ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.18, No.10, October 2007, pp.2538−2550
DOI: 10.1360/jos182538

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

利用直连网络实现可扩展路由器*

乐祖晖+, 赵有健, 吴建平

(清华大学 计算机科学与技术系,北京 100084)

Implementation of Scalable Routers with Direct Networks

YUE Zu-Hui+, ZHAO You-Jian, WU Jian-Ping

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

+ Corresponding author: Phn: +86-10-62795818 ext 6854, E-mail: yuezuhui@gmail.com, http://www.tsinghua.edu.cn

Yue ZH, Zhao YJ, Wu JP. Implementation of scalable routers with direct networks. *Journal of Software*, 2007, 18(10):2538−2550. http://www.jos.org.cn/1000-9825/18/2538.htm

Abstract: In the Internet, the exponential growth of user traffic has been driving routers to run at higher capacity. Traditional routers consist of line cards and centralized switching fabrics. The centralized switching fabric in such a router, however, is becoming the bottleneck for its limited port numbers and complicated scheduling algorithms. In addition, the fabric is the single point of failure (SPF) in the router. Direct networks, such as 3-D Torus topology, have been successfully applied to the design of scalable routers. They show good scalability and fault tolerance. Unfortunately, its scalability is limited in practice. This paper introduces another type of direct network, called cellular router (CR). With a little modification, this network shows excellent topological properties. Based on this network, two classes of minimal routing algorithms are introduced. The load-balanced minimal routing (LBMR) algorithm makes use of path diversity and shows low latency and high throughput on both uniform random (UR) and Tornado traffic. This paper also discusses some other aspects of the routing algorithms, such as effects of queue length and scheduling algorithms. The CR architecture is a promising choice for the design of scalable routers.
Key words: scalable router; cellular router; cobweb; switch fabric; direct network

摘 要: Internet 的迅速发展直接表现为用户流量的迅速增长,这就要求路由器必须提供更大的容量.传统的路由器由线卡和集中式交换网络构成.集中式交换网络只能支持有限的端口数目,而且随着端口数目的增加,调度算法也变得越来越复杂,所以交换网络正成为整个路由器的性能瓶颈.集中式交换网络还是路由器的单一失效点,无法提供令人满意的容错性能.直连网络具有良好的扩展性和容错性.其中,3-D Torus 拓扑结构已被成功应用到可扩展路由器的设计当中.但是在实际应用中,3-D Torus 结构受到等分带宽的约束,限制了扩展规模.介绍了一种新型的直连网络结构,称为蜂巢式结构.将对蜂巢结构作简单的改动,修改后的拓扑表现出很好的拓扑属性.基于该结构,提出了两类最短路径路由算法.其中,负载均衡的最短路径路由算法较好地利用了直连网络路径多样性的特点,针对均匀随机和Tornado两种类型的流量都表现出较低的分组延时和较高的吞吐量.另就队列长度和单节点调度算法等方面对路由算法的影响进行了讨论.蜂巢结构为可扩展路由器的设计提供了新的选择.

* Supported by the National Natural Science Foundation of China under Grant No.90604029 (国家自然科学基金); the National Basic Research Program of China under Grant No.2003CB314801 (国家重点基础研究发展计划(973))

Received 2006-04-11; Accepted 2006-07-26

关键词: 可扩展路由器;蜂巢式路由器;蛛网;交换网络;直连网络
中图法分类号: TP393 文献标识码: A

路由器是计算机网络中非常重要的设备,主要负责将到达输入链路的分组转发到输出链路上.因此,路由器从逻辑上可以划分为两个组成部分:1) 查找.根据分组中包含的目的地址信息,在转发表中查找,得到对应的输出端口号;2) 交换.将分组由输入链路交换到相应的输出链路.在实际的路由器中,由线卡完成查找(当然,线卡还需要完成很多其他的任务,例如分组内容处理、分片和重组等),交换网络完成交换任务.

Internet 发展迅速,网络用户对网络容量的需求(通过对用户流量进行测量)按每年大约提升 1 倍的速度增长[1],而商用路由器的容量增长速度只是接近摩尔定律,即每 18 个月增加 1 倍[2].为了满足不断增长的用户需求,路由器需要提供更大的容量,也就意味着路由器需要支持更多的端口数目或更高的端口速率.但是,很多因素制约了端口速率的增长[3].因此,路由器的研究重点正由单纯提高端口速率逐渐转移到交换网络如何支持更多的端口数目.

本文第 1 节介绍与本文相关的工作.第 2 节在概述直连网络中一些基本概念的基础上,介绍蜂巢式路由器及其变体.第 3 节定义两类最短路径路由算法.第 4 节用实验来评测这两类路由算法的性能.最后是对全文的总结,指出进一步的研究工作.

1 相关工作

早期的路由器多采用背板总线或 Crossbar 结构来实现交换网络.总线技术受到带宽限制,扩展能力较差.由于 Crossbar 结构具有简单、无阻塞的特点,在端口数目较少的情况下,多采用 Crossbar 来构建交换网络.但是,Crossbar 结构的开销与端口数目的平方成正比,这在很大程度上限制了它的扩展能力.

多级交换网络可以支持中、大规模的端口数目,自从电信时代[4]就有人开始研究这类网络.Banyan 网[5]的开销为 $N\cdot\log N$(其中 N 表示端口数目,下同),在输入、输出端口间提供大量可选路径.但是,Banyan 网在实现交换时会出现内部阻塞问题.Benes 网[6]的开销为 $N\cdot 2\log N$,不会出现内部阻塞问题.但是,Benes 网要实现严格意义上的无阻塞就必须重新配置,这会破坏现有的连接关系.Clos 网[7]在理论研究和实际应用中,都具有非常重要的意义.但是,目前 Clos 网中仍然存在一些无法解决的理论问题.

目前,绝大多数的商用路由器都采用了上述的某类交换网络,它们存在一个共同的特点是都属于集中式交换网络.众所周知,集中式结构是不利于扩展的.另外,集中式交换网络将成为路由器的单一失效点,一旦交换网络出现故障,整台路由器将陷入瘫痪.而且,这些交换网络要达到较高吞吐率,就必须配合使用复杂的调度器.例如,应用于 Crossbar 结构的最大权重匹配(maximum weight matching,简称 MWM)算法,如 LQF(longest queue first)和 OCF(oldest cell first)等[8],在允许的流量下,不需要内部加速比就可以达到 100%的吞吐率.但是,这些算法的计算复杂度为 $O(N^3)$,很难利用硬件实现.目前,在实际的路由器中多采用一些利用近似算法实现的调度器,虽然计算复杂度有所降低,但是仍然和端口数目相关,这就意味着调度器的复杂度将随着端口数目的增加而增加,不利于路由器的持续扩展.

Chang 等人[9]提出了一类新型的交换网络,称为负载均衡型交换网络(load-balanced switches),该交换网络取消了调度器,只需通过简单的轮寻就可达到较高的吞吐率.另外,该结构特别适合利用光技术加以实现[2].但是,文献[2]中提到的交换网络用到了一些目前无法实现的技术.

另外,实现路由器的平滑升级也是一个非常实际但又容易被忽视的问题.路由器要提供更大的容量,就必须增加线卡数目或者提高端口的处理速率.以 Crossbar 结构为例,交换网络无法支持线卡数目的持续增加.多级交换网络虽然可以支持端口数目的不断增加,但是扩展粒度太大,容易造成部分资源的闲置.

直连网络(direct network)最初主要应用于处理器和存储器之间的互连,或是 I/O 端口间的互连.后来,这一技术又成功应用于可扩展路由器的设计.其中,基于 3-D Torus(又称为 k-ary 3-cube)结构[10]的交换网络在 Avici 公司的 TSR 路由器中得到应用[11].TSR 中的每个线卡作为该拓扑结构的一个节点,在源节点和目的节点之间存在

多条可选路径.这一设计具有下述优点:扩展粒度小、规模大;有利于实现负载均衡;具有较高的容错性.

虽然 3-D Torus 拓扑结构本身具有较好的扩展性能,但是在 TSR 的具体实现中,线卡数目达到 560 后[11],由于整个交换网络的等分带宽不再增加,为了提供所需的加速比,再增加线卡就会降低整个路由器的性能,因此而限制了 TSR 的扩展规模.

本文提出的蜂巢式路由器是一类新型的交换网络,能够支持更多数量的线卡,特别适用于可扩展路由器.而且在源节点和目的节点之间存在更多的可选路径.通过一些修改,该结构具有很好的拓扑属性.

2 蜂巢结构及其变体

在介绍蜂巢结构之前,首先给出直连网络中的一些基本概念.

2.1 直连网络基本概念

文献[12]给出了直连网络的定义:直连网络中的每一个节点,既是网络终端,同时又是信息转发节点.区别于传统的集中式交换网络,直连网络中,线卡(为了叙述方便,后面统一称为节点)除了负责分组输入、输出和内容处理外,还需要承担分组在各个线卡间转发的任务;分组由源节点到达目的节点的过程中可能需要经过多个中间节点.

直连网络中,用 V 表示节点集合,用 E 表示节点间的链路集合.$|V|$代表节点数目,$|E|$代表链路数目.$v_i(i=0,1,\ldots,|V|-1)$表示第 i 个节点,$e_j(j=0,1,\ldots,|E|-1)$表示第 j 条链路.记 $e_j=(v_m,v_n)\in E$,其中,$v_m,v_n\in V$,称 v_m 为链路 e_j 的起点,v_n 为链路 e_j 的终点.节点 v_i 的邻居数目称为节点 v_i 的度,记为 $deg(v_i)$.源节点 s 到目的节点 d 的路由 r,可以表示为节点序列 $v_0v_1\ldots v_i\ldots v_k(v_i\in V,\forall i\in\{0,1,\ldots,k\})$,且满足下述条件:1) $v_0=s,v_k=d$;2) 节点 v_i 和 $v_{i+1}(\forall i\in\{0,1,\ldots,k-1\})$之间存在一条链路 $e=(v_i,v_{i+1})$.路由 r 的长度记为 $len(r)$,s 和 d 的路由集合记为 R_{sd}.源节点 s 到目的节点 d 的最短路径路由为 $r_{\min}$,满足:$len(r_{\min})\le len(r),\forall r\in R_{sd}$;$len(r_{\min})$又称为节点 s 到节点 d 的距离.

在设计一个直连网络时,需要兼顾拓扑、路由和流控 3 个方面.其中,拓扑主要表示直连网络中数据通道和节点间的静态连接关系;基于选定的拓扑结构,路由算法为每个分组选择一条可以到达目的节点的路由;在分组路由过程中,流控负责资源分配.为了简化设计,一般假设两个相邻节点间存在两条单向链路.对于外部注入流量,路由器不提供反压机制,这就意味着路由器无法控制注入其中的流量.所以,要求路由算法和流控方案合理配合,尽可能地提高分组吞吐率,降低分组延时.

在评价网络性能时,直径和等分带宽是两个重要的参数.网络直径定义为网络中所有节点对的距离的最大值;将网络 N 的节点集合 V 划分为两个集合 V_1 和 V_2(满足:① $V_1\cap V_2\neq\varnothing$;② $V_1\cup V_2=V$;③ $||V_1|-|V_2||\le 1$),所需要去除的边的数目称为网络 N 的等分带宽,记为 B_N.较小的网络直径可以有效降低分组交换的平均延时;增加网络的等分带宽,可以缓解整个交换的瓶颈.

在直连式交换网络中,传统的交换调度问题转换为路由问题.路由算法的好坏将直接影响到整个交换网络的性能.文献[12]中给出了网络容量的定义:$C(N)=2B_N/|V|$.如图 1 所示,对于均匀随机类型的流量,由 V_1 注入网络的流量中,将有$|V_1|/2$ 的流量经过等分截面;同理,对于 V_2,也会有$|V_2|/2$ 的流量经过等分截面,所以,总计有$|V|/2$ 的流量将经过等分截面,每个节点允许向网络中注入的最大流量为 $B_N/(|V|/2)=2B_N/|V|$.在讨论吞吐率和注入速率时,都将针对 $C(N)$作归一化处理.由此可以得到下面的命题:

命题 1. 节点数目相同的各类直连式交换网络,当注入均匀随机类型流量时,每个节点允许注入的最大流量与交换网络的等分带宽成正比.

最坏情况下,$V_1(V_2)$中所有节点注入的流量都去往 $V_2(V_1)$,这时,每个节点允许注入的最大流量为 $B_N/|V|=C(N)/2$,这种类型的流量称为直径流量[12](diameter traffic,简称 DIA).所以,对于任意路由算法 r 而言,其所能达到的最大吞吐量不会超过网络容量的 50%,即$\Theta(r,DIA)\le 0.5$.

Valiant 提出了一种完全随机的路由算法[13](VAL).该算法分为两个阶段:1) 随机选择一个中间节点 v_j,分组由源节点 s 路由到节点 v_j;2) 分组由节点 v_j 路由到目的节点 d.基于 VAL 算法可以提出下面的命题:

命题 2. 在直连式交换网络中,当采用 VAL 路由算法时,2 倍的加速比可以达到 100%的吞吐率.

证明:VAL 算法的每个阶段都等价于均匀随机类型流量的路由,假设各个节点的分组注入速率为α,则两个阶段经过等分截面的总流量为 $N\alpha$,即 $N\alpha=B_N$,可得$\alpha=B_N/N=C(N)/2$.所以,$\Theta(DAL)=0.5$.

所以,当采用 VAL 算法时,2 倍的加速比就可以达到 100%的吞吐率. □

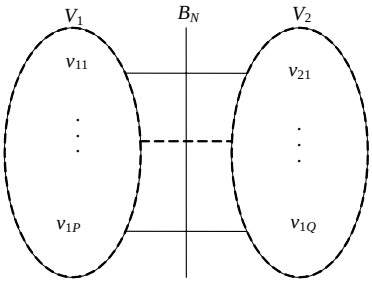


Fig.1 The node set V of network N is divided into two disjointed node set V_1 and V_2

图 1 网络 N 的节点集合 V 被等分截面划分为两个不相交的节点集合 V_1 和 V_2

2.2 蜂巢结构及其变体

蜂巢结构是基于增加了中心点的正六边形拓扑而提出的,文献[14]中详细介绍了蜂巢结构,这里只作简单介绍,并对蜂巢结构的变体:H-Mesh 和 H-Torus 进行详细的讨论.

2.2.1 蜂巢结构

蜂巢结构的基本单元就是一个增加了中心点的正六边形,如图 2 所示.图 2 中的数字 0~6 表示节点添加到基本单元的次序,每增加一个节点,相应的边也会同时增加.文献[14]给出了基本单元在单机架单层结构中的具体部署方案.

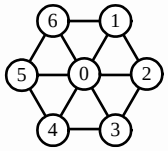


Fig.2 Basic element with 7 nodes

图 2 由 7 个节点构成的基本单元

路由器在实际部署时,为了有效利用空间,经常采用多层结构.针对蜂巢结构,可以考虑在单机架的每层部署一个基本单元,层间的对应节点通过数据链路相连.为了提高系统的容错性,可以将机架最顶层和最底层的对应节点也互相连接起来,如图 3 所示.

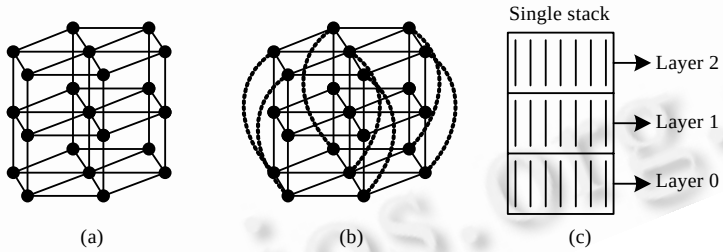


Fig.3 Multi-Layers in a single stack

图 3 单机架中部署的多层结构

显然,单机架结构无法容纳持续增加的线卡.多机架结构是目前普遍采用的路由器扩展方案,文献[14]中提出的蛛网结构可以解决蜂巢结构在多机架情况下的部署问题.

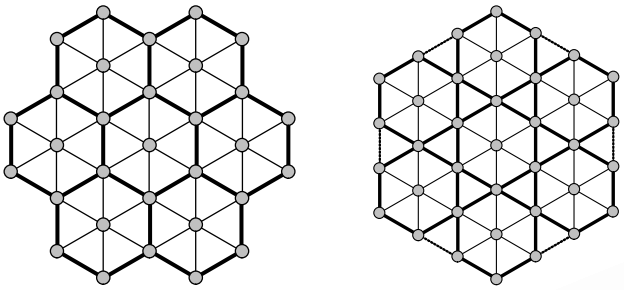
2.3 蜂巢结构的变体

蜂巢结构以正六边形作为基本扩展单元,系统中每个内部节点的度均为 6,具有良好的容错性能.但是,基本蜂巢结构的扩展粒度较大,而且在处理外围节点的连接时无法给出节点对称方案.文献[15]针对 H-Mesh 结构,给出了一种外围节点的连接方案,称为连续类型的连接(continuous type wrapping).改造后的 H-Mesh 结构具有

出色的拓扑属性.

2.3.1 蜂巢结构到 H-Mesh 结构的“转变”

如图 4 所示,如果将蜂巢结构中的基本单元(单层或者多层)进行重新组织(图 4(b)中的虚线也是实际存在的数据链路),就可以很容易得到 H-Mesh 结构.H-Mesh 结构中的外围节点位于同一个正六边形上.



(a) (b)

Fig.4 Cellular router to H-Mesh

图 4 蜂巢结构到 H-Mesh 结构的“转变”

2.3.2 H-Torus 结构

H-Mesh 结构中,外围节点的度均小于 6,所以该拓扑结构不属于规则图.文献[15]针对 H-Mesh 结构,对外围节点进行了统一处理,即连续类型的连接.如图 5 所示,在 H-Mesh 结构中引入了 x,y 和 z 坐标轴,在 3 个方向上,分别定义第 0 行、第 1 行、…、第 $2t$–2 行.每个方向上,第 i 行的尾节点与第 $j(j=(i+t-1)\bmod(2t-1))$行的首节点相连.由文献[15]可知,经过这样的处理,所得到的拓扑结构为同构图,该拓扑结构记为 H-Torus.

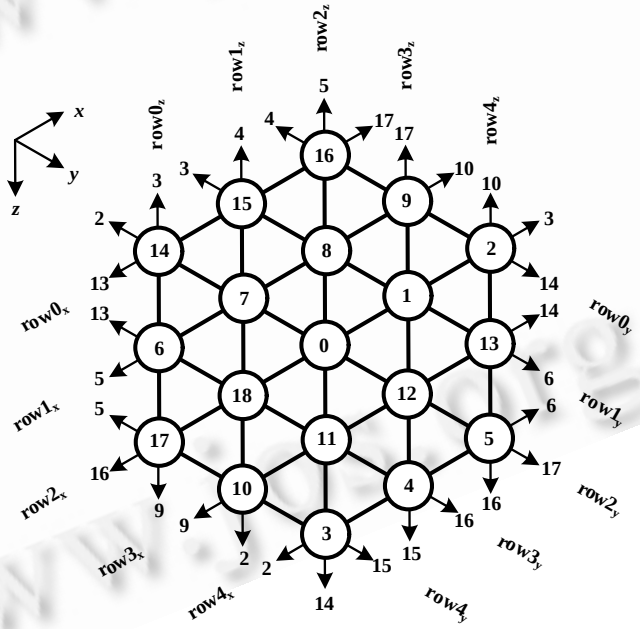


Fig.5 H-Torus

图 5 H-Torus 结构

2.3.3 H-Mesh 和 H-Torus 结构的拓扑属性

为了讨论方便,首先定义只包含一个节点的 H-Mesh 结构的圈数为 1,记为 HM_1,相应的 H-Torus 结构记为

HT_1.圈数为 t 的 H-Mesh 结构记为 HM_t,相应的 H-Torus 结构记为 HT_t.

命题 3. $HM_t(HT_t)$的节点数目为 $3t^2-3t+1$;令 $HM_t(HT_t)$的节点数目为 n,则 $t=\frac{3+\sqrt{12n-3}}{6}$.

证明:HM_t 与 HT_t 的节点数目相同,所以只需要考虑 HM_t 中的节点数目即可.

很容易发现,HM_1 的节点数目为 1,$HM_t(t\geq 2)$中,最外圈节点的数目为 $6\times(t-1)$,所以可得 $HM_t(t\geq 2)$的节点数目为 $1+6+\ldots+6\times(t-1)=1+3\times(t-1)=3t^2-3t+1$.

令 $n=3t^2-3t+1$,可得: $t=\frac{3+\sqrt{12n-3}}{6}$. □

命题 4. HM_t 的链路数目为 $9t^2-15t+6$,HT_t 的链路数目为 $9t^2-9t+3$.

证明:$HM_t(t\geq 2)$中最外圈节点的数目为 $6\times(t-1)$,其中,度为 3 的节点数目为 6,度为 4 的节点数目为 $6\times(t-2)$;HM_t 中,内部节点的数目为$(3t^2-3t+1)-(6\times(t-1))=3t^2-9t+7$,这些节点的度均为 6.所以,总的链路数目为 $[6\times(3t^2-9t+7)+3\times 6+4\times 6(t-2)]/2=9t^2-15t+6$.

HT_t 的节点数目为 $3t^2-3t+1$,所有节点的度均为 6,所以总的链路数目为 $6\times(3t^2-3t+1)/2=9t^2-9t+3$. □

命题 5. HM_t 的直径为 $2t-2$,HT_t 的直径为 $t-1$.

证明:$t=1$ 时,HM_1 的直径为 $0,2\times 1-2=0$,结论成立;假设 $t=k(k\geq 1)$时结论成立,即 HM_k 的直径为 $2k-2$;由拓扑结构易知,对于 HM_{k+1} 中任意两点 A 和 B,当且仅当 A 和 B 的度均不为 6 时(即 A 和 B 位于拓扑结构的最外圈上),A 和 B 之间的距离才可能是直径.可以在 HM_k 的外围上找到两个节点 M 和 N,使得$|AM|\leq 1$,$|NB|\leq 1$,所以有

$$|AB|\leq|AM|+|MN|+|NB|\leq 1+|MN|+1\leq 1+2k-2+1=2(k+1)-2.$$

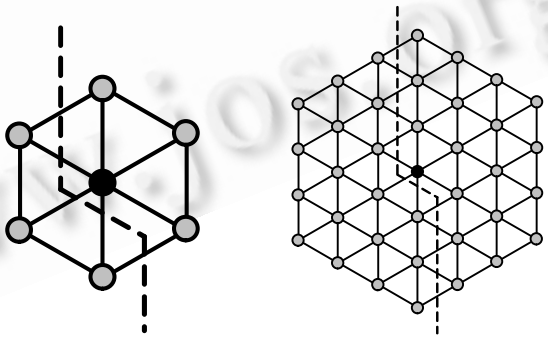
所以,HM_{k+1} 的直径为 $2(k+1)-2$,即当 $t=k+1$ 时结论也成立.所以,HM_t 的直径为 $2t-2$.

由于 HT_t 属于节点对称的拓扑结构,所以只需考虑 HT_t 的中心点即可.很容易发现,中心点到达最外圈节点的距离即为该拓扑结构的直径,所以 HT_t 的直径为 $t-1$.证毕. □

命题 6. $HM_t(t\geq 2)$的等分带宽为 $4t-3$,$HT_t(t\geq 3)$的等分带宽为 $10t-10$.

证明:HM_1 的等分带宽为 0,HM_2 的等分带宽为 5(如图 6(a)所示),由图 6(b)易知,每增加一圈,等分带宽增加 4,所以,$HM_t(t\geq 2)$的等分带宽为 $5+4\times(t-2)=4t-3$,结论成立.

HT_1 的等分带宽为 0,HT_2 的等分带宽为 12.当 $t\geq 3$ 时,相比 HM 结构,HT 结构增加了外围节点的连接.HT_t 按图 6(b)进行分割时,x 轴方向上,每一行的终点都无法连接到对应下一行的起点,所以共计断开了 $2t-1$ 条边;y 轴方向上,除了第 t 行的终点和第 0 行的起点的连接、第 $2t-2$ 行的终点和第 $t-2$ 行的起点的连接没有断开外,其余的外围边都被切断,所以共计断开 $2t-3$ 条边;z 轴方向上,除了第 0 行的终点和第 $t-1$ 行的起点的连接、第 $t-1$ 行的终点和第 $2t-2$ 行的起点的连接没有断开外,其余的外围边都被切断,所以共计断开了 $2t-3$ 条边.由前面得出的 HM_t 的等分带宽可知,HT_t 的等分带宽为$(4t-3)+(2t-1)+(2t-3)+(2t-3)=10t-10$. □



(a) (b)

Fig.6 Bisection width

图 6 等分带宽

2.3.4 几类常见拓扑结构的性能比较

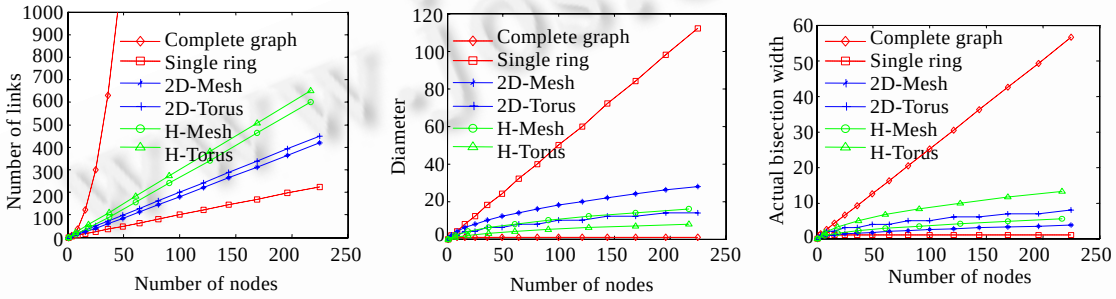
表 1 列出了 6 类常见拓扑结构在节点数目、度、链路数目、网络直径和等分带宽等几个方面的比较.

Table 1 Comparison of various topologies

表 1 几类拓扑结构的比较

	Number of nodes	Degree of each node	Number of Links	Network diameter	Bisection width (number of links)
Complete graph	n	n–1	$n(n-1)/2$	1	$n^2/4$, n is even $(n^2-1)/4$, n is odd
Single ring	n	2	n	$n/2$, n is even $(n-1)/2$, n is odd	2
2D Mesh	$n=t^2$	4 (some nodes have degree of 2 or 3)	$2n-2t$	$2\sqrt{n}-2$	$\sqrt{n}$
2D Torus	$n=t^2$	4	$2n$	$\sqrt{n}$, n is even $\sqrt{n}-1$, n is odd	$2\sqrt{n}$, n is even $2(\sqrt{n}+1)$, n is odd
H-Mesh	$n=3t^2-3t+1$	6 (some nodes have degree of 3 or 4)	$9t^2-15t+6$	$\frac{\sqrt{12n-3}}{3}-1$	$\frac{2\sqrt{12n-3}}{3}-1 (t\geq 2)$
H-Torus	$n=3t^2-3t+1$	6	$3n$	$\frac{\sqrt{12n-3}-3}{6}$	$\frac{5\sqrt{12n-3}}{3}-5 (t\geq 3)$

假设网络中每个节点的总交换能力均为 1,即:$deg(v)\times b_c=1$,这里,b_c 表示单条链路的带宽.因此,网络 N 的实际等分带宽 $B_N^*=B_N b_c=B_N/deg(v)$.图 7 给出了上述 6 类拓扑结构的性能比较:链路数目、网络直径和实际等分带宽.



(a) Number of links (a) 链路数目 (b) Diameter (b) 网络直径 (c) Actual bisection width (c) 实际等分带宽

Fig.7 Comparison of various topologies

图 7 几类拓扑结构的比较

由图 7 不难发现,在节点数目相同的情况下,完全图拓扑的网络直径最小、实际等分带宽最大,但是所需的链路数目太大,在实际应用中很难部署.H-Torus 拓扑的链路数目增长不快,而且网络直径和实际等分带宽仅次于完全图,具有良好的拓扑属性.

3 最短路径路由算法

3.1 H-Torus中节点标识

文献[15]介绍了一种最短路径路由算法,但该算法较为复杂,不利于在高速硬件中实现.本文给出一种节点标识方案,可以在很大程度上简化路由算法的复杂度.

首先,根据文献[15]提出的方案,沿 x 轴方向对所有节点进行编号(如图 5 所示),该编号可以唯一标识各个节点.H-Torus 属于节点对称结构,所以,拓扑结构中的任意一个节点都可以视为 H-Mesh 结构的中心节点.如图 8 所示,以节点 s 为中心,定义 6 条轴线:$axis_{s,0},axis_{s,1},\ldots,axis_{s,5}$.另外,定义 6 个区域:$region_{s,0},region_{s,1},\ldots,region_{s,5}$.显然,$s$ 的 6 个邻居节点 $n_0,n_1,\ldots,n_5$ 分别位于 $axis_{s,0},axis_{s,1},\ldots,axis_{s,5}$ 上.除 s 外的任意节点 d 必然位于某条轴线 $axis_{s,i}(i\in\{0,1,\ldots,5\})$上或某个区域 $region_{s,j}(j\in\{0,1,\ldots,5\})$中.例如,图 8 中的 d_1 位于 s 的 $region_{s,4}$ 中,d_2 位于 s 的

$axis_{s,3}$ 上.

实际系统中,任意给定节点 s 除了记录自己的全局编号(记为 $GlobalID_s$)外,还需要记录其余各个节点 d(全局编号为 $GlobalID_d$)相对 s 的位置信息:(t_{ds},pos_{ds}),其中,t_{ds} 表示 d 到 s 的距离,pos_{ds} 表示 d 相对节点 s 所处的轴线或区域.例如,在图 8 中,节点 s 记录的 d_1 的位置信息为(2,$region_4$),d_2 的位置信息为(2,$axis_3$).

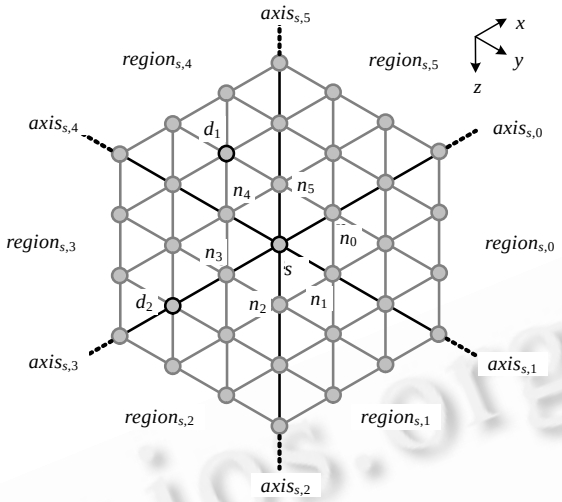


Fig.8 Division of topology based node s

图 8 基于节点 s 的拓扑分割

3.2 位置信息的建立

对于 H-Torus 中任意给定节点 s,其余的节点可以分为两类:s 的邻居和非 s 的邻居.下面将分别介绍如何根据 s 的 $GlobalID$ 来确定这两类节点相对于节点 s 的位置信息.

3.2.1 邻居节点信息的建立

文献[15]中定义$[b]_a \equiv b \bmod a$,HT_t 中任意给定两个节点 $s(x_1,y_1,z_1)$和 $d(x_2,y_2,z_2)$,存在下述关系(其中,$p=3t^2-6t+1$):1) $[x_2-x_1]_p=[(3t^2-6t+3)(y_2-y_1)]_p$;2) $[x_2-x_1]_p=[(3t^2-6t+2)(z_2-z_1)]_p$.若沿 x 轴方向对节点进行标记,则 $GlobalID_s=x_1$,$GlobalID_d=x_2$.由 $GlobalID_s$ 很容易得到 $Global_{n0},Global_{n1},\ldots,Global_{n5}$,具体算法如图 9 所示.其中,输入 s 代表 $GlobalID_s$,$axis$ 代表邻居位于 s 的哪条轴上,t 表示 H-Torus 的圈数,p 表示 H-Torus 的节点数目.

```
NeighborID(s,axis,t)
1)    IF (axis=0) THEN
2)        BEGIN nid=(s+1)modp END
3)    ELSE IF (axis=1) THEN
4)        BEGIN nid=(s−3×t+2)modp END
5)    ELSE IF (axis=2) THEN
6)        BEGIN nid=(s−3×t+1)modp END
7)    ELSE IF (axis=3) THEN
8)        BEGIN nid=(s−1)modp END
9)    ELSE IF (axis=4) THEN
10)       BEGIN nid=(s+3×t−2)modp END
11)   ELSE IF (axis=5) THEN
12)       BEGIN nid=(s+3×t−1)modp END
```

Fig.9 Computing the neighbor's $GlobalID$

图 9 计算邻居的 $GlobalID$

3.2.2 非邻居节点信息的建立

如图 10 所示,按箭头所示方向,通过不断调用 *NeighborID* 方法,可以得到沿途各个节点的 *GlobalID*,而且还可以得到各个节点 *d* 相对于节点 *s* 的位置信息.在路由器线卡中,可以利用 SRAM(static random access memory)芯片存放这些信息,每一条记录的具体格式为(*GlobalID*,*t*,*pos*).

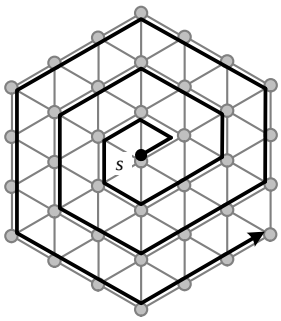


Fig.10 Computing the information of all the nodes based on node *s*

图 10 基于节点 *s* 计算所有节点的信息

3.3 两类最短路径路由算法

假设分组由节点 *s* 注入,目的节点为 *d*,例如,图 8 中的节点对(s,d_1).节点 *s* 上记录的关于节点 d_1 的信息为$(GlobalID_{d_1},2,region_4)$,记节点 n_4 为节点对(s,d_1)的左邻居(left neighbor,简称 LN),节点 n_5 为节点对(s,d_1)的右邻居(right neighbor,简称 RN).节点对(s,d_2)的左邻居和右邻居重合,即为节点 n_3.如果在分组的路由过程中,始终选择当前节点和目的节点的左(或右)邻居,记为左(或右)邻居优先最短路径路由算法,简记为 LNF(或 RNF)算法.类似 Torus 结构中的维度优先算法[16](dimension-order routing,简称 DOR),LNF(left neighbor first)(right neighbor first,简称 RNF)算法硬件实现简单,但是因为该算法不关心实际网络的拥塞情况,针对特定类型的流量无法提供较高的吞吐率.

在采用最短路径路由的过程中,可能出现左邻居和右邻居不重合的情况,如果从中选择队列拥塞情况较轻的节点,可以缓解网络的拥塞情况,有效利用直连网络路径多样性的特点,该算法记为负载均衡的最短路径路由算法(load-balanced minimal routing,简称 LBMR).

4 性能分析

本节将对 LNF,RNF 和 LBMR 算法进行比较,选择 HT_3 作为直连网络的拓扑结构,流量类型选择均匀随机(uniform random,简称 UR)类型流量和龙卷风(tornado)类型流量,重点分析分组延时和吞吐量这两个性能指标.

4.1 实验设置

为了评估系统性能,本文设计了一套仿真系统.该系统采用基于周期的方案,即每个时间片(slot)到达系统中每个节点的分组数目为 $t(t\leq 1)$.节点输入侧的缓存按照 VOQ(virtual output queue)进行组织(如图 11 所示),分组到达节点(来自系统外部的注入或邻居节点)时,节点将根据分组的目的地址选择下一跳节点,然后将分组存放在相应的 VOQ 中.每个节点都包含一个 7×7(分别对应节点自身和 6 个邻居)的小型 Crossbar,分组调度时采用 WFA(wave-front arbiter)[17]算法.由于仿真系统刚启动时各个节点的队列均为空状态,这将影响测量的准确性.所以,所有的测量均在网络状态稳定时进行.仿真系统注入的流量类型见表 2.为了更准确地模拟实际系统,假设各个节点间不存在反压,单条链路的每个时间片能够传输一个分组,单个 VOQ 的长度(用 VOQ-Len 表示)限定为 10.

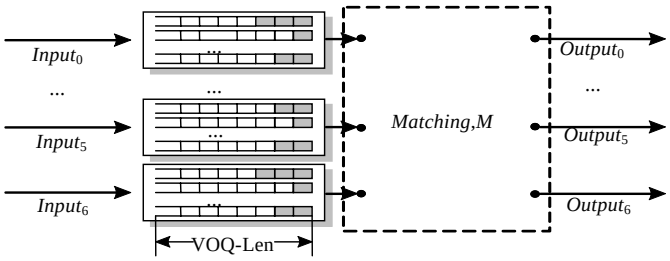


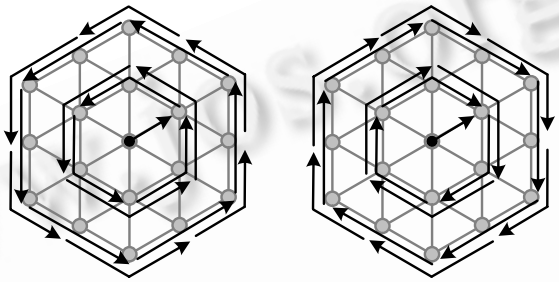
Fig.11 Model of simulation system

图 11 仿真系统模型

Table 2 Traffic patterns for evaluation of routing algorithms

表 2 用于评估路由算法的流量类型

Traffic pattern	Comments
UR	Uniform random traffic pattern, packets are injected from each node to a randomly selected destination node
Tornado	Tornado traffic pattern, as shown in Fig.12



(a) Counter-Clockwise tornado (b) Clockwise tornado

(a) 逆时针龙卷风类型流量 (b) 顺时针龙卷风类型流量

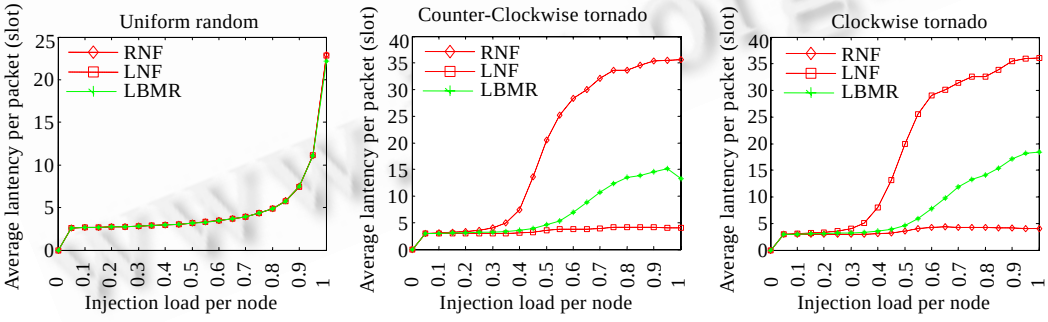
Fig.12 Tornado traffic patterns

图 12 龙卷风类型流量

4.2 实验结果

4.2.1 RNF,LNF 和 LBMR 算法在各类流量下的延时和吞吐量

图 13 给出了 RNF,LNF 和 LBMR 算法针对均匀随机类型流量和逆时针(顺时针)龙卷风类型流量的延时曲线.



(a) Uniform random (b) Counter-Clockwise tornado (c) Clockwise tornado

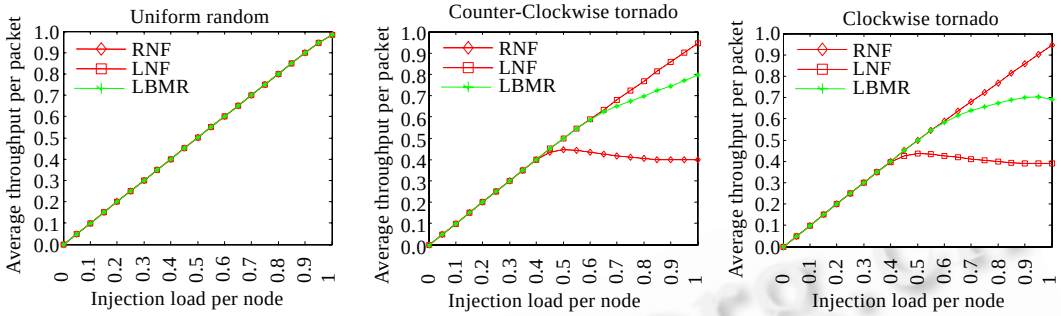
(a) 均匀随机类型流量 (b) 逆时针龙卷风类型流量 (c) 顺时针龙卷风类型流量

Fig.13 Latency curves for each algorithm on three types of traffic

图 13 各类算法针对 3 类流量的延时曲线

由于均匀随机类型流量本身具有很好的负载均衡特性,所以,3 种算法的延时情况类似.RNF 算法在处理逆时针龙卷风类型流量时,会导致流量过于集中在某些链路上,从而造成较大的分组延时(特别是当注入负载较大时).类似地,LNF 算法在处理顺时针龙卷风类型流量时,同样会引起较大的分组延时.而 LBMR 算法可以有效实现负载均衡,在处理逆时针(顺时针)龙卷风类型流量时,仍然可以表现出较好的特性.

图 14 给出了 3 种算法在各类流量下的吞吐量曲线,LBMR 算法都表现出较高的吞吐量.而 RNF(LNF)算法在处理逆时针(顺时针)龙卷风类型流量时,饱和吞吐量只能维持在 0.4 左右.



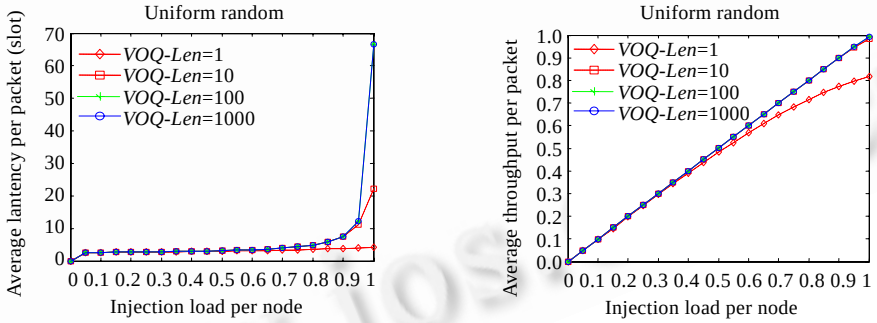
(a) Uniform random (b) Counter-Clockwise tornado (c) Clockwise tornado
(a) 均匀随机类型流量 (b) 逆时针龙卷风类型流量 (c) 顺时针龙卷风类型流量

Fig.14 Throughput curves for each algorithm on three types of traffic

图 14 各类算法针对 3 类流量的吞吐量曲线

4.2.2 缓存对延时和吞吐量的影响

在设计路由器时,缓存起到了重要的作用,缓存的有效设置可以缓解由于流量突发造成的分组丢弃.但是,并不是缓存越大越好,图 15 给出了 LBMR 算法在不同队列长度下表现出的性能(注入均匀随机类型流量).不难看出,增加队列长度会增加分组延时,但是同时能够提高吞吐量.当 VOQ-Len 分别取 10,100 和 1 000 时,吞吐量的提升并不明显.所以,设置合适大小的缓存,既有利于降低路由器的实现成本,又能够保持较高的吞吐量和较低的分组延时.



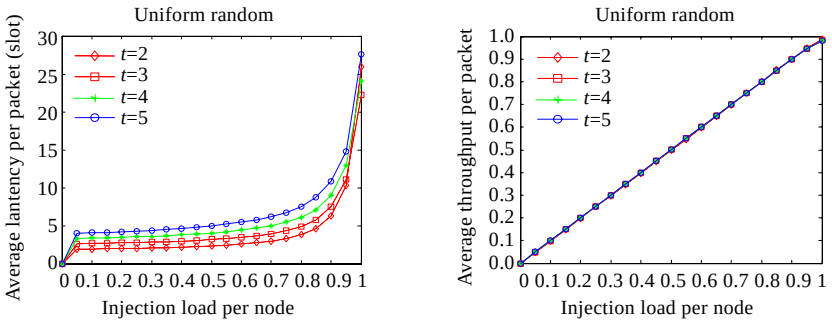
(a) Latency (b) Throughout
(a) 延时 (b) 吞吐量

Fig.15 Comparisons of different VOQ-Len

图 15 不同 VOQ-Len 的比较

4.2.3 网络规模对延时和吞吐率的影响

利用直连式交换网络实现可扩展路由器,突出的优点在于路由器本身具有良好的扩展性.图 16 给出了 LBMR 算法在不同规模的 H-Torus 结构下所表现出来的性能.随着圈数的增加,分组平均延时由于平均跳数的增加而略有增加,而吞吐量并没有因为拓扑规模的增大而降低.



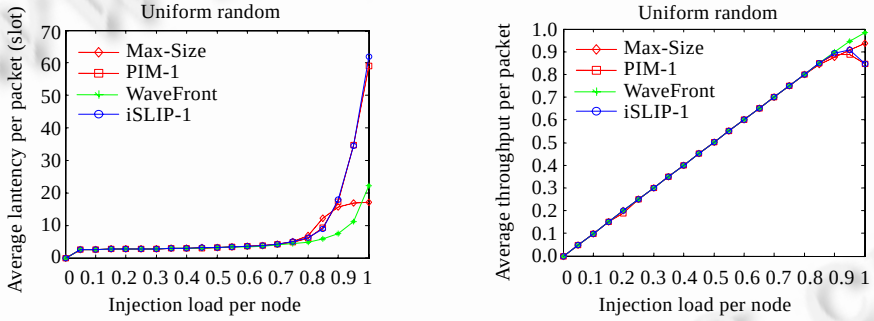
(a) Latency (a) 延时 (b) Throughout (b) 吞吐量

Fig.16 Comparison of different topology scales

图 16 不同拓扑规模的比较

4.2.4 匹配算法对延时和吞吐率的影响

在前面的讨论中,单个节点的调度算法使用了 Wave-Front 算法.图 17 给出了 4 类常用调度算法的性能比较:Max Size Matching[18]算法、PIM[19](parallel iterative matching)算法、Wave-Front 算法和 iSLIP(iterative round robin matching with slip)[18]算法.其中,PIM 算法和 iSLIP 算法都采用了单次迭代的策略.在注入均匀随机类型流量时,WFA(wave front arbiter)算法表现出较低的延时和较高的吞吐量.因此,单节点调度算法的巧妙设计有利于提升整个系统的性能.采用 H-Torus 结构时,在网络扩展过程中,单个节点的交换网络始终保持不变(端口数目恒为 7),有利于实现高效的调度算法,这与传统集中式交换网络有很大区别.



(a) Latency (a) 延时 (b) Throughout (b) 吞吐量

Fig.17 Comparisons of different scheduling algorithms

图 17 不同调度算法的比较

5 结论和未来工作

将直连网络引入路由器,极大地提升了路由器的扩展能力.本文提出了蜂巢结构的变体,详细分析了相应的拓扑属性.其中,H-Torus 结构具有网络直径小、等分带宽高的优良特性,超过了目前普遍采用的 2D Torus 结构.

本文以 H-Torus 结构为基础,提出了一套节点标识的方案.该方案为 RNF,LNF 和 LBMR 这 3 种最短路径路由算法提供了支持.在 3 种最短路径路由算法中,LBMR 算法表现出较好的性能,能够有效缓解不良流量对路由算法的冲击.本文还针对实际路由器中缓存设置以及单节点的调度算法进行了讨论,并给出了它们对路由算法的性能影响.

拓扑结构是直连式交换网络的研究基础,在今后的研究中,研究重点将会集中在路由算法和流控方案的设

计上.针对 H-Torus 结构,还可以进行组播和 QoS 交换的研究.另外,还可以针对拓扑容错性和不规则拓扑等方面展开研究.

References:

[1] Odlyzko AM. Internet traffic growth: Sources and implications. In: Dingel BB, ed. Proc. of the SPIE Optical Transmission Systems and Equipment for WDM Networking II. Orlando, 2003. 1−15.

[2] Keslassy I, Chuang ST, Yu K, Miller D, Horowitz M, Solgaard O, McKeown N. Scaling Internet routers using optics. In: Feldmann A, ed. Proc. of the Special Interest Group on Data Communication (SIGCOMM). Karlsruhe: ACM Press, 2003. 189−200.

[3] Chiussi FM, Francini A. Scalable electronic packet switches. IEEE Journal on Selected Areas in Communications, 2003,21(4): 486−500.

[4] Marcus M. The theory of connecting networks and their complexity: A review. Proc. of the IEEE, 1977,65(9):1263−1271.

[5] Narasimha MJ. The batcher-banyan self-routing network: Universality and simplification. IEEE Trans. on Communications, 1988, 36(10):1175−1178.

[6] Sapountzis G, Katevenis M. Benes switching fabrics with *O*(*N*)-complexity internal backpressure. IEEE Communications Magazine, 2005,43(1):88−94.

[7] Jajszczyk A. Nonblocking, repackable, and rearrangeable clos networks: Fifty years of the theory evolution. IEEE Communications Magazine, 2003,41(10):28−33.

[8] McKeown N, Mekkittikul A, Anantharam V, Walrand J. Achieving 100% throughput in an input-queued switch. IEEE Trans. on Communications, 1999,47(8):1260−1267.

[9] Chang CS, Lee DS, Jou YS. Load balanced Birkhoff-von Neumann switches, part I: One-stage buffering. Computer Communications, 2002,25(6):611−622.

[10] Dally WJ. Performance analysis of k-ary *n*-cube interconnection networks. IEEE Trans. on Computers, 1990,39(6):775−785.

[11] Dally WJ. Scalable switching fabrics for Internet routers. http://www.avici.com/technology/whitepapers/TSRfabric-WhitePaper.pdf

[12] Dally WJ, Towles B. Principles and Practices of Interconnection Networks. San Francisco: Morgan Kaufmann Publishers, 2004. 45−55.

[13] Valiant LG, Brebner GJ. Universal schemes for parallel communication. In: Proc. of the ACM Symp. on the Theory of Computing. New York: ACM Press, 1981. 263−277.

[14] Yue ZH, Zhao YJ, Wu JP, Zhang XP. Designing scalable routers with a new switching architecture. In: Proc. of the Autonomic and Autonomous Systems and Int'l Conf. on Networking and Services. Piscataway, 2005. 1−1.

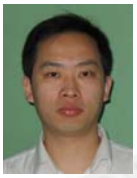
[15] Chen, MS, Shin, KG, Kandlur DD. Addressing, routing, and broadcasting in hexagonal mesh multiprocessors. IEEE Trans. on Computers, 1990,39(1):10−18.

[16] Sullivan H, Bashkow TR. A large scale, homogeneous, fully distributed parallel machine (I). In: Proc. of the 4th Annual Symp. on Computer Architecture. New York: IEEE, 1977. 105−117.

[17] Tamir Y, Chi HC. Symmetric crossbar arbiters for VLSI communication switches. IEEE Trans. on Parallel and Distributed Systems, 1993,4(1):13−27.

[18] McKeown N. The iSLIP scheduling algorithm for input-queued switches. IEEE Trans. on Networking, 1999,7(2):188−201.

[19] Anderson T, Owicki S, Saxe J, Thacker C. High speed switch scheduling for local area networks. ACM Trans. on Computer Systems, 1993,11(4):319−352.



乐祖晖(1978－),男,湖北孝感人,博士,主要研究领域为可扩展路由器体系结构,交换网络,调度算法.

赵有健(1969－),男,副教授,CCF 高级会员,主要研究领域为高速路由器硬件体系结构,高速大容量交换结构,IP 调度算法,混洗交换高速背板.

吴建平(1953－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为计算机网络体系结构,计算机网络协议测试,形式化技术.