

网络灾备的传输补偿技术研究^{*})

李钟华^{1,2} 李伟华² 武 鲁²

(西北工业大学计算机学院 西安 710072)

摘 要 本文研究了当前计算机网络中进行重要数据事故灾备时存在速度慢、可靠性低等问题,提出利用信息学编码理论中的线性纠错码 LECC 来改善此状况。线性纠错码方法对数据进行分块冗余编码,在不可靠信道(如 Internet)上传输编码块。接收端只要接收到足够数量的编码块,就可解码出初始数据信息,而不管哪些数据包丢失了,缩短了包应答及丢失包重传的时间,提高了信道的可靠性及传输效率。

关键词 LECC 编码,网络灾备,有损信道,补偿传输

Research on Compensative Transmission of Network Backup and Recovery

LI Zhong-Hua^{1,2} LI Wei-Hua² WU Lu²

(School of Computer Science, Northwestern Polytechnic University, Xi'an 710072)

Abstract Linear error correction code(LECC) of the informatics encoding theoretics is used to solve the problems of slowness and low reliability on network disaster backup of crucial data in this paper. LECC partitions the data into blocks and encodes it redundantly, then the encoded packages are transmitted in the erasure channel (Internet for example). If enough encoded packages are received, the receiver can decode the original data, regardless of which package is lost. It saves the time for package acknowledging and retransmitting, and improves the reliability of channel and the efficiency of data transmission.

Keywords LECC code, Network disaster backup, Erasure channel, Compensative transmission

世界正在经历巨大的变革,信息已经成为公司拥有的最有价值的资产,而且是制约企业发展的重大因素之一。企业对数据的存储管理及容灾储备已经成为对日益紧张的 IT 资源另一需求来源。多点远程网络在线实时备份容灾成为保护用户数据的首要选择。Internet 的传输问题研究结果表明,目前 Internet 上传送数据,当考虑组播(Multicast)情况下的数据报的丢失概率时,一个发送者同时向 11 个地理位置分散的接收端发送同样的数据,在 1 小时之内,每个接收端的数据报的平均丢失概率为 9.3%,同时有 46.5%的数据报曾经被一个或一个以上的接收端所丢失。传统的 TCP/IP 处理丢包问题的应答重传机制耗时长、可靠性低,难以保证各备份点的数据的完整性和可用性。

针对以上问题,我们提出线性纠错码(Linear Error Correction Code,简称 LECC)^[1,2]方法,来保证网络灾备中的传输问题。LECC 方法对需要灾备传输的数据信息进行码,接收端只要接收到足够数量的编码块,就可解码出初始数据信息,而不管哪些数据包丢失了。而且,接收到哪些信息包、信息包到达的次序,甚至有多少信息包丢失,都不重要。接收端不需要对数据包进行应答,有效地解决了信息包丢失或受到破坏等问题,提高了传输的效率和稳定性。

1 数据传输错误控制

网络在线实时备份容灾中,网络环境本身存在一定工作负载。出现灾难故障或遭受黑客攻击(特别是分布式拒绝服务 DDOS 攻击)时,可用于灾备传输的可用信道带宽是有限的,因此传输过程中的错误控制方式是保证灾备效率及副本

可靠性的重要环节之一。本节对自动重发请求(Automatic Repeat reQuest,简称 ARQ)方式及前向纠错控制(Forward Error Control,简称 FEC)方式进行简要比较。

1.1 自动重发请求方式 ARQ

在保证了物理连接线路无故障的情况下,要保障信息可靠地接收,最常用的纠错方法就是让接收端向发送端反馈信息(ACK 或 NAK 信号),告诉发送端已经收到了什么样的数据。ACK 信号表明所发送信息已经被正确接收,发送端继续进行新的信息的发送;NAK 信号则需要发送端重新发送刚才发送的数据,在规定的时间内,若未能收到反馈结果(称为超时),则发送方可以认为传输出现差错,进而执行重传动作,这种机制称之为 ARQ 错误纠正机制。

ARQ 方式的优点有:简单、信元速率可控,同时对信道干扰的变化不敏感等。但它有一系列的缺点:(1)必须有一个反馈通道;(2)只能用在点对点的单一通信方式当中,不能实现一对多广播(Broadcast)或组播(Multicast)通信方式;(3)如果信道干扰大,则系统经常处于反馈-重发状态,效率降低,不能连贯工作。ARQ 在处理大时延系统时,需要反馈信道来传送重传请求,必须有两个来回的时间(RTT)来完成一次重传请求,从而有较高的时延。在广播传输情况下,若每个接收端都随机地发出重传的请求时,常常会由于重传请求的增多而引起所谓的反馈拥堵现象,从而使传送受到限制甚至停止。这些缺陷使 ARQ 方式在处理大型应用(如大型文件发布、多媒体应用、分组广播等)中受到限制。

1.2 前向纠错方式 FEC

FEC(Forward Error Correction, or Forward Erasure Cor-

^{*})国家 863 发展计划项目资助(编号:2003AA142060)。李钟华 博士生,主要研究方向:网络安全、多媒体通信技术。

rection, 前向纠错^[3]技术中, 发送者对数据进行编码, 然后发送所得到的纠错码, 使得接收者将接收到的数据经过解码后能够自行纠正出现的差错, 无需反馈信道。FEC 方式常用于误码率高的场合(如无线信道、modem 通信)或能容忍一定差错的实时业务中。

FEC 的优点: (1) 冗余。对需灾备传输的数据信息进行编码, 每一编码块组成了原始信息块的一个随机子集, 有关每个原始信息包的信息冗余被散布在一些编码信息包上, 通过传输冗余信息来防止信息丢失和进行差错控制; (2) 无需反馈信道。FEC 方式无需向发送端传递反馈信息, 因而在需要对数据进行快速修复时, FEC 能帮助简化协议, 甚至在某些情况下可以取消反馈信道。

FEC 错误控制方式由于它的可靠性以及无需反馈信道, 使得它在事故灾备传输领域具有极大的优势。但 FEC 在面对大文件传输时, 编码效率低, 解码时间长。本文提出的 LECC 方法是对 FEC 方法的一种改进编码方式, 能提高编解码效率, 为大规模数据的网络备份恢复提供快速、可靠的传输。

2 线性纠错码 LECC

在远程网络在线实时备份容灾中, 纠错码用于解决数据备份传输的可靠性及效率问题, 因此在线实时容灾要求纠错码算法能快速编解码及高效数据可解码恢复。我们采用的 LECC 编码方法是利用图论中二分图相关知识和二进制数的 XOR 运算而得到的一种纠错编码方法, 具有快速的编解码速率和高效的数据恢复效率。

对于一个长度为 M 的数据文件, 首先对其按一定长度 l 进行分块, l 可任意选择, 但考虑到网络传输中 IP 碎片问题, 我们一般选择 l 的大小略低于一个 IP 包的长度, 因此源数据文件可被分为 $m = M/l$ 信息块。信息块标记为 x_1, x_2, \dots, x_m , 对 x_1, x_2, \dots, x_m 进行冗余编码得到 n 个编码块, 编码率 $e = n/m$, 编码块记为 c_1, c_2, \dots, c_n 。这里,

$$\begin{cases} c_1 = x_{11} \oplus x_{12} \oplus \dots \oplus x_{1d_1} \\ c_2 = x_{21} \oplus x_{22} \oplus \dots \oplus x_{2d_2} \\ \vdots \\ c_n = x_{n1} \oplus x_{n2} \oplus \dots \oplus x_{nd_n} \end{cases} \quad (1)$$

其中, $1 \leq d_i \leq m$, ‘ \oplus ’表示异或加运算。

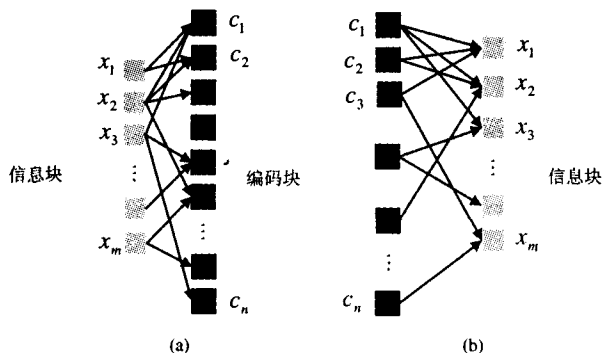


图 1 用二分图表示线性纠错码的编、解码过程。(a) 编码过程, (b) 解码过程

LECC 编码方法中的有关生成编码信息和利用编码信息来修复丢失了的原始信息的过程利用图的方式来表达, 可以得到图 1。其中图(a)表示的是编码过程, 图(b)表示的是解码过程。图 1 中, 二分图 G 为含有 m 个左节点和 n 个右节点, 左右节点之间的连接数不限制。 m 个左节点分别代表 m

个源数据块, n 个右节点则表示了由 m 个左节点生成的所有编码数据块。

2.1 LECC 编码设计

由图 1 可知, LECC 编码对初始信息块进行 XOR 运算, 生成包含初始信息块冗余信息的编码信息块, 实现信息的编码冗余传输。LECC 编解码设计的过程就是构建图 1 的二分图的过程。图 1 顶点 c_i 邻接的边的数量称为该顶点的度 (Degree of Node), 如 c_1 的度为 3, c_2 的度为 2。每一个编码块 c_i 通过以下方式由源文件 $x_1 x_2 x_3 \dots x_m$ 产生:

1. 编码块 c_i 在度分布 $\rho(d)$ 下随机选择度 d_i ;
 2. 从 x_1, x_2, \dots, x_m 中随机均匀选择 d_i 个不同的初始信息包 $x_{i1}, x_{i2}, \dots, x_{id_i}$, 计算
- $$c_i = x_{i1} \oplus x_{i2} \oplus \dots \oplus x_{id_i} \quad (2)$$

因此, 各个编码块都独立于其他编码块, 恢复过程只须关心编码块的数量, 而不需要在意具体哪些编码数据包丢失了。LECC 的编码过程可以看成是一个随机独立地往 m 个箱子里扔球的问题, 其中边的数量 ($\bar{d}n$) 相当于球的个数, 初始信息块可以看作是一个个的箱子。由典型的球与箱子问题的分析可知, 平均需要 $m' = m \cdot \ln(m/\delta)$ 个球才能保证每个箱子以概率 $1-\delta$ 至少有一个球。

编码块 c 的度概率分布 $\rho(d)$ 设计是 LECC 设计的关键部分。 $\rho(d)$ 设计须保证图 1(a) 的所有信息节点都有边连接 (即 $\delta \rightarrow 0$), 而且须保证图 1(b) 的解码过程可顺利完成。 $\rho(d)$ 设计还制约着编解码效率。LECC 中我们采用改进的 Soliton^[4] 概率分布:

$$\rho(d) = \frac{\omega(d) + \tau(d)}{\beta}$$

其中,

$$\omega(d) = \begin{cases} 1/m & d=1 \\ \frac{1}{d(d-1)} & d=2, \dots, m \end{cases}$$

$$\tau(d) = \begin{cases} \frac{R}{m} \frac{1}{d} & d=1, \dots, m/R-1 \\ \frac{R}{m} \ln(R/\delta) & d=m/R \\ 0 & d=m/R+1, \dots, m \end{cases}$$

$$\beta = \sum_d (\omega(d) + \tau(d))$$

$R = a \cdot m \ln(m/\delta)$, a 为大于零的常数。

2.2 LECC 解码过程

在异或加运算中, 有下列性质:

$$0 \oplus 0 = 1 \oplus 1 = 0$$

$$1 \oplus 0 = 0 \oplus 1 = 1$$

我们以图 1 中校验信息元 c_1 为例, $c_1 = x_1 \oplus x_2 \oplus x_3$ 。如果在传输过程中, 原始数据 x_3 丢失, 而我们正确接收到了 x_1, x_2 和 c_1 , 就可以利用异或加法本身的特点, 得到 $x_3 = x_1 \oplus x_2 \oplus c_1$, 从而将 x_3 修复。

由公式(1)可知, 若知道编码块的度及其邻居节点集, 我们就可以构建出图 1(b) 的解码二分图。解码器就可以迭代使用以下规则来恢复源信息块。解码流程如图 2 所示。

解码恢复规则: 在图 1(b) 中, 如果至少一个编码块只有一个邻居 (度 $d=1$), 则可直接恢复其对应的初始信息块。若已恢复的输入信息块还作为其他编码块的一个邻居, 则将其从这些编码块的邻居中移除 (即删除连接的边), 同时将对应的编码块度减 1, 从而简化了二分图。继续找 $d=1$ 的编码块, 重复以上过程, 直到恢复过程结束。

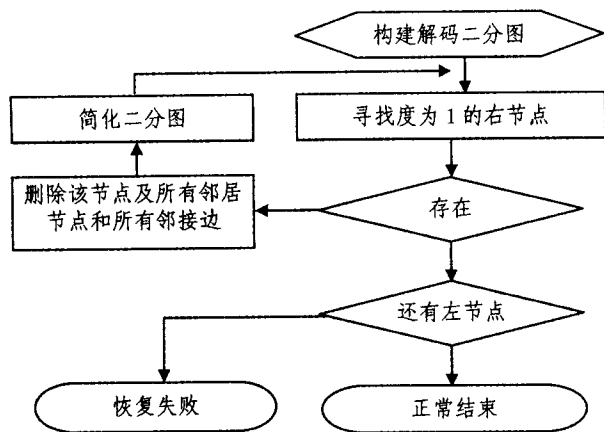


图2 LECC的解码流程

LECC的编解码过程是基于传输层的,编码数据包在灾备传输中采用图3数据结构进行数据传输。在编码数据前面加上该编码块的度信息和邻居节点的信息,接收端主机收到编码包后,根据编码块的度信息及邻节点信息实时构建解码二分图。当接收到一定数量的编码包,完成解码二分图构建后,对数据进行解码恢复。

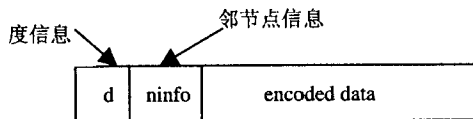


图3 LECC编码包的结构

3 LECC编解码性能仿真测试

本节对LECC编码进行原型系统性能测试。对于 $m=10000$ 初始信息文件,经过500次仿真实验,统计结果表明,接收端平均只需接收到10400个编码块,就能完全解码出初始信息文件,编码率为4%。接收端解码原始信息文件需要的接收的编码包数量分布柱状图如图4所示。

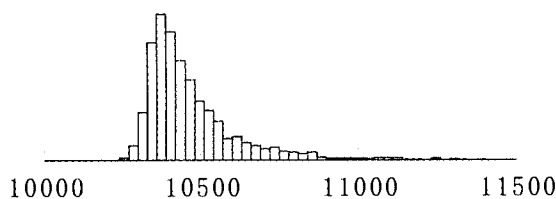


图4 LECC解码所需编码包数量分布

4 LECC在网络灾备传输中的应用

LECC编码传输具有编码速度快、编码效率高、传输无需反馈信道等特性,在网络化灾备中具有很大的应用前景,可极

大地提高恶劣传输环境(高网络工作负载或带宽攻击)下的网络灾备效率及可靠性。我们在局域网环境下对LECC进行数据传输性能测试,测试环境为100M以太网,人为加载高附加传输负载,发送主机及接收主机皆为P4,CPU为2.66GHz,内存512MB,主机系统环境RED HAT 9。对不同大小的文件的LECC编码传输和一般网络传输进行性能对比,结果如图5所示。

单位:秒

传输方式 文件大小	远程拷贝传输	LECC编码传输
1k	0.38	0.56
16k	0.40	0.60
512k	0.53	0.64
1M	0.85	0.92
512M	115.38	86.32
1G	221.62	120.56
2G	389.29	285.36

图5 LECC编码传输与远程拷贝性能对比

图5的结果表明,LECC在大文件网络灾备传输上有非常大的优势,例如传送1G的灾备数据,在我们的实验环境下,使用一般的远程拷贝方式需要211s,而LECC只需120s(包括LECC编解码时间及网络传输时间),仅为前者的54%。LECC对于1M以下的数据备份体现不出效率优势,但能很好地提高副本的传输可靠性。

另外,LECC编码在Broadcast及Multicast方面也有很大的应用前景。

结论 LECC编码意在保护大型文件(1GB或更大的文件),而且这种技术以软件方式实现,速度非常快,编码恢复原始数据所需的数据开销只占原始内容的4%。LECC编码技术是解决内容传送问题的一种创新方式。采用LECC编码传输方式时,发生信息包丢失或受到破坏等问题时带来的影响很小,重传数据或对原始信息包增加特大冗余以补偿丢失的信息包的需要减至最小或完全消除,很好地提高了网络灾备的效率及可靠性。

参考文献

- 1 王育民,梁传甲. 信息与编码. 西安:西北电讯工程学院出版社,1985
- 2 肖国镇,卿斯汉. 编码理论. 北京:国防工业出版社,1993
- 3 Elias P. Coding for Two Noisy Channels. In: Information Theory, Third London Symposium. Butterworth's Scientific Publications, 1955. 61~76
- 4 Scott J. Report on Waves. Report of the fourteenth meeting of the British Association for the Advancement of Science, York, 1844 (London 1845). 311~390