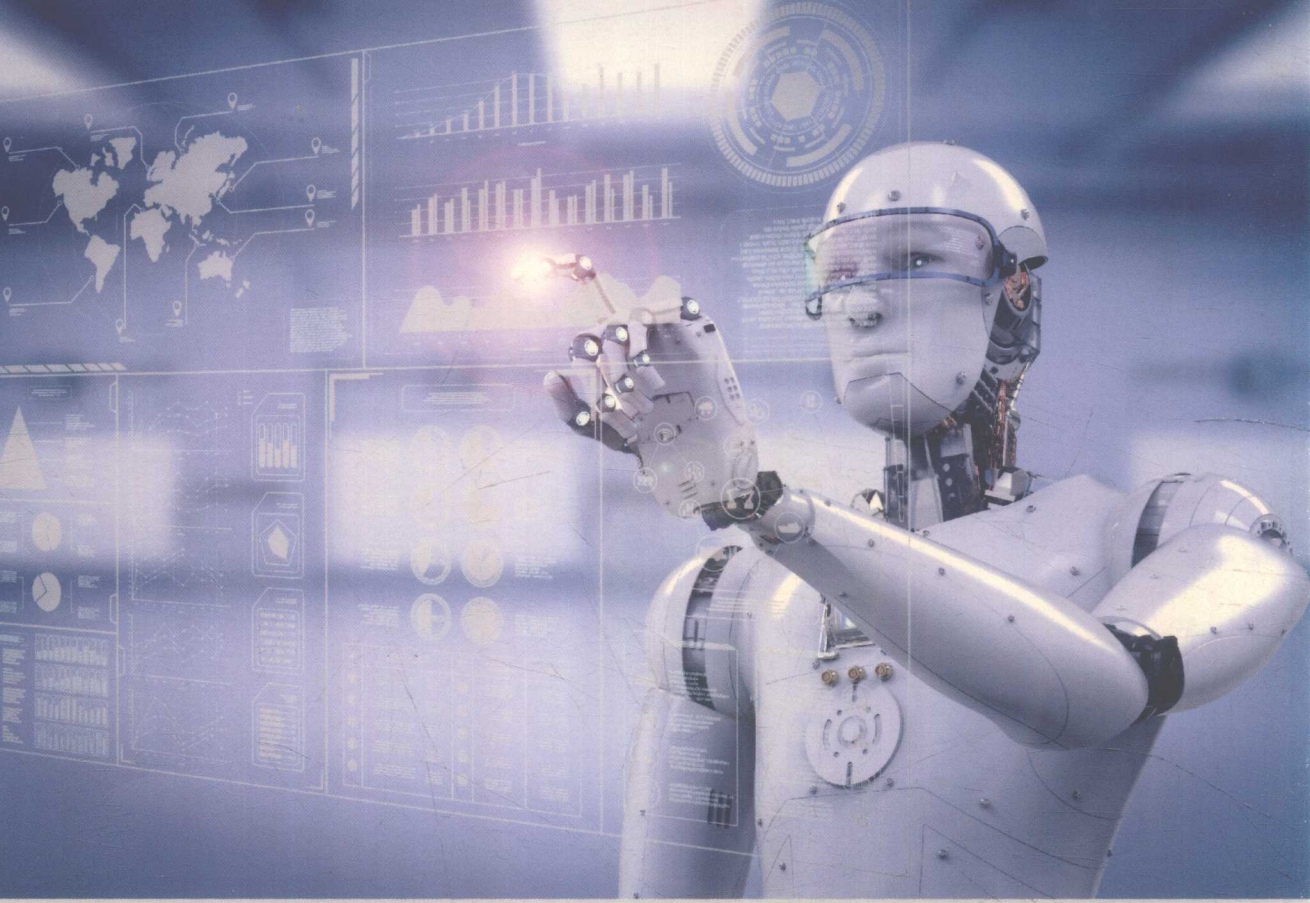


基于负载均衡结构的高性能交换技术及仿真研究

申志军 编著

西安电子科技大学出版社
http://www.xduph.com

基于负载均衡结构的高性能交换技术及仿真研究

申志军　编著

西安电子科技大学出版社

内 容 简 介

本书以交换技术领域中最新的研究成果为核心，系统地介绍了基于负载均衡结构的高速交换技术方案和网络仿真的一般方法。全书分为三篇，第一篇介绍交换技术概况，分析交换技术的应用领域和发展状况；第二篇首先分析负载均衡结构的起源、发展和面临的问题，随后系统地阐述了该领域最新的研究进展和成果；第三篇重点介绍 Opnet 软件仿真中数据流模型的创建和交换技术仿真案例。

本书可作为网络科研和相关工程技术人员的参考资料。

图书在版编目(CIP)数据

基于负载均衡结构的高性能交换技术及仿真研究/申志军编著. —西安：西安电子科技大学出版社，2018.8

ISBN 978-7-5606-4976-4

Ⅰ. ① 基… Ⅱ. ① 申… Ⅲ. ① 通信交换—研究 ② 计算机网络—计算机仿真—研究

Ⅳ. ① TN91 ② TP393.01

中国版本图书馆 CIP 数据核字(2018)第 159830 号

策划编辑 刘玉芳
责任编辑 杨 薇
出版发行 西安电子科技大学出版社(西安市太白南路 2 号)
电 话 (029)88242885 88201467 邮 编 710071
网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com
经 销 新华书店
印刷单位 北京虎彩文化传播有限公司
版 次 2018 年 8 月第 1 版 2018 年 8 月第 1 次印刷
开 本 787 毫米×960 毫米 1/16 印张 8.5
字 数 167 千字
印 数 1～1000 册
定 价 25.00 元
ISBN 978-7-5606-4976-4/TN
XDUP 5278001-1

前　言

Internet 中多媒体业务流的激增，云计算、5G 等一系列新型网络服务和新型通信技术的不断涌现都使得 Internet 面临着越来越大的数据传输压力。随着未来高清数字电视和“三网合一”的逐步推进，通信网络必将承载更多的用户数据，因此，实现高速的数据传输已经成为下一代 Internet 的核心问题之一。

能否实现网络终端之间的高速通信，关键就在于由通信介质和中继设备组成的数据传输通路能否提供这种高速的数据传输能力。在传输介质方面，光通信技术，特别是密集波分复用 DWDM (Dense Wavelength Division Multiplexing)技术的发展极大地提高了光纤的数据传输带宽。但交换技术的发展却远远滞后于光通信技术的发展，这使之成为制约 Internet 性能的瓶颈。因此，提高中继设备的数据交换速率成为提高 Internet 数据传输能力的关键。

基于这种背景，本书重点介绍基于负载均衡结构的高速交换技术和相关研究领域的仿真技术。期望通过将最新、最准确的信息传递给读者来进一步推动交换技术的发展，从而提高中继系统的交换能力，提高 Internet 的数据传输带宽以造福广大网络用户。

在本书编写过程中，作者尽可能把相关知识进行归纳和总结，但由于时间仓促和作者水平所限，书中难免存在疏漏和不当之处，欢迎读者批评指正。

作　者

2018 年 1 月

目　录

第一篇　交换技术概况

第1章 绪 论

1.1 背景和意义

随着互联网用户数量的迅猛增长和多媒体业务流的激增，Internet 面临着越来越大的数据传输压力。据国际数据公司预测，到 2018 年全球网民数将达到 30 亿人[1]，相当于世界总人口的 40%；2015 年 2 月 3 日，中国互联网络信息中心(CNNIC)发布的《第 35 次中国互联网络发展状况统计报告》[2]显示：截至 2014 年 12 月底，中国网民规模达到 6.49 亿人，互联网普及率为 47.9%。与此同时，多媒体业务数据在 Internet 数据流中的比重越来越大，其中视频，特别是高清视频业务对网络传输带宽的消耗是惊人的。这使得网络用户对大量数据高速传输的需求与接入速率过低的矛盾日益突出。此外，随着高清数字电视和“三网合一”的逐步推进，未来的通信网络将要承载更多的视频数据，Internet 必然会面临更大的数据传输压力。因此，实现高速的数据传输已经成为下一代 Internet 的核心问题之一。

图 1-1 所示为 Internet 数据传输通路的概念化模型。从图中可以看出：网络终端存在大量的高速数据传输需求，能否实现网络终端之间的高速通信，关键就在于由通信介质和中继设备组成的数据传输通路能否提供这种高速的数据传输能力。

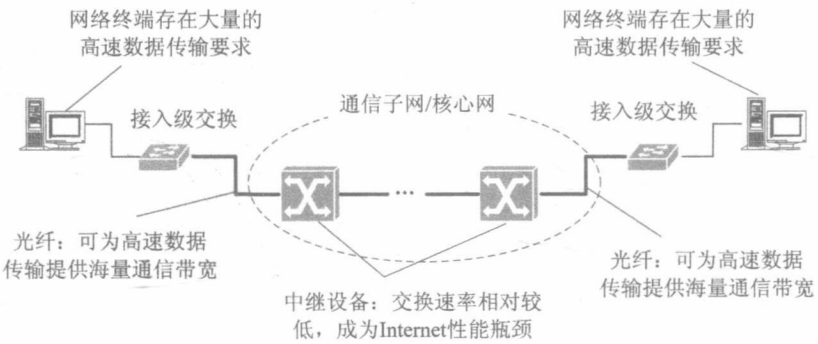


图 1-1 Internet 数据传输通路的概念化模型

在传输介质方面：光通信技术，特别是密集波分复用 DWDM (Dense Wavelength Division Multiplexing)技术的发展极大地提高了光纤的数据传输带宽。阿尔卡特公司已经

在单根光纤上实现了 256 个波长复用，NEC 公司甚至实现了 274 个波长复用。每个波长的数据传输率以 80 Gb/s(OC－1536 标准)计算，256 个波长复用的单根光纤所能提供的数据传输带宽高达 20 Tb/s(256×80 Gb/s)，这就意味着光纤已经在传输介质方面为 Internet 的高速数据传输提供了可能。

在中继设备方面：截至目前中继设备的最高端口速率是由华为公司保持的 400 Gb/s[3]。这一数据表明：现有中继设备的数据交换速率远远低于光纤所能提供的数据传输率(400 Gb/s≪20 Tb/s)，这使之成为制约 Internet 性能的瓶颈。因此，有必要开展与高性能中继设备相关的技术研究以提高其数据交换速率，进而提高 Internet 数据传输带宽。

传统的交换结构中，每个时隙均需进行一次调度过程来选择合适的数据包并将其转发出去。在包长一定的情况下，提高转发速率势必要缩短时隙长度，这就意味着交换结构所容许的算法执行时间缩短了。倘若算法的复杂度高于 $O(1)$，不妨设为 $O(N)$(N 为交换结构输入端口/输出端口数)，则端口速率的提高所导致的算法的时域空间的缩减必然导致两个可能的结果：其一是为适应高速交换所导致算法时域空间的缩减而减少交换规模；其二是为保证一定的交换规模而限制交换速率的提高。反之，若交换结构的算法复杂度为 $O(1)$，则算法耗时与交换规模 N 无关，从而可以有效缩短时隙长度，进而支持更高的交换速率和更大的交换规模。

基于上述分析可知，若要实现高速的数据转发，就必须采用全流程 $O(1)$复杂度的新型交换结构。然而传统的交换结构因复杂度或加速比的原因无法有效满足未来的高速交换需求。张正尚教授等人提出的负载均衡结构 LB－BvN(Load Balanced Birkhoff-von Neumann switch architecture)[4, 5]采用两级 crossbar 和必要的缓冲组成，其两级 crossbar 均采用确定的、周期性的连接模式，这种具有 $O(1)$复杂度的连接模式可以有效缩短时隙长度从而使高速转发成为可能。此外，其第 1 级 crossbar 能够将到达输入端口的数据流均匀散布到中间缓存，从而使得该结构能够较好地适应自相似业务流。负载均衡结构的以上两点优势使之成为交换技术领域的研究热点。但负载均衡交换结构中数据包有可能通过不同的转发路径到达输出端，这样就可能因为中间缓存的队列长度差异而导致数据包在输出端失序。国内外研究机构为解决这一问题做了大量研究，但现有解决方案[6—13]或者复杂度过高，或者交换性能不够理想。一方面若解决失序问题的复杂度高于 $O(1)$，则必然会使整个交换流程迟滞，进而使得负载均衡结构原本的高速交换能力失去了意义。另一方面，若为片面追求全流程 $O(1)$复杂度而付出过高的性能代价也是不可取的。

基于 Internet 对高速交换的迫切需求和负载均衡结构的研究现状，本书以全流程 $O(1)$复杂度为约束条件，介绍能够满足未来高速交换需求的负载均衡交换结构。

1.2 主要内容

本书首先介绍交换技术的基本概念和应用领域，随后介绍交换技术的几类主流解决方案并针对现有解决负载均衡结构中数据包失序问题的方案所存在的复杂度过高和交换性能不够理想的现象，分别从基于时延戳[14]、基于 Flow Splitter[15, 16]和基于反馈机制[17—20]的角度研究能够实现全流程 $O(1)$ 复杂度且具有更优交换性能的负载均衡结构及相关算法。

本书重点介绍的内容如下：

(1) 基于时延戳的方法提出一种“智能维序”的负载均衡交换结构(Smart Load Balanced switch Architecture，SLBA)[14]，SLBA 通过引入 crossbar 的反向通信模式和“智能维序”的重排序机制实现了全流程 $O(1)$ 复杂度并有效解决了数据包失序问题。

(2) 分析相关文献指出 Byte-Focal 存在复杂度、伪队首阻塞(Pseudo-Head-of-Line blocking，PHOL)和惯性服务模式等问题，在此基础上提出将 Flow Splitter 和 Byte-Focal 显式结合的负载均衡交换结构 CFSB (Combine Flow Splitter with Byte-Focal)[15]，CFSB 实现了全流程 $O(1)$ 复杂度并有效解决了数据包失序问题。和 SLBA 相比，CFSB 具有更简单的交换结构，且无需在交换结构和线卡之间进行额外的通信，从而得以避免在超大规模和多机柜交换环境中的长往返时间(Round Trip Time，RTT)问题。

(3) CFSB 中所采用的 Flow Splitter 和 Byte-Focal 显式结合方案无法保证数据包离开第 1 级时保持先入先出特性，这一缺陷会导致两个结果：其一是 CFSB 的重排序时延和系统时延会增加，在某些流量环境中可能会恶化；其二是 CFSB 需要在输出端设置更大容量的重排序缓存。为解决这一问题，本书提出将 Flow Splitter 和 Byte-Focal 隐式结合的负载均衡交换结构 LB-IFS(Load Balanced switch based on Implicit Flow Splitter)[16]，LB-IFS 采用双缓冲模式和两步调度策略克服了 CFSB 的缺陷，LB-IFS 同样在线卡和交换结构之间无需额外通信的前提下以全流程 $O(1)$ 复杂度解决了数据包失序问题，且其时延性能优于 CFSB 和 Byte-Focal。

(4) 尽管 LB-IFS 结构以相对简单的结构实现了较为优异的交换性能，但相对于迄今为止理论性能最优的负载均衡交换结构 FTSA (Feedback-based Two-stage Switch Architecture)[13]而言，其交换性能仍存在明显不足。然而 FTSA 结构自身也存在着若干缺陷，如该结构需要在交换结构和线卡之间进行额外的通信，这使之无法有效应用于超大规模和多机柜交换环境，但考虑到在较为一般的交换环境中，较大的性能优势仍使其具有可观的实践价值。本书经分析发现 FTSA 还存在算法复杂度较高以及要求算法在极短的时间内完成等问题。针对这些问题，本书提出“开源”和“节流”两种方案。所谓“开源”即通过拓展算法的时域空间缓解 FTSA 对算法执行时间的苛刻限制。所谓“节流”即在算法有限的调度时间内尽可能降

低算法复杂度，进而降低算法调度耗时。本书首先基于“二次反馈”的思想提出“开源”方案——DFTS (Double-Feedback-based Two-stage Switch architecture)[17]结构。相对于FTSA结构，DFTS能够有效拓展算法的时域空间，且在理论上二者具有等价的交换性能。

(5) 本书将嵌入式系统中的优先级位图算法(Priority Bitmap Algorithm，PBA)与FTSA中的最早离去者优先算法[13](Earliest Departure First，EDF)相结合提出“节流”方案——PB-EDF(Priority Bitmap-based Earliest Departure First)[18]算法，该算法利用EDF按固定的顺序检索所有 N 个队列的特点，将各缓冲队列映射为具有不同优先级的任务，在此基础上利用PBA以 $O(1)$ 复杂度实现调度过程。引入PB-EDF算法使得在反馈制负载均衡交换结构中实现了全流程 $O(1)$ 复杂度，同时PB-EDF算法还继承了PBA调度耗时为定值的优点。因PB-EDF的判决过程完全遵循文献[13]中的EDF算法，故在相同的交换环境中二者具有等价的调度性能。

(6) 由于作为“开源”方案的DFTS结构需要在其第1级调度中尽可能获得两个调度结果，故该结构无法和作为“节流”方案的PB-EDF算法一起协同工作。为解决这一问题，本书基于“前置反馈”的思想提出能够和PB-EDF协同工作的“开源”方案FFTS (Front-Feedback-based Two-stage Switch architecture)[19]结构，FFTS通过将反馈操作提前到数据包传输之前的方法有效拓展了算法的时域空间，但其为解决由此而带来的数据包冲突和失序问题而导致其交换性能略低于FTSA的理论性能。尽管如此，理论和仿真表明其时延性能依然远优于其他非反馈制负载均衡交换结构。在FFTS的基础上，本书还通过引入一种2-错列对称的crossbar连接模式(2-Staggered Symmetry connection pattern，2-SS)提出一种改进的“开源”方案FTSA-2-SS (FTSA using 2-Staggered Symmetry connection pattern)[20]结构，FTSA-2-SS在获得与FFTS等价交换性能的前提下能够为算法拓展更大的时域空间。

(7) 使用Opnet网络仿真软件对交换技术仿真的一般方法，包括Opnet的工作机制，数据流模型建模，以及关于OQ、iSLIP结构的仿真示例等。

1.3 结构安排

第1章对交换技术的研究背景、意义、对象和目标等进行概括性的介绍，第2章介绍交换结构领域的研究进展，重点介绍国内外研究机构对负载均衡结构的数据包失序问题所提出的各种解决方案并分析其优势与不足。

第3～7章是本书的主要内容，重点介绍负载均衡结构的交换技术方案。第3章介绍“智能维序”的负载均衡交换结构SLBA；第4章介绍基于Flow Splitter的负载均衡交换结构CFSB和LB-IFS；第5章介绍FTSA的“开源”方案DFTS；第6章介绍“节流”方案PB-EDF算法；第7章介绍“开源”方案FFTS和FTSA-2-SS；第8章介绍Opnet软件的

特性和工作机制；第 9 章介绍 Opnet 中的常用数据流模型；第 10 章和第 11 章简介 OQ 和 iSLIP 的仿真模型。

本书最后列出了参考文献和 Opnet 仿真常见错误的解决方法以及缩略语说明。

1.4 相关约定

为便于讲述，本书做以下约定：

(1) 交换结构的输入端口和输出端口数均记为 N，两级 crossbar 分别记为 XB1，XB2；

(2) 在不引起混淆的情况下，“输入端口”均指 XB1 的输入端口，“中间端口”均指 XB2 的输入端口，“输出端口”指 XB2 的输出端口；

(3) 序号为 i 的输入端口记为 I_i，序号为 j 的中间端口记为 M_j，序号为 k 的输出端口记为 O_k；

(4) I_i与 M_j相连记为 I_i-M_j，M_j与 O_k相连记为 M_j-O_k，I_i通过 M_j与 O_k相连记为 I_i-M_j-O_k；

(5) M_j在 t 时隙起始时刻的缓存队列状态数据记为 $QS_j(t^b)$，M_j在 t 时隙结束时刻的缓存队列状态数据记为 $QS_j(t^e)$，$QS_j(t^b)$和 $QS_j(t^e)$都仅有 N 个 bit，其第 v 位为“1”表示 $VOQ2(j,v)$非空，反之表示 $VOQ2(j,v)$为空。QS 意为 Queue Status。

(6) t 时隙到达 M_j的数据包信息记为 $ToM_j(t)$，$ToM_j(t)$仅有 N 个 bit 且最多只能有 1 个 bit 为“1”，其第 v 位为“1”表示到达的是输出端口为 v 的数据包，若 $ToM_j(t)=0$ 则表示 t 时隙无任何数据包到达 M_j。

(7) t 时隙到达输入端口 I_i的数据包信息记为 $ToI_i(t)$，$ToI_i(t)$仅有 N 个 bit 且最多只能有 1 个 bit 为“1”，其第 v 位为“1”表示到达 I_i的是输出端口号为 v 的数据包，$ToI_i(t)=0$ 表示 t 时隙无数据包到达 I_i。

(8) I_{i-2}在 t 时隙开始的调度结果记为 $SR_{i-2}(t)$，SR 意为 Schedule Result，对 $SR_{i-2}(t)$的终裁结果记为 $FD_{i-2}(t)$，FD 意为 Final Decision。$SR_{i-2}(t)$和 $FD_{i-2}(t)$仅有 N 个 bit 且最多只能有 1 个 bit 为“1”，其第 v 位为“1”表示调度或终裁选择的是输出端口号为 v 的数据包，其值为 0 表示未选择任何数据包。

(9) crossbar 重配置时间记为 T_R，交换端口发送 N 个 bit 的数据传输至下一个端口的发送和传播时延之和记为 T_N；输出端口将 N 个 bit 的数据反馈至位于同一线卡的输入端口的传输时延记为 T_F，输入端口进行数据处理的耗时记为 T_P；一个数据包在 XB1 或 XB2 上的传输时延和传播时延之和记为 T_X。因 T_R、T_N、T_F、T_P等均耗时极短，故记 $T^*=\max(T_R, T_N, T_F, T_P)$。

(10) 假定交换机内传输的数据包具有相同的长度。交换结构各类端口号的加减操作都要对 N 取模，即 $i-1$ 实质上表示是 $(i-1) \bmod N$。

第 2 章　交换技术的发展历程

作为交换机和路由器的关键组件，交换结构的发展已历经从时分交换到空分交换，从单级 crossbar 到多级 crossbar，从单平面交换到多平面交换等多个阶段。作为后续研究工作的基础，本章首先介绍中继系统和交换结构，而后分两大类分别讨论各种典型的交换结构及其相关算法，其中特别针对本书的重点内容——负载均衡交换技术进行深入分析。

2.1　中继系统和交换结构

将不同的网络连接在一起时必须使用相应的网络互联设备，ISO 将这类设备统称为中继系统。现有的网络互联设备依据其所工作的 OSI 层次的不同主要分为以下四种[21]：

(1) 中继器(Repeater)：中继器是物理层的互联设备，其主要功能在于恢复和放大数据信号从而物理地延长数据传输的距离。

(2) 网桥(Bridge)：网桥是数据链路层的中继设备，最初的网桥设备只能用于同类型的两个局域网之间互联，且一般以软件的形式实现其“存储转发”功能。随着技术的发展，网桥逐步演变为多端口的数据链路层交换设备，即二层交换机。二层交换机以硬件为支撑，易于实现高速交换，是局域网、园区网和城域网的主要交换设备之一。

(3) 路由器(Router)：路由器是网络层的存储转发设备，通常特指以 IP 为基础的存储转发设备。低速通信环境中的路由器主要以软件为依托进行转发。随着交换技术的发展，出现了在网络层进行交换的三层交换机，现代路由器与三层交换机之间的界限正在淡化，从而出现了路由交换机或交换路由器之类的称呼。

(4) 网关(Gateway)[22]：网关是一个较为模糊的概念，它是网络层之上的协议转换或封装设备。一个网关具体属于哪一层取决于它涉及的协议转换与封装层次。

通常，二层以上的中继设备都采用存储转发的工作方式，如图 2-1 所示，其结构[23]可以用输入单元、输出单元和交换结构(包括调度与仲裁机制)来描述。从某种程度上讲，交换结构的性能优劣决定着其转发性能，因此，研究能够适应未来高速交换环境的交换结构就成为 NGI 的核心技术之一。

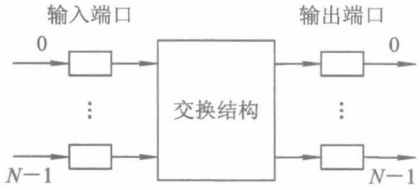


图 2-1 交换结构示意图

交换结构可分为时分交换结构(Time - Division Switching, TDS)和空分交换结构(Space - Division Switching, SDS)[23]，如图 2-2 所示。

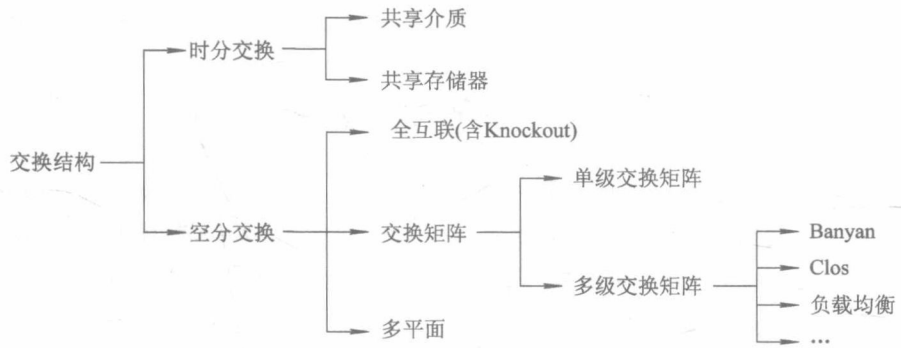


图 2-2 交换结构的分类

TDS 结构中各输入端口的信元通过时分复用的方法通过一个公共的数据通道转发数据包，这就决定了该数据通道必须与所有输入端口和输出端口相连，共享介质结构和共享存储器结构[24—27]是两种典型的 TDS 结构。

SDS 结构的典型特征是在无冲突的情况下，输入端口和输出端口均不同的多个信元可在同一时刻经不同的转发路径到达输出端。理论上其传输带宽等同于单个传输通路的传输带宽与通路数量的乘积，但实践中 SDS 结构往往受到芯片引脚数目以及背板连接等问题的限制。

2.2 时分交换结构

2.2.1 共享介质

共享介质型的交换结构如图 2-3 所示，所有输入端口均直接与一条公共的高速总线(环)相连，信元到达各输入端口后通过时分复用汇集到该总线(环)，同时与之相连的地址

过滤器(Address Filter，AF)检测到达公共总线上的所有信元并只允许目的地址为本端口的信元进入相应的缓存中等待转发。图 2-3 中的输出端缓存以 FIFO(First In First Out)模式为例。

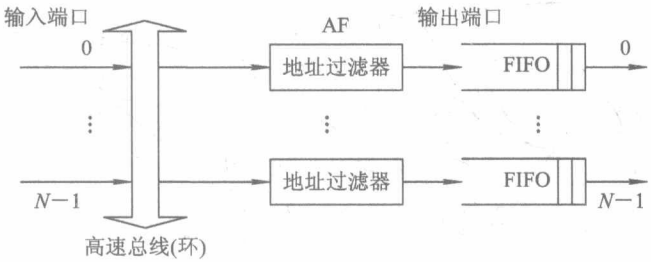


图 2-3　共享介质型交换结构

TDS 结构具有结构简单、易于实现的优点，同时能够方便地支持多播操作。但由于 TDS 结构要对所有到达输入端口的信元进行时分复用的处理，故对其内部通信带宽要求较高，这一缺陷限制了其交换规模的扩大。在极端情况下，可能会有 N 个目的端口相同的信元同时汇聚到公共总线。此时，在一个时隙内输出端口的 FIFO 必须完成 N 个信元的写入和 1 个信元的读出操作，若将一个时隙的时间记为 T_{SLOT}，将存储器的存取周期记为 T_{MEM}，则该存储器必须满足：

$$(N+1)\leqslant\frac{T_{\mathrm{SLOT}}}{T_{\mathrm{MEM}}} \tag{2-1}$$

公式(2-1)表明 T_{SLOT} 和 T_{MEM} 决定了共享介质结构的交换规模。对于信元长度为 64 字节、端口速率为 40 Gb/s 的共享介质结构而言，T_{SLOT} 仅为 12.8 ns，若 T_{MEM} 为 2 ns，必有 $N\leqslant 5.4$，即交换端口数不能超过 5。

2.2.2　共享存储器

共享存储器结构[24—27]将到达各输入端口的信元通过时分复用的方法缓存于公共的存储器，而后通过集中式调度算法将存储器中的信元分别调度至各自的输出端口，其结构如图 2-4 所示。共享存储器结构的优势同样是逻辑简单，易于实现，且由于其存储器为所有端口共享，故其利用率相对较高。其缺点是对存储器的存取速率要求较高，极端情况下需要在一个时隙的时间内完成 N 个信元的写入和 N 个信元的读出操作，即共享存储器必须满足：

$$T_{\mathrm{MEM}}\leqslant\frac{T_{\mathrm{SLOT}}}{2N} \tag{2-2}$$

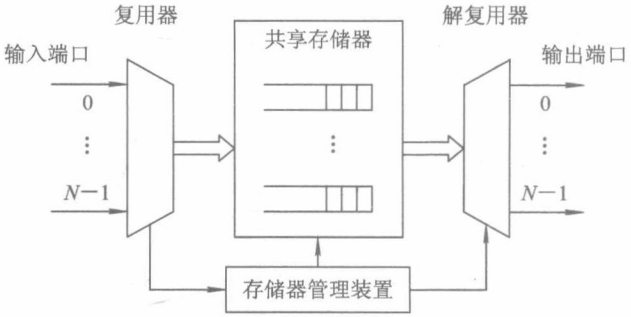


图 2-4 共享存储器型交换结构

若信元长度为 128 Byte，端口速率为 10 Gb/s，交换端口数 N 为 64 时，则要求 $T_{MEM} \leqslant 0.8$ ns。典型的共享存储器交换结构如 Growable switch[28]、Multinet switch[29]、Siemens switch[30]和 Alcatel switch[31]等。

2.3 空分交换结构

2.3.1 全互联型交换结构

所谓全互联(Fully Inter-connected)即在每个输入端口和每个输出端口之间都有一条独立的数据转发通路，其两种实现方式如图 2-5 所示。

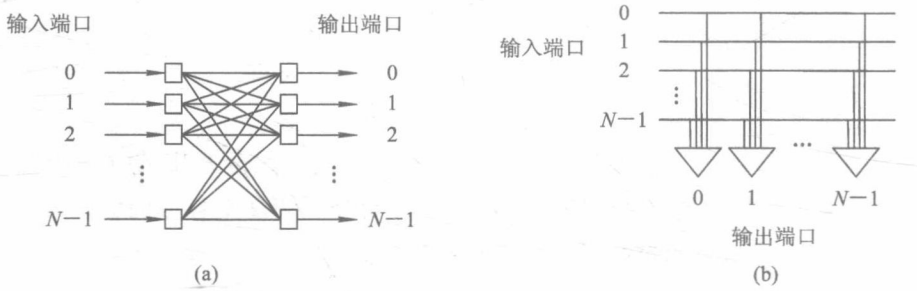


图 2-5 全互联型交换结构

全互联结构的优势在于其结构简单，内部无阻塞。然而对于 $N \times N$ 的交换规模而言，交换结构必须具有 N^2 个数据转发通路，这种 $O(N^2)$ 的硬件复杂度限制了该结构的交换规模。此外该结构对存储器的存取速率要求较高，极端情况下其输出端存储器同样需要在一

个时隙内完成 N 个信元的写入和 1 个信元的读出操作。

考虑到全互联结构的代价过高，且对于输入输出端口数均为 N 的交换结构而言，当 N 较大时，如 $N=128$，现有的存储技术无法满足 128 个信元在同一个时隙内写入输出端缓存，反之发生这种极端情况的概率极小。因此贝尔实验室提出一种 Knockout 结构[32-38]，该结构中每个输入端口都与一条广播总线相连，每个输出端口都通过一个总线接口与所有 N 条广播总线相连，如图 2-6 所示。总线接口包含地址过滤器、集中器和输出端缓存。其中集中器有 N 条入线和 L 条出线($L \leqslant N$)，若一个时隙内有 K 个信元到达，则当 $K \leqslant L$ 时 K 个信元全都可经集中器到达输出端缓存，否则最多只有 L 个信元可从集中器到达输出端缓存，其余信元被丢弃。

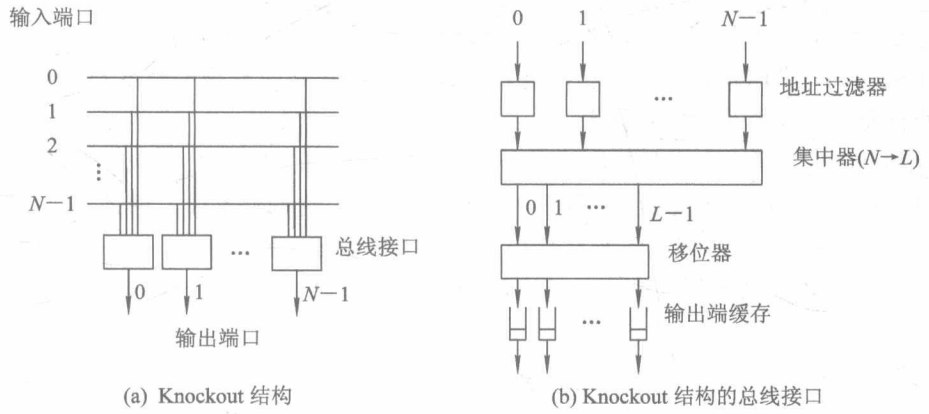
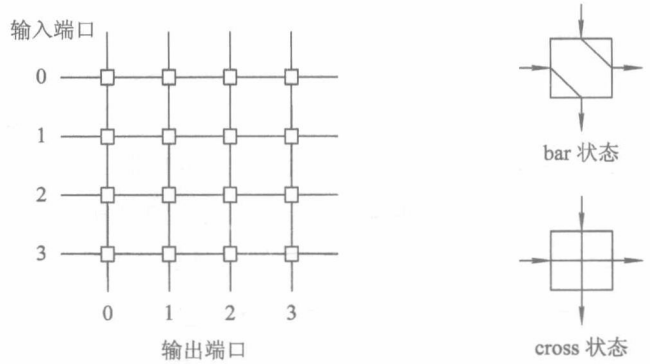


图 2-6　Knockout

Knockout 结构虽然在一定程度上降低了交换结构对存储器存取速率的要求，但同时也会导致一定的丢包率。理论分析表明在均匀业务流环境中，若取 $K=12$，则无论交换规模 N 如何，丢包率都会低于 10^{-10}。然而这仅仅是在理想环境中获得的结论，考虑到 Internet 中的数据流具有自相似特性，故该结论仅存在理论意义。

2.3.2　基于单级 crossbar 的交换结构

在空分交换结构中，除全互联结构和多平面结构之外，其余大多基于交换矩阵，即 crossbar 来实现，甚至在部分多平面结构中，其单个交换平面也都采用 crossbar 来实现，$N \times N$ 的 crossbar 具有 N^2 个交叉开关(也称之为交叉点)，图 2-7(a)所示为 crossbar 的实现方式(以 4×4 为例)。文中对交换矩阵和 crossbar 不做区分。



(a) 交换矩阵的实现方式　　(b) 交叉开关的两个状态

图 2-7　交叉开关结构示意图

crossbar 的交叉开关具有两个状态，bar 和 cross 状态，bar 状态时，横向输入信号从纵向输出端输出，纵向输入信号从横向输出端输出。cross 状态时横向输入信号和纵向输入信号均直通输出，如图 2-7(b)所示。

初始状态时，所有交叉开关均处于 cross 状态，此时任意输入线和输出线均不连通，若输入端口 i 向输出端口 j 转发信元，则需在转发前将输入线 i 和输出线 j 的交叉开关置于 bar 状态，同时输入线 i 和输出线 j 的所有其他交叉开关应置于 cross 状态。在一个时隙的时间内，最多可有 N 个信元从不同的输入端口被转发到不同的输出端口。crossbar 有三个优点：

(1) 内部无阻塞；

(2) 结构简单；

(3) 模块化。

其缺点是内部交叉点数随交换规模 N 的增加而以指数级增长。

crossbar 工作机制决定了在一个时隙内任意输入端口至多转发 1 个信元、任意输出端口至多只能接收 1 个信元。然而同一个时隙内可能会有多个具有相同目的端口的信元同时到达不同的输入端口，这种情况下为避免信元被简单丢弃必须为交换结构设置信元缓冲装置，根据缓冲的位置和数量不同，基于单级 crossbar 的交换结构又可细分为以下 5 类：

(1) 输出排队(Output Queuing，OQ)[39, 40]：缓存仅设置于输出端口；

(2) 输入排队(Input Queuing，IQ)：缓存仅设置于输入端口；

(3) 联合输入输出排队(Combined Input and Output Queuing，CIOQ)：在输入端口和输出端口均设置缓存；

(4) Buffered Crossbar：缓存仅设置于交叉开关处；

(5) 联合输入和交叉点排队(Combined Input and Crosspoint Queuing，CICQ)：在交叉点处和输入端口均设置缓存。