



| 网络编码丛书

Network Coding: An Introduction

网络编码导论

[新加坡] Tracey Ho [澳] Desmond S. Lun 著

北京跟踪与通信技术研究

冯贵年 叶建设 刘国胜 译

出版社

| 网络编码丛书

网络编码导论

[新加坡] Tracey Ho [澳] Desmond S. Lun 著

冯贵年 叶建设 刘国胜 译

清华大学出版社
北京

内 容 简 介

本书以网络编码的起源开篇,对网络编码技术进行了全面系统的论述。内容包括网络编码的基本概念、无损多播网络编码、会话网络编码、有损网络中的网络编码、子图选择以及网络编码的安全性等。

本书内容全面,示例丰富,深入浅出,是网络编码领域不可多得入门读物,对相关领域科研人员 and 工程技术人员都具有较大的参考价值。

This is a translation edition of the following title(s) published by Cambridge University Press:

Network Coding: An Introduction

ISBN: 978-0-521-87310-9

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press.

This Simplified Chinese translation edition for the People's Republic of China (excluding Hong Kong SAR, Macau SAR and Taiwan) is published by arrangement with the Press Syndicate of the University of Cambridge, Cambridge, United Kingdom.

© Cambridge University Press and Tsinghua University Press 2015

This translation edition is authorized for sale in the People's Republic of China (excluding Hong Kong SAR, Macau SAR and Taiwan) only. Unauthorized export of this reprint edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of Cambridge University Press and Tsinghua University Press.

此版本仅限在中华人民共和国境内(不包括中国香港、澳门特别行政区及中国台湾地区)销售。

北京市版权局著作权合同登记号 图字:01-2016-3556

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

网络编码导论/(新加坡)何翠屏,(澳)伦兆民著;冯贵年,叶建设,刘国胜译.—北京:清华大学出版社,2016
(网络编码丛书)

书名原文:Network coding: An introduction

ISBN 978-7-302-43918-9

I. ①网… II. ①何… ②伦… ③冯… ④叶… ⑤刘… III. ①计算机网络—编码程序—程序设计
IV. ①TP393

中国版本图书馆 CIP 数据核字(2016)第 111158 号

责任编辑:石磊 赵从彤

封面设计:傅瑞学

责任校对:刘玉霞

责任印制:何芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:三河市君旺印务有限公司

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×230mm 印 张:10.25 字 数:207千字

版 次:2016年12月第1版 印 次:2016年12月第1次印刷

印 数:1~2000

定 价:39.00元

译者序



REFACE

网络编码起源于 2000 年,主要用于提高组播网络的吞吐量,解决实际网络传输无法达到最大理论容量的问题。其核心思想是突破传统网络中只允许中继节点转发数据的限制,使用中继节点对接收到的数据包进行组合、编码等智能化处理。网络编码彻底改变了通信网络中信息处理和传输的方式,是信息理论研究领域的重大突破。目前,网络编码已成为网络、信息论及编码等领域最受关注的研究课题之一。

本书原著 *Network Coding: An Introduction* 由剑桥大学出版社于 2008 年 4 月出版,作者为加州理工学院的 Tracey Ho 教授和哈佛-麻省理工博德研究所的 Desmond S. Lun 研究员。目前,Ho 教授和 Lun 研究员均是网络编码理论研究领域的领先人物,对网络编码理论研究具有很深的造诣。

该书为通信与信息系统方向专著。全书采用由简入深的方法,将新兴的相对“晦涩”的网络编码理论娓娓道来。内容上,本书侧重于从工程角度引出网络编码理论知识,浅显易懂,非常适用于通信相关领域内网络编码的初学者了解基本的网络编码概念;同时,该书也基于工程实例辐射至相关理论基础,深入推演了网络编码理论的由来,并提供了丰富的导读信息,对需要进一步学习网络编码理论的相关技术人员具有较强的参考作用,可以作为通信与信息系统相关专业本科生和研究生的教材。

本书的翻译过程遇到很多困难和阻碍,得到了众多人员的帮助和支持。首先,特别感谢帮我们把关技术的张胜利副教授、孙奇福副教授,感谢他们的专业帮助。其次,也要

感谢热心帮助和支持我们的网络编码研究团队成员：孙威、李海涛、陈凌晖。同时，衷心地感谢清华大学出版社的专业团队，没有他们就没有这本书。最后，要感谢我们的家人，谢谢他们的爱与支持。

译者

2016年9月

前言

F

FOREWORD

网络编码背后的基本思想非常简单。正如本书中给出的定义,网络编码不过是对包的内容进行编码操作——不再像传统的存储转发体系结构中典型采用的严格复制转发方式,而是对包的内容进行任意映射。不过,虽然网络编码简单,却没能在网络发展历史中占据一席之地。理由很显然:主导网络发展历史的传统有线技术中,网络编码不是非常实用或者说非常有利。

今天,我们看发展,不仅看新技术,也看新服务。在针对这些新技术、新服务设计新的网络协议时,我们必须小心谨慎,不能因为熟悉而简单移植旧的协议。相反地,我们必须考虑一些其他迄今未使用的思想,它们有可能会对新情况提供更好的解决方案。

网络编码正是这样一种思想。各种理论和实验研究均表明,在多跳无线网络多播会话业务中,采用网络编码可以获得明显增益,这正是迅速崛起的技术或服务的典型代表。下面,以微软的 Avalanche 项目为例进行说明。Avalanche 项目旨在建立端到端的文件分发系统。这些研究促使微软在该项目中采用网络编码作为核心技术来发掘其在多播业务中的优势。至此,网络编码的时机终于成熟。

因此,本书的动机在于:我们认为在未来分组网络设计中网络编码潜能巨大,我们将推进这种潜能的实现。同时,我们也鼓励在该新兴领域开展更多的研究。为此,本书面向两类(不一定完全区分)读者:第一类,从业者,致力于应用;第二类,理论家,致力于网络编码特性的深入研究。针对两类读者,虽然本书内容上是理论的,但我们却倾向于满足第一类应用需求的从业者。我们将采用网络编码应用人员容易理解的方式对理论进行详细论述,进而弥补当前这方面书籍不够的现状。相较于这类书籍,针对理论家的书籍则比较泛滥。除了本书以外,有关网络编码的重要理论结果还可以参考另两本优秀的

概述资料：作者 Yeung 等的网络编码理论^[144,145]和作者 Fragouli、Soljanin 的网络编码基础^{[45]①}。对于从业者，作者 Fragouli、Soljanin 即将出版的网络编码应用^{[46]②}可以作为补充。

本书内容充分参考了我们的博士论文^[55,90]，突出了个人开展工作的偏好。不过，我们仍然努力在正文或总结中涵盖网络编码相关的绝大部分有意义的工作，每章最后一节的备注及导读即为总结。书中难免存在疏漏，提前敬请谅解。

宽泛地说，本书适用于电子工程或计算机科学背景的任何读者。我们使用的一些数学方法，比如代数算法、最优化方法等可能对这类读者而言不熟悉，所以对这些方法不介绍，仅提供适当的相关参考书目。

本书的出版要感谢许多人的帮助和支持。首先要特别感谢我们俩的博士、博士后导师——Medard 教授和 Koetter 教授，他们是我们学习的楷模和专业导师。同时，也要感谢和我们一起进行网络编码研究的优秀团队成员：Ahmed、Chang、Deb、Effros、Eryilmaz、Fragouli、Gerla、Han、Harvey、Jaggi、Karger、Katabi、Katti、Kliwer、Langberg、Lee、Leong、Maymounkov、Pakzad、Park、Ratnakar、Ray、Shi、Traskov、Vishwanath、Viswanathan 和赵方。总体上，整个网络编码团队充满欢笑、友爱，并互相激励，为此我们要感谢所有成员。还要感谢崔涛、Dikaliotis 和 Erez，他们对本书的初稿提出了建设性意见和评价。此外，我们还要感谢另外两个团队，没有他们就没有这本书。第一个是剑桥大学出版社的专业团队，他们伴随着本书的出版，尤其要感谢 Meyler、Littlewood、Barton 和 Karpinski。第二个是我们的家人，感谢他们在我们研究生阶段和本书的写作过程中给予的爱与支持。

Tracey Ho

Desmond S. Lun

于 2007 年秋

① 原书中误写为[44]。

② 原书中误写为[45]。

第 1 章 简介	1
1.1 网络编码的定义	1
1.2 网络编码的效用	2
1.2.1 吞吐量	3
1.2.2 鲁棒性	5
1.2.3 复杂度	7
1.2.4 安全	7
1.3 网络模型	8
1.4 本书概要	10
1.5 备注及导读	11
第 2 章 无失真多播网络编码	12
2.0 符号规定	12
2.1 基本网络模型和多播网络编码问题推导	13
2.2 无延迟标量线性网络编码	14
2.3 可解性和吞吐量	16
2.3.1 单播情况	16
2.3.2 多播情况	17
2.3.3 多信源节点多播	17

2.3.4	最大吞吐量增益	18
2.4	多播网络编码构造	19
2.4.1	集中式多项式时间构造	20
2.4.2	随机线性网络编码	22
2.5	分组网络	25
2.5.1	分组网络中的分布式随机线性编码	26
2.6	有环网络和卷积网络编码	28
2.6.1	卷积网络编码的代数表达	29
2.7	相关信源过程	31
2.7.1	联合信源-网络编码	32
2.7.2	信源编码与网络编码的分离	33
2.8	备注及导读	34
2.A	附录: 随机网络编码	35
第3章	会话间网络编码	44
3.1	标量与向量线性网络编码	45
3.2	分数编码问题公式化	46
3.3	线性网络编码的非充分性	47
3.4	信息论方法	50
3.4.1	多路单播网络	52
3.5	构造方法	53
3.5.1	有线网络中的成对异或编码	53
3.5.2	无线网络中的异或编码	54
3.6	备注及导读	58
第4章	失真网络的网络编码	59
4.1	随机线性网络编码	61
4.2	编码定理	62
4.2.1	单播连接	63
4.2.2	多播连接	74

4.3 有独立同分布(i. i. d)丢包的泊松流错误因子	75
4.4 备注及导读	77
第5章 子图选择	78
5.1 基于流的方法	79
5.1.1 会话内编码	79
5.1.2 计算受限编码	100
5.1.3 会话间编码	100
5.2 基于队列长度的方法	103
5.2.1 多个多播会话的会话内网络编码	104
5.2.2 会话间编码	115
5.3 备注及导读	116
第6章 对抗差错的安全性	117
6.0 符号惯例	118
6.1 差错修正	118
6.1.1 集中式网络编码的纠错界	118
6.1.2 分布式随机网络编码和多项式复杂度纠错	128
6.2 对抗差错检测	133
6.2.1 模型和问题描述	133
6.2.2 检测概率	135
6.3 备注及导读	137
6.A 附录: 对抗差错检测结果的证明	137
参考文献	142

第 1 章

简 介

网络编码是一个新兴的研究领域。2000 年, Ahlswede、Cai、Li 和 Yeung 发表了开创性论文^[3], 标志着网络编码的诞生。网络编码像许多其他新兴领域一样, 存在一定程度的混乱性, 人们既对其可能性充满兴奋, 又对其潜能存在一定争议。本书的首要目的之一是澄清这种混乱。因此, 我们严谨地以网络编码定义开篇。

1.1 网络编码的定义

网络编码的定义不是简明易懂的。目前使用的定义方法有以下几种。

开创性论文[3]中, Ahlswede、Cai、Li 和 Yeung 指出他们将网络中节点进行的编码称为网络编码, 其中编码指由输入到输出的任意因果映射。这是网络编码最广义的定义。然而, 该定义并没有将网络编码的研究与历史悠久的、存在诸多难题的网络信息论或多用户信息论区别开来。由于不希望本书专注于网络信息论(已有一些很好的涵盖网络信息论的文献, 如文献[27]的第 14 章), 我们将对网络编码作进一步的定义。

Ahlsvede 等的文章区别于大多数网络信息论文章的特性之一在于：他们不是研究本质上各节点对其他节点均有任一概率影响的一般性网络，而是明确研究由节点通过端到端无差错链路所连接而成的网络。因此，Ahlsvede 等的网络模型为常规网络信息论研究网络的一个特例，而这种模型与当前网络非常相关，本质上所有的有线网络一旦将物理层抽象为传输比特的无差错管道便均属于该模型。

因此，网络编码的另一个可能定义为无差错链路网络中节点上的编码。该定义区分了网络编码与有噪链路下信道编码的功能；类似地，我们可以区分网络编码与信源编码的功能。这里，网络编码可视为独立、不可压缩信源编码。此定义将被频繁使用，且基于此可将网络编码的研究简化为网络信息论的一个特例。事实上，该特例在 2000 年前已经得到了一些研究（比如，参考文献[51, 131]）。这虽然减少了网络编码的部分创新性，但我们仍能够在该定义下作进一步的研究。

网络编码的大量研究工作集中在网络编码的一个特定形式上，即随机线性网络编码。随机线性网络编码由文献[58]引入，是一种简单的随机编码方法，它为每一个信源过程维持一个系数向量，该系数向量由每个编码节点进行更新。换句话说，随机线性网络编码要求网络传输消息时附加一定程度的额外信息——这里即为系数向量。当今通信网络中，存在一种广泛使用的网络，不仅可以简单地提供这样的额外信息，而且还是由无差错链路组成：分组网络。经分组处理后，这样的额外信息或边信息可以放置在包头，而将边信息放置在包头也的确是现在的通用惯例（比如，经常将序列号放置在包头以保持顺序）。

网络编码的第三种定义为分组网络（对数据进行分组，网络编码应用于分组的内容）节点编码，或者更广义地定义为物理层以上的编码。这与网络信息论不同，后者通常关注物理层编码。本书采用此种定义。局限于分组网络有时确实会不必要地限制我们的研究范围，而且超出分组网络的相关研究成果也不宜作介绍。然而，该定义还是非常有用的，因为它使得我们的论述根植于与实际相关的具体设置。

1.2 网络编码的效用

定义完网络编码，现在我们讨论网络编码的效用。网络编码可以改善吞吐量、鲁棒性、复杂性和安全性。下面依次讨论这些性能因素。

1.2.1 吞吐量

网络编码最为人熟知且最易解释的效用是提高吞吐量。吞吐量的改善由更有效的分包传输实现,即使用更少的分包传输获取更多的信息。Ahlswede 等的论文[3]给出了这种改善的最著名例子,他们考虑有线网络中的多播问题。他们的例子通常被称为蝴蝶网(如图 1.1 所示),该网具有一个信源节点到两个信宿或目的节点的多播特性。两个信宿都想获取信源节点发送的全部信息。在他们考虑的约束网络中,只有当一个中间节点(非信源/信宿节点)打破分组网络中传统的复制转发路由方式,并进行编码操作后,才能够建立所需的多播连接。该中间节点对接收到的两个包进行二进制加法或者异或处理得到一个新包,并将该新包输出。因此,如果两个接收包的内容为比特向量 b_1 和 b_2 ,则输出包为 $b_1 \oplus b_2$,即向量 b_1 和 b_2 的逐位异或。两信宿节点分别对其接收到的数据包作进一步的编码处理实现解码。信宿节点 t_1 对 b_1 和 $b_1 \oplus b_2$ 进行异或处理解出 b_2 ,同样的,信宿节点 t_2 对 b_2 和 $b_1 \oplus b_2$ 进行异或处理解出 b_1 。传统路由模式下,我们可以传输信息 b_1 和 b_2 到节点 t_1 ,但是同时只能够传输信息 b_1 或 b_2 中的一个到节点 t_2 。

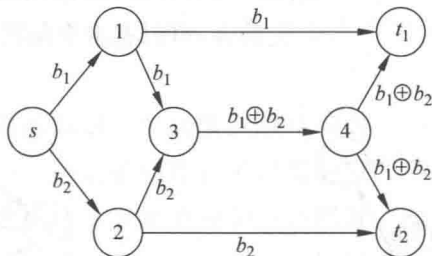


图 1.1 蝴蝶网。该网络中,每条弧代表能够可靠传输一个包的有向链路。

信源节点 s 有两个包 b_1 和 b_2 待传输,我们希望将这两个包的内容同时通信至信宿节点 t_1 和 t_2 。

设计该蝴蝶网说明了重要的一点:网络编码能够增加有线网络的多播吞吐量。蝴蝶网中 9 个包传输完成了两个信息包的通信。不进行编码的情况下,通过 9 个包的传输无法完成同样多信息的通信,必须通过补充额外的传输(比如,节点 3 到节点 4 的额外传输)来实现。

虽然网络编码能够增加有线网络的多播吞吐量,但是其对吞吐量的改进不仅仅局限于多播或者有线网络。对蝴蝶网进行简单修改,可以得到一个涉及两个单播连接的例

子,它在编码情况下可以实现,而无编码则做不到,如图 1.2 所示。本示例包括两个单播连接。目前考虑的无失真有线网络单播传输,至少需要两个单播连接网络编码才能够带来吞吐量增益。在 2.3 节中我们将详细证明,网络编码在无失真有线网络仅存在一个单播连接下不能带来吞吐量增益。

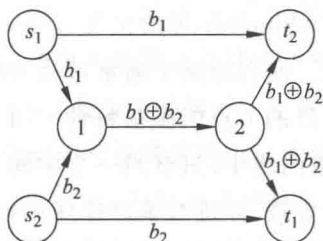


图 1.2 改变式蝴蝶网。该网络中,各条弧代表能够可靠传输一个包的有向链路。源节点 s_1 有一个包 b_1 需要传输给宿节点 t_1 ,而源节点 s_2 有一个包 b_2 需要传输给宿节点 t_2 。

网络编码也可以扩展到无线网络。无线网络中,更容易找到例子说明网络编码较之传统路由会带来吞吐量增益。确实,蝴蝶网所对应的无线网络(如图 1.3 所示)和改变式无线蝴蝶网(如图 1.4 所示)分别只涉及 6 个以及 3 个节点,相比于原来的 7 个和 6 个要少一些。和前述一样,这些例子说明使用路由方式无法达到所需通信目标,而采用编码方式能够实现。和有线网络不同的地方在于,这些无线例子中不再假设从单一节点到另一单一节点的包传输,而是允许从单一节点到多个节点的包传输。因而,我们不再使用弧表示传输,而使用超弧表示之,这里超弧是更广义的弧,可以有多个信宿节点。

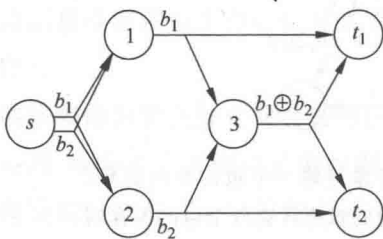


图 1.3 无线蝴蝶网。该网络中,各超弧代表能够可靠传输一个包到一个或者多个节点的有向链路。信源节点 s 有两个包 b_1 和 b_2 需要同时传输给宿节点 t_1 和 t_2 。

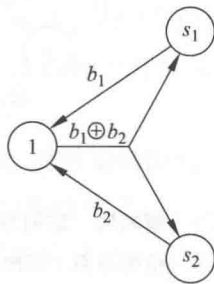


图 1.4 改变式无线蝴蝶网。该网络中,各超弧代表能够可靠传输一个包到一个或者多个节点的有向链路。信源节点 s_1 有一个包 b_1 需要传输给节点 s_2 ,而信源节点 s_2 有一个包 b_2 需要传输给节点 s_1 。

到目前为止,我们已经讨论的例子说明,即使在没有丢失和误码的情况下,只要应用于一个或多个同时传输的多播连接或者两个或多个同时传输的单播连接,网络编码就能

够带来吞吐量增益。该结论对信息包从一个单一节点传输到另一单一节点(有线网络)或者从一个单一节点传输到一个或多个节点(无线网络)都成立。然而,这些例子看起来像是设计的微型示例,不具备一般性,所以很容易使人产生疑问,网络编码是否可以推广呢?即使可以,又会是什么情况呢?本书后续大部分内容将致力于推广前述结论到更一般的情况。

1.2.2 鲁棒性

1.2.2.1 丢包鲁棒性

在进一步论述之前,先强调分组网络尤其是无线分组网络中一个到目前为止我们一直忽略的重要问题——丢包。丢包由网络中的多种原因造成,包括缓存溢出、链路中断和冲突。处理这些丢包的方法很多。其中,传输控制协议(TCP)使用的机制可能是最直接的。TCP中设置应答(Ack)系统,信宿节点接收到数据包后传输一个Ack信息给信源节点,如果信源节点没有收到某特定包的Ack信息,则重传该数据包。另一个有时使用的可选方法是信道编码,或者更确切的称为纠错码。信源节点运用纠错码对数据包进行一定程度的冗余,这样即使信宿节点仅接收到信源节点发送的部分数据包,也能恢复出全部信息。

纠错码是信源节点采用的编码。中间节点怎样编码呢?即如何进行网络编码呢?网络编码是否可以帮助对抗丢包呢?确实可以。下面通过一个简单的例子说明其原因。考虑简单双链路序列网络,如图1.5所示。该网络中,节点1到节点2链路的丢包率为 ϵ_{12} ,节点2到节点3链路的丢包率为 ϵ_{23} 。节点1使用纠错码后,可以以每单位时间 $(1-\epsilon_{12})(1-\epsilon_{23})$ 个数据包的速率进行通信。本



图 1.5 双链路序列网络。节点1和节点2分别可以在其输出链路上每单位时间传输1个包。

质上,节点1到节点3之间为一个删除概率为 $1-(1-\epsilon_{12})(1-\epsilon_{23})$ 的删除信道,其容量为 $(1-\epsilon_{12})(1-\epsilon_{23})$,可由一个适当设计的编码达到。然而,系统的真正容量可以更大。如果在节点1和节点2之间的链路上采用一个纠错码,在节点2和节点3之间的链路上采用另一个纠错码,即节点2分两步进行完全译码和再编码,则节点1和节点2之间可以以每单位时间 $(1-\epsilon_{12})$ 个数据包的速率进行通信,节点2和节点3之间可以以每单位时间 $(1-\epsilon_{23})$ 个数据包的速率进行通信。因此,节点1和节点3之间可以以 $\min(1-\epsilon_{12}, 1-\epsilon_{23})$ 的速率进行通信,这通常比 $(1-\epsilon_{12})(1-\epsilon_{23})$ 要大。

那么为何分组网络中不使用这种解决方案呢? 关键的原因是延迟。无论使用分组码或循环码, 每一步纠错码都将产生一定的延迟, 因为每步译码器都需要接收到一定数量的包后才能开始译码。这样, 如果每条链路或者连接均采用纠错码, 将带来巨大的总延迟。然而, 使用纠错码这样的额外步骤仅仅是一种特殊形式的网络编码——这里中间节点进行了编码。因此, 网络编码可以用于提高对丢包的鲁棒性, 从而可以转化为吞吐量增益。不过, 我们需要的网络编码解决方案不仅仅能够提高吞吐量, 而且要超越单单使用纠错码的额外步骤, 即中间节点采用额外编码且不进行译码。本书第 4 章将讨论随机线性网络编码如何满足这种编码方案的要求。

丢包在网络编码问题中增加了一个额外的因素。丢包存在的情况下, 即使仅有一个单播链路也可以获得增益。丢包与无线网络非常相关, 并且考虑丢包因素使得网络编码与无线应用联系更紧密。我们讨论过无线网络的另一个特性是存在广播链路, 即链路可以传输到一个以上的终端节点, 因此还需要将丢包和广播链路结合起来。

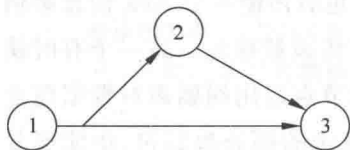


图 1.6 分组中继信道。节点 1 和节点 2 分别可以在其输出链路上单位时间内传输 1 个包。

图 1.6 中给出了一种改进的双链路序列网络, 我们称之为分组中继信道。这里, 节点 1 的输出链路不仅到达节点 2, 而且到达节点 3。然而, 由于存在丢包, 节点 1 发出的包是否节点 2 和节点 3 都未接收、仅节点 2 接收、仅节点 3 接收或者节点 2 和节点 3 都接收是由概率分布决定的。现在, 假定节点 1 传输的包仅被节点 2 接收的概率为 $p_{1(23)2}$, 仅被节点 3 接收的概率为 $p_{1(23)3}$, 被节点 2 和节点 3 同时接收的概率为 $p_{1(23)(23)}$ (全部丢失

的概率为 $1 - p_{1(23)2} - p_{1(23)3} - p_{1(23)(23)}$)。对节点 2 的包传输, 假定节点 2 传输的包被节点 3 接收的概率为 p_{233} (彻底丢失的概率为 $1 - p_{233}$)。该设置下, 网络编码, 特别是随机线性网络编码能够实现最大可达吞吐量, 即最小割容量, 为 $\min(p_{1(23)2} + p_{1(23)3} + p_{1(23)(23)}, p_{1(23)3} + p_{1(23)(23)} + p_{233})$ 。

这是相当好的结论: 首先, 从网络信息论的观点, 是否存在一种简单的、可达容量的网络编码甚至还不清楚; 其次, 对无线分组网络而言, 它提供了主流方法的一个重大替代。主流的路由方法提倡将无线分组网视为有线分组网的扩展。因而, 它提倡经路由传输信息; 即在前述例子中, 要么节点 1 传输给节点 2 后再传给节点 3, 要么直接从节点 1 传输至节点 3, 要么使用更复杂的方案将前面两种方式结合。采用网络编码后, 不再有这样的路径概念, 即节点对特定连接提供传输, 但这些节点不一定必须沿着一条路径。因

此,有必要重新思考路由。这种思考带来了子图选择,我们将在第5章讨论。

1.2.2.2 链路中断鲁棒性

除了具备对随机丢包的鲁棒性,网络编码还可以用于对非遍历链路中断的保护。实时路径保护对每个连接提供主备流传输,这样不需要重新路由,从而可以快速恢复链路中断。然而,这种方法使得网络流量倍增。网络编码通过在不同流之间共享网络资源,能够提高资源利用率。任一多播会话下对可通过任意重新路由机制恢复的任何中断模式集合,总是存在一种静态网络编码方案,使得该集合中的任意中断模式均不使用重新路由即可恢复^[82]。

1.2.3 复杂度

某些情况下,虽然最优路由可能达到近似于网络编码的性能,然而最优路由解却很难获得。比如,多播路由的最小代价子图选择涉及 Steiner 树,该问题即使在集中设置下也很复杂,而相应的网络编码问题则仅是线性最优化问题,它可生成低复杂度分布式解。这些将在 5.1.1 节中进一步讨论。

有些场景下,受实际限制必须选择次优方案,例如,基于闲谈的数据分发^[32]以及 802.11 无线自组织网络^[74]。经论证,网络编码在这些情况下也能够大幅提高性能。这些将在 3.5.2.2 节讨论。

1.2.4 安全

从安全性角度看,网络编码既有利处也有弊端,以蝴蝶网(如图 1.1 所示)为例说明。假如对手仅仅获取了包 $b_1 \oplus b_2$,单单只有包 $b_1 \oplus b_2$,对手不能获取 b_1 或 b_2 。因此,我们具备安全通信的可能方案。这个例子说明网络编码能够提供安全增益。

另一方面,假设节点 3 为恶意节点,不传送真实的包 $b_1 \oplus b_2$,而是传送伪装成 $b_1 \oplus b_2$ 的包。由于数据包是由编码而非路由得到,对数据包的篡改很难发现。这个例子说明网络编码带来了潜在的安全隐患。后续将在第 6 章中讨论网络编码在安全方面的应用。

目前,我们已经给出了一些微型示例以说明网络编码带来的好处。这些例子很明显与分组网络相关,而它们说明的原理在实际情况下如何使用则可能不同。下一节中我们将给出模型说明更一般的情况。