曙光 5000 高性能计算机多播网络的设计

曹 政[1,2],王达伟[1,2],刘新春[1,2],孙凝晖[1,2]
(1.中国科学院计算技术研究所国家智能计算机研究开发中心,北京 100190;
2.中国科学院计算机系统结构重点实验室,北京 100190)

摘 要: 本文介绍了曙光 5000 高性能计算机多播网络设计的关键技术.减少多播与单播/多播与多播间的网络竞争是降低多播延迟的主要途径之一,而无死锁则是保障多播操作正确完成的前提.为了解决死锁问题,本文提出了一种基于全局资源公告的死锁避免方法;为了获得较低的多播延迟,本文充分利用胖树拓扑特点,提出了一种基于重载交换机去除的多播路径选择策略.测试结果表明,在网络重载情况下,相比于已有多播路径选择算法,本文的路径选择策略可以获得近三倍的性能提升.对于 many-to-many 多播通信,曙光 5000 多播网络可以获得 90% 以上的多播吞吐率.

关键词: 高性能计算机; 多播; 死锁; 路由算法; 胖树
中图分类号: TP303 **文献标识码:** A **文章编号:** 0372-2112 (2011) 02-0481-08

Design of Multicast Network of Dawning 5000 High Performance Computer

CAO Zheng[1,2], WANG Da-wei[1,2], LIU Xin-chun[1,2], SUN Ning-hui[1,2]
(1. *National Research Center for Intelligent Computing Systems, Institute of Computing Technology, Chinese Academy of Sciences, Beijing* 100190, *China*; 2. *Key Laboratory of Computer System and Architecture, Chinese Academy of Sciences, Beijing* 100190, *China*)

Abstract: This paper proposed the design of Dawning 5000 multicast network based on fat-tree topology. Multicast's latency can be lowed by reducing the interference between unicast and multicast, while the completion of multicast can only be guaranteed by freeing the deadlock. To solve deadlock problem, this paper proposed a deadlock avoidance design based on Resource Bulletin Board. To reduce the interference, this paper proposed a multicast routing algorithm which selects light- weighted multicast path by eliminating heavy-loaded switches. Compared with existing load balancing routing algorithm, the algorithm proposed in this paper can fully use light-weighted paths and achieve three times performance improvement. Besides, under many-to-many multicast communication, Dawning5000 multicast network can achieve more than 90% multicast throughput.

Key words: high performance computer; multicast; deadlock; routing algorithm; fat-tree

1 引言

多播通信是某进程将数据复制多份,分发给其他进程的过程,它被广泛地应用于并行图形算法、并行搜索以及高性能计算中的一些基本操作,如矩阵乘法和 LU 分解.更为关键的是,多播是其他一些集合操作(如 Gather 和 All-reduce 操作等)的子过程,优化多播性能,便间接优化了这些集合通信的性能.

多播的优化工作可以分为两大类:一类是基于软件的多播优化,也被称为基于单播的(unicast-based)多播优化;另一类是基于硬件支持的多播优化.通过提高消息输出并发度[1,2],降低网络竞争[3],基于软件的多播优化也可以获得较低的多播延迟和较高的多播带宽,但软件进行数据拷贝转发,仍有较大开销,制约了性能的提高.

现有支持硬件多播的机群网络中,大多采用了基于树的硬件多播算法,如 IBM SP2[4], NEC Cenju-3[5], Quadrics[6]和 Infiniband 网络.其中 IBM SP2 实现了 90% 以上的多播吞吐率[4],Quadrics 的硬件多播也可以获得高于软件 10 倍的性能[6].为获得较高的多播性能,曙光 5000 高性能计算机采用内嵌式硬件多播网络对多播性能进行优化.所谓内嵌式多播网络,即通过向曙光 5000 互连网络部件(网卡和交换机)中增加多播功能模块实现多播网络的构建.

曙光 5000 互连网络采用胖树拓扑结构,具有扩展性好,等分带宽,确定路由无死锁的特点.曙光 5000 高性能网络是面向 1024 个处理机节点的大规模互连网络,目标在于实现节点间高带宽低延迟通讯,为提高并

收稿日期:2009-09-15;修回日期:2010-03-15
基金项目:国家 863 计划高技术研究发展计划(No.2006AA01A102);国家自然科学基金重点项目(No.60633040)

行应用性能服务.曙光 5000 交换芯片为 16 端口全双工(单端口带宽 5Gb/s),交叉开关设计,支持多虚通道,使用基于绝对信用的流量控制机制,虚切入(VCT)交换方式.同时,曙光 5000 互联网络采用带外管理的方式,通过另一套监控网对各网络部件进行配置和监视.本文的多播网络设计将围绕曙光 5000 数据网络的特征展开.

2 关键问题

胖树拓扑中结点与结点间只能通过交换机互连,而基于路径的多播需要结点转发消息,导致通信路径过长,无法获得最优的性能.此外在大规模网络环境下,基于路径的多播携带的地址包头开销过大,降低了有效数据的传输效率.相反的,基于树的多播算法易于实现源结点与所有目的结点间以最短路径通信,更适合于胖树拓扑.实现基于树的多播操作,树形结构的建立策略,即多播路径选择策略成为影响多播性能的关键.实现树形多播,要考虑如下三个问题:

(1)解决死锁:实现基于树的硬件多播,存在发生死锁的危险.图 1 中为两种发生死锁风险的情况.一种如图 1(a)所示,多播 A 占据交换机 0 的 P3 端口,请求交换机 1 的 P1 端口,多播 B 占据交换机 1 的 P1 端口,请求交换机 0 的 P3 端口,形成交换机间循环依赖,发生死锁,称为交换机间死锁;另一种如图 1(b)所示,多播 A 占据端口 P1,请求端口 P0,多播 B 占据端口 P0,请求端口 P1,形成交换机内部输出端口循环依赖,发生死锁,称为交换机内死锁.解决这两种死锁问题,既可通过多播路径选择策略避免,也可通过特定硬件支持得以解除.

(2)缩短最长通信路径:当所有结点均收到多播数据包后,我们称一次多播操作完成,因此在不考虑网络竞争的情况下,多播通信的延迟与最长通信路径长度直接相关.

(3)降低网络竞争:在多播路径中的每一级均会涉及多个输出端口,大大增加了发生网络竞争的几率,大大降低了多播的性能.网络竞争一方面来自于多播与多播通信,另一方面来自于多播与单播通信.因此,多播路径选择策略涉及静态和动态两个方面,即一方面

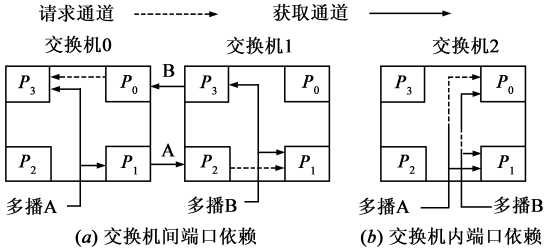


图1 多播死锁示意图

在多播路径建立初始即加入降低网络竞争的考虑,另一方面可以根据网络拥塞状态对多播路径进行合理调整.

本文以胖树拓扑为背景,重点解决死锁和多播路径选择问题,以实现最优的多播网络性能.

3 相关工作

3.1 多播死锁

虫蚀交换的网络中,消息头的阻塞会造成整个路径的阻塞,因此交换机间死锁常发生在基于虫蚀交换的网络中.剪枝机制[7]是解决此类死锁的常见方法,当发生死锁后,交换机主动将被阻塞的分支剪掉,被剪掉的分支将消息转为单播发送,实现死锁恢复.但在异步网络中,因为依赖不同交换机的端口而导致的死锁很难发现,所以难以采用死锁恢复机制.

Quadrics 是基于虫蚀交换的网络,为避免多播死锁发生,在网卡和交换机中设置了严格限制[8].首先,所有包含重叠目的结点的多播都要在源结点中排队输出,以保证同一个结点发送的多播操作之间不会产生死锁.其次,Quadrics 只设置一个广播树,所有的多播树均为该广播树的子树,使得不同结点发出的多播操作在广播树的根结点处串行处理,以保证不同源结点发送的多播包不发生死锁.Quadrics 中的死锁避免方法大大降低了多播操作的吞吐率.

在虚切入交换网络中,多播包被完整缓存于下一级缓冲区中,不会继续占用之前的路径,避免了交换机间死锁的发生,但仍会发生交换机内死锁.文献[9]提出了利用流控信息,由网卡进行死锁发现,进而死锁恢复的机制,但这种方法解决死锁的效率很低.文献[10]提出了设定固定优先级的死锁避免策略,当来自不同端口的多播操作请求的输出端口发生重叠时,交换机只响应输入端口号大的请求,当该请求得到满足后,再响应其他请求.解决死锁的最优方案是支持扇出分割(fanout-splitting),即多播包的输出不必等待所有输出端口有效,只要输出端口满足输出条件,就复制一份数据从该端口输出,解除了多播对多个端口的同时依赖,文献[11]中采用了扇出分割方式.扇出分割可以获得最佳的多播输出带宽,但会增加硬件的复杂度,因此无分割(no-splitting)方式仍被广泛使用.

通过上述分析可知,解决死锁的主要方式是死锁避免和死锁恢复,由于死锁恢复需要相应的死锁发现机制,导致实现复杂度高,死锁避免更易于实现.然而具体的死锁避免策略要兼顾多播性能和实现复杂度.

3.2 多播路径选择

死锁避免保证了多播的正常完成,而多播操作的性能则与多播路径选择策略直接相关.胖树拓扑具有

固有的树形结构,非常易于实现最短路径通信,因此降低网络竞争是多播路径选择的目标.在胖树拓扑中,路由可以分为两阶段,上行阶段(ascending phase)和下行阶段(descending phase).上行阶段即由结点到顶层交换机的路径,下行阶段即由顶层交换机到结点的路径.

文献[12]中提出了一种基于 Infiniband 网络的轮转(Cyclic)路径选择策略,即在上行阶段,源结点 $P(p = p_0p_1p_2\cdots p_l\cdots p_n)$发送单播包至顶层交换机,在第 l 层交换机中选用交换机的 $k + p_l$ 端口输出(k 为交换机端口数目的一半),在下行阶段再由各级交换机以多播包转发.使用轮转选路策略,可以实现不同源结点发出的多播操作,在上行阶段不发生网络竞争,进而获得较低的多播延迟.但该算法使得同一源结点发出的,不同目的结点的多播包在整个上行阶段都串行排队,无法充分利用胖树中的并行路径,因此更适用于广播通信.

为提高多播的并行性,利用胖树中多条并行路径,基于负载均衡的路径选择算法被提出.文献[13]提出了基于局部负载信息,选择最小负载交换机作为父结点的算法 LLP(least loaded parent),即每个交换机均选择与之相连的下一级交换机中,负载最低的作为父结点.文献[14]在 LLP 算法的基础上,进一步提出了基于全局负载信息,选择最小负载路径的算法 LLP_EC(least loaded parent using equivalence classes),即为每个交换机评估所有可行路径的负载,然后选择最轻负载路径上的交换机作为其父结点.根据判定路径负载轻重的依据不同,LLP_EC 分为 LLP_EC_MAX 和 LLP_EC_SUM 两种,其中 LLP_EC_MAX 中选择上行下行路径负载最大值最小的路径,LLP_EC_SUM 选择上行下行路径负载加和最小的路径.

LLP 和 LLP_EC 的评测结果表明,基于负载均衡的方法可以实现较高的多播吞吐率.然而当多个结点同时计算多播路由时,由于获得了相同的负载信息,LLP 和 LLP_EC 算法会导致多个结点作出相似的路径选择,反而加重网络拥塞.其次,根据计算多播路由时的负载信息进行精确选径,没有考虑通信时的负载,增加了负载均衡的失效概率.

此外,在许多支持硬件多播的网络中,只支持连续目的结点的多播.对于存在空洞的多播操作,需要多个多播树共同完成.在此类网络中,多播的路径选择以最优化多播树数目为目标,即使用有限的多播树,为更多多播操作服务.文献[6]提出了基于贪心算法的多个多播树建立策略,但真正解决问题的方法是在多播网络设计中,避免多播树的建立或支持不连续目的地址多播树的建立.

通过上述分析可知,相比于确定路径选择算法,基于负载均衡的策略可以在一定程度上降低网络竞争,提高多播吞吐率.现有的方法只关注于路由计算时刻的负载,并使用精确选径方式,没有真正做到负载均衡.

4　曙光 5000 多播网络设计

曙光 5000 多播网络基于曙光 5000 互连网络单播网络实现,其拓扑、流控和交换方式均与单播网络相同.但是由于多播涉及多个目标地址,因此其路由方式与单播有所不同.

多播路由方式有源址路由和查表路由两种.文献[15]提出了在包头中携带各级交换机输出端口列表的源址路由方法(以位图模式),该方法可以避免多播树的建立,具有较好的扩展性,但在跳步数较多的网络中,会带来较大的包头开销,而且要求树形结构对称,不能够实现灵活的多播建立.查表路由方式,即通过配置路由表实现多播树的建立,之后的多播包均按照路由表的配置进行转发,多播包中只需携带对应的多播树序号,因此曙光 5000 多播网络采用包头中携带输出端口列表的源址路由方式.

为进一步提高多播网络性能,本节针对之前多播网络设计的不足,着重解决死锁和多播路径选择两个关键问题,提出了基于全局资源公告(RBB:resource bulletin board)的死锁避免方法,和基于重载交换机去除策略(HLSE:heavy-loaded switch eliminating)的多播路径选择机制.

4.1　基于全局资源公告的死锁避免机制

曙光 5000 互连网络采用 FIFO 方式对包缓冲区进行管理,数据包只能被读出一次,因此只能实现 no-splitting 方式的多播,有发生死锁的风险.曙光 5000 互连网络中采用虚切入交换方式,因此需要解决交换机内死锁问题.

本节提出了 RBB 死锁避免机制,该机制具有实现复杂度低和多播吞吐率高的特点.RBB 机制的核心思想是撤销无效请求,所谓无效请求,即输出端口不能全部得到满足的多播请求.RBB 是一个全局资源寄存器,采用位图格式,记录交换机中无多播包传输的输出端口.

采用 RBB 机制,交换机中多播处理流程如图 2 所示,分为五个步骤:

(1)多播发起请求之前,首先查询 RBB 寄存器;

(2)当多播包请求的所有输出端口在 RBB 寄存器中均有效,且都有足够流控信用时,发起请求;

(3)满足条件的请求经过矩阵仲裁器(Matrix Arbiter)进行公平仲裁,给出仲裁满足信号;

(4)RBB 把获得仲裁满足的多播包中包含的目的

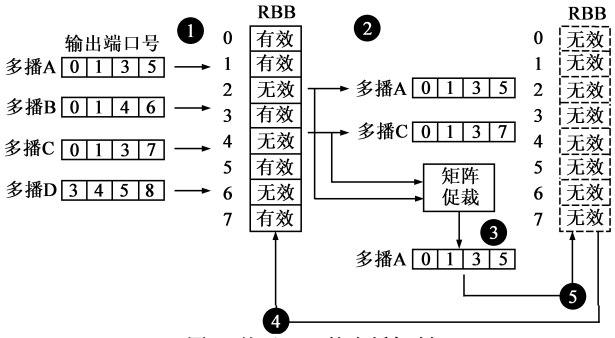


图2 基于RBB的多播机制

端口从 RBB 中除去;

(5)当发送完成,RBB 将第 4 步中去除的端口重置为有效.

从上述的第(2)步可以看到,通过使用 RBB,完全避免了一个多播操作既占有部分输出端口,又请求其他输出端口的情况,避免了循环依赖的发生,实现了多播包输出的原子性.由于采用了基于公平的仲裁策略,相比于文献[10]更有利于实现负载均衡.仲裁器给出仲裁满足信号后,即可进行下一个包的仲裁,因此仲裁和数据传输流水执行,可以获得很高的多播传输效率.

4.2 基于 HLSE 的多播路径选择机制

当多个结点发起多播通信时,所有结点获得的负载信息相同,因此 LLP 和 LLP_EC 精确选择路径的方法,会导致相似的路径选择,加剧网络拥塞,无法充分利用胖树中的并行路径.针对这一问题,本节提出了基于重载交换机去除策略的多播路径选择机制.

4.2.1 基于胖树拓扑的多播路由

首先给出胖树 $FT(m,n)$的三个定义[16],其中 m 为交换机端口数,n 为交换机层数:

定义 1 在 $FT(m,n)$网络中,结点标记为 $P(p=p_0p_1\cdots p_{n-1})$,其中 $p\in\{0,1,\cdots,m-1\}\times\{0,1,\cdots,(m/2)-1\}^{n-1}$.交换机标记为 $SW\langle w=w_0w_1\cdots w_{n-2},l\rangle$,其中 l 为交换机所在层数,$l\in\{0,1,\cdots,n-1\}$. w 的取值范围是:

$$w\in\begin{cases}\{0,1,\cdots,(m/2)-1\}^{n-1}, & \text{if } l=0\\ \{0,1,\cdots,m-1\}\times\{0,1,\cdots,\\ (m/2)-1\}^{n-2}, & \text{if } l\in\{1,2,\cdots,n-1\}\end{cases}$$

交换机 $SW\langle w,l\rangle$的第 k 个端口标记为 $SW\langle w,l\rangle_k$,其中 $k\in\{0,1,\cdots,m-1\}$.

定义 2 若交换机端口 $SW\langle w,l\rangle_k$ 与交换机端口 $SW\langle w',l'\rangle_{k'}$直接相连,当且仅当:

$$\begin{cases}l'=l+1\\ w_0w_1\cdots w_{n-3}=w_0'w_1'\cdots w_{l-1}'w_{l+1}'\cdots w_{n-2}'\\ k=w_l'+1\\ k'=w_{n-2}+(m/2)+1\end{cases}$$

定义 3 若交换机端口 $SW\langle w,l\rangle_k$ 与结点 $P(p)$直接相连,当且仅当:

$$\begin{cases}w_0w_1\cdots w_{n-2}=p_0p_1\cdots p_{n-2}\\ k=p_{n-1}+1\end{cases}$$

胖树中有关多播有如下三个定义[12]:

定义 4 给定胖树 $FT(m,n)$,对于结点 $P(p=p_0p_1\cdots p_n)$和结点 $P'(p=p_0'p_1'\cdots p_n')$,若 $p_0p_1\cdots p_{a-1}=p_0'p_1'\cdots p_{a-1}'$,且 $p_ap_{a+1}\cdots p_{n-1}\neq p_a'p_{a+1}'\cdots p_{n-1}'$,则两结点的最大公共前缀 $gcp(P(p),P'(p'))=p_0p_1\cdots p_{a-1}$,其中 a 为 $gcp(P(p),P'(p'))$的长度,记为$|gcp(P(p),P'(p'))|$.当 a 为 0 时,则标志两个结点没有共同前缀.

定义 5 给定胖树 $FT(m,n)$,结点 P 和结点 P'的最大公共前缀 $gcp(P(p),P'(p'))=p_0p_1\cdots p_{a-1}$,则结点 P 和结点 P'的所有最近公共祖先交换机集合 $lca(gcp(P(p),P'(p')))=\{SW\langle w,l\rangle\mid w_0w_1\cdots w_{a-1}=p_0p_1\cdots p_{a-1},l=a\}$.

定义 7 给定胖树 $FT(m,n)$,定义最大公共前缀组 $gcpg(x,a)$为一系列拥有公共前缀 x,且$|x|=a$ 的结点集合.$|gcpg(x,a)|$为其包含的结点数,最多包含 $(m/2)^{n-a}$个结点,一个多播组对应一个 $gcpg_g(x,a)$,一个 $gcpg_g(x,a)$对应一个 $lca_g(x)$,其中 g 为多播组组号,用以区分不同的多播树.

基于上述定义,可以得出定理 1.

定理 1 给定胖树 $FT(m,n)$,对于源结点 $P(p=p_0p_1\cdots p_n)$和目的结点 $P'(p=p_0'p_1'\cdots p_n')$,若选定 $lca(gcp(P(p),P'(p')))$中的一个交换机为根,则结点 P 到结点 P'的路径确定且唯一.

证明 选定 $lca(gcp(P(p),P'(p')))$中的一个交换机 $SW\langle w=p_0p_1\cdots p_{a-1}w_a\cdots w_{n-2},a\rangle$,根据定义 2,与之相连的交换机为 $SW^1\langle w=p_0p_1\cdots p_{a-1}w^1w_a\cdots w_{n-3},a+1\rangle$,依次轮推,在上行阶段与结点 P 相连的交换机为 $SW^b\langle w=p_0p_1\cdots p_{a-1}w^1\ w^2\cdots w^bw_a\cdots w_{n-2-b},a+b\rangle$,根据定义 3 可知:

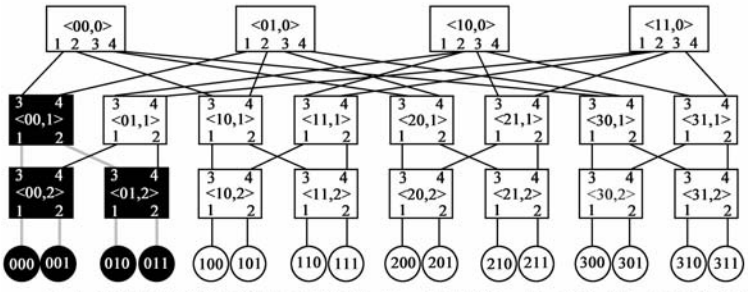
$$w^1\ w^2\cdots w^bw_a\cdots w_{n-2-b}=p_ap_{a+1}\cdots p_{n-2}$$

即
$$w^b=p_{a+b-1}\qquad(1)$$

代入式(1)可知,上行阶段在 $a+b$ 层经过的交换机为 $SW^b\langle w=p_0p_1\cdots p_{a-1}p_a\ p_{a+1}\cdots p_{a+b-1}w_a\cdots w_{n-2-b},a+b\rangle$,其中 $a+b<n$.同理,下行阶段在 $a+b$ 层经过交换机 $SW^b<w=p_0p_1\cdots p_{a-1}p_a'p_{a+1}'\cdots p_{a+b-1}'w_a\cdots w_{n-2-b},a+b\rangle$,其中 $a+b<n$.由 SWb 的表达式可知上行下行阶段的交换机编号由结点 P、结点 P'和交换机 SW 的编号确定,因此路径确定且唯一.

由定理 1 可以很容易的获得推论 1.

推论 1 给定胖树 $FT(m,n)$,对于 $gcpg_g(x,_a)$,若选定 $lca_g(x)$中的一个交换机为根,则连接 $gcpg_g(x,_a)$



$gcpg_g(x,a)$={P(000),P(001),P(010),P(011)}则x=0,a=1,相应的$lca_g(x)$={SW<00,1>SW<01,1>}

图3　胖树中多播路径示意图

中所有结点的多播路径确定且唯一.

以 $FT(4,3)$网络为例,对上述定义和定理进行说明,如图 3 所示.给定的结点集合 $gcpg_g(x,a)=\{P_0(000),P_1(001),P_2(010),P_3(011)\}$,则这四个结点的最大公共前缀 $gcp(P_0,P_1,P_2,P_3)=0$,即 $x=0,|x|=a=1$,$lca_g(x)=\{SW\langle 00,1\rangle,SW\langle 01,1\rangle\}$.若选定 $SW\langle 00,1\rangle$为根,则 $gcpg_g(x,a)$的多播路径唯一,该多播路径中包含的交换机为 $gcpg_g(x,a)$中任意两个最大公共前缀长度为 a 的结点以 $SW\langle 00,1\rangle$为根,所经过交换机的并集.

```
CalcRoute(FT(m,n),gcpg_g(x,a),lca_g(x),P)
FT(m,n):        m-port n-tree胖树
gcpg_g(x,a):    多播结点集合
lca_g(x):       根交换机集合
P(p=p0p1…pn-1): 源结点
Begin

    //获取根交换机rootSwitchID = <w0w1…wn-2,a>
    rootSwitchID = GetRootID(lca_g(x));

    for (i = 0; i < |gcpg_g(x,a)|; i=i+1)
    Begin
        //到结点P'(p'=p'0p'1…p'n-1)的路由计算
        //MulticastRoute[跳步号][端口号]
        a = |gcp(P,P')|;

        //顶层路由
        downPort = p'a+1;
        MulticastRoute[rootSwitchID][downPort] = 1;

        //上行路由
        For(j=a+1; j<=n-1; j=j-1)
        Begin
            mediaSwitchID = <w0w1…wa-1pa pa+1…pj-1wa…wn-2-j+a, j>;
            upPort = wn-1-j+a+(m/2)+1;
            MulticastRoute[mediaSwitchID][upPort] = 1;
        End

        //下行路由
        For(j=a+1; j<=n-1; j=j-1)
        Begin
            mediaSwitchID = <w0w1…wa-1p'a p'a+1…p'j-1wa…wn-2-j+a, j>;
            downPort = p'j+1;
            MulticastRoute[mediaSwitchID][downPort] = 1;
        End
    End
End
```

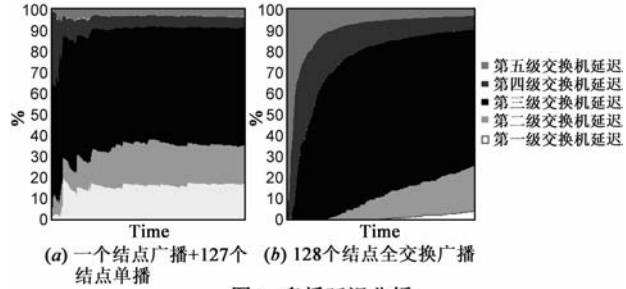
图4　计算多播路由伪代码

由推论 1 可知,在胖树拓扑中,只要选定根交换机,即可确定多播树结构.相应地,曙光 5000 多播网络的多播路由计算如图 4 所示,首先获取根交换机编号,然后以该交换机为根,为源结点 P 计算到达所有目的结点 P'所需的顶层路由,上行路由和下行路由.图 4 中的根交换机选择过程 GetRootID 将在下一节中详细介绍.

4.2.2　HLSE 多播路径选择机制

在曙光 5000 互连网络中,交换机负载信息通过为每个交换机端口设置统计寄存器实现.端口统计寄存器实时统计的输入输出带宽和平均输出延迟信息,均通过曙光 5000 互连网络的带外管理网络读取,并作为多播路径选择的依据.

使用 HPPNetSim 网络模拟器[17],以胖树 $FT(8,3)$为例,对多播通信行为进行分析.在一个结点发起广播,其他结点以随机均匀分布模式进行单播通信的情况下,上行阶段占据了多播包传输 90% 以上的延迟,其中根交换机(第三级交换机)的延迟最高,如图 5(a)所示.同样的现象也存在于全交换广播(all-to-all broadcast)情况下,如图 5(b)所示.因此在胖树拓扑中,根交换机一方面可以确定多播路径,另一方面是多播通信的瓶颈.根交换机的负载可以反映出整个多播路径的负载情况,多播路径的选择就是对根交换机的选择.



(a) 一个结点广播+127个结点单播　(b) 128个结点全交换广播

图5　多播延迟分析

与 LLP 和 LLP_EC 的精确选择方式不同,本节采用基于随机的 HLSE 多播路径选择算法.HLSE 算法的伪代码如图 6 所示,在根交换机的选择过程中,首先获取负载小于平均负载的交换机,然后随机作出最终的交换机选择.HLSE 算法的主要思想就是去除负载高的交换机,然后采用随机方式选择负载低的交换机.采用 HLSE 算法可以在发送时刻将多播通信散布于多条轻载的并行通信路径,很好的避免了精确选择方式造成的多播路径冲突,且可缓解负载消息滞后造成的影响.

下面以一个例子具体说明,整个过程涉及四个结点的多播路由计算.给定多播结点集合 $gcpg_0(x,0)=\{P(000),P(001),P(100),P(101)\}$,则根交换机集合为 $lca_0(x)=\{SW\langle 00,0\rangle,SW\langle 01,0\rangle,SW\langle 10,0\rangle,SW\langle 11,0\rangle\}$,在多播路由计算时刻,各根交换机的负载如图 7 所示,分别为 3,4,2 和 5.若采用 *LLP* 算法,则在四个结点的多播路径选择过程中,均会选择负载最轻的 $SW\langle 10,0\rangle$作为根交换机,导致 $SW\langle 10,0\rangle$的负载加重.而采用

```
GetRootID(lca_g(x))
lca_g(x): 根交换机集合
Begin

    meanLoad = 0;
    //计算根交换机平均负载
    for each SW  lca_g(x)
    Begin
        meanLoad += GetSWLoad(SW);
    End
    meanLoad = meanLoad / |lca_g(x)|;

    /* 记录负载小于平均值的交换机，将其插
入轻负载交换机列表llList */
    for each SW  lca_g(x)
    Begin
        if (GetSWLoad(SW) < meanLoad)
            InsertToLeastLoadList(SW, llList);
    End
    //在轻负载交换机列表中随机选取
    return RandomGetRootID(llList);
End
```

图6 HLSE算法伪代码

HLSE,则有 50% 的概率将四个结点的多播负载分摊在两个根交换机 $SW\langle 00,0\rangle$ 和 $SW\langle 10,0\rangle$ 中.系统规模越大,可供随机选择的根交换机越多,越有利于实现负载均衡.

5 多播性能评价

本节以 LLP,LLP_EC_MAX,LLP_EC_SUM 和轮转多播路由选择机制为对比,对曙光 5000 多播网络进行评测,所有评测均基于 FT(8,3)胖树网络,其中单播包长为 256 字节,多播包长为 1024 字节.

5.1 广播性能

本节针对存在单播通信的情况下,单结点广播的性能进行测试.本测试中,单播通信采用 Uniform random(均匀随机分布)、Sphere(50% 的单播数据包发往邻近的 k^2 个结点,其中 k 为交换机端口数的一半,其它 50% 的单播数据包按均匀随机分布发往其他结点)、Decay Probability(目标结点获得单播数据包的概率,随着与源结点的距离的增加而均匀降低)和 Hot-region(25% 的单播数据包发往 12.5% 的结点,其它 75% 单播数据包的目标地址采用随机均匀分布)四种模式,单播和广播均为 100% 注入率.

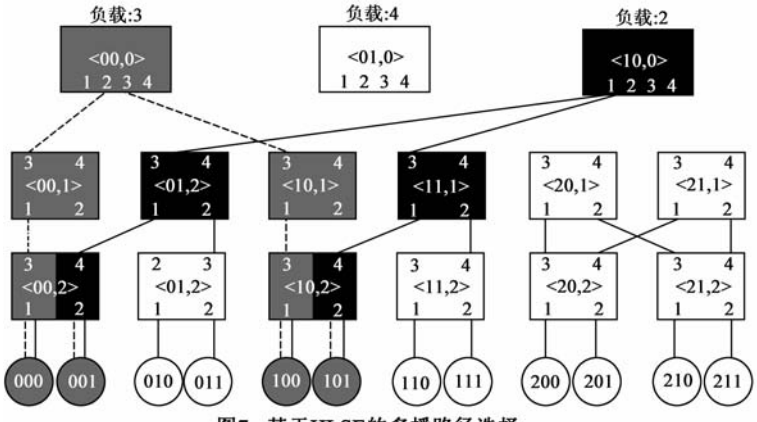


图7 基于HLSE的多播路径选择

曙光 5000 多播网络的广播性能如图 8 所示,横轴为单播通信模式,纵轴为 1000 次广播通信延迟的平均值.可以看到,广播延迟与单播通信模式相关,因为 Sphere 模式下,有 50% 的数据不会经过顶层交换机,因此顶层交换机负载较轻,可以获得最低的广播延迟.

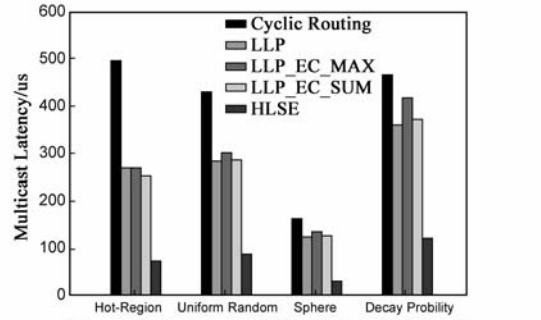


图8 各种路径选择机制下的广播通信延迟

由于没有进行负载均衡,轮转路由算法的广播延迟最高,在 Hot-region 通信模式下,128 个结点规模的广播延迟达到 500μs.与轮转路由算法相比,采用了负载均衡策略的 LLP、LLP_EC_MAX 和 LLP_EC_SUM 路由选择方式可以获得较低的延迟,相比于轮转路由最高有 1.9 倍的性能提升.

但是由于这些负载均衡策略采用精确选径的方式,导致突发的网络拥塞,因此延迟仍然较高.而本文的 HLSE 多播路径选择策略,不仅利用了胖树中的并发路径,而且规避了负载较高的交换机,因此获得了很低的延迟.除降概率通信模式外,HLSE 机制下的广播延迟均低于 100us,相比之前的负载均衡策略有近 3 倍的性能提升.

5.2 many-to-many 多播性能

曙光 5000 多播网络的 many-to-many 多播性能曲线如图 9 所示,四个图中采用了不同的多播扇出系数,多播目标地址均随机产生.可以看出,在曙光 5000 多播网络中,多播吞吐率随着扇出系数的增加而增加,由于轮转 路由算法保证来自不同结点的多播包在上行阶段不发生竞争,因此具有最好的性能,当扇出系数为 64 时,已经可以获得 90% 以上的多播吞吐率.

而 LLP 路由算法只根据当前交换机负载信息选择路径,因此性能最差,当扇出系数为 96 时,也仅能获得不到 70% 的多播吞吐率.LLP_EC_MAX、LLP_EC_SUM 和 HLSE 均基于全局负载信息进行路径选择,可以获得较高的

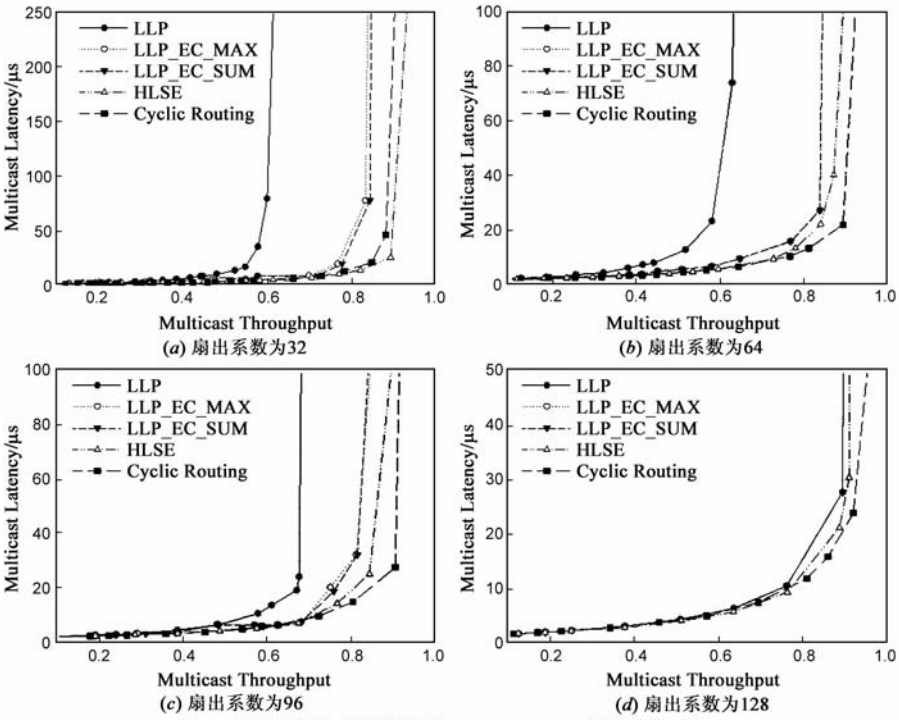
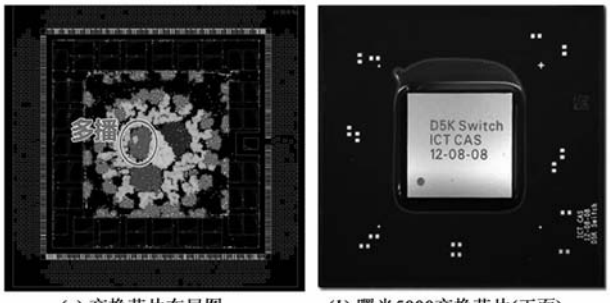


图9　各种路径选择机制下的many-to-many多播通信延迟

路由算法,提出了基于重载交换机去除策略(HLSE)的路径选择机制.测试结果表明,使用 HLSE 机制,在网络重载的情况下,相比于之前的负载均衡策略有近三倍的提升.对于 many-to-many 多播通信,曙光 5000 多播网络可以获得 90%以上的多播吞吐率.

本文的设计在曙光 5000 网络验证平台上通过了验证,其中支持硬件多播的 16 端口曙光 5000 交换芯片已经流片成功.曙光 5000 交换芯片如图 10 所示,采用 UMC HJ's generic 0.18μm/6M CMOS 工艺标准单元实现,集成了 20M 晶体管,共 1053 个管脚(其中 690 个 I/O 管脚),采用 Flipchip 封装.多播模块面积为 905,265μm^2,占芯片面积的 2.4%.



(a) 交换芯片布局图　　(b) 曙光5000交换芯片(正面)

图10　曙光5000交换芯片ASIC

参考文献:

[1] Y-D Gui, H Xu, L M Ni. Optimal software multicast in wormhole-routed multistage networks[A]. Proceedings of the Supercomputing Conference[C]. New York: ACM, 1994. 703 – 712.

[2] C-M Chiang, L M Ni. Efficient software multicast in wormhole-routed unidirectional multistage networks[A]. Symposium on Parallel and Distributed Processing[C]. Washington DC: IEEE Computer Society, 1995. 106 – 113.

[3] Sameer Kumar, Laxmikant V Kale. Scaling all-to-all multicast on fat-tree networks[A]. Proceedings of the Parallel and Distributed Systems[C]. Washington DC: IEEE Computer Society, 2004. 205 – 214.

[4] C B Stunkel, D G Shea, B Aball, et al. The SP2 high-performance switch[J]. IBM Systems J, 1995, 34(2): 185 – 204.

[5] N Koike. NEC Cenju-3: A microprocessor-based parallel computer[A]. Proceedings of the 8th International Parallel Processing Symposium[C]. Washington DC: IEEE Computer Society, 1994. 396 – 401.

[6] Salvador Coll, José Duato, Fabrizio Petrini, et al. Scalable hardware-based multicast trees[A]. Proceedings of the 2003 ACM/IEEE conference on Supercomputing[C]. Washington DC: IEEE Computer Society, 2003. 54 – 74.

[7] J Duato, M P Malumbres, J Torrellas. An efficient implementation of tree-based multicast routing in distributed shared-memory multiprocessors[A]. Proceedings of the 8th IEEE Symposium on Parallel and Distributed Processing[C]. Washington DC: IEEE Computer Society, 1996. 186 – 189.

[8] Petrini F, et al. Hardware and software-based collective communication on the Quadrics network[A]. IEEE International

6　总结

本文介绍了基于胖树拓扑的曙光 5000 多播网络设计.曙光 5000 多播网络基于胖树拓扑实现树形多播算法,多播数据在交换机中进行复制转发.此外,通过使用路由表路由方式,曙光 5000 多播网络实现了对不连续结点多播通信的支持.针对多播死锁问题,提出了基于全局资源公告(RBB)的死锁避免机制.而对于影响多播网络性能的多播吞吐率.由于本文的 HLSE 路径选择机制可以降低多个结点同时选择相同路径的概率,因此获得了与轮转路由相近的性能.当进行全交换广播时(扇出系数为 128),网络很快达到饱和,各种多播路径选择机制均可以达到 90%以上的吞吐率.

Symposium on Network Computing and Applications[C]. Washington DC: IEEE Computer Society, 2001. 24 – 35.
[9] Raoul Bhoedjang, Tim Rühl, Henri E Bal. Efficient multicast on Myrinet using link-level flow control[A]. Proceedings of the 1998 International Conference on Parallel Processing[C]. Washington DC: IEEE Computer Society, 1998. 381 – 389.
[10] Jaehyung Park, Lillykutty Jacob, Hyunsoo Yoon. Performance analysis of a multicast switch based on multistage interconnection networks[A]. Proceedings of the INFOCOM[C]. Washington DC: IEEE Computer Society, 1997. 939 – 946.
[11] Jay Herring, Craig B Stunkel, Bulent Abali, Rajeev Sivaram. A new switch chip for IBM RS/6000 SP systems[A]. Proceedings of the ACM/IEEE conference on Supercomputing[C]. Washington DC: IEEE Computer Society, 1999. 16 – 32.
[12] Jiazheng Zhou, Xuan-Yi Lin, Yeh-Ching Chung. Hardware supported multicast in fat-tree-based InfiniBand network[J]. The Journal of Supercomputing, 2007, 40(3): 333 – 352.
[13] Sameer Kumar. Optimizing Communication for Massively Parallel Processing[D]. Urbana Champaign: University of Illinois at Urbana Champaign, 2005.
[14] Quanbao Sun, Minxuan Zhang, Liquan Xiao. Hardware-based multicast with global load balance on k-ary n-trees[A]. International Conference on Parallel Processing[C]. Washington DC: IEEE Computer Society, 2007. 21 – 27.
[15] Rajeev Sivaram, Dhabaleswar K Panda, Craig B Stunkel. Efficient broadcast and multicast on multistage interconnection networks using multiport encoding[A]. Eighth IEEE Symposium on Parallel and Distributed Processing[C]. Washington DC: IEEE Computer Society, 1996. 1004 – 1028.
[16] Chung YC, Lin XY, Huang TY. A multiple LID routing scheme for fat-tree-based infiniband networks[A]. Proceedings of IEEE international parallel and distributed proceeding symposiums[C]. CD-ROM, 2004.
[17] Zheng Cao, Jianwei Xu, Mingyu Chen, et al. HPPNetSim: A parallel simulation of large-scale interconnection network[A]. 42nd Annual Simulation Symposium[C]. San Diego: Society for Computer Simulation International, 2009. 32 – 39.

作者简介:

曹 政 男,1982 年生于山东济宁,博士,中国科学院计算所助理研究员,主要研究方向为分布式计算、计算机体系结构、高性能互联网络等.
E-mail: cz@ncic.ac.cn

王达伟 男,1980 年生于河北保定,博士,中国科学院计算所助理研究员,主要研究方向为分布式计算、计算机体系结构、高性能互联网络等.

刘新春 男,1968 年生于湖南衡阳,博士,中国科学院计算所副研究员,主要研究领域为可重构计算、计算机体系结构、高性能互联网络等.

孙凝晖 男,1968 年生于安徽寿县,博士,中国科学院计算所研究员,博士生导师,主要研究领域为并行体系结构、分布式操作系统、高性能计算等.