第 5 章 多处理机的数据通信与同步操作

由于多处理机并行性高、并行级别多样，并行任务之间必然需要同步；而对于非共享主存储器的多处理，并行任务之间的数据通信量往往很大，所以数据通信时延和同步操作开销成为制约多处理机有效发挥性能效率的重要因素。本章讨论多处理机数据通信的协议结构及其底层实现方法、路径选择及其算法、流量控制及其控制策略、同步操作与同步原语概念、同步原语种类及其实现，介绍多处理机的通信性能指标、商品化高性能通信网络、数据通信（含存储访问）时延处理策略及其容忍技术、各种同步原语的性能。

5.1 数据通信协议结构与高性能通信网络

5.1.1 数据通信的性能指标及其影响因素

1. 数据通信的性能指标

对于消息传递模型的多处理机，数据通信性能直接决定并行计算的性能。两台计算机相连接是最简单的通信网络，从一台计算机向另一台计算机发送数据来看，度量数据通信的基本性能指标可分为时延和带宽两方面。

1）时延性能指标

时延包含通信时延和网络时延。

通信时延指多处理机中，从源节点向目的节点传送一条消息所需要的全部时间，它由软件开销、通路时延、选路时延和竞争时延 4 部分组成。软件开销是源节点发送消息与目的节点接收消息的软件执行时间；通路时延是消息通过通路传送消息所花费时间且通路时延＝消息长度/通路带宽；选路时延是消息传送过程中所有选路决策所花费时间；竞争时延是消息传送过程中为避免或解决网络资源争用冲突所花费时间。

网络时延指多处理机中，通路时延与选路时延的和。在负载较轻时，网络时延（通常为 1μs）远小于软件开销与竞争时延（几十到几百微秒）。

软件开销取决于源节点和目的节点收发消息的 OS 内核，通路时延由瓶颈链路或通路决定，选路时延与源节点和目的节点之间的距离有关，影响竞争时延的因素比较复杂，关键在于通道流量。可见，软件开销与竞争时延由软件行为特性（即通路流量）决定，通路时延与选路时延即网络时延由网络硬件特性决定，即通信时延是收发消息的 OS 内核、消息长度、通路带宽、源目节点间距离、通路流量的函数。

2）带宽性能指标

数据通信依赖于互连网络，互连网络的带宽参数即数据通信的带宽指标，而互连网络的

带宽参数主要有端口带宽、聚集带宽、对剖带宽、网络带宽等，单位均为 Mb/s。

端口带宽指互连网络中任一端口到另一端口传输消息的最大速率。对称网络的端口带宽与其位置无关，非对称网络的端口带宽是所有端口带宽中的最小值。

聚集带宽指互连网络中从一半节点到另外一半节点传输消息的最大速率。对于对称网络，聚集带宽＝端口带宽×节点数/2，如端口带宽为 10Mb/s，那么 512 个节点的聚集带宽为(10Mb/s×512)/2≈2.5Gb/s。

对剖带宽指互连网络中最小对剖平面上传输信息的最大速率。对剖平面是一组连接线，它将互连网络分成节点数相等的两部分；一个互连网络有许多个对剖平面，其中连接线数最少的为最小对剖平面。

网络带宽指消息进入互连网络后传输信息的最大速率。网络带宽是数据通信的综合性指标，它与网络拓扑结构、通路宽度、网络规模(端口数量)、通路数量、时钟频率等有关，完全由网络硬件特性决定。

2. 影响数据通信性能的因素

影响数据通信性能的因素包含通信硬件、通信软件和通信服务 3 方面。

通信硬件包含节点及其存储器、缓冲区、I/O 接口和通信网络等，在 I/O 接口中，NIC 是不可缺的。具有 DMA 功能的网络接口电路含有通信处理器(协处理器)、存储器和先进先出缓冲区等，协处理器用于初始化、装配/拆卸、校验编译码等，存储器用于存储 NIC 代码和暂存消息，缓冲区用于缓冲消息包。对于松散耦合多处理机，消息一般从源节点存储器发出，通过 DMA 缓冲区、存储总线、I/O 总线、网络接口电路，送到通信网络上；消息经过通信网络到达目的节点，反方向地传输到存储器接收。在消息传输的路径上，通信带宽受限于路径上窄带宽部件，窄带宽部件往往是存储部件而不是网络部件，因为数据通信时对存储器的访问通常采用 DMA 方式，DMA 缓冲区复制的带宽远小于存储总线的峰值带宽。

对于现代机群多处理机，软件开销在通信时延中占比很大，所以数据通信必须有良好的通信软件支持，否则即使通信网络与通信硬件非常有效，也难以使通信时延明显减少。通信软件包含通信协议组成结构及其算法等，软件开销主要来源于：①消息通过多层协议解释所需时延；②消息传输路径上存储复制所需时延；③消息传输路径上多次跨越保护边界(如从 NIC 进入通信网络等)所需时延。

通信服务对数据通信性能影响也是很大的。所期望的通信服务包括：①可靠传输，使消息准确无误而完整地从源节点传送到目的节点；②流量控制，使消息传输避免死锁、拥堵和缓冲溢出；③失效处理，以实现错误校验、重发等；④有序传输，使接收信息的顺序正确。

5.1.2　数据通信协议结构及其低层实现

1. 数据通信协议的组成结构

多处理机节点之间的数据通信是利用多层次协议来实现的，这些协议的组成结构如图 5-1 所示。在源节点发送端，用户通过通信函数(如在 PVM 和 MPI 等数据通信库中调用)，将消息向下传输到应用编程接口(application programming interface，API)层(套接字层)、TCP/IP 或 UDP/IP 层、OS 内核层、网络接口电路层，直至设备驱动器和通信网络。在目的节点接收端，以相反次序向上传输到通信函数中。特别地，套接字层和通信函数均可以在低级基本通信层(basic communication layer，BCL)上实现，而旁路掉 TCP/IP 或 UDP/IP 层。

低级基本通信层更多反映的是通信硬件性能，数据通信性能更多是由通信库或套接字来反映。

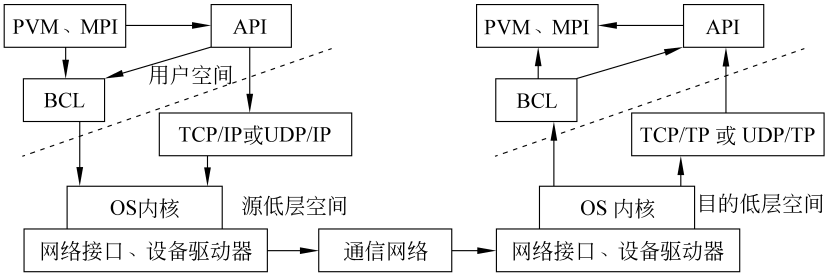

图 5-1　数据通信多层次协议的组成结构

在多处理机节点之间的通信网络可以分为通道和网络两种类型，通道可以认为是一种特殊的网络。通信通道是多处理机节点之间直接的点到点的固定连接，偏向物理性，主要用于处理机与 I/O 设备或 I/O 设备之间的连接。通信通道的数据传输速率高、时间开销小，可以处理任务的范围有限，目前 HiPPI、SCSI 等均定义了通信通道标准。通信网络是多处理机节点之间间接的多到多的可变连接，偏向逻辑性，主要用于处理机之间的连接。通信网络的数据传输速率低、时间开销大，可以处理任务的范围很宽泛，一般都有自己的通信协议。

2. TCP/IP 通信协议族

TCP/IP 通信协议族包含应用层、传输层、网络层和网络接口层等，其组成结构如图 5-2 所示。为了使其与具体的物理介质无关，TCP/IP 通信协议族没有对数据链路层和物理层进行规范而取名为网络接口层，所以可以认为它仅包含应用层、传输层和网络层 3 层。TCP/IP 通信协议族的特点是两端大中间小，即应用层有很多协议，如依赖于 TCP 的有超文本传输协议 HTTP、电子邮件协议 SMTP、文件传输协议 FTP 等，依赖于 UDP 的有内部网关协议 RIP、动态主机配置协议 DHCP 等，依赖于 TCP 和 UDP 的有通信管理信息协议 CMOT 等，而网络接口层无规范，传输层有传输控制协议 TCP 和用户数据报文协议 UDP，网络层仅有网络协议 IP。

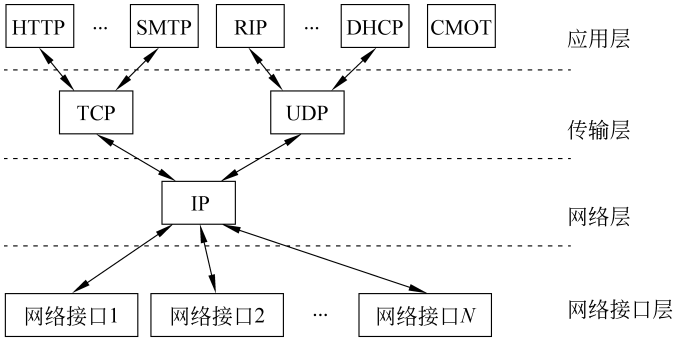

图 5-2　TCP/IP 通信协议族的组成结构

应用层协议是向用户提供常用的应用程序，一般分为 3 类：依赖于 TCP 的、依赖于 UDP 的、依赖于 TCP 与 UDP 的。网络层协议的主要功能有路径选择、数据报文分段与装配、错误通告等，还提供流量控制、拥堵处理等服务。传输层提供端到端（即应用进程之间）的通信服务，其主要功能为格式化数据流、端到端可靠传输、不同应用程序的识别等。UDP

是面向非连接的，发送端仅将消息分割成消息包发送并不建立连接，接收端对消息包接收装配和检查，但将错误数据丢弃而没有重发机制。TCP 是面向连接的，发送端将消息分割成消息包并建立连接后发送，接收端对消息包接收装配和检查，并利用应答和重发机制以确保消息可靠地传输。网络接口层的作用是向特定网络发送或从特定网络接收 IP 数据报文的物理帧。

应用编程接口是应用 TCP/IP 通信协议族的一组数据类型和操作，目前基本成为 UNIX 和 Windows 平台的标准。一个应用编程接口即一个套接字，就是一个通信端口，当两个进程应用套接字通信时，各自生成一个套接字，以指明是采用协议 TCP 还是 UDP，通过读/写相应的套接字来实现通信。

3. 基本通信层的实现方法

低级基本通信层对通信性能影响很大，其实现目前主要有双复制、单复制和零复制 3 种方法。

1）双复制

双复制即用户空间协议，IBM SP 多处理机的低级基本通信利用了该方法实现，所以此处以 IBM SP 多处理机的低级基本通信过程为例来介绍双复制方法。

IBM SP 多处理机的低级基本通信过程如图 5-3 所示，其中消息层和管道层为基本通信库。消息层是非阻塞的、点到点的简单通信库，如 MPI、PVM 等，高级消息传输的通信函数功能均由其原语实现；管道层维持成对发/收进程之间的具有流量控制的、有序可靠的字节流。当源节点在消息层执行一条数据发送操作时，便将数据从发送缓冲器复制到源管道缓冲器；再在消息层调用管道层代码，将数据从源管道缓冲器复制到输出队列；最后由网络接口采用 DMA 方式将输出队列数据传输到源网络接口（源适配器）。源网络接口通过通信网络传输到目的网络接口（目的适配器），目的网络接口采用 DMA 方式将数据传输到输入队列；之后目的节点在消息层调用管道层代码，将数据从输入队列复制到目的管道缓冲器；最后在消息层执行一条数据接收操作，将数据从目的管道缓冲器复制到接收缓冲器。

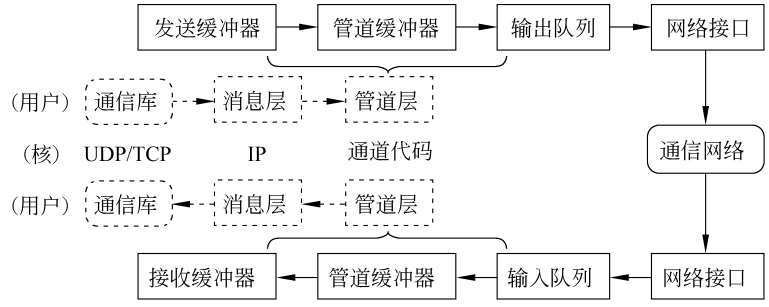

图 5-3　IBM SP 多处理机的低级基本通信过程

2）单复制

单复制又称快速消息，主要支持机群与大规模并行多处理机，目的在于为消息层提供足够功能，使得高级原语（如 MPI 和 PVM 的套接字）实现的通信性能接近硬件极限。

单复制数据结构和功能均比较简单，仅含几个通信原语，如初始化的、发送存储器长消息的、发送寄存器多字消息的、抽取接收消息的、调用处理程序的等。单复制是一个低级通信层，但初始化后，网络接口和队列对用户可见，随后的发送和接收均不必跨界面保护。当

发送节点从缓冲器取出数据并组装成消息包后，则直接将其传输到输出队列中，而后在DMA控制下传输到通信网络。当消息包到达目的节点时，则仍由DMA控制传输到输入队列，而后调用处理程序，将数据存入目的存储空间。

3）零复制

零复制又称为虚拟存储映射通信。当节点之间进行数据通信时，节点中的守护程序可以为源节点和目的节点建立输入-输出关系，发送节点的进程可以将缓冲器中的数据直接发送到目的缓冲器，而不通过核缓冲器。

5.1.3 商品化高性能通信网络

1. Myrinet通信网络

Myrinet是一项高效分包通信与交换技术，可以用于构建适用于机群多处理机的、虫蚀全双工的多级交叉开关商品化通信网络，由其建构的网络拓扑结构如图5-4所示。Myrinet网络拓扑结构不受限，数据链路层消息包可变长且可以实现流量和错误控制，光纤链路长度可达3m，峰值速率为(1.28+1.28)Gb/s。特别地，其中的节点处理机可以是PC、工作站、超级计算机、服务器等。

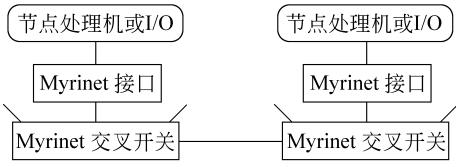

图5-4 Myrinet通信网络的拓扑结构

Myrinet交叉开关通过链路同其他交叉开关或Myrinet接口相连，具有输入缓冲功能，采用无死锁选路策略，可以同时流动多个消息包。对于8个端口的Myrinet交叉开关，对剖带宽可达10.24Gb/s、通信时延为300ns，功耗为6～11W。

Myrinet接口用于将节点处理机与Myrinet交叉开关互连，是一个32位的用户定制的VLSI处理器，带有DMA引擎和快速SRAM。SRAM用于存储Myrinet控制程序和消息包缓冲，由于控制程序在接口处理器上运行(设备驱动程序仍在节点处理器上运行)，从而避免了OS开销。特别地，目前还推出了标准TCP/IP和UDP/IP的Myrinet接口及其API。

2. HiPPI通信网络

高性能并行接口(high performance parallel interface，HiPPI)技术是单工点到点的、覆盖物理和数据链路层的数据传输接口，主要用于异构超级计算机及其I/O设备的组网，由其建构的网络拓扑结构如图5-5所示。基本HiPPI的线宽50位，其中数据线32位、控制线18位，通信时延40ns，峰值速率为800Mb/s。50对双绞线链路(实现双向通信)长度为25m，标准多模光纤链路长度可达300m；使用光纤扩展器时，支持远距离10km(多模光纤)和20km(单模光纤)光纤扩展器之间的链路。若位采用4根电缆来实现全双工通信，则峰值速率为1.6Gb/s。特别地，其中的节点处理机可以是PC、工作站、超级计算机等。

HiPPI交叉开关与HiPPI通道在物理层实现了基于信用的流量控制策略，使得不同速率的节点之间的通信可靠有效；兼容双绞线和光纤链路，且链路长度可长可短；由帧协议将数据格式化，不使用任何高层协议，使得协议具有独立性，同其他网络或节点处理机或I/O

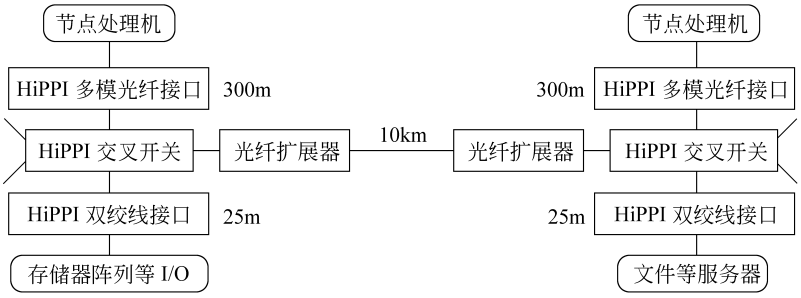

图 5-5　HiPPI 通信网络的拓扑结构

设备组网时，都是自适应的。HiPPI 交叉开关是面向连接的，且连接简单，允许多个消息包同时流动，所以其聚集带宽为 800Mb/s 或 1.6Gb/s 的整数倍。HiPPI 通道不支持组播。

后来，人们成功开发出峰值速率比 HiPPI 高 8 倍达 6.4Gb/s，且通信时延很低的“超级 HiPPI”，但在实现上与 HiPPI 完全不同且相互不兼容。

3. FDDI 通信网络

光纤分布式数据接口(fiber distributed data interface，FDDI)技术是采用双向光纤令牌环，利用方向相反的旋转环来提供冗余通路，以使通信网络具有可靠性高、互连能力强、共享介质、容错性优的特点，主要用于需要增加、删除、移动所连节点(含处理机和 I/O 设备)且不会导致网络崩溃的组网，由其建构的网络(环)拓扑结构如图 5-6 所示。FDDI 的数据传输速率为 100～200Mb/s，双绞线链路长度为 100m，多模光纤链路长度可达 2km，单模光纤链路长度可达 60km。特别地，其中的节点处理机可以是 PC、工作站、超级计算机等。

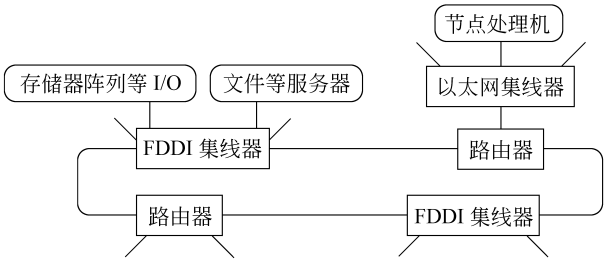

图 5-6　双向 FDDI 通信网络的拓扑结构

光纤通道是通道与网络标准的集成，用以减轻制造商在支持现有的不同通道和网络所带来的负担，满足通道用户和网络用户的需求。光纤通道的数据传输速率为 100～800Mb/s，双绞线光纤通道长度为 50m，多模光纤通道长度可达 2km，单模光纤长度可达 10km。光纤通道除支持集线器环路互连外，还支持点到点、星状互连。星状互连时，采用带缓冲的虫蚀传输方式，在不存在冲突时，8×8 的交叉开关可以使 8 个数据片在 1/40μs 内通过。

4. SCI 通信网络

节点处理机特别是基于总线的 SMP 通常是通过 I/O 总线连接到高性能通信网络上，而 I/O 总线不仅时延大，还难以利用由硬件生成的高速缓存一致性的信息。存储总线(局部总线)则不存在 I/O 总线的缺陷，所以为实现通过存储总线连接到高性能通信网络上，有人便开发出了可扩展一致性接口(scalable coherent interface，SCI)。SCI 协议内容极其丰富，它将主板总线扩展为全双工、点到点的互连，该互连独立于高性能通信网络，并提供基于

目录分布共享存储一致性协议的实现。

SCI是利用若干组链路(一组包含16位的输入与输出链路各一条),每条链路带宽为1Gb/s的节点与外部互连的接口,由其建构的网络拓扑结构如图5-7所示。显然,SCI通信网络是由若干SCI环组成的二维网孔,每个方块即为SCI节点,且SCI将处理器、存储器、I/O设备、处理机、计算机等均视为节点。每条环即是一条两端相连的局部总线,环中的网孔通过SCI连接在一起。SCI通信网络采用点到点链路,从而不存在T形接头,极大地缓解了信号噪声反射,提高了信号传输速率;采用单向环链路,使得驱动电流可以保持一个常数,从而降低信号噪声。另外,SCI通信网络并行性较高,因为所有节点可以同时发送和接收消息包。特别地,环中的SCI逻辑上是同步驱动的(时钟频率为500MHz),为了处理不同节点的时钟差异,接收节点设置了弹性缓冲器,可以对存入消息包中的空闲字符进行删除。

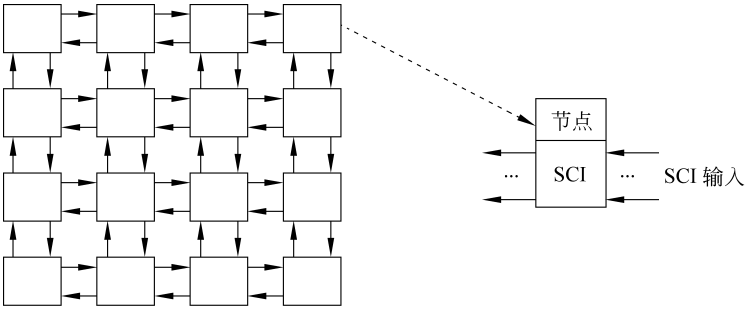

图5-7 SCI通信网络的拓扑结构

5. 高速以太通信网络

以太网是一种基带总线局域网,采用无源电缆总线来传输数据,其网络拓扑结构如图5-8所示。以数据传输速率为标志,以太网的发展经历了3个时代:数据传输速率小于10Mb/s的为第一代,称为传统以太网;数据传输速率大于100Mb/s、小于1Gb/s的为第二代;数据传输速率大于1Gb/s的为第三代。第二、三代以太网统称为高速以太网,即把数据传输速率达到或超过100Mb/s的以太网统称为高速以太网。

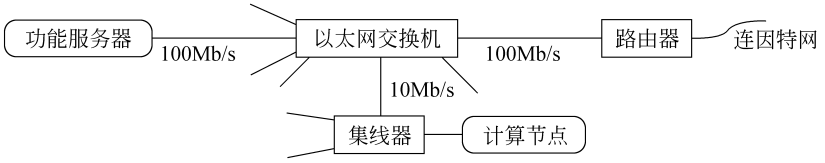

图5-8 以太通信网络的拓扑结构

(1) 百兆位以太网。百兆位(又称100BaseT)以太网使用双绞线或光纤为传输介质,采用星状拓扑结构,数据传输速率达100Mb/s,标准为IEEE 802.3u。百兆位以太网的主要特点有:①性能价格比高,数据传输速率是传统以太网的10倍,价格仅是它的2倍;②组网简单容易,利用成型通用的网卡、集线器、传输介质等就可以直接建构局域网。

(2) 吉比特以太网。吉比特以太网的组网与百兆位以太网基本相同,且技术兼容百兆位以太网,但比特发送间隔由100ns降低为1ns,即数据传输速率可达到1Gb/s,标准有IEEE 802.3z和IEEE 802.3ab两种。吉比特以太网的主要特点有:①与低速以太网连接简单容易,通过交换机或路由器连接即可;②具有流量控制功能,可以有效避免冲突和拥堵。

(3) 10吉比特以太网。10吉比特以太网的组网也没有大的变化，帧格式也仍然采用IEEE 802.3，但许多技术得到突破，而不仅是数据传输速率提高到10Gb/s，标准为IEEE 802.3ae。10吉比特以太网的主要特点有：①传输介质仅使用光纤，而不再使用双绞线；②通信方式仅采用全双工，而不再采用单双工；③物理层分为局域网物理层和广域网物理层。

5.2　数据通信的路径选择与流量控制

5.2.1　路径选择与虚拟通道

1. 路径选择及其度量参数

对于系统域互连网络，当两个节点之间没有直接的链路相连接时，消息包就需要通过中间节点进行传输。当消息包在源节点或中间节点时，数据通信可能存在多条有效链路。为了充分利用互连网络的带宽，应该选择若干合适的链路来构成数据通信的通路。两个节点之间所谓合适的链路包含两方面的含义：路径最短且传输时延少和避免链路选择冲突。链路选择冲突是指多对节点之间利用互连网络进行数据通信时，在某节点交叉开关上发生输出端口争用的现象。这就是路径选择需要解决的问题。

路径选择即是链路选择或路由选择，有时简称为寻径，是指节点对之间经中间节点进行数据通信时，对中间节点的选择。路径选择的操作就是为源节点或中间节点上的交叉开关有效地建立输入端口与输出端口之间的连接，即对于从任一输入端口输入的消息包，均为其选择一个合适的输出端口输出。使多处理机节点之间高效率地实现数据通信是系统域互连网络路径选择的根本目的，而路径选择效率的常用度量参数为通道流量和传输时延。

通道流量采用数据通信所使用的链路数来表示，它反映网络通信负载。传输时延采用数据通信所需要的时间来表示，它反映网络通信速度。显然，路径选择的基本目标就是使网络以最小流量和最短时延来实现节点之间的连接通信，但通道流量与传输时延是相互关联并相互制约的。要使通道流量小，传输时延就会长；要使传输时延短，通道流量就会大。可见，路径选择的优化类似多目标优化，仅能在最小流量和最短时延两个目标之间进行折中。对于不同的数据通信方式时，优化有所侧重，如存储转发偏重于时延短(因其传输时延长)，而虫蚀则偏重于流量小(因其占用链路多)。

2. 虚拟通道

链路是指一段物理通路，但在虫蚀传递方式中，链路被许多节点对所共享，由此，便引出虚拟通道的概念。虚拟通道指多节点对共享一条物理链路而形成的各节点对之间的逻辑链路，所以节点对之间的链路有物理链路与逻辑链路之分，或节点对之间的通路有物理通路与虚拟通路之分。节点对之间的虚拟通路由源节点片缓冲区、节点间的物理链路和目的节点片缓冲区组成，如图5-9所示为4条虚拟通道共享一条物理链路，源节点和目的节点各有4个片缓冲区。当物理链路分配于某缓冲区对时，这对缓冲区和物理链路便构成一条虚拟通道，且赋予其一种状态，即不同虚拟通道是通过状态来表示的。源缓冲区存放等待使用物理链路的数据片，目的缓冲区存放由物理链路刚传送过来的数据片，物理链路是它们之间进行数据通信的媒介。

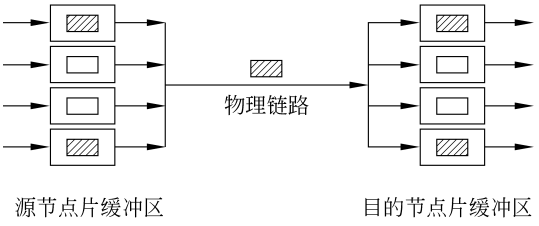

图 5-9　4 条虚拟通路分时共享一条物理链路

当然，虚拟通路会使节点对之间可用的有效带宽降低，一条物理链路配置多少虚拟通路需要综合权衡。虚拟通路有单向与双向之分，两条单向通路组合在一起可以构成一条双向通路。双向通路可以增加物理链路利用率与通路带宽，但双向通路的仲裁复杂，从而增加了时延与成本。

5.2.2　路径选择算法及其分类

路径选择算法指节点对之间经中间节点进行数据通信时，中间节点的选择方法，它可以分为确定性路径选择算法与自适应路径选择算法两种。

1. 确定性路径选择算法

确定性路径选择算法指节点对之间的连接路径在数据通信之前就已确定，在通信过程中不可能改变，节点对之间仅有一条通信路径。显然，由确定性路径选择算法建立的通信通路，路径最短，但不一定最合理，传输时延不一定最少，还可能存在链路选择冲突等问题。确定性路径选择算法实现简单方便，但当发生链路选择冲突或通路存在阻塞有故障时，无法改变连接路径，只会沿着原先选择的路径进行通信。确定性路径选择算法，一方面单个连接失效会使网络断开，导致数据通信失败；另一方面会给网络带来大量竞争，当多对节点对使用相同的链路时，多个通信仅能串行顺序进行。

确定性路径选择算法主要有 3 种：算术选路、源选路和查表选路。

算术选路算法指所有节点对之间的连接路径由源地址和目的地址完全确定，与网络负载及其状态无关。最典型的算术选路算法是维序选路算法，即根据物理链路的坐标维来决定消息包相继流过的链路。特别地，当维序选路算法用于二维网格网络时则称为 X-Y 选路算法，用于超方体网络时则称为 E-立方选路算法。

源选路算法指源节点为消息包建立一个头部，其包含连接路径上经过的所有交叉开关的输出端口，交叉开关从消息包头部可以得到输出端口号。源选路算法的交叉开关简单、通用性强，但消息包头部长且长度不固定。

查表选路算法指网络中每个交叉开关维护一张选路表 R，消息包头部包含一个选路域 I，以 I 为索引查找选路表来得到输出端口号 R[I]。查表选路算法的选路表容量可能很大，且要求节点之间的连接路径相对稳定，对系统域互连网络不太合适。

2. 自适应路径选择算法

自适应路径选择算法指根据网络资源及其状态来选择节点对之间连接所需要的链路，以躲避拥堵或故障节点，提高网络资源利用率。对于自适应路径选择算法，路径上的寻径器可以根据各链路的通道流量，动态地进行链路选择。若交叉开关期望的某输出端口被阻塞或失效，寻径器可以选择另外的链路进行数据通信。自适应路径选择算法的路径选择宽松，

允许节点对之间的连接存在多条合法路径。一方面当存在链路失效时，可以绕开故障点进行数据通信，提高容错能力；另一方面可以在可用链路上更广泛地分布通道流量，将网络负载分布到多个链路上，提高网络的利用率。若有 4 个消息包从不同的源节点向不同的目的节点传输，当采用确定性路径选择算法时，可能均被强迫沿同一路径传输，这时该路径则可能形成通信瓶颈，而其他最短路径上的链路却闲置未用。

自适应路径选择算法的交叉开关复杂，从而降低了通信速度；另外，路径选择灵活且具有动态性，但容易产生死锁。对于系统域互连网络，由于每隔几个时钟周期，交叉开关就需要为所有输入的数据进行路径选择。可见，路径选择算法应尽可能简单快速，自适应路径选择算法极其复杂，在系统域互连网络中不常使用。

3. E-立方选路算法

E-立方选路算法主要用于拓扑结构为立方体或超立方体的互连网络。设有一个 $N=2^N$ 个节点的 N 方体，源节点编号 $S=S_{N-1}\cdots S_1S_0$，目的节点编号 $D=D_{N-1}\cdots D_1D_0$，$V=V_{N-1}\cdots V_1V_0$ 为物理链路中间节点编号。维号 $i=1,2,\cdots,N$，其中第 i 维对应于节点编号中的第 $i-1$ 位，选择一条 S 到 D 路径距离最短的 E-立方选路算法为：

(1) 计算方向位：$r_i=S_{i-1}\oplus D_{i-1}$；

(2) $i=1$，$V=S$；

(3) 若 $r_i=1$，则从当前节点 V 选择下一节点 $V=V\oplus 2^{i-1}$；若 $r_i=0$，则跳过；

(4) $i=i+1$，若 $i\leqslant N$，则转第(3)步，否则退出。

E-立方选路算法可以在源节点和目的节点之间建立多条距离最短的路径，而最短路径判定依据为：路径所经过的最少链路数与源节点和目的节点间的海明距离相等(源节点与目的节点二进制编号位号相同位值不同的位数)。在如图 5-10 所示的 3-立方网络中，当源节点为(011)、目的节点为(110)时，其方向位向量为(101)；由于 $r_2=0$，则 2 维方向上源节点没有距离，所以有两条距离最短的路径。①先沿 1 维将消息由源节点(011)传送至中间节点(011)$\oplus 2^0$=(010)，再沿 3 维将消息传送至节点(010)$\oplus 2^2$=(110)。②先沿 3 维将消息传送至节点(011)$\oplus 2^2$=(111)，再沿 1 维将消息传送至节点(111)$\oplus 2^0$=(110)。显然，这两条路径的最少链路数均为 2，源节点与目的节点间的海明距离也为 2，所以两条路径的距离均为最短。

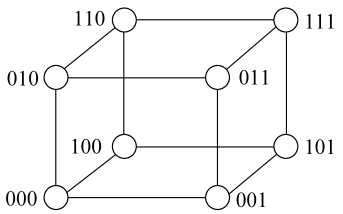

图 5-10　3-立方体网络的 E-立方选路算法

设源节点与目的节点间的海明距离为 h，则最短路径有 $h!$ 条。如节点(000)与节点(111)之间的海明距离为 3，则它们之间共有 $3\times2\times1=6$ 条最短路径，且分别是 000→001→011→111、000→001→101→111、000→010→011→111、000→010→110→111、000→100→101→111 和 000→100→110→111。除最短路径外，还有多条非最短路径。所以立方体网络的可选路径很多，可靠性比较高；若某个或某些节点出现故障，剩下节点仍可以进行源目节点间的通信。

5.2.3　死锁及其解除避免方法

1. 死锁及其表现形态

死锁指消息包在网络中传送时，等待一个不可能发生的事件，这个事件多数是网络资源