层进行编码操作，得到一个编码结果，循环进行 m 次此操作，则会得到编码后的数据包集合 $A=\{a_1,a_2,\dots,a_m\}$ 。

为了能够降低缓冲延迟，我们采取边传输边播放的策略。在编码时根据网络带宽及媒体播放速率确定优先层数据的大小，即确定参数 n 的值。假设网络的可用带宽为 B，媒体的播放速率为 R，则需要满足剩余数据层的传输时间不大于优先层数据的播放时间。数据总大小为 L，则优先层数据和后续层数据的大小分别为 $(L/m) \times n$ ， $(L/m) \times (m-n)$ 。则满足：

$$\frac{\frac{L}{m} \times n}{R} \geq \frac{\frac{L}{m} \times (m-n)}{B}$$

即

$$n \geq \frac{m \times R}{B + R} \quad (1)$$

确定了参数 n 的大小后，就可以进行数据传输了。在数据接收端，根据解压出来的数据层进行判断，如果是优先层且该数据层还未被传输过，则送入缓冲区等待播放。

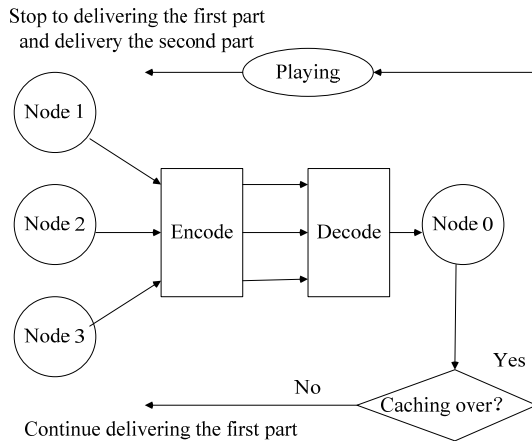


Figure 1. System model
图 1 系统模型图

如果该数据层已经存在则丢弃。等到所有优先层数据传输完毕，则通知源节点停止传输优先层数据，开始播放并且源节点开始传输数据。系统模型图见图 1。

3 算法描述

3.1 编码/解码策略

基于网络编码的 P2P 流媒体传输要求每个 Peer 对数据进行编码，其基本原理可见图 2。

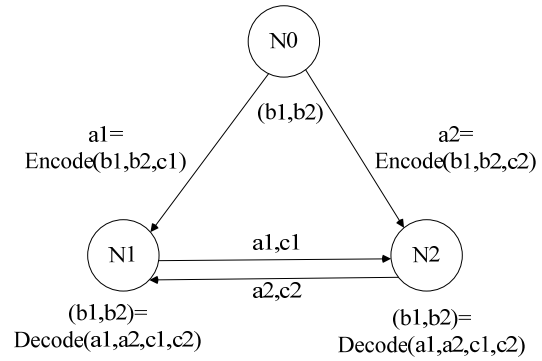


Figure 2. P2P network coding theory

图 2 P2P 网络编码原理

在图 2 中，N0 为数据发布源节点，N1 和 N2 是数据请求节点。c1 和 c2 分别为 N0 与 N1 及 N2 之间的编码参数。N1 和 N2 同时向 N0 请求下载一个数据。N0 用编码操作 $Encode(b_1,b_2,c_1)$ 产生了数据包 a1 并发送给 N1，同理用 $Encode(b_1,b_2,c_2)$ 产生了数据包 a2 并发送给 N2。由于网络编码产生的数据包相同性较小，所以 N1 和 N2 之间可以相互交换彼此的数据块和编码参数。当 N1 和 N2 获得 a1、a2、c1、c2 这些必需的数据后就可以进行解码操作 $Decode(a_1,a_2,c_1,c_2)$ ，分别解码出数据块 b1 和 b2。

在介绍本文的调度算法之前，首先来了解基于网络编码技术的编码和解码原理。

在满足长度不变、运算可逆的前提下，编码函数可被任意设定。本文我们使用加洛瓦有限域 (Galois field, GF) 上的随机现行编码函数 (random linear code, RLC)。其中，GF 是一个非常合适满足数据大小不变而且保证运算可逆的运算域。RLC 的特点在于较小的运算消耗和重编码。

定义 1 重编码是指对元素进行任意次编码操作后所生成的元素，其形式不变。

RLC 编码操作可以看做对一组由数据块组成的向量左乘以一个随机矩阵，如图 3 所示：

$$\begin{bmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \dots & \dots & \dots & \dots \\ r_{m1} & r_{m2} & \dots & r_{mm} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_m \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_m \end{bmatrix}$$

Figure 3. RLC coding operation
图 3. RLC 编码操作

图 3 可简写为 $R \times B = A$ 。如果 $B = \{b_1, b_2, \dots, b_m\}$ 是所有原始数据组成的向量, $R = (r_{ij})$ 是在 $GF(2^k)$ 上的一个随机矩阵, 则根据矩阵的运算性质, 我们有:

$$a_i = r_{i1} \times b_1 + r_{i2} \times b_2 + \dots + r_{im} \times b_m \quad (2)$$

通常, 原始数据块是通过对文件分块而直接得到的, 所以 a_i 可以用 $R_i (r_{i1}, r_{i2}, \dots, r_{im})$ 组成的数据串来唯一标识。这样我们有如下定义:

定义 2 编码参数值 c 为 R 矩阵的一次编码向量 $R1$ 。

同理, 我们可以进行二次编码, 即重复上面的步骤 $R1 \times A1 = A2$ 。

编码和解码虽然本质上是互逆的过程, 但是它们之间仍然存在的差别: 编码不需要所有 m 个数据层全部参与, 但是解码通常需要所有的 m 个数据包全部参与。通过对 RLC 编码函数分析可知, 假设我们获得 m 个数据包 A_m 和相应的编码参数 C_m , 如果存在矩阵 R^{-1} , 使得满足:

$$R^{-1} \times R = I \quad (3)$$

其中 I 是单位矩阵, 那么只要通过 A_m 和相应的 R 就能解码出原始数据。

3.2 算法描述

分层编码后的数据在优先层和后续层分别进行网络编码, 由于基于网络编码的 P2P 系统各个发送端的数据是彼此独立的, 所以在接收端看来, 这些网络编码后的数据就如同来自一个发送端一样。因此接收端不需要协调每个发送端的工作, 而只需要根据接收端的接受速率统一控制发送端的编码质量。

本文利用分层编码和网络编码技术相结合进行数据传输。数据请求节点并不对每个数据包都发送请求, 也不对发送节点所拥有的数据进行询问。首先确定优先层的数据块个数 $n (0 < n < m)$, 然后要求发送节点发送编码后的数据包, 同时对发送来的数据包进行解码。若发现优先层的数据已经解码完毕, 则开始播放数据并且通知发送节点停止发送优先层数据同时继续

发送后续层的数据包, 同时解码送入缓冲区。具体的代码描述如下:

```
Algorithm1 SeverPeer(B,A,C,n,m,w,sig)
// B={b1,b2,...,bm},all data blocks
// A={a1,a2,...,am},datas after coding
// C={c1,c2,...,cm},Parameters of NC
//sig is a Whole variable
// w is a random variable
// bj[w] means that there are w bj
1. if sig=0 then//优先层数据已经发送完毕
Begin
1) stop to delivery the data of Priority layer;
2) for i:=1 to m do
Begin
a) ai:=Encode(bj[w],ci)
//random j ∈ [n+1,m]
b) push ai to A;
End for;
3) wait for request;
End if;
2. if sig=1 do
Begin
1) for i:=1 to m do
Begin
a) ai:=Encode(bj[w],ci);
//random j ∈ [1,m]
b) push ai to A;
End for;
2) wait for request;
End if;
End.
```

```
Algorithm2 RequesetPeer(B,A,C,N,M,sig)
Begin
1. for i:=1 to m do
Begin
a) bi:=Decode(a1,a2,...,am,c1,c2,...,cn);
b) push bi to B;
End for;
2. for j:=1 to n do
Begin
if bj ∈ B and 1<=j<=n then
count:=count+1;
End for;
3. if count=n then
sig:=0;
else sig:=1;
4. if all data decoded over then
Sig:=3;
End.
```

本系统采用网状的拓扑结构。由于某些节点会突然从系统中退出, 索引服务器为了把已退出的节点及

时从列表中清除，要求所有注册节点每隔 T 时间发送心跳消息，若在一定时间内系统未收到节点的心跳消息，则会认为该节点已经退出系统。

4 仿真实验及结果分析

本文的仿真实验中，我们利用 NS2 和 GT-ITM 建立 P2P 网络拓扑。对于每个数据，分层后共有 200 个数据层，其中优先层有 150 个数据层，后续层有 50 个数据层。所有数据层大小均为 5K，播放缓冲大小为 15s。网络由 100 个节点构成，心跳监测时间 $T=15s$ 。网络解码缓冲大小为 6s，发送节点的发送速率为 200 $packes/s$ 。

假设平均一个节点在一个时间段内得到的数据块的数目为 $Q1$ ，而播放的数据块的数量为 $Q2$ 。我们令播放质量 $Q=Q1/Q2$ 。

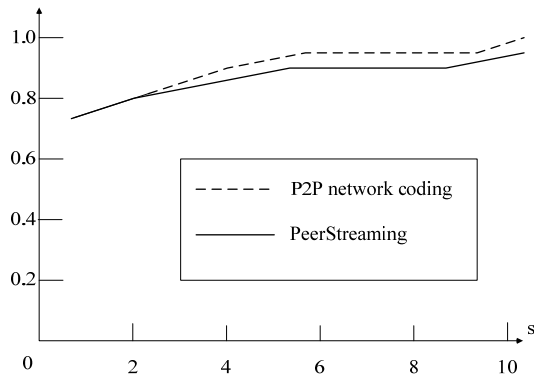


Figure 4. Play Quality Comparison Chart
图 4 播放质量对比图

图 4 反映了在动态网络带宽的情况下，随着时间的增加，两类算法的播放质量对比情况。从结果可以看出，基于网络编码的 P2P 流媒体系统在相等条件下具有更好的数据服务质量。

图 5 反映了两类算法的 CPU 占用率情况。对于网络编码系统而言，所需 CPU 占用率和 Peer 的上传速率成正比例。由结果可以看出，基于网络编码的 P2P 流媒体传输系统能够有效的节省内存空间，从而降低 CPU 占用率，随着 Peer 上传速率的不断增加，这种优势更加明显。

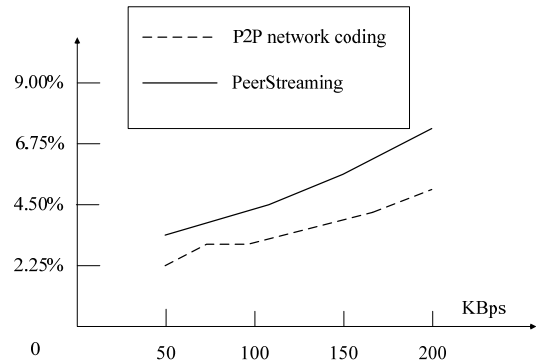


Figure 4. CPU
图 5. CPU

5 结语

本文提出了一种基于网络编码技术的 P2P 流媒体数据传输算法。分析了网络编码技术应用于 P2P 流媒体领域的巨大优势，由于发送节点的数据彼此独立，因此并不需要对传输的任务进行分配，大大简化了对传输的控制过程，并且由于发送数据的独立性等特点使得基于网络编码的传输是具有较强的动态适应性。仿真实验表明，该算法在提高数据的播放质量和降低 CPU 占用率两方面具有较高的性能。

References (参考文献)

- [1] Wu Yun nan, Kung Sun Yuan, Reduced complexity network coding for multicasting over adhoc networks[A]. Proceedings of IEEE International Conference on Acoustics, Speech, and Signal processing[C], 2005, pp:501-504.
- [2] Chen Y, Li J. Wireless Diversity through Network Coding. Proceedings of IEEE Wireless Communications and Networking Conference[C], 2006, pp:1681-1688.
- [3] AHLWEDE R, CAI N, LI S R. Network information flow [J]. IEEE Transactions Information Theory, 2000, Vol.46 (4): 1204-1216.
- [4] LI S R, YEUNG R W, CAI N. Linear network coding. IEEE Transaction Information Theory[J], 2003, Vol.49(2): 371-381.
- [5] WIDMER J, FRAGOULI C, LE Boudec JY. Low-complexity energy-efficient broadcasting in wireless ad-hoc networks using network coding [EB/OL]. <http://www.netcod.org/final.pdf>
- [6] WEI Yi-fei, GUO Xiang-li. High throughput route selection in multi-rate wireless mesh networks[J]. The Journal of China Universities of Posts and Telecommunications, 2008, Vol.15(3), pp:13-18.
- [7] P.A.Chou, Ywu, and K.Jain, Network coding for the Internet, IEEE Communication The Workshop, Italy, May, 2003.
- [8] C.Gkant sidis and P.Rodri. Network coding for large scale content distribution. IEEE Incofom, Miami, FL, 2005.