



Y1751907

浙江大学研究生学位论文独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得 浙江大学 或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名： 解群 签字日期： 2010 年 8 月 12 日

学位论文版权使用授权书

本学位论文作者完全了解 浙江大学 有权保留并向国家有关部门或机构送交本论文的复印件和磁盘，允许论文被查阅和借阅。本人授权 浙江大学 可以将学位论文的全部或部分内容编入有关数据库进行检索和传播，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

(保密的学位论文在解密后适用本授权书)

学位论文作者签名： 解群 导师签名： 谢振云

签字日期： 2010 年 8 月 12 日

签字日期： 2010 年 8 月 12 日

致 谢

衷心感谢我的导师谢磊副教授。本课题的选题、设计以及最终论文的撰写、校审和定稿都得到导师的精心指导。同时，在研究生学习的四年时间里，导师对我授业解惑，关怀备至。导师渊博的知识、敏锐的洞察力和高瞻远瞩的科研思维将是我一生追求的目标、学习的榜样。

感谢陈惠芳副教授的指导。她为人随和热情，治学严谨细心。在论文的写作方面，她一直以“专业标准”严格要求，论文的反复修改和润色，陈老师始终认真负责地给予我细致地指导。正是陈老师的无私帮助，我的毕业论文才能够顺利完成。

特别感谢王政师兄在论文完成过程中给予的热心帮助。正在撰写博士研究生毕业论文的王政师兄，在百忙之中抽出时间帮助我搜集文献资料，帮助我理清论文写作思路，对我的论文提出了许多宝贵的意见和建议。

最后感谢我的父母在我求学生涯中给予的无私奉献与关爱。

感谢所有帮助过我、关心过我的人！

摘 要

网络编码是一种新颖的网络传输技术。与传统网络中只具备存贮转发功能的节点不同，中间节点可以将接收到的数据进行编码。作为通信网络技术中重要的研究内容，网络编码能够在带宽受限的通信网络中提供更大的传输速率和更好的传输性能。此外，运用网络编码能够更充分的利用网络资源，达到在许多情况下传统网络传输无法实现的组播速率，甚至达到组播传输速率的上限值。

但是，运用网络编码，首先要求中间节点是具备网络编码功能的网络转发设备。更新硬件设备往往需要大量的资金投入，网络编码的应用受限于硬件设备。如何在应用网络编码技术的同时，尽可能的减少付出的各种开销，是网络编码在实际应用中所要解决的重要问题之一。

本文首先详细介绍了网络编码的基本理论，并对代数网络编码做了深入的分析，给出了单发送节点多播模型，并将其进行了推广。在此基础上，对网络编码节点数量受限的网络传输性能进行了深入的研究，提出了一种简单通用的组播传输网络的生成方法。通过仿真，对使用该方法前后的性能做出了比较，从而证明了此方法达到了需要的效果。

关键词：网络编码，组播传输，骨干节点

ABSTRACT

Network Coding is a novel network technology introduced in recent years. With network coding, the intermediate nodes are allowed to encode the received messages rather than only storing and forwarding messages. As an important research area in communication and network, network coding can improve transmission rate and transmission performance in bandwidth constrained channels. Furthermore, by using network coding, network resources are fully utilized, and the transmission rate is even reach to the upper limit of the multicast capacity of the network.

However, using network coding requires that intermediate nodes equipped the hardware with network coding capability. Hence, the usage of network coding is restrained to the device of the network hardware. How to apply network coding with low cost is one of critical problems in the application of network coding.

This thesis first introduces the basic theory of network coding and algebraic network coding. Then, the model of a single source node multicast is given, and the extended model is also introduced. The transmission performance with restricted number of coding nodes is investigated. Finally, an universal multicast transmission network generation method is proposed. The simulation results validate the proposed method.

KEY WORDS: Network Coding, multicast, key node

目 录

致 谢	I
摘 要	II
ABSTRACT	III
目 录	I
图表索引	III
第 1 章 绪论	1
1.1 引言.....	1
1.2 网络编码技术概述.....	1
1.3 网络编码的特点.....	3
1.4 网络编码的应用.....	6
1.5 本文的主要工作和篇章结构.....	6
第 2 章 网络编码概述.....	8
2.1 组播.....	8
2.1.1 IP 组播.....	8
2.1.2 应用层组播	8
2.2 网络编码理论研究.....	9
2.3 网络编码模型.....	10
2.4 线性代数编码数学模型的建立.....	11
2.5 数学模型的推广.....	13
2.5.1 单发送节点多播模型	13
2.5.2 多发送节点多播模型	13
2.5.3 不同接收速率多播模型	15
2.6 随机网络编码.....	16
2.7 网络编码应用方面的研究.....	16
2.7.1 P2P 文件传输.....	16
2.7.2 P2P 视频流传输.....	18
2.7.3 视频编码技术与网络编码	19
2.8 编码节点数量受限的网络编码.....	22

2.9 本章小结.....	22
第3章 编码节点数量受限的网络编码传输性能研究.....	23
3.1 简化网络模型.....	23
3.2 设计过程.....	23
3.3 性能仿真与分析.....	24
3.3.1 全部节点随机编码.....	24
3.3.2 网络编码节点数量受限的网络编码.....	26
3.3.3 全部节点无编码.....	27
3.4 仿真结果.....	29
3.5 本章小结.....	34
第4章 组播链路构建的研究.....	35
4.1 设计准则.....	35
4.2 网络简化模型及相关概念.....	36
4.3 接收节点的路径集的构建.....	37
4.3.1 发送节点 S 到各个接收节点 T_i 的路径发现.....	37
4.3.2 构建接收节点 T_i 的可用路径集 $\text{Path}(T_i)$	38
4.3.3 简化接收节点 T_i 的可用路径集 $\text{Path}(T_i)$	38
4.4 使用网络编码的组播链路构建方法.....	40
4.4.1 第一接收节点路径选择.....	40
4.4.2 其他接收节点路径选择.....	42
4.5 无网络编码的组播链路构建.....	43
4.6 仿真结果与分析.....	44
4.7 本章小结.....	45
第5章 总结与展望.....	46
参考文献.....	48
作者简介及攻读硕士学位期间的研究成果.....	51

图表索引

图 1 网络编码举例.....	2
图 2 网络编码在无线中的应用.....	4
图 3 网络编码的特性.....	5
图 4 单发送节点的多播模型.....	14
图 5 多发送节点多播模型.....	14
图 6 不同接收速率多播模型.....	15
图 7 P2P 文件传输模式示意图.....	17
图 8 采用网络编码的 P2P 文件传输模式示意图.....	18
图 9 分层视频编码与网络编码结合.....	20
图 10 分层视频编码与网络编码结合 2.....	21
图 11 编码节点与本地编码系数.....	25
图 12 不可编码节点.....	26
图 13 可编码节点转发数据包.....	27
图 14 部分网络编码性能研究仿真流程图.....	28
图 15 Graph50-1 仿真结果.....	30
图 16 Graph50-2 仿真结果.....	30
图 17 Graph50-3 仿真结果.....	31
图 18 Graph50-4 仿真结果.....	32
图 19 Graph50-5 仿真结果.....	33
图 20 Graph50-6 仿真结果.....	33
图 21 数据包 FP 结构示意图.....	37
图 22 发送节点到接收节点的可用路径发现算法.....	39
图 23 发送节点到第一接收节点的可用路径发现算法.....	42
图 24 构建发送节点到其他接收节点之间的传输网算法的示意图.....	44

第 1 章 绪论

网络编码 (Network Coding), 其特点是路由交换技术和信息编码技术的融合, 其核心思想是网络中的各个节点在收到的各路径的信息后, 进行线性或非线性的处理, 然后转发给其他节点, 中间节点扮演编码器或信号处理器的作用。根据图论中最大流最小割定理, 数据发送节点和接收节点不能超过最大流 (或最小割), 如果使用传统组播的方式, 一般不能达到的上限。而通过使用网络编码, 组播传输可达到最大流, 从而提高了信息传输的效率。

经过十年的网络编码的理论研究以及应用, 网络编码的应用得到了极大的扩展, 覆盖了信息论、编码理论、网络交换、无线通信、网络安全以及许多其他学科领域。

1.1 引言

21 世纪, 全球进入了信息时代, 信息的产生和传递非常迅速。网络已经影响了社会生活的方方面面。

随着各种网络应用以及数据业务的不断发展, 不论是有线网络的带宽资源还是无线网络的频带资源, 变得越来越紧张。迅猛发展的网络应用及业务和有限带宽资源之间的矛盾越来越突出。在此情况下, 如何高效利用这些有限的通信资源, 成为通信技术发展的焦点之一。研究表明, 网络编码能够充分利用网络资源, 在不增加系统带宽情况下, 可以有效对抗网络的动态变化, 大大提高传输网络的带宽利用率, 是大规模信息组播传输的优选技术之一, 也是缓解该矛盾的有效办法。

1.2 网络编码技术概述

网络编码是网络通信领域的一项前沿科技。在网络编码出现以前, 有线通信网络和无线通信网络都是采用存储转发的机制 (store-and-forward) 来实现数据的传输, 即信息传输都是由发送节点通过中间节点, 以存储转发的方式传送到接收

节点。网络中的节点除了复制接收到的数据以外，一般来说，并不需要做任何数据处理。这样信息传递的方式类似于邮递系统，是最自然的通信方式。但是，在信息量急剧增大的情况下，如实时视频传输，容易产生网络 congestion 的问题。

举个简单的例子，如图 1 所示，图中的圆表示节点，线段表示单位链路（单位时间传送单位数据包）。其中， S 表示发送节点， R_1 、 R_2 、 R_3 、 R_4 表示中间节点， T_1 、 T_2 表示接收节点。根据最大流最小割原理，节点 S 与节点 T_1 、节点 S 与节点 T_2 之间的最小割为 2，因此理论上可以一个单位时间，节点 S 可以向节点 T_1 或节点 T_2 传输 2 个单位数据包。

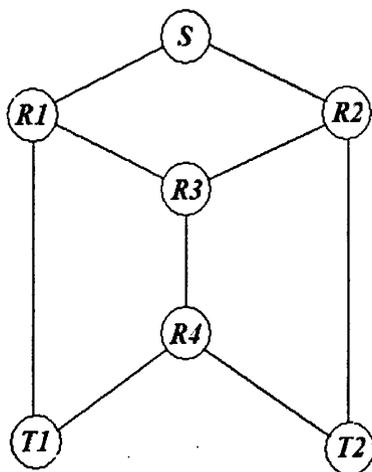


图 1 网络编码举例

在一个单位时间内，节点 S 向节点 T_1 传送数据包 x ，节点 S 向节点 T_2 传送数据包 y 。可见，此时节点 S 与节点 T_1 之间、节点 S 与节点 T_2 之间的数据流的速率为 1。因此，网络资源没有充分利用。

在一个单位时间内，节点 S 向节点 T_1 、节点 T_2 同时传送数据包 x 和数据包 y 。由于中间节点 R_3 与中间节点 R_4 之间的链路具有单位带宽，在单位时间只能传递单位数据包，所以节点 R_3 只能先传输数据包 x 再传输数据包 y ，或者先传输数据包 y 再传输数据包 x 。如果节点 S 需要在一定时间内向节点 T_1 、节点 T_2 发送 2 个数据包，节点 R_3 将成为瓶颈。当节点 R_3 的缓冲区溢出，必然发生丢包。在这种情况下，使用传统组播的方式，显然不能达到的网络传输的上限。除了增大节点 R_3 与节点 R_4 之间链路的信道容量外，似乎没有其他的解决办法。

如果使用网络编码，上面的问题就能得到解决。节点 R_3 将接收到的数据包 x 和数据包 y 进行数据处理——异或操作。节点 R_4 将接收到的处理过的数据包 $x \oplus y$ ，分别发送给节点 T_1 和节点 T_2 。节点 T_1 和节点 T_2 分别对自己接收到的数据包处理，可以很方便地得到节点 S 需要传送的数据包。此时，传输速率达到理论上的最大流。因此，传统传输方式无法解决的问题，网络编码可以很好的解决。

网络编码概念的提出，改变了人们的传统观念。节点所做的不仅仅是复制转发信息，而且可以对信息进行编解码。这里说的编解码，不同于信源信道编码。那什么是网络编码呢？简而言之，就是网络中的节点将接收到的信息按照一定的函数关系进行编码，并传送函数的输出。在传统传输方式中，节点的输出是该节点之前的输入信息。从这个意义上说，传统传输方式只是网络编码的一个简化模型，即其编码函数为 $f(x) = x$ 的对等函数。做个简单的比喻，在我们生活中的邮件运输系统中，每份邮件（包裹）都是独立的个体，收发的整个过程包裹都没有变化，他们在共同的道路上独立运输。而采用网络编码后，在道路上运输的包裹不再是原始的包裹，是所有包裹的碎片组合。

1.3 网络编码的特点

网络编码和传统的传输方式相比，提高了能量的效率，减小了时延，节约了无线带宽。这里简单介绍网络编码的特点。

1. 提高网络的吞吐量

如果不采用网络编码，而是使用传统的路由方式，那么组播速率最大化的问题就等同于 Steiner 树问题，是一个 NP 问题。如果用 C 表示网络的信源到接收节点的最大流的最小值，用 T 表示接收节点的集合， $|T|$ 表示集合 T 中元素的个数，那么已知的解决 Steiner 树问题的算法只能保证 $C/|T|$ 速率。相比之下，使用了网络编码的网络理论上可以以速率 C 进行传输。

如果采用网络编码，那么实际组播传输的速率就可能到达组播速率的上限。Ahlswede 提出的蝴蝶网^[1]就是表达网络编码概念最著名的例子。从 1.2 节的分析我们很清楚的看到网络编码可以带来更大的吞吐量，传输速率的提高成为可能。

2. 带宽利用率的提高

同样是蝴蝶网的例子，相同的信息量，采用网络编码后所需要的带宽比传统传输方式需要的带宽少。

3. 传输时间的减少

单位时间传输的信息量大，则所需要的传输时间就少。

4. 能耗的降低

这个是针对无线传输而言。电池一直是移动设备设计的制约因素，网络编码可以降低移动设备的功耗。如图 2 所示，A、C 通过无线网络交换信息。传统方式下，需要 4 步完成的信息交换，如果采用网络编码则需要 3 步完成。发射信号次数的减少，带来的是能耗的下降。

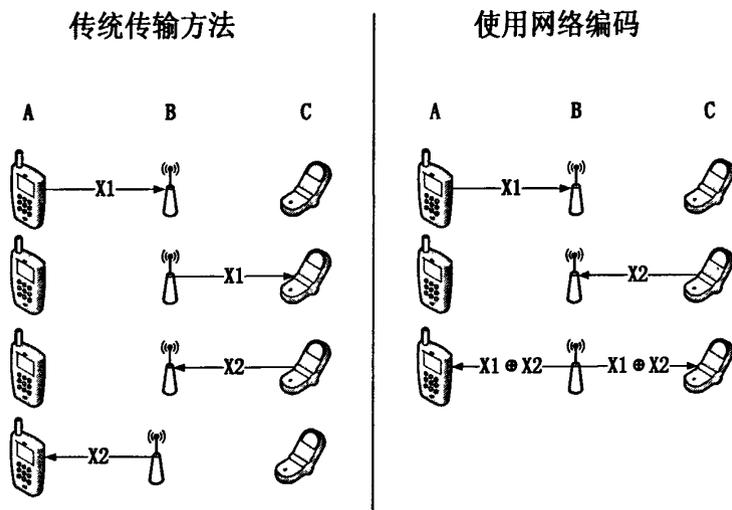


图 2 网络编码在无线中的应用

5. 网络编码的普适性

发送节点传输信息的最大速率只和发送节点到接收节点的最大流值有关，在构造网络编码时不需要考虑发送节点的具体位置。

6. 网络的鲁棒性提高

只要发送节点与接收节点之间的最小割能够满足信息流的传递，那么接收节点就能够接收到足够的信息进行译码，即使网络中的某些节点或链路失效，仍然可以正常通信。

7. 传输的可靠性增加

即使网络中的数个链路发生错误，接收节点仍可以通过编码纠正错误。

8. 提高网络安全性能

Cai 等人最先研究了单发送节点有向无环网络中数据安全多播问题^[20]，给出了搭线窃听的网络通信模型，并且构造了提升网络安全的网络编码。模型指出，当窃听者只能偷听某给定链路，那么窃听者无法恢复出发送节点发送的原始信息。如图 3 所示，从发送节点 S 发出的信息中， m 是消息本身，而 k 是为了保证信息安全传输的随机数。如果单某一个时间内，只允许窃听者偷听整个网络的一条链路，这样接收节点 T_1 和 T_2 能够安全接收到发送节点发送来的消息 m 。

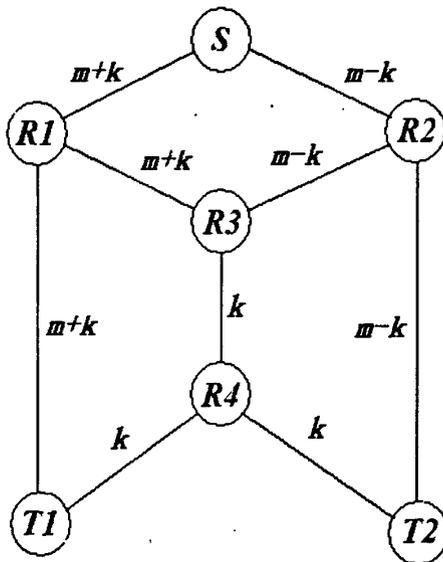


图 3 网络编码的特性

1.4 网络编码的应用

正如上面介绍的，网络编码能带来很多的好处。例如其组播传输速率达到网络容量的上限，节省网络带宽资源消耗，均衡网络负载，并且可以提高网络鲁棒性。网络编码的理论创新具有普遍意义，应用前景十分广阔。近年来，网络编码的理论及应用在信息论、编码理论、网络交换、无线通信网络、网络安全以及许多其他学科领域，都受到人们的普遍关注。其中，网络编码与对等网络 P2P 的结合^{[5], [13-15], [26]}，以及网络编码在无线通信上的应用方面的研究成果不断涌现^[8]。网络编码的新应用不断涌现。

1.5 本文的主要工作和篇章结构

在实际网络中，大部分路由器只具备储存转发的路由功能。如果要应用网络编码，那就必须对网络的基础设施进行换代升级。而大型交换设备的升级必然需要大量的资金投入。因此，逐步更新网络设备有较大的实际意义。如果仅通过更新网络中部分的骨干节点的硬件设备，使其具有网络编码的功能，但是从一定程度上提升整体网络传输性能，那么这对网络编码的普及应用，是有积极的推动作用的。

本文的研究工作，旨在实现在有线网络中，合理设置部分节点使其具备网络编码的功能。通过部分网络编码，达到提高网络带宽利用率和降低更新升级网络硬件设备的代价。

具体内容如下：

第一章为绪论，简述网络编码的概念及特点，使读者对本文的研究方向有初步了解。

第二章介绍了目前网络编码的理论研究成果以及数学模型，并对网络编码部分应用进行了说明。最后，还介绍了部分网络编码的概念。

第三章着重研究部分网络编码的传输性能。通过对部分网络编码与完全网络编码的对比分析，以及部分网络编码与完全不网络编码的对比分析，发现（1）设置一定比例的节点具有网络编码功能，能够达到完全网络编码的传输性能；（2）

采用部分网络编码的传输网络，与不使用网络编码的传输网络比较，完成组播时占用的网络资源更少。

第四章着重研究组播链路构建机制。首先分析了组播传输网络建立的基本要求，据此，提出了一种新的组播链路构建准则以适应网络中部分节点具备网络编码功能的情况。

最后，对全文进行总结，并对一些有待继续研究的内容和方向进行展望。

第 2 章 网络编码概述

在进行部分网络编码研究之前，首先要对网络编码和部分网络编码做一个简要的介绍。

2.1 组播

在介绍网络编码之前，先介绍一下什么是组播。传统单播路由必须确定每一个目的地址，并分别向这些地址发送数据包。而组播通过一个组播地址只发送一个数据包的拷贝到多个接收目的地址。因此，组播能够有效地利用网络带宽并减少服务器的负载。网络编码正是应用于组播的网络环境下。

2.1.1 IP 组播

IP 组播的概念最早在 1988 年出现于一博士论文。IP 组播是利用一种协议将 IP 数据包从一个发送节点（源）传送到多个接收节点（目的地），其实现方法是将信息的拷贝发送到一组想要接收它的接收节点处。IP 组播是将 IP 数据包“尽最大努力”传输到一个构成组播群组的节点集合，群组的各个成员可以分布于各个独立的物理网络上。IP 组播群组中成员的关系是动态的，主机可以随时加入和退出群组，群组的成员关系决定了主机是否接收送给该群组的组播数据包，不是某群组的成员主机也能向该群组发送组播数据包。

2.1.2 应用层组播

应用层组播(application layer multicast)实现组播业务的思想是将组播作为一种叠加的业务，实现为应用层的服务。应用层组播网的节点是组播成员主机，数据路由、复制、转发功能都由成员主机完成，成员主机之间建立一个叠加在 IP 网络之上的、实现组播业务逻辑的功能性网络，称为叠加网 (overlay network)，主机

基于自组织算法建立和维护叠加网。IP 组播的数据沿着物理链路复制和转发，而应用层组播的数据则在主机实现复制和转发，数据报沿着逻辑链路转发，多跳逻辑链路可能经过同一条物理链路。

自组织算法是端系统组网的核心功能和机制，自组织算法的主要功能包括：周期性地交换节点状态信息，通报组成员状态；周期性地收集网络逻辑连接的带宽、时延等动态参数；动态地调整叠加网拓扑。

直观上，应用层组播功能可以避免 IP 组播功能的许多难题：一是应用层组播的状态在主机系统中维护，不需要路由器保持组的状态，解决了业务的扩展性问题，网络可以支持大量的组播组。二是组播应用可以随时部署，不需要网络设备的升级和功能扩展。三是可以简化组播的控制、可靠等功能的实现，建立在网络连接之上的应用层组播可以使用 TCP、UDP 服务，如可以利用 TCP 的可靠和拥塞控制简化组播的可靠和拥塞控制。

当然，应用层组播也有许多局限：一是端系统对 IP 网络的了解有限，节点参与组网时，只能通过探测获得一些网络性能参数，选取的逻辑链路难以优化；二是主机不了解 IP 网络的拓扑结构，只能通过带宽和时延等外在的特性参数，以启发式的方式建立叠加网，逻辑链路不能较好地利用质量较好的底层网络资源，叠加网的多条链路可能经过同一条物理链路。

2.2 网络编码理论研究

自 Ahlswede 发表“network information flow”^[1]以来,网络编码逐渐成为一个研究热点。文献[1]提出了网络编码的概念，并从信息论的角度出发证明了：当符号域接近无穷时，信源可以以其到任意接收机的最小割的速率进行组播。网络编码的提出从本质上打破了通信网络中传统的信息处理方式，最大限度的利用网络的现有资源。然而，文献[1]虽然提出了网络编码理论并说明了它的优越性，但并没有给出具体的实现方式和数学模型。2003 年，Ahlswede 发表了文献[2]，给出了网络编码的具体实现方式。从此，网络编码不再仅仅是一种理论和设想，而成为了可以实现的技术。

各国的科研机构、大学院校纷纷加入到研究队伍中。对网络编码的研究可以分

为理论和应用两个方向。

网络编码的构造方法很多，有线性和非线性之分。Li 指出^[2]：只要一个网络编码问题有解则一定有线性解，也即线性编码就可以实现网络编码。线性编码的关键在于节点编码系数的选取，只有在所有节点的编码系数两两线性无关的情况下，接收节点才能完全恢复原始数据。这个结论是很有用的。线性的操作大大降低了计算量，使网络编码的应用成为了可能。

在线性编码方面，根据网络拓扑结构是否已知可以分为确定性和随机性。如果网络拓扑结构已知，那么称在此情况下的网络编码为确定性网络编码；如果网络拓扑结构未知，那么称在此情况下的网络编码为随机性网络编码。

确定性编码方式又有集中式和分布式之分。集中式编码方法要求整个网络结构已知，通过计算得到各个节点的编码函数。确定性编码方案适用于稳定的网络环境。Li 的线性网络编码 (Linear Network Coding, LNC)^[2]、Koetter 和 Médardn 的代数构造法^[3-4]，以及 Jaggi 的信息流算法 (Linear Information Flow, LIF)^[6]都是集中式网络编码的代表。LNC 和代数构造法虽然都是线性网络编码，但其时间复杂度很高，为指数级。而 LIF 算法，其复杂度仅为多项式级。分布式编码方法算法，其核心思想是将网络拓扑分解成多个子树，并保证每个子树的编码系数属于其父节点编码系数的扩张空间 (span space)。以上对网络编码的研究，可以看做是在静态网络编码方向的研究，适用于较为稳定的网络。

随机性编码方式与确定性编码方式不同，它不需要了解整个网络结构的知识，各个节点可以独立随机的在某一域内选择编码系数。Médardn 的随机编码^[9-11]和 Chou 的实用网络编码^[7]是随机性网络编码的代表。网络的不稳定性以及节点流动性几乎不会对随即编码产生影响。随机性网络编码扩大了网络编码应用的范围。

下面简单介绍线性代数编码和随机编码。

2.3 网络编码模型

为了讨论的方便，我们这里只考虑最简单的无延时无噪声的无环网络 $G=(V, E)$ 。V 是一个有限集合，它的元素 V 被称为节点 (node)。E 是 $V \times V$ 的子集，其

元素 L 称为链路 (link)。对于任意链路 $L (R_i, R_j) \in E$, 称 R_i 和 R_j 是链路 $L (R_i, R_j)$ 的节点。一般来说, R_i 和 R_j 分别称为链路 L 的输入节点和输出节点, 链路 L 称为节点 R_i 的输入边, 链路 L 称为节点 R_j 的输出边。

为了研究线性网络编码, 对网络模型 $G=(V, E)$ 进行以下简化:

第一, G 中每条链接均具有相同的单位信道容量。如果某一链路 L 的容量为 N , 则将此边看作是 N 条具有平行单位容量的链路的组合, 即 $L=L_1, L_2, \dots, L_N$;

第二, 每条链路的时延为 0;

第三, 假设所有输入是独立的;

第四, 节点所有的操作是线性的。

第五, 对网络拓扑中的节点和链路分别进行编号。使信息流同时满足: ① 由编号低的节点流向编号高的节点; ② 由编号低的链路流向编号高的链路。

同时, 为了后文讨论上的方便, 对本文使用的符号做以下约定:

- ① L , 粗体大写, 表示集合, 如 $L \in \mathbf{L}$ 。
- ② $|\mathbf{L}|$, 表示集合 \mathbf{L} 所含元素的个数。
- ③ $\mathbf{L}/\{L\}$, 表示集合 \mathbf{L} 删去元素 L 。
- ④ $\mathbf{M}_{k \times j}$, 表示大小为 $k \times j$ 的矩阵。

2.4 线性代数编码数学模型的建立

设网络中的节点集合 V 由三类节点构成: 发送节点 S (source)、中间节点 R (relay) 和接收节点 T (terminal)。发送节点 S 构成集合 \mathbf{S} , 中间节点 R 构成集合 \mathbf{R} , 接收节点 T 构成集合 \mathbf{T} , 且 $|\mathbf{S}|=1$, $|\mathbf{R}|>1$, $|\mathbf{T}|>1$ 。

设发送节点 S 需要传递到接收节点 T_k 的消息 x 为空间 $\mathbf{F}_q (q=2^m)$ 上的 ω 维的向量。而在各个链路上传递的信息是消息 x 的组合, 同样也是 \mathbf{F}_q 上的 ω 维的向量。

设发送节点 S 共需发送 r 个消息: $\{x_1, x_2, \dots, x_r\}$ 。由于网络中所有节点进行的

操作都是线性的，所以网络边 $L(R_i, R_j)$ 上传输的消息 $y(L)$ 可以表示为发送节点 S 的输入 x_i 和节点 R_i 输入边的输入 $y(L')$ 的线性组合，即

$$y(L) = \sum_{x_i} a_{i,L} x_i + \sum_{L'} f_{L,L'} y(L')$$

其中， $a_{i,L}$ 表示发送节点 S 到节点 R_i 之间传送的消息 x_i 的系数。如果某个 $a_{i,L}=0$ ，表示发送节点 S 未向节点 R_i 发送消息；如果所有的 $a_{i,L}=0$ ，表示发送节点 S 与节点 R_i 之间无直接链路连接。 $f_{L,L'}$ 表示链路 L 传输的信息 $y(L)$ 与链路 L' 传输的信息 $y(L')$ 之间的比例关系。

定义编码矩阵 (Encoding matrix):

$$\mathbf{EM}_{|\mathbf{E}| \times |\mathbf{E}|} = [f_{L,L'}]_{|\mathbf{E}| \times |\mathbf{E}|} \quad (1)$$

其中，编码矩阵 $\mathbf{EM}_{|\mathbf{E}| \times |\mathbf{E}|}$ 中的元素 $f_{L,L'}$ 表示网络中链路之间的关系。

基于网络模型简化假设的第四点，编码矩阵 $\mathbf{EM}_{|\mathbf{E}| \times |\mathbf{E}|}$ 为上三角矩阵，且对角线的元素均为 0。对于某一网络而言，链路的变化（如信道容量，链路的失效或增加等）会引起编码矩阵 $\mathbf{EM}_{|\mathbf{E}| \times |\mathbf{E}|}$ 的变化。

定义信源输出矩阵 (Source matrix):

$$\mathbf{SM}_{r \times |\mathbf{E}|} = [a_{i,L}]_{r \times |\mathbf{E}|} \quad (2)$$

信源输出矩阵 $\mathbf{SM}_{r \times |\mathbf{E}|}$ 表示发送节点 S 到网络 $\mathbf{G}=(\mathbf{V}, \mathbf{E})$ 的输入关系。

定义接收节点 T_k 接收矩阵 (Terminal matrix):

$$\mathbf{SM}(T_k)_{r \times |\mathbf{E}|}^T = [b(T_k)_{L,i}]_{|\mathbf{E}| \times r} \quad (3)$$

其中，接收节点 T_k 接收矩阵 $\mathbf{SM}(T_k)_{r \times |\mathbf{E}|}^T$ 中的元素 $b(T_k)_{L,i}$ 表示网络中链路与接收节点 T_k 的输入链路之间的传输关系。矩阵 $\mathbf{SM}_{r \times |\mathbf{E}|}(T_k)^T$ 表示网络到输出节点 T_k 的输出关系。

定义单接收节点传输矩阵:

$$\mathbf{M}(T_k)_{r \times r} = \mathbf{SM}_{r \times |\mathbf{E}|} (\mathbf{I}_{|\mathbf{E}| \times |\mathbf{E}|} - \mathbf{EM}_{|\mathbf{E}| \times |\mathbf{E}|})^{-1} \mathbf{SM}(T_k)_{r \times |\mathbf{E}|}^T \quad (4)$$

其中， $\mathbf{I}_{|\mathbf{E}| \times |\mathbf{E}|}$ 为的单位矩阵。 $\mathbf{M}(T_k)_{r \times r}$ 反映了发送节点 S 到接收节点 T_k 的传输关系。只有当 $\mathbf{M}(T_k)_{r \times r}$ 满秩， T_k 才能完全接收并回复发送节点发送的原始信息。

定义全网络传输矩阵 (Transmission matrix):

$$\mathbf{TM}_{r \times kr} = [\mathbf{M}(T_1)_{r \times r}, \mathbf{M}(T_2)_{r \times r}, \dots, \mathbf{M}(T_k)_{r \times r}] \quad (5)$$

$\mathbf{TM}_{r \times kr}$ 反映了整个网络的传输性能。

2.5 数学模型的推广

前面的讨论均为单发送节点多接收节点情况下，网络编码的数学模型的构建。下面对这个模型进行修改，以使其适用于其他网络模型。

2.5.1 单发送节点多播模型

单发送节点多播模型，当网络中有 k 个接收节点 T (terminal)，且节点接收相同的 r 个数据时，全网络传输矩阵 $\mathbf{TM}_{r \times kr}$ 的大小为 $r \times kr$ 。网络编码的问题只要分析 $\mathbf{TM}_{r \times kr}$ 就可以了。代数编码将网络编码转化为了数学上的矩阵问题，使网络编码问题的分析更为简单。

如图 4 所示， T_1 、 T_2 、 T_3 表示接收节点， X_1 、 X_2 、 X_3 表示信源 S 所要传递的消息，全网传输矩阵 $\mathbf{TM}_{r \times kr}$ 的大小为 3×9 。

2.5.2 多发送节点多播模型

文献[2]提出了多发送节点多播问题的解决办法。构造一个虚拟信源，它将信息发送到实际的信源。原本的多信源多播问题就转化为了单信源多播问题。并且对于信源的相关性没有要求。

如果采用代数模型，多发送节点多播问题也很好建模。当网络有 N 个信源节点 S (source) 和 k 个接收节点时，如果信源节点发送的消息总数为 r ，那么全网络传输矩阵 $\mathbf{TM}_{r \times kr}$ 的大小仍为 $r \times kr$ 。

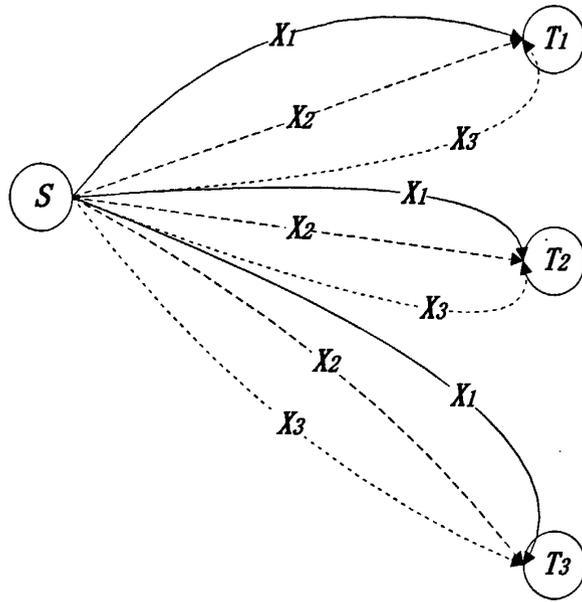


图 4 单发送节点的多播模型

如图 5 所示，图中 S_1 、 S_2 表示发送节点， T_1 、 T_2 、 T_3 表示接收节点， X_{11} 、 X_{12} 、 X_{13} 表示发送节点 S_1 所要传递的信息， X_{21} 、 X_{22} 表示发送节点 S_2 所要传递的信息。

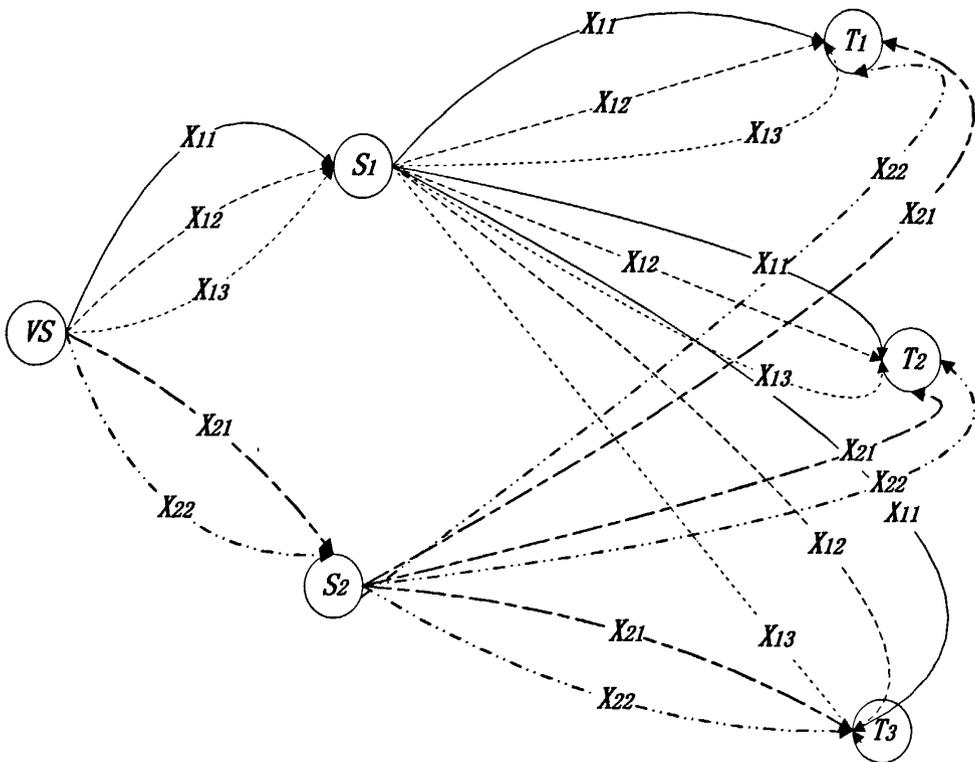


图 5 多发送节点多播模型

接收节点 T_1 、 T_2 、 T_3 需要接收发送节点 S_1 与发送节点 S_2 发送的全部消息。假设网络中存在一虚拟信源 VS ， VS 传递消息 X_{11} 、 X_{12} 、 X_{13} 到节点 S_1 ， VS 传递消息 X_{21} 、 X_{22} 到节点 S_2 ，那么多发送节点多播问题就转化为单发送节点多播问题。图 5 所示的例子中，全网络传输矩阵 $\mathbf{TM}_{r \times kr}$ 的大小为 5×15 。

2.5.3 不同接收速率多播模型

此外，对于网络中存在不同接收速率节点的情况，代数模型也同样可以建模。

当网络仅一个信源节点，但接收节点以不同速率接收信息时，全网络传输矩阵 $\mathbf{TM}_{r \times (k_1 r_1 + k_2 r_2)}$ 的大小为 $r \times (k_1 r_1 + k_2 r_2)$ ， r 表示发送节点 S 的总的传输速率， k_1 表示接收速率为 r_1 的接收节点数量， k_2 表示接收速率为 r_2 的接收节点数量。

如图 6 所示，图中 X_1 、 X_2 、 X_3 、 X_4 、 X_5 表示发送节点 S 所要传递的消息。不同的接收节点接收不同的信息：接收节点 T_1 需要接收消息 X_1 、 X_2 、 X_3 ，接收节点 T_2

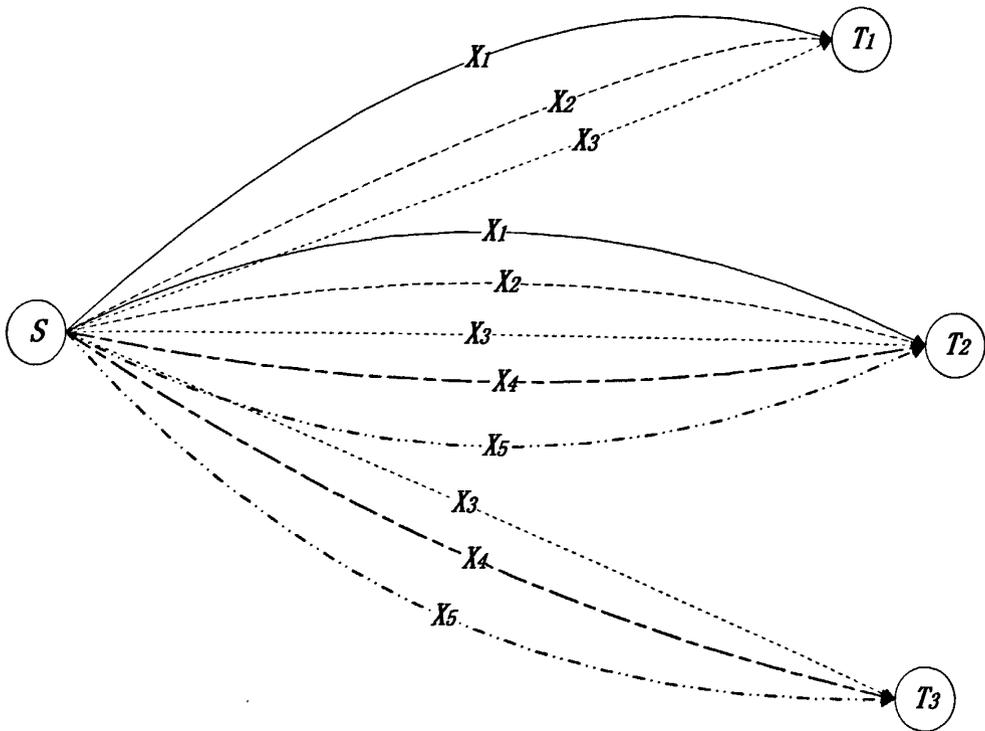


图 6 不同接收速率多播模型

需要接收消息 X_1, X_2, X_3, X_4, X_5 , 接收节点 T_3 需要接收消息 X_3, X_4, X_5 。全网网络传输矩阵 $\mathbf{TM}_{r \times (k_{r1} + k_{r2})}$ 的大小为 $5 \times (3 + 5 + 3)$, 即 5×11 。

2.6 随机网络编码

前面提到的网络编码的算法, 都有一个基本的前提, 即网络拓扑结构全部或部分已知。在网络的规模不大或网络结构简单的时候, 集中式编码方法需要的这个条件比较容易满足。但是随着网络复杂度的提高, 节点和链路动态变化的时候, 要了解全局拓扑结构就比较难。如果采用集中式编码方法, 则需要花费大量的时间资源用于全局编码系数的重分配。分布式网络编码在掌握局部拓扑信息时即可进行编码操作, 具有良好的拓扑适应性, 但需要一定的通信开销。不论是集中式编码还是分布式编码, 在网络拓扑不稳定的时候, 其优势都不明显, 甚至还不如不采用网络编码。

随机编码可以很好应付上述这些问题。与确定性编码的不同, 随机编码的编码向量不是一成不变的, 而是在一个域内随机的选择。文献[9]给出了采用随机编码, 接收机恢复出原始数据的概率, 并将随机编码与随机转发机制比较。结果表明, 当构建节点的随机编码系数的域足够大时, 采用随机编码节点恢复原始数据的比率明显高于随机转发机制。

2.7 网络编码应用方面的研究

目前, 网络编码的应用主要集中在无线网络传输、P2P 传输以及网络安全上。下面, 我们将结合 P2P 文件传输以及流媒体播放系统中的应用, 简单介绍网络编码的应用。

2.7.1 P2P 文件传输

BitTorrent 协议是当前最为流行的提供文件共享的 P2P 网络协议。BitTorrent 是一种无结构的网络协议, 具备了高扩展形、差错容忍性和独立性, 易于部署应用, 得到了大范围的使用。

在典型的 BitTorrent 协议中，节点通过目录服务器来查找目标节点。如图 7 所示。我们将拥有全部文件的节点称为种子节点(Server)。当一个节点把文件共享为种子 (seed) 时，BitTorrent 协议把共享的文件按一定的大小分成 n 个数据块，并将文件和分片信息保存在一个 Torrent 文件中，同时把共享的文件信息发布在目录服务器上。其他对该内容感兴趣的用户节点(节点 A 等)，只需要点击种子信息 (Torrent 文件)，即可通过源节点和所有用户节点构成的 P2P 网络传输数据。其中种子信息包含了所有参与节点已经下载的数据和尚未下载数据的情况。

BitTorrent 协议允许有多个发送点，某节点下载文件的同时也在上传文件，节点处在同步传送的状态。节点根据自己的本地信息选择转发的分片，这适用于小规模传输场合，且节点对整个系统有较好的了解的情况下。在大规模系统中，每个节点仅知道相邻节点的情况，用户 A 可能发送了重复信息(重复信息包括用户 B 已有信息和其他节点正在转发到 B 的信息)而占用过多的网络资源。此外，BitTorrent 由于采用 Rarest First 的策略，所以在文件下载的最后阶段需要的时间比较长。种子中途离开也是文件下载面临的问题之一。

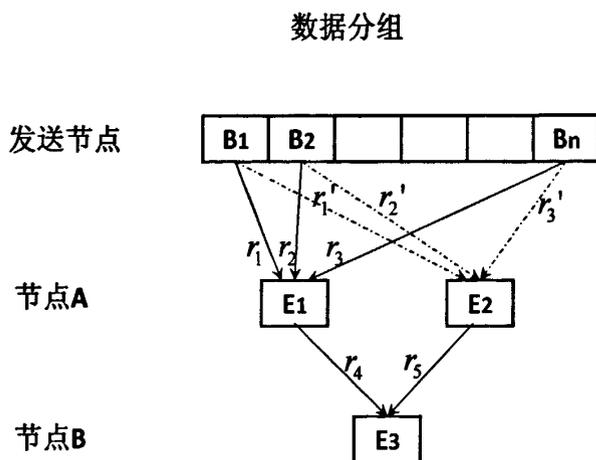


图 7 P2P 文件传输模式示意图

如果采用了网络编码就可以克服上述问题。用户 A 每次发送的信息都是与已经更新的分片的线性组合，其中总能有节点 B 感兴趣的分片。只有当编码后的分片和

用户B已有的分片线性相关，传输才是无效的。这在编码的符号域足够大时，发生的可能性很小，各节点以概率1恢复原文件。如图8所示。

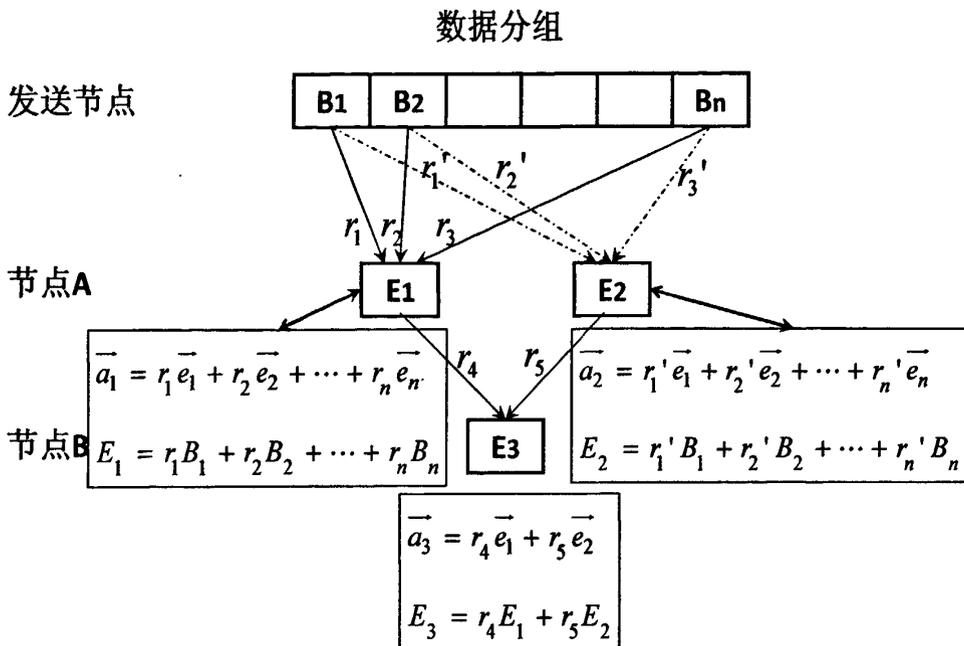


图 8 采用网络编码的 P2P 文件传输模式示意图

文献[22]首次将网络编码应用于 P2P 文件传输系统。具体做法是将要传输的文件划分为 1000 到 2000 个分片，所有传输的分片可以是原始分片的组合。为了保证较少的编码解码时间，将 50 到 100 个分片作为一组 (segment 或称 generation)，并只在同一个组里的分片进行组合。采用网络编码可以提高种子的效率，减少用户的下载时间。

2.7.2 P2P 视频流传输

使用 P2P 原理进行流媒体传输与文件传输十分相似。

文献[22]中提出，为了使网络编码的复杂性降低，将文件划分为不同的 segment，并在同一 segment 内部进行网络编码。与文件传输相似，流媒体传输需要按照图像

时间上的先后顺序，将整个文件划分为不同的 segment（或称 GOP）。一般来说这些 segment 拥有相同的播放时间。此时的 segment 就相当于文件传输的原始文件，我们将一个 segment 划分为 n 个 block，同一 segment 内的 block 之间进行网络编码。

某一节点同时和 3 到 4 个节点交换数据。节点接收到的数据放在缓冲区中，当缓冲区中有足够的数据就进行网络编码的解码操作，如 Gaussian 消除法，或者在接收数据的同时进行解码工作，如 Lava 采用的 Gauss-Jordan 消除法^[23]；节点从缓冲区中选择需要发送的数据，进行编码后发送给接收节点。

2.7.3 视频编码技术与网络编码

如果以视频流的速率为标准，那么视频编码技术大致可以分为二类：单码率与可扩展性编码。

在单码率编码中，服务器始终以单一速率向所有接收端发送流媒体数据。如文献[18]、[19]、[23]和[24]讨论的都是单一速率。如果使用网络编码，对应的网络编码模型，即为 2.4.1 节中的单发送节点多播模型。该编码方式的控制粒度较粗，不能同时公平有效的对待多个接收节点。在接收节点的网络状况差异很大时，接入带宽高于发送速率的接收节点的接收能力没有得到充分利用，带宽低于该速率的接收端处会发生拥塞。

在可扩展性编码方式中，为了适应网络异构的特性，将视频内容编码分为若干个互不相交的视频层，接收端只需要接收一定数量的视频层，就可以解码还原得到视频画面，质量取决于接收到视频层的数量。由于各个层的内容之间并不重叠，接收到多个层的信息可以大大提高视频的接收质量，从而更有效地节省了带宽。

作为可扩展编码的一种，分层编码将视频内容编码分为占用少量带宽的基本层和若干可以提高视频质量的增加层，接收节点可以根据自己的网络状况选择要接收的层，解码产生相应质量的视频图像。针对分层编码这种视频编码技术，出现了许多不同的网络编码策略来构建流媒体传输系统。

根据文献[2]可知，线性网络编码可以满足拥有不同带宽的节点，使其达到自身的最大流最小割速率，而非整个网络的最大流最小割速率。如果使用网络编码，

对应的网络编码模型，即为 2.4.3 节中的不同接收速率的多播模型。

文献[16]将随机网络编码（Random Linear Network Coding, RLNC）应用于使用了分层编码的流媒体传输系统。信源采用可伸缩视频编码（Scalable Video Coding, SVC），对不同的层给予不同的优先级，并采用非均匀错误保护（Unequal Error Protection, UEP）的策略对数据进行打包。中间节点采用随机网络编码。

文献[25]采用了优先级编码传输（Priority Encoding Transmission, PET）^[27]对不同层的数据包进行打包。图 9 所示，信源将一个 GOP（Group of Pictures）的数据进行分层编码，优先级由低到高依次为 Layer 1, Layer 2, ..., Layer N 。不同层对应的数据依次为 R_1, R_2, \dots, R_N 。然后对数据按照优先级进行打包。之后再采用随机网络编码进行传输。当接收端接收到 k 个数据包后，即可以恢复 R_1 到 R_k 的数据。

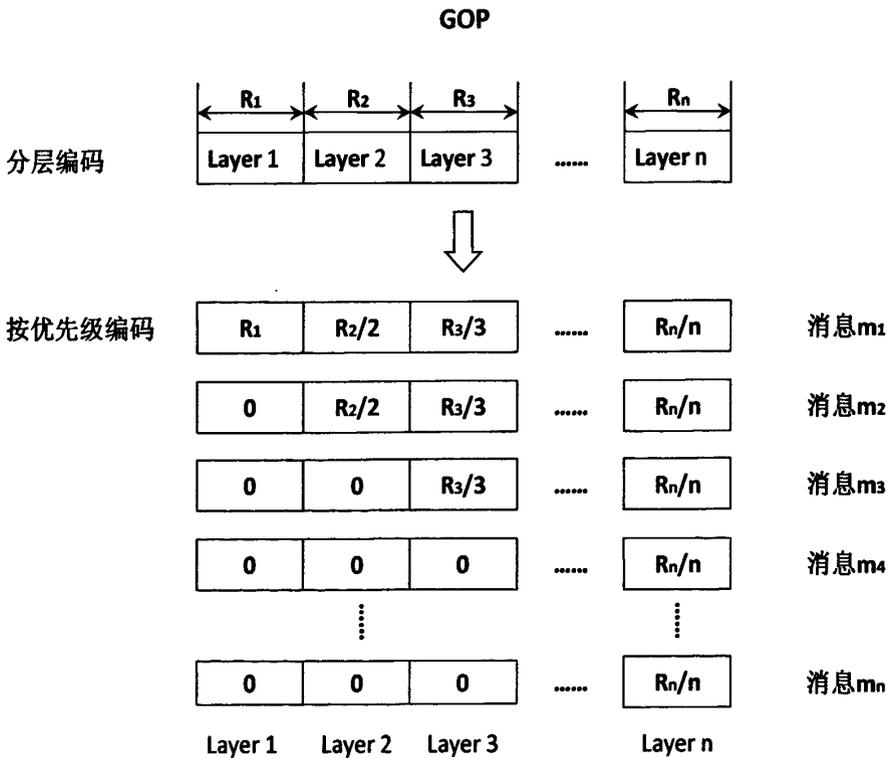


图 9 分层视频编码与网络编码结合

文献[12]也采用了 PET 对 GOP 进行分块。如图 10 所示。



图 10 分层视频编码与网络编码结合 2

文献[21]将层分离网络编码 (Layer Separated Network Coding, LSNC) 来实现分层编码的传输, 即只将位于同一层的数据进行网络编码。

文献[17]提出的层次网络编码 (Hierarchical Network Codes, HNC) 也是基于视频的分层编码。首先将流媒体文件按照重要性划分为不同的子集 **A**、**B**、**C**。假设 a_1 、 a_2 属于子集 **A**, b_1 、 b_2 属于子集 **B**, c_1 、 c_2 属于子集 **C**。按照如下的规则产生编码块:

$$\begin{aligned}
 N_1 &= f_1^1 a_1 + f_2^1 a_2 \\
 N_2 &= f_1^2 a_1 + f_2^2 a_2 + f_3^2 b_1 + f_4^2 b_2 \\
 N_3 &= f_1^3 a_1 + f_2^3 a_2 + f_3^3 b_1 + f_4^3 b_2 + f_5^3 c_1 + f_6^3 c_2
 \end{aligned}$$

其中, f_i^j 表示编码系数。采用这种编码结构, 可以很明显的看到, 恢复 a_i 的概率总是大于 b_i 、 c_i 的。这是因为只需要二个 N_1 类型的包就可以恢复 a_i , 而需要四个 N_1 或 N_2 才能恢复 a_i 、 b_i 。采用这种结构, 可以通过的调整包 N_i 的产生概率, 从而调整接收端接收重要数据 a_i 概率, 以利于节点较快的恢复出最重要的信息。

虽然文献[12]、[16]和[25]将网络编码应用于了采用分层编码的流媒体系统中，但它并不是通过网络编码来对不同优先级的数据进行保护。对于不同层的数据，网络编码时并没有区分。但是，文献[17]和[21]是通过网络编码的方法，对不同的数据实行保护。

2.8 编码节点数量受限的网络编码

虽然网络编码可以带来诸如提高信道利用率、提高鲁棒性等好处，但是需要硬件设备的升级换代，此外编码的复杂性可能带来编码、解码造成时延。鉴于此，部分学者开始研究，在网络中只有部分节点具备网络编码功能的环境下信息传递的效率问题。这里把不具备网络编码功能的节点称为路由节点，把具备网络编码功能的节点称为编码节点。所谓编码节点数量受限的网络编码，即网络中路由节点和编码节点同时存在，编码节点完成网络编码的编解码功能。

文献[8]讨论了无线环境下受限网络编码的性能问题。假设在无线网络中一些传统的路由节点被可编码的网络节点所替代。并讨论了以下两个问题：(1) 网络需要多少编码节点，(2) 如何设置编码节点的位置。通过计算机模拟进行量化计算，发现：需要进行网络编码的节点数随着拓扑复杂度的增加而增多，其中拓扑复杂度的增加指的是节点数量或链路数量的增加；多次生成拓扑后其编码节点的位置相对固定。

2.9 本章小结

网络编码技术由于其较高的带宽利用率，成为解决网络大流量传输的有效技术之一。本章对线性代数网络编码、随机网络编码以及编码节点数量受限的网络编码的理论以及部分应用进行了介绍，为后续的研究奠定了一定的理论基础。

第3章 编码节点数量受限的网络编码传输性能研究

在有线环境中,通过对全部节点网络编码、编码节点数量受限的节点网络编码和全部节点转发三种情况的下网络传输性能的比较,讨论了网络编码对带宽利用率的影响。

3.1 简化网络模型

在已有的传输拓扑内,设定部分节点具有网络编码功能,讨论网络编码对带宽的影响。这里的传输拓扑,满足如下条件:(1)拓扑中所有的边都参与信息的传输;(2)信源与各个接收节点之间的最小割,均等于仿真设定的传输速率。(3)如果删去拓扑中任意一条边,那么发送节点与某一个或某几个接收节点之间的最小割将小于传输速率。

为了讨论方便,我们使用2.3节的网络模型,即对拓扑模型进行以下假设:

第一,网络无噪声。

第二,网络中的所有链路传输信息时无时延。

第三,网络中不存在环路。

第四,假设所有的链路均具有相同的单位信道容量,即单位时间一条链路只能传递一个单位容量的信息。如果某一链路的信道容量为 n ,则将此链路看作是由 n 条具有单位信道容量的平行链路构成。

第五,网络中只有一个发送节点,若干接收节点。如果存在两个发送节点,则假设存在一个虚拟信源,实际发送节点作为虚拟信源的接收节点,而虚拟信源将所需传输的信息进行线性编码后输出。

第六,拓扑中所有节点对信息的操作均是线性的。

3.2 设计过程

1、建立网络拓扑

由于仿真所需要的网络环境，是无环、无噪，且所有具有单位信道容量的链路均参与到仿真的过程中。因此，可以对拓扑生成器生成的拓扑，进行进一步的简化，以达到我们的设计准则。

对链路的有效性进行判断的具体方法是：如果删去链路后，发送节点与接收节点之间最小割没有变化，那么认为此链路对仿真来说是无效边。我们将拓扑中的无效边移去，即得到了仿真所需要的网络拓扑。

2、计算可网络编解码节点

通过前面我们知道，可以在时域或空间域进行网络编码。对于实时传输，主要是进行空间网络编码。如果采用空间网络编码，那只有一条输入边的节点是不能进行编码的。除去源节点和接收节点，只有 2 条及 2 条以上输入边的节点才可以进行空间网络。根据这个原则，如果存在链路 $L(R_1, R_3)$ 、链路 $L(R_2, R_3)$ ，那么说明节点 R_3 有 2 条输入边，于是将节点 R_3 标记为可编码节点。计算节点的输入边的数量，确认节点是否为可编码节点。通过这个方法找出所有可以编码的节点。

3.3 性能仿真与分析

3.3.1 全部节点随机编码

在该仿真场景下，所有节点都使用随机网络编码。

第一步，各节点分别产生本地编码系数。

由于所有节点均使用网络编码，这里我们使用随机数生成函数 `random` 产生本地编码系数。如图 11(a)所示，节点 R_4 为编码节点， x_1 、 x_2 、 x_3 分别表示链路 (R_1, R_4) 、 (R_2, R_4) 、 (R_3, R_4) 的数据包，即节点 R_4 的输入数据。 y_1 、 y_2 分别表示链路 (R_4, R_5) 、 (R_4, R_6) 的数据包，即节点 R_4 的输出数据。采用随机网络编码后， y_1 、 y_2 均为 x_1 、 x_2 、 x_3 的线性组合，如图 11(b)所示，随机生成的本地编码系数 $a_1=[1\ 0\ 1]$ ，表示输出链路 (R_4, R_5) 的编码向量，于是得到链路 (R_4, R_5) 的输出为：

$$y_1 = [1 \ 0 \ 1] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_1 + x_3$$

同样, 我们得到链路(R_4, R_6)的本地编码系数 $a_1 = [1 \ 1 \ 0]$, 可以计算出链路(R_4, R_6)的输出为:

$$y_2 = [1 \ 1 \ 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_1 + x_2$$

对于只有一条输入边的节点, 我们将其视为可编码节点。因为转发只是编码的特殊情况, 即本地编码系数为 $[1]$ 。

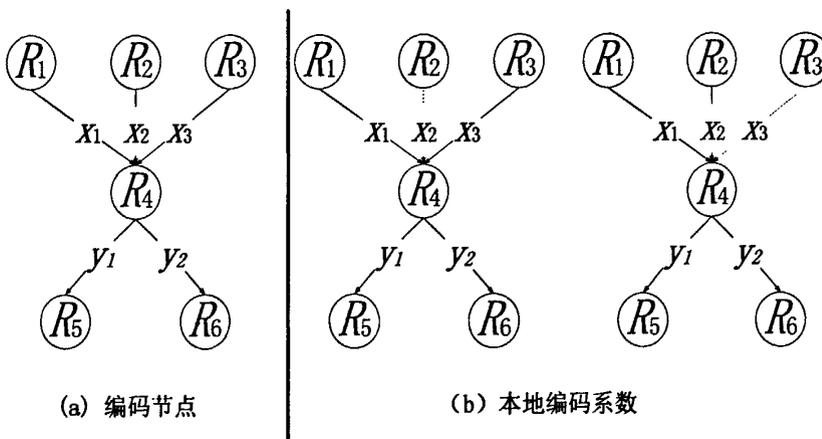


图 11 编码节点与本地编码系数

第二步, 计算编码矩阵。

由于我们采用的随机网络编码是线性编码, 可以通过线性计算得到全局编码系数, 从而得到整个网络的编码矩阵。

第三步, 计算采用网络编码后获得的数据传输量。

编码矩阵包含了所有的编码信息, 根据编码矩阵秩的大小, 可以知道理想状况下接收节点所能获得的最大数据量。

由于随机网络编码并不是最优编码, 所以需要多次仿真以获得最大数据传输量。

3.3.2 网络编码节点数量受限的网络编码

首先，设定编码比例。

我们已经知道，只有输入边大于或等于 2 的点才能进行网络编码的点。通过设置一定的编码比例，只选取部分节点进行网络编码，从而降低网络编码带来的开销。

其次，设置本地编码系数。

对于部分使用网络编码的节点，使用随机数生成函数产生本地编码系数。

对于只有一条输入边的节点，我们将其视为可编码节点。因为转发只是编码的特殊情况，即本地编码系数为 $[1]$ 。如图 12 所示，节点 R_2 只有 1 条输入边 (R_1, R_2) ，即为不可编码节点。输出边 (R_2, R_3) 和输出边 (R_2, R_4) 只需要转发输入边 (R_1, R_2) 的数据包即可。

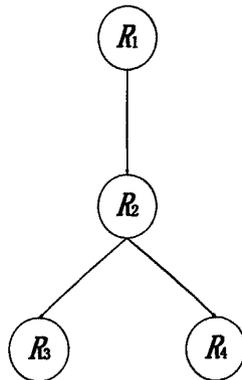


图 12 不可编码节点

对于条输入边数量大于或等于 2 的不编码节点，这些节点单次只能转发某条输入边的数据包，而转发数据包的选择将影响接收节点的接收效果。如图 13 所示，节点 R_4 即为可编码节点，它有 3 条输入边 (R_1, R_4) (R_2, R_4) (R_3, R_4) ，2 条输出边 (R_4, R_5) (R_4, R_6) 。如果节点 R_4 进行数据的转发，那将有 A_3^2 （共 6 种）转发的可能。对于拓扑中存在的所有输入边大于或等于 2、但不使用网络编码的节点，均需要按照上面的规则遍历所有情况。例如网络中存在 3 个可编码的节点，每个节点分别存在 2、3、6 种转发数据包的可能，那么根据乘法原则，整个网络需要遍历 36 种情况。

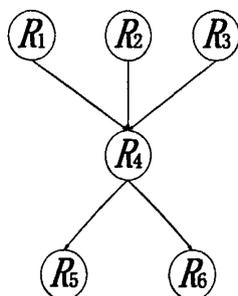


图 13 可编码节点转发数据包

再次，计算编码矩阵。

由于我们采用的随机网络编码是线性编码，我们可以通过线性计算得到全局编码系数，从而得到整个网络的编码矩阵。

最后，计算采用网络编码后获得的数据传输量。

编码矩阵包含了所有的编码信息，根据编码矩阵秩的大小，我们可以知道理想状况下接收节点所能获得的最大数据量。

由于随机网络编码并不是最优编码，所以需要多次仿真以获得最大数据传输量。

3.3.3 全部节点无编码

之前，我们已经计算出可编码的节点。对于不可编码的节点来说，只有一条输入边，只需要转发输入边的数据包。而对于可编码的节点来说，有两条及两条以上的输入边。在无编码条件下，这些可编解码节点单次只能转发某条输入边的数据包，数据包的选择将影响接收节点的接收效果。正因如此，对于可编码的节点就需要遍历所有的转发数据包的可能，以求出可以获得的传输最大速率。

首先，各节点设置本地编码系数。

其次，计算编码矩阵。

由于采用的随机网络编码是线性编码，可以通过线性计算得到全局编码系数，从而得到整个网络的编码矩阵。

最后，计算采用网络编码后获得的数据传输量。

编码矩阵包含了所有的编码信息，根据编码矩阵秩的大小，可以知道理想状况下接收节点所能获得的最大数据量。

具体的流程如图 14 所示。

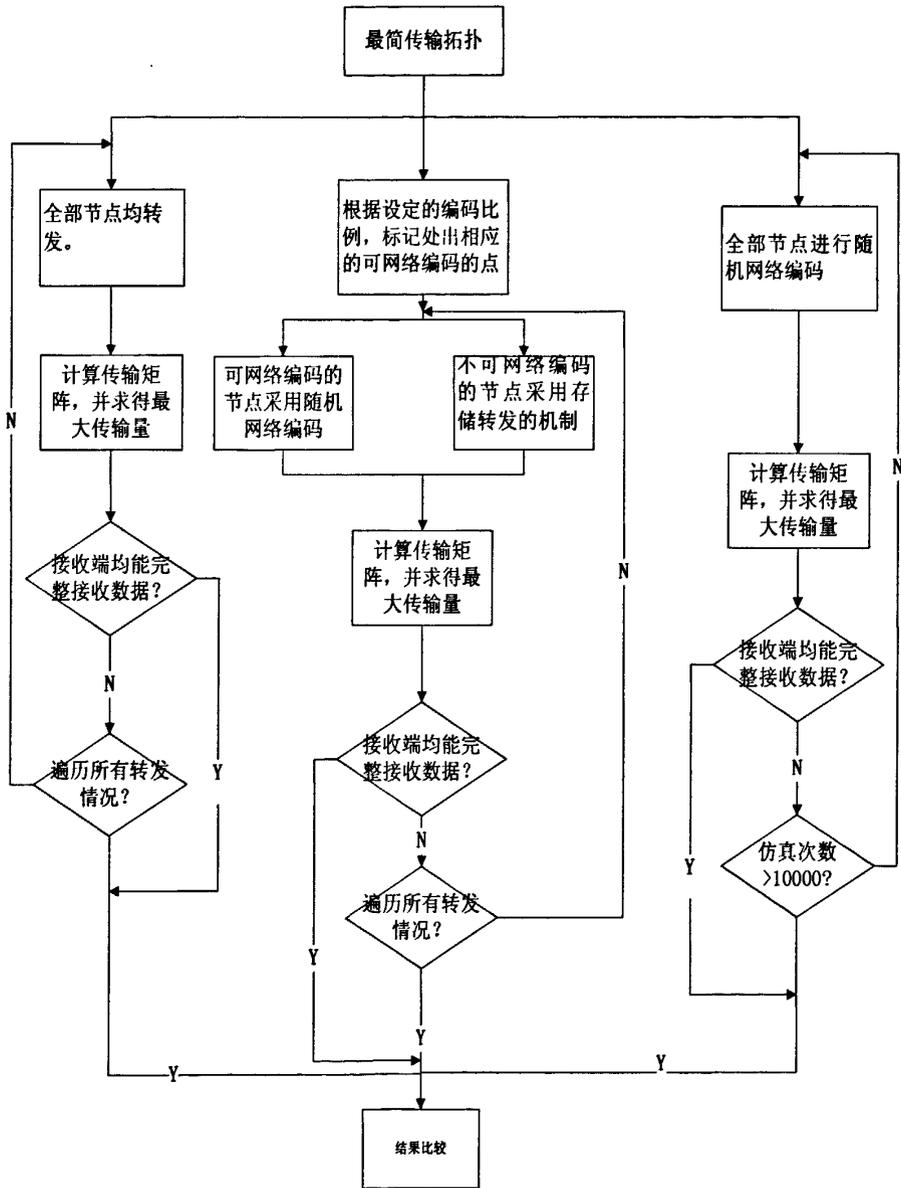


图 14 部分网络编码性能研究仿真流程图

3.4 仿真结果

我们将从以下两个方面衡量部分网络编码的性能：

1. 传输效率

设置参数编码比例：

$$p = \frac{\text{实际编码节点总数}}{\text{可以进行网络编码的节点总数}}$$

将部分网络编码的传输效果与完全网络编码时的传输效果进行比较。在相同的网络条件下，设置不同的编码比例 p ，使网络部分节点使用网络编码后，达到全部节点网络编码的传输效果。

2. 链路增加量

设置参数

$$\delta = \frac{\text{增加的链接数}}{\text{原图的链接总数}}$$

将部分网络编码的传输效果与完全不使用网络编码时的传输效果进行比较。在原网络条件下，完全不实用网络编码可能无法完成发送节点到多接收节点之间的信息传递。因此，通过增加原网络的有效链路，使不使用网络编码也能完成信息的组播。 δ 用于衡量增加的链路数占原网络的链路总数的比例。 δ 越大，表示所需要增加的链路越多。 δ 从另一个角度，说明了网络编码节约了网络的带宽资源。

随机生成五组含 50 个节点的网络拓扑结构图，Graph50-1、Graph50-2、Graph50-3、Graph50-4、Graph50-5、Graph50-6。具体仿真结果与性能比较结果如下。

图 15 表示针对 Graph50-1 的仿真结果。Graph50-1，共有 50 个节点，其中 1 个发送节点、3 个接收节点。各个接收节点的最大接收速率为 3，即单位时间某一接收节点最多能够收到 3 个不同的数据包。Graph50-1 的 50 个节点中，共有 22 个输入边大于 2 的节点。Graph50-1 共有 80 条有效链路参与信息的传输。

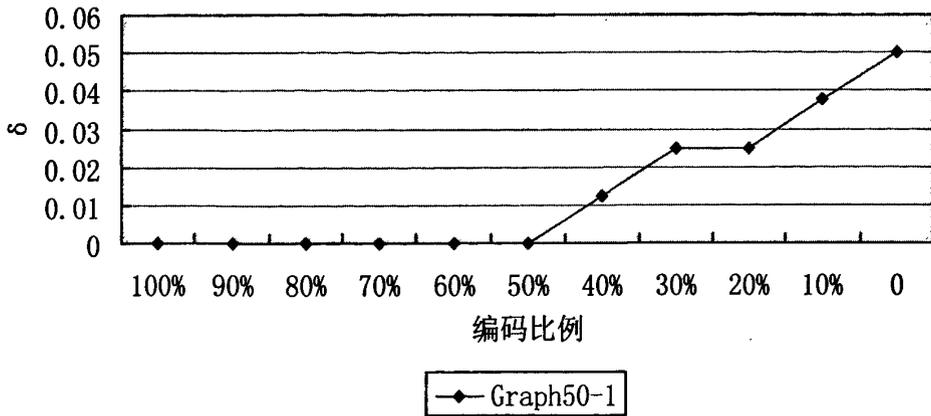


图 15 Graph50-1 仿真结果

仿真结果表明，当设置编码比例为 50% 时，可以达到全部节点进行编码的传输效果。部分编码只需要 80 条链路即可完成发送节点到接收节点之间的信息传递。如果不使用网络编码，全部节点进行转发，则无法实现信息的完全传递。通过对 Graph50-1 增加一定的链路，使完全不使用网络编码进行信息传递成为可能。当整个拓扑增加到 84 条单位链路时，全部节点均转发接收到的数据包，此时 3 个接收节点才能完全接收发送节点传递的消息。

图 16 表示针对 Graph50-2 的仿真结果。Graph50-2 共有 50 个节点，其中 1 个

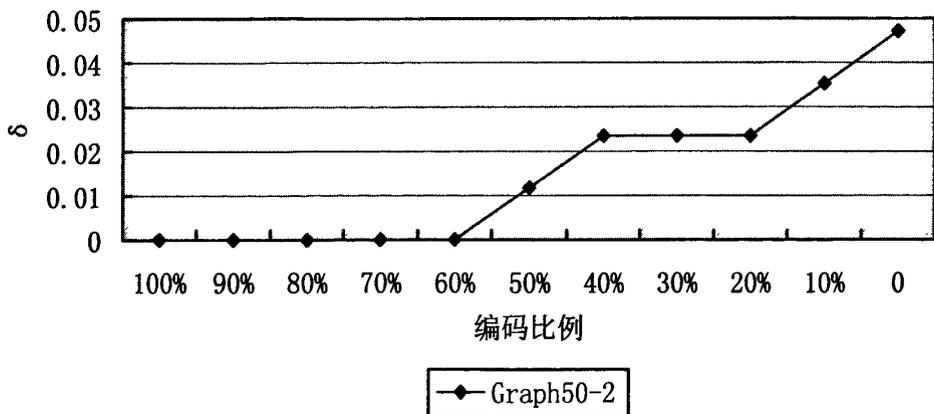


图 16 Graph50-2 仿真结果

发送节点、3个接收节点。各个接收节点的最大接收速率为3。Graph50-2的50个节点中，共有24个可编码节点。Graph50-2共有85条有效链路参与信息的传输。

仿真结果表明，当编码比例为60%时，可以达到全部节点进行编码的传输效果。部分编码只需要85条链路即可完成发送节点到接收节点之间的信息传递。如果不使用网络编码，全部节点进行转发，则无法实现信息的完全传递。通过对Graph50-2增加一定的链路，使完全不使用网络编码进行信息传递成为可能。当整个拓扑增加到88条单位链路时，全部节点均转发接收到的数据包，此时3个接收节点才能完全接收发送节点传递的消息。

图17表示针对Graph50-3的仿真结果。Graph50-3共有50个节点，其中1个发送节点、3个接收节点。各个接收节点的最大接收速率为3。在Graph50-3的50个节点中，共有22个可编码节点。Graph50-3共有83条有效链路参与信息的传输。

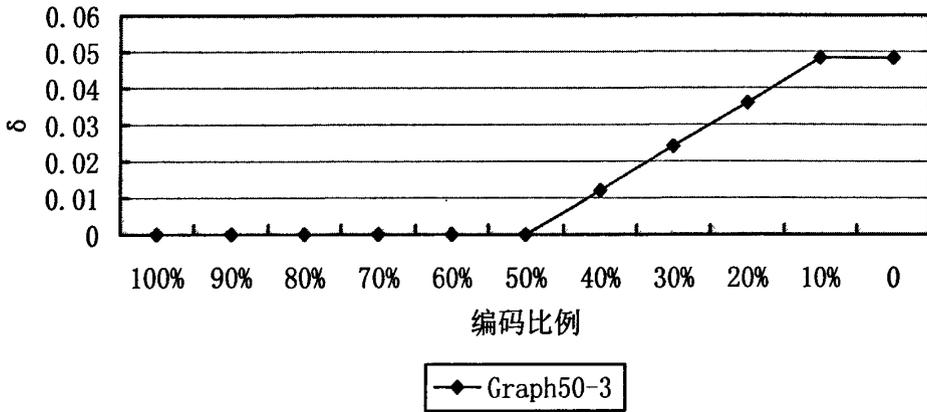


图 17 Graph50-3 仿真结果

仿真结果表明，当编码比例为50%时，可以达到全部节点进行编码的传输效果。部分编码只需要83条链路即可完成发送节点到接收节点之间的信息传递。如果不使用网络编码，全部节点进行转发，则无法实现信息的完全传递。通过对Graph50-3增加一定的链路，使完全不使用网络编码进行信息传递成为可能。当整个拓扑增加到87条单位链路时，全部节点均转发接收到的数据包，此时3个接收节点才能完全接收发送节点传递的消息。

图 18 表示针对 Graph50-4 的仿真结果。Graph50-4 共有 50 个节点，其中 1 个发送节点、3 个接收节点。各个接收节点的最大接收速率为 3。Graph50-4 的 50 个节点中，共有 23 个可编码节点。网络中共有 84 条有效链路参与信息的传输。

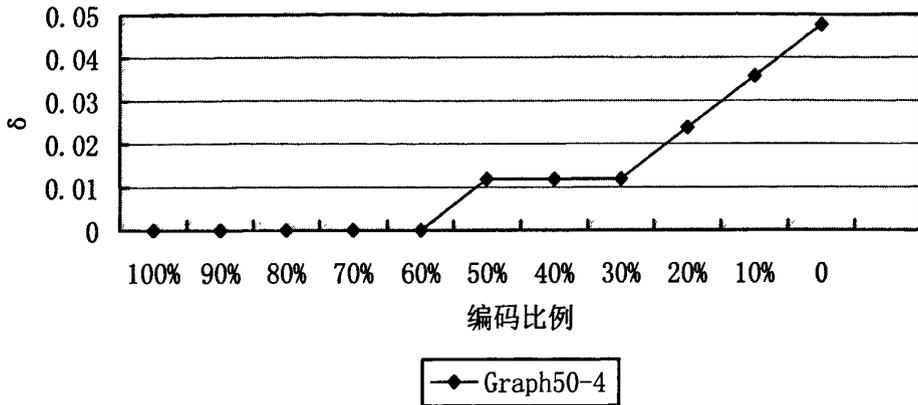


图 18 Graph50-4 仿真结果

仿真结果表明，当编码比例为 60% 时，可以达到全部节点进行编码的传输效果。部分编码只需要 84 条链路即可完成发送节点到接收节点之间的信息传递。如果不使用网络编码，全部节点进行转发，则无法实现信息的完全传递。通过对 Graph50-4 增加一定的链路，使完全不使用网络编码进行信息传递成为可能。当整个拓扑增加到 88 条单位链路时，全部节点均转发接收到的数据包，此时 3 个接收节点才能完全接收发送节点传递的消息。

图 19 表示 Graph50-5 的仿真结果。Graph50-5 共有 50 个节点，其中 1 个发送节点、3 个接收节点。各个接收节点的最大接收速率为 3。Graph50-5 的 50 个节点中，共有 25 个可编码节点。Graph50-5 共有 86 条有效链路参与信息的传输。

仿真结果表明，当编码比例为 60% 时，可以达到全部节点进行编码的传输效果。部分编码只需要 86 条链路即可完成发送节点到接收节点之间的信息传递。如果不使用网络编码，全部节点进行转发，则无法实现信息的完全传递。通过对 Graph50-5 增加一定的链路，使完全不使用网络编码进行信息传递成为可能。当整个拓扑增加到 90 条单位链路时，全部节点均转发接收到的数据包，此时 3 个接收节点才能完全接收发送节点传递的消息。

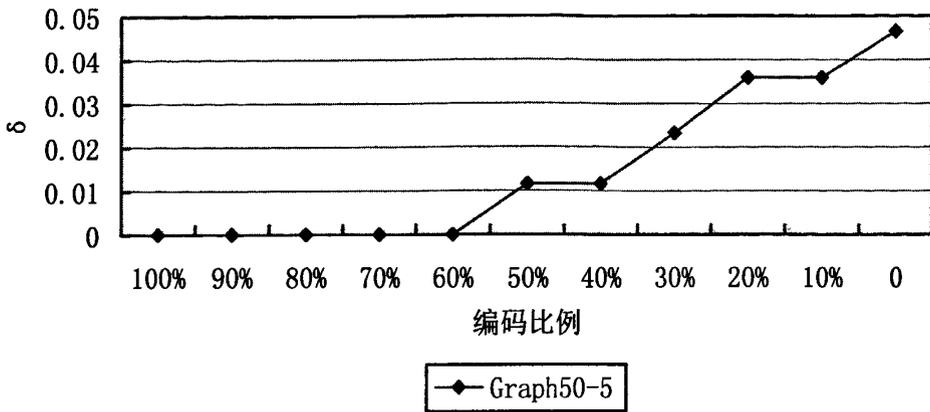


图 19 Graph50-5 仿真结果

图 20 是针对 Graph50-6 的仿真结果。Graph50-6 共有 50 个节点，其中 1 个发送节点、3 个接收节点。各个接收节点的最大接收速率为 4，即单位时间某一接收节点最多能够收到 4 个不同的数据包。Graph50-6 的 50 个节点中，共有 32 个可编码节点。Graph50-6 共有 100 条有效链路参与信息的传输。

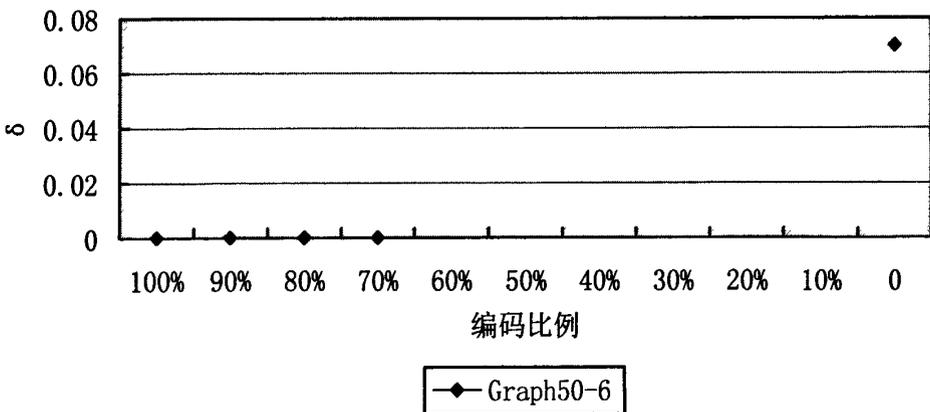


图 20 Graph50-6 仿真结果

仿真结果表明，当编码比例为 70% 时，可以达到全部节点进行编码的传输效果。部分编码只需要 100 条链路即可完成发送节点到接收节点之间的信息传递。如果不使用网络编码，全部节点进行转发，则无法实现信息的完全传递。通过对 Graph50-6 增加一定的链路，使完全不使用网络编码进行信息传递成为可能。当整个拓扑增加到 107 条单位链路时，全部节点均转发接收到的数据包，此时 3 个接

收节点才能完全接收发送节点传递的消息。

汇总仿真结果数据如表 1 所示。

表 1 汇总的仿真结果

(编码比例, 占用链路数)	全部节点编码	部分节点编码	无编码
Graph50-1	(100%, 80)	(50%, 80)	(0%, 84)
Graph50-2	(100%, 85)	(60%, 85)	(0%, 88)
Graph50-3	(100%, 83)	(50%, 83)	(0%, 87)
Graph50-4	(100%, 84)	(60%, 84)	(0%, 88)
Graph50-5	(100%, 86)	(60%, 86)	(0%, 90)
Graph50-6	(100%, 100)	(70%, 100)	(0%, 107)

分析以上实验结果, 得到结论如下:

1、在一定比例的编码系数的情况下, 网络中部分节点使用网络编码达到的传输效果, 与全部节点使用网络编码达到的传输效果一样。即使用相同的网络资源, 发送节点向若干接收节点发送相同的数据包, 接收节点均可以完整的恢复发送节点发送的信息。

2、当网络中所有节点均未采用网络编码时, 虽然理论上各个接收节点与发送节点之间的最小割等于发送节点需要传送的信息流的速率, 但是实际上组播无法达到这个速率的上限。只有为传输网络增加一定的带宽资源, 才能完成传输。网络编码的优越性也得到了证明。

3、随着网络组播的规模的增大, 如接收节点占总的传输网络的比例上升, 所需要的编码节点也增多。

3.5 本章小结

本章首先介绍了测试编码节点数量受限的网络编码传输性能的系统方案, 并对不同编码比例下, 编码节点数量受限的网络编码的传输性能进行了仿真, 为后续的研究奠定了一定的理论基础。

第 4 章 组播链路构建的研究

在实际网络中,大部分路由器只具传统的路由功能。如果要应用网络编码,那就必须对网络的基础设施进行换代升级,从而需要投入大量资金用于大型交换设备的升级。

基于第 3 章的分析,我们发现,一定比例的节点具有部分网络编码功能即可达到全部网络节点具有网络编码功能的传输性能。受到这个启发,可以逐步更新网络设备,使骨干网中的部分节点具备网络编码功能。这样既能提高网络的整体传输性能,又不会需要太大的投入。这对网络编码的普及应用,是有积极的推动作用的。

网络编码的应用效果与组播网络的构建有一定的关联。本章讨论网络中可编码节点的设置问题以及组播网络构建问题。首先,讨论如何设置网络编码节点。接着,对如何建立组播数据传输网这个问题进行研究。

4.1 设计准则

目前,网络中组播树构建主要有两种思路,一是最短路径树(Shortest Path Tree, SPT),一是汇节点树(Rendezvous Point Tree, RPT),也称共享树。

最短路径树的树根在源 S 。网络中所有组成员根据到源的单播最短路径获得组播数据。按照这种分发树模型,组成员可以更快的接收到组播数据,同时还能避免在网络中某些点形成拥塞。

共享树的树根是网络中某一个做核心的路由器。源首先将组播数据发到核心路由器,组成员再根据到核心路由器的单播路径获得组播数据。

由此可见,不论是最短路径树还是共享树,均没有考虑到网络编码节点的位置,所以不能很好的利用网络编码在组播时具有节约带宽上的特点。这里我们提出一种新的组播树的构建方法,在网络编码节点数量受限的情况下,利用网络编码降

低带宽的占用量，从而达到提高网络传输效率的目标。

4.2 网络简化模型及相关概念

网络中存在骨干节点和普通节点。骨干节点的输入（或输出）的带宽大于某阈值。较普通节点而言，骨干节点的信息交换能力更强。假设网络中的所有骨干节点具有网络编码功能。

为了讨论方便，我们将网络做以下规定：

网络由一定数量的节点以及连接节点的链路构成，用 $G(V, E)$ 表示网络拓扑。其中， V 表示网络拓扑的节点的集合，网络中的节点均有唯一的 ID 表示。 E 表示网络拓扑的链路的集合。

网络存在三种节点：一是唯一的发送节点，用 S 表示发送节点的集合；二是若干接收节点，用 T 表示接收节点的集合；三是若干中间节点，用 R 表示中间节点的集合。其中， $S \in S$, $T_i \in T$, $R_i \in R$ 。

网络中存在超级节点 N_{super} 。超级节点具有以下特性：（1）节点的输入带宽（或输出带宽）大于或等于某阈值；（2）节点具有线性网络编码功能。

网络中的链路 $L(R_i, R_j)$, $L(R_i, R_j) \in E$, $L(R_i, R_j)$ 是一条连通 R_i 与 R_j 的单向链路， $L(R_i, R_j)$ 为节点 R_i 的输出链路，为 R_j 的输入链路。如果节点 R_i 与节点 R_j 之间存在链路 $L(R_i, R_j)$ 或链路 $L(R_j, R_i)$ ，那么称 R_j 是 R_i 邻居节点。约定 $L(R_i, R_j)$ 简写为 L_{ij} 。

对于某一条链路 L 来说，具有 2 个属性：一是链路 L 的带宽，记为 $B(L)$ ；二是链路的时延，记为 $D(L)$ 。

对于某一个节点 R 来说，具有属性最大流，记为 $F(R)$ 。定义单位时间通过节点 R 的流量的最大值为节点的最大流 $F(R)$ 。 $F(R)$ 等于节点 R 的输入链路和输出链路的带宽之和。

路径 $P(S, T_i)$ 是链路的集合，表示节点 S 与节点 T_i 的一条通路。如果 T_i 为 S 的邻居节点，那么 $P(S, T_i) = \{L(S, T_i)\}$ 。

定义参数 $B_total(P(S, T_i))$, 表示路径 $P(S, T_i)$ 占用的链路的带宽总数。此参数用于衡量路径 $P(S, T_i)$ 占用资源多少。设 $P(S, T_i) = \{L_1, L_2, \dots, L_k\}$, L_i 的带宽为 $B(L_i)$,

$$\text{得到 } B_total(P(S, T_i)) = \sum_{i=1}^k B(L_i)。$$

4.3 接收节点的路径集的构建

4.3.1 发送节点 S 到各个接收节点 T_i 的路径发现

在已知的网络拓扑 $G(V, E)$ 中, 发送节点 S 运用泛洪算法, 向所有的邻居节点发送数据包 FP , 以寻找发送节点 S 到各个接收节点 T_i 的路径。

数据包 FP 的结构如图 21 所示, 包含五部分的信息:

- ① 数据包的头信息 $Head$, 表示此数据包为发送节点寻找到接收节点的路径的数据包 FP ;
- ② ID , 表示数据包 FP 的唯一标识号。
- ③ 数据包经过的节点信息 $NodeID$, 包括发送及转发此数据包的节点的唯一标识号;
- ④ 带宽信息 $Band$, 表示传输此数据包经过的所有链路中最小的链路带宽。
- ⑤ 时延信息 $Delay$, 表示数据包从发送节点到接收节点经过的所有链路的时延的总和。

<i>Head</i>	<i>ID</i>	<i>NodeID</i>	<i>Band</i>	<i>Delay</i>
-------------	-----------	---------------	-------------	--------------

图 21 数据包 FP 结构示意图

当中间节点 R_i 接收到 R_j 发送过来的 FP 后, 首先检查节点 R_i 的 ID 是否在数据包 FP 中。如果节点 R_i 的 ID 在数据包 FP 中, 中间节点 R_i 不转发数据包 FP 。这样可以防止环路的产生。如果节点 R_i 的 ID 不在数据包 FP 中, 节点 R_i 将自身的 ID

添加到数据包 FP 的 $NodeID$ 中。如果链路 $L(R_i, R_j)$ 的带宽 $B(L)$, 小于数据包 FP 中带宽信息的值, 更新数据包 FP 中的带宽信息。同时, 数据包 FP 中时延信息的值更新为 $Delay+D(L)$ 。最后, 节点 R_i 向除节点 R_j 外的所有邻居节点发送更新后的数据包 FP。

需要注意的是, 如果节点 R_i 接收到多个来自不同节点发送的数据包 FP, 那么节点 R_i 对每个数据包都要进行以上的操作。

当接收节点 T_i 接收到邻居节点转发来的数据包 FP 后, 首先将节点 T_i 的 ID 加入到数据包 FP 中, 并更新数据包 FP 中时延信息。如果链路 $L(R_i, T_i)$ 的带宽 $B(L)$, 小于数据包 FP 中带宽信息的值, 更新数据包 FP 中的带宽信息。数据包 FP 的转发过程, 终止于接收节点。

发送节点到接收节点的可用路径发现算法的流程如图 22 所示。

4.3.2 构建接收节点 T_i 的可用路径集 $Path(T_i)$

接收节点 T_i 的可用路径集 $Path(T_i)$ 中, 包含了所有发送节点 S 到接收节点 T_i 之间的可用路径。

由于数据包 FP 包含经过节点的 ID 信息, 所以每个数据包 FP 表示一条从发送节点 S 到接收节点 T_i 的路径 $P(S, T_i)$ 。接收节点 T_i 记录所有发送节点到节点 T_i 的可用路径 $P(S, T_i)$ 。如果接收节点 T_i 接收到 n 个不同 ID 的数据包 FP, 那么接收节点 T_i 的可用路径集 $Path(T_i)$ 的有效元素数为 n , 即 $Path(T_i) = \{ P_1(S, T_i), P_2(S, T_i), \dots, P_n(S, T_i) \}$ 。

其中, 数据包 FP 表示的路径为 $P(S, T_i)$ 的最小带宽 $Band$, 等于数据包 FP 中的带宽信息 $Band$, 路径 $P(S, T_i)$ 的时延等于数据包 FP 中的时延信息 $Delay$ 。

4.3.3 简化接收节点 T_i 的可用路径集 $Path(T_i)$

由于第 3 章的仿真所用的网络均是由具有单位信道容量的链路组成。在下一

```

1  发送节点  $S$ :
2      init FP :  $\text{NodeID}=\emptyset, \text{Band}=0, \text{Delay}=0$ ;
3      forward FP to neighbours
4  中间节点  $R_i$ :
5      if (  $ID(R_i) \in \text{NodeID}$  ) ;
6      else
7           $\text{NodeID} = \text{NodeID} \cup \{ ID(R_i) \}$ ;
8           $\text{Delay} = \text{Delay} + D(L)$ ;
9          if (  $\text{Band} == 0$  )
10              $\text{Band} = B(L)$ ;
11             else if (  $B(L) < \text{Band}$  )
12                  $\text{Band} = B(L)$ ;
13             forward FP to other node;
14 接收节点  $T_i$ :
15      $\text{NodeID} = \text{NodeID} \cup \{ ID(R_i) \}$ ;
16      $\text{Delay} = \text{Delay} + D(L)$ ;
17     if (  $B(L) < \text{Band}$  )  $\text{Band} = B(L)$ ;
18     update  $\text{Path}(T_i)$ ;

```

图 22 发送节点到接收节点的可用路径发现算法

步，我们将进一步简化网络的模型，使每条链路具有单位带宽。假设链路均有相同的单位带宽，拓扑中的链路的实际带宽均为单位带宽的整数倍。将带宽为 n 的链路 L ，视为 n 条具有单位带宽的平行链路，即 $L = \{L_1, L_2, \dots, L_n\}$ 。

在此基础上，更新节点 T_i 的可用路径的集合 $\text{Path}(T_i)$ 。如果最小带宽 Band 为 n 的路径 $P_i(S, T_i)$ ，视为 n 条具有单位带宽的平行路径，即 $P_i(S, T_i) = \{P_{i1}(S, T_i), P_{i2}(S, T_i), \dots, P_{in}(S, T_i)\}$ 。任意一条从发送节点 S 到接收节点 T_i 的可用路径的带宽为单位带宽。

4.4 使用网络编码的组播链路构建方法

在能完成组播的条件下,使用尽可能用较少的带宽,构建一个完整的组播数据传输网。

接收节点 T_i 运用 ford-fulkerson 算法,计算发送节点 S 与接收节点 T_i 的带宽的最大值 N 。由于任意一条从发送节点 S 到接收节点 T_i 的路径的带宽为单位带宽,根据最大流最小割原理,发送节点 S 到接收节点 T_i 之间的最小割等于最大传输速率。

由于我们已经假设网络中的链路均为单位容量,所以如果从发送节点 S 到接收节点 T_i 之间存在 $N(T_i)$ 条不重叠的路径,那么发送节点 S 和接收节点 T_i 的带宽即为 $N(T_i)$ 。简称发送节点 S 和接收节点 T_i 的带宽,即 $N(T_i)$ 为接收节点 T_i 的带宽 $N(T_i)$ 。

如果发送节点 S 的传输速率恒定为 F ,那么对于带宽大于 F 的接收节点来说,将只利用部分带宽用于构建组播数据传输网。而对于带宽等于 F 的接收节点来说,所有的带宽都将用于构建组播数据传输网。为了达到组播传输网占用的链路资源总数的最小化的目的,必须整体考虑发送节点到接收节点之间的传输网络构建。但是,这需要较多的资源用于计算才能达到最优。这里,采用分步构建组播数据传输网以趋近最优。我们分两步构建组播传输网:第一步构建发送节点 S 到某一接收节点之间的组播传输网;第二步构建发送节点 S 到其他接收节点之间的组播传输网。

4.4.1 第一接收节点路径选择

构建发送节点 S 到接收节点 T_i 之间的组播数据传输网,即是从接收节点的可用路径集 $\text{Path}(T_i)$ 中,选择一定数量的不重合路径。基于假设的网络模型,网络中所有的链路均具有单位信道容量,因此,只要发送节点到接收节点之间的路径总数和组播数据的流大小相同,从理论上接收节点就可以接收到发送节点发送的全部数据。

在接收节点集 \mathbf{T} 中,任意选取一个节点 T ,作为第一接收节点。构建发送节点

S 到第一接收节点 T 的组播数据传输网的过程中,不仅提高骨干节点(即编码节点)的利用率,同时在一定程度上减少链路资源的占用。

设路径 $Path$ 占用链路总数为路径集中链路的总数,且发送节点 S 到接收节点 T 之间的路径占用的链路总数的最小值为 H_{min} 。针对第一接收节点,路径的选择上应遵从以下原则:

(1) 选取占用链路总数为 H_{min} 且经过骨干节点数目较多的路径用于构建组播树;

(2) 如果满足原则(1)的路径数量小于组播数据的流大小,那么选取占用链路总数小于或等于 $H_{min} + \delta$ 且经过骨干节点数目较多的路径用于构建组播树;

(3) 如果满足原则(1)和(2)的路径数量小于组播数据的流大小,那么选取占用链路总数为较少的路径用于构建组播树。

原则(1)的优先级最高,这是因为它不仅利用了网络编码节点,而且占用的网络资源最少,这是最理想的情况。

但是网络中,满足原则(1)的路径不存在或者满足原则(1)的路径数量少于组播数据的流大小,根据这种情况,我们根据网络拓扑的大小定义 δ 的大小(一般情况下, $\delta \leq 2$),在拓扑中选择占用链路总数小于或等于 $H_{min} + \delta$ 且包含骨干节点数量较多的路径。 δ 越大,选中路径占用的骨干节点数目越多,但是占用网络资源的数量也越多。所以限制 δ 的大小,也就是在尽可能利用网络编码节点与尽量少占用网络资源之间寻找平衡。

由于限制了 δ 的大小,可能出现满足原则(1)和(2)的路径总数仍少于组播数据的流大小的情况,所以提出原则(3)。

这样我们就能完成发送节点到第一接收节点之间的路径选择,构建两者之间的组播数据传输网。具体算法如图 23 所示。

```

1  随机选取第一接收节点  $T$ 
2  定义链路集合  $LL$ , 初始化为空集合
3  计算发送节点  $S$  到接收节点  $T$  之间的路径占用的链路总数的最小值为
     $H_{min}$ 
4  for 占用链路总数为  $H_{min}$  的路径
5      计算路径经过骨干节点数目
6      经过骨干节点数目较多的路径用于构建组播树
7      将选中路径包含的链路添加到集合  $LL$  中
8  if 选中路径数量小于组播数据的流大小
9      for 占用链路总数小于或等于  $H_{min} + \delta$ 
10         计算路径经过骨干节点数目
11         经过骨干节点数目较多的路径用于构建组播树
12         将选中路径包含的链路添加到集合  $LL$  中
13 if 选中路径数量小于组播数据的流大小
14     选取占用链路总数为较少的路径用于构建组播树
15     将选中路径包含的链路添加到集合  $LL$  中

```

图 23 发送节点到第一接收节点的可用路径发现算法

4.4.2 其他接收节点路径选择

最后, 分别构建发送节点 S 与其他接收节点之间的组播数据传输网。

构建组播数据传输网的目标, 是用尽可能用较少的带宽, 构建一个完整的组播数据传输网。所以在路径的选择上, 要求尽可能的占用较少的网络的资源。衡量一条路径 $P(S, T_i)$ 占用网络资源的多少, 即是计算路径 $P(S, T_i)$ 占用链路的带宽总数 $B_{total}(P(S, T_i))$ 的过程。基于前面的步骤, 可以做出以下的简化:

① 由于假设链路具有单位带宽, 那么链路 $P(S, T_i)$ 占用链路的带宽总数即为链路 $P(S, T_i)$ 占用的链路总数 k , $B_{total}(P(S, T_i)) = \sum B(L_i) = k$;

② 如果链路 $P(S, T_i)$ 使用 m 条链路 L , 其中链路 L 包含于除接收节点 T_i 外其他接收节点选中的路径中, 那么链路 $P(S, T_i)$ 占用链路的带宽总数应减去 m 条链

路 L 的带宽, 即 $B_{total}(P(S, T_i)) = \sum B(L_i) = k - m$ 。

构建发送节点 S 和接收节点 T_i 之间的组播数据传输网, 具体步骤如下:

(1) 选择 F 条路径 $P(S, T_i)$, 用于构建 S 到 T_i 的组播数据传输网。选中的 F 条不重叠的可用路径, 构成集合 $Path_F(T_i)$ 。选则路径的原则如下:

- ① 路径 $P(S, T_i)$ 的 B_{total} 的值较小。
- ② 如果 B_{total} 达到某值的路径有多条, 选择路径的时延较小路径。
- ③ 如果 B_{total} 达到某值的路径有多条, 且路径的时延均相等, 那么随机选则路径。

(2) 接收节点 T_i 发送请求信息到发送节点 S , 请求保留挑选出的 F 条路径。记 $Path_F(T_i)$ 。

(3) 发送节点 S 接收到来自接收节点 T_i 发送的请求信息, 用集合 $LL \cup Path_F(T_i)$ 更新链路集合 LL 。

(4) 发送节点将链路集合 LL 包含的链路信息, 发送给除接收节点 T_i 外、节点集合 T_2 中所有元素。

通过以上步骤, 这样我们就能完成发送节点到其他接收节点之间的路径选择, 构建两者之间的组播数据传输网。具体算法如图 24 所示。

对于恒定网络, 即不存在节点加入或退出和链路加入或失效的情况, 一次计算后得到组播路径。对于变化网络, 即存在节点加入或退出、链路增加或失效的情况, 需要在网络变化后, 重复以上步骤。

4.5 无网络编码的组播链路构建

这里, 我们选择使用最短路径树的构建的组播网络来进行无网络编码的组播传输。

最短路径树的树根在源 S 。网络中所有组成员根据到源的单播最短路径获得组

```

1  init  $T_1 = \emptyset, T_2 = \emptyset, LL = \emptyset;$ 
2  if( $N(T_i) > F$ )
3       $T_2 = \{T_i\} \cup T_2;$ 
4  elseif( $N(T_i) = F$ )
5       $T_1 = \{T_i\} \cup T_1;$ 
6  for  $T_i \in T_1$ 
7       $LL = LL \cup \text{Path\_F}(T_i);$ 
8       $T_1 = T_1 / \{T_i\};$ 
9  for  $T_i \in T_2$ 
10     for  $P(S, T_i) \in \text{Path}(T_i)$ 
11          $B_{total}(P(S, T_i)) = \sum B(L_i) = k - m;$ 
12          $\text{Path\_F}(T_i) = \min_{B_{total}} \min_{Delay} \{P_x(S, T_i), P_y(S, T_i), P_z(S, T_i)\};$ 
13          $LL = LL \cup \text{Path\_F}(T_i);$ 
14      $T_2 = T_2 / \{T_i\};$ 

```

图 24 构建发送节点到其他接收节点之间的传输网算法的示意图

播数据。按照这种分发树模型，组成员可以更快的接收到组播数据，同时还能避免在网络中某些点形成拥塞。基于我们对网络的假设，对于一对发送节点和接收节点来说，需要寻找 3 条没有重叠边的最短路径用于构建两者之间的组播树。

4.6 仿真结果与分析

通过网络编码节点数量受限的情况下组播传输所占用的链路数量与无网络编码情况下组播传输所占用的链路数量的对比，考察部分网络编码的性能。

考虑 50 个节点构成的网络，其中 10% 的节点具有网络编码的功能。这 10 个编码节点，均匀的分布在网络中。需要通过仿真，我们发现，如果采用网络编码，只需要 19 个节点（其中，1 个发送节点、3 个接收节点）、25 条链路参与传输，就

可以完成组播。而如果不采用网络编码，则需要 21 个节点、26 条链路参与传输。

4.7 本章小结

本章通过模拟真实网络环境，对网络编码节点数量受限情况下数据传输占用网络资源的数量，与完全不进行网络编码情况下数据组播占用网络资源的数量进行比较，发现其仿真的结果与第 3 章的仿真结果基本一致。

第5章 总结与展望

网络编码技术由于其较高的带宽利用率，成为解决大规模视频、文件传输的有效方案之一。同时，由于网络编码的特点，近年来也在网络加密传输方面发展成一种有效的方法。

但是，通信网络如何使用网络编码提高网络容量或增加可靠性，必然会涉及网络建设成本的提高，同时需要影响到一些已部署网络的基础设施。如果能使只是部分节点使用网络编码，即可达到全部网络节点使用网络编码的性能，就能有效地降低升级网络硬件设施的代价，也为更好地推广网络编码的应用降低要求。在这个研究动机基础上，本文通过对部分网络编码进行研究分析，提出了一些个人见解。

首先，针对同一拓扑网络，通过全部节点使用网络编码、部分节点使用网络编码、无节点使用网络编码三种情况进行信息传输，分析网络传输的整体性能。仿真结果表明，部分节点网络编码，完全可以达到全部节点网络编码时的传输性能；部分节点网络编码的传输性能与全部节点均未使用网络编码的传输性能比较，部分节点网络编码能够节约更多的网络资源，提高网络的利用效率。

然后，着重研究在网络中存在部分编码节点的情况下，组播传输网络的建立的问题。目前现有的组播网络的构建，或者以占用资源数最少为目标，或者以最小时延为目标。这些构建算法，都未能考虑到编码节点对网络组播带来的积极影响。本文提出的基于链路优先等级的组播网络构建方法，合理利用网络中存在的编码节点。与使用最短路径树构建方法形成的组播网络比较，本文提出的方法占用的网络资源更小。

但是，限于本人目前的理论水平和实践水平，虽然对网络编码以及编码节点数量受限情况下的网络编码进行了研究，并通过仿真模拟真实网络环境得到了一些结果，但是缺少理论上的分析，未能给出理论上的最优值或理论上的边界。虽然得到了一些关于编码节点数量受限的网络编码特性的结论，但是由于自身能力的局限，未能更为深入的理解本质。

网络编码有许多问题有待研究。根据既有研究证据和经验，本文的研究工作在

将来可以从以下方面入手加以深化:

1) 目前的研究是在网络无环、无延时的前提下进行的,而这和实际传输差别很大。另外,网络编码处理的延迟、存储空间的限制以及网络丢包等因素也没有考虑。要让网络编码从实验室仿真走向实际应用,必须将考虑这些实际因素的影响。网络编码的仿真环境需要更接近实际网络。

2) 本文根据节点的单位时间最大流量的大小来设置编码节点,使编码节点均匀分布于网络中。但是在实际网络环境中,可能出现有流量大的节点密集分布于网络某一区域的现象,所以使编码节点均匀分布的方法不能适应这种情况。网络编码节点的设置的原则,还有待进一步研究分析。

3) 进行网络编码之前必须构建相应的组播网络时。本文仅提出了一种考虑网络编码节点的组播网络构建方法。与采用最短路径构建组播网络的方法相比,本文提出的方法在一定程度上降低了网络资源的消耗量,但可能存在更优的组播网络构建方法。在网络中存在部分编码节点的情况下,如何构建最优的组播传输网络,有待进一步研究分析。

参考文献

- [1] R. Ahlswede, N. Cai, S-Y. Li, and R. Yeung, Network information flow, *IEEE Trans. on Information Theory*, July 2000, **46** (4): 1204-1216.
- [2] S.-Y. Li, R.W. Yeung, and N. Cai, Linear network coding, *IEEE Trans. on Information Theory*, February 2003, **49**(2): 371-381.
- [3] R. Koetter and M. Médardn, Beyond routing: an algebraic approach to network coding, *Proc. of IEEE INFOCOM 2002*, pp. 122-130, 2002.
- [4] R. Koetter and M. Médard, An algebraic approach to network coding, *IEEE/ACM Trans. on Networking*, October 2003, **11**(5):782-795.
- [5] C. Gkantsidis and P. Rodriguez, Network coding for large scale content distribution, *Proc. of IEEE INFOCOM 2005*, pp. 2235-225, 2005.
- [6] S. Jaggi, P. Sanders, P.A. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen, Polynomial time algorithms for multicast network code construction, *IEEE Trans. on Information Theory*, June 2005, **51**(6):1973-1982.
- [7] P.A. Chou, Y. Wu, and K. Jain, Practical network coding, <http://research.microsoft.com/apps/pubs/default.aspx?id=78167>.
- [8] M. Kim, M. Medard, and U. O'Reilly, Integrating network coding into heterogeneous wireless networks, *Proc. of MILCOM 2008*, pp.1-7, 2008.
- [9] T. Ho, R. Karger, M. Médardn, D. Karger, and M. Effros, The benefits of coding over routing in a randomized setting, *Proc. of IEEE ISIT*, pp. 442, 2003.
- [10] T. Ho, M. Médardn, J. Shi, M. Effros, and D.R. Karger, On randomized network coding, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.58.1873&rep=rep1&type=pdf>.
- [11] T. Ho, M. Médardn, R. Karger, D.R. Karger, M. Effros, Jun Shi, and Ben Leong, A random linear network coding approach to multicast, *IEEE Trans. on Information Theory*, October 2006, **52**(10):4413-4430.

- [12] H. Liu, X. Tu, and J. Xie, Network coding for P2P live media streaming, *Proc. of IFIP International Conference on Network and Parallel Computing 2008*, pp. 399-398, 2008.
- [13] 马冠骏, 许胤龙, 林明宏, 宣颖, 基于网络编码的P2P内容分发性能分析, 《中国科学技术大学学报》, 2006, 36(11): 1237-1240.
- [14] D. Nguyen, T. Tran, T. Pham, and V. Le, Internet media streaming using network coding and path diversity, *Proc. of IEEE GLOBECOM 2008*, pp. 1-5, 2008.
- [15] T. Ho, D. R. Karger, M. Médard, and R. Koetter, Network coding from a network flow perspective, *Proc. of IEEE International Symposium on Information Theory*, pp. 441, 2003.
- [16] H. Wang, S. Xiao, and C.-C. Kuo, Robust and flexible wireless video multicast with network coding, http://w2cn.cis.nctu.edu.tw/wimax_paper/Multicast/Robust%20and%20Flexible%20Wireless%20Video%20Multicast%20with%20Network%20Coding.pdf.
- [17] K. Nguyen, T. Nguyen, and S. Cheung, Peer-to-peer streaming with hierarchical network coding, *Proc. of ICME 2007*, pp. 396-399, 2007
- [18] M. Wang, and B. Li, Network coding in live peer-to-peer streaming, *IEEE Trans. on Information Theory*, December 2007, 9(8):1554-1567.
- [19] M. Wang, and B. Li, R^2 : Random push with random network coding in live peer-to-peer streaming, *IEEE Journal on Selected Areas in Communications*, December 2007, 25(9): 1655-1666.
- [20] N. Cai, and R. Yeung, Secure network coding, *Proc. of 2002 IEEE ISIT 2002*, pp. 323, 2002.
- [21] C. Xu, Y. Xu, C. Zhan, R. Wu, and Q. Wang, On network coding based multirate video streaming in directed networks, *Proc. of Performance, Computing, and Communications Conference 2007*, pp. 332-339, 2007.
- [22] C. Gkantsidis, J. Miller, and P. Rodriguez, Comprehensive view of a live network coding P2P system, <http://research.microsoft.com/apps/pubs/default.aspx?id=69452>.
- [23] M. Wang, and B. Li, Lava: a reality check of network coding in peer-to-peer live streaming, *Proc. of IEEE INFOCOM 2007*, pp. 1082-1090, May 2007.

- [24] Y. Liu, and Y. Peng, Network coding for peer-to-peer live media streaming, *Proc. of GCC 2006*, pp. 149-155, 2006.
- [25] Y. He, I. Lee, and L. Guan, Distributed throughput maximization in P2P VoD applications, *IEEE Trans. on Multimedia*, April 2009, **11**(3): 509-522.
- [26] 尹浩, 林闯, 文浩, 陈治佳, 吴大鹏, 大规模流媒体应用中关键技术的研究, 《计算机学报》, 2008, **5**: 755-774.
- [27] A. Albanese, J. Blomer, and J. Edmonds, Priority encoding transmission, *IEEE Trans. on Information Theory*, November 1996, **42**(6): 1737-1744.

作者简介及攻读硕士学位期间的研究成果

作者简介

舒群，女，1985年5月出生，江西南昌人。2006年获得浙江大学信息学院信息工程专业学士，2006年9月~2010年8月攻读浙江大学信息与电子工程学习信息与通信工程硕士学位。研究方向是网络编码的理论及其相关应用。

攻读硕士学位期间的研究成果

谢磊，陈惠芳，舒群，王政，一种部分节点进行网络编码的传输网络生成方法，
201010242828.5