保证 100%吞吐率的两级组播交换结构

周　婷*①　　赵有健①　　王瑞生②
①(清华大学计算机科学与技术系　北京　100084)
②(南加州大学电子工程系　洛杉矶　CA90089)

摘　要：在路由器或交换机的交换结构中实现组播是提高组播应用速度的重要途径之一。传统的交叉开关结构(crossbar)组播调度方案有两种缺陷，一种是性能较低，另一种是实现的复杂度太高，无法满足高速交换的需要。该文提出了一个新的基于交叉开关的两级组播交换结构(TSMS)，第 1 级是组播到单播的交换结构，第 2 级是联合输入和输出排队(CIOQ)交换，并为该结构设计了合适的最大扇出排队(FCN)优先-均匀分配中间缓存调度算法(LFCNF-UMBA)。理论分析和仿真实验都显示在该结构中，加速比低于 $2-2/(N+1)$ 倍时吞吐率不可能实现 100%；而采用 LFCNF-UMBA 调度算法，2 倍加速比就可保证在任意允许(admissible)组播的吞吐率达到 100%。

关键词：交换结构；组播；调度；交叉开关；吞吐率
中图分类号：TP393.2　　**文献标识码**：A　　**文章编号**：1009-5896(2012)01-0082-07
DOI: 10.3724/SP.J.1146.2011.00257

Achieving 100% Throughput in a Two-stage Multicast Switch

Zhou Ting①　　Zhao You-jian①　　Wang Rui-sheng②
①(*Department of Computer Science & Technology*, *Tsinghua University*, *Beijing* 100084, *China*)
②(*Department of Electrical Engineering*, *University of Southern California*, *Los Angeles* CA90089, *USA*)

Abstract: The Internet growth coupled with the variety of multicast services is creating an increasing need for multicast traffic support by routers and packet switches. However, the traditional crossbar-based multicast scheduling schemes are unable to meet the needs of high-speed switching for the low performance and high implementation complexity. In this paper, a Two-Stage Multicast Switch (TSMS) is proposed, which is a serial combination of a Multicast To Unicast (MTU) switch to copy input cells from various sources simultaneously and a Combined Input and Output Queueing (CIOQ) switch to deliver copies of multicast cells to their final destinations. Based on MTU switch, a novel Largest Fanout Cardinal Number First-Uniform Middle Buffer Allocation (LFCNF-UMBA) scheduling algorithm is designed to determine how to copy multicast cells into unicast cells. By coordinately using Maximal Matching scheduling algorithm in CIOQ switch, it is proved that speedup of $2-2/(N+1)$ is necessary and 2 is sufficient for a $M\times N$ TSMS to achieve 100% throughput under any admissible multicast traffic pattern, which is also verified by the simulation results.

Key words: Switch; Multicast; Scheduling; Crossbar; Throughput

1　引言

近年来，新型多媒体业务成为互联网发展的重要方向，如大容量数据分发、视频会议、远程教育等，带来了组播技术的广泛应用。支持组播的核心路由器交换设备为成为研究者关注的一个重要问题。

基于交叉开关(crossbar)的交换结构由于其低损耗、可扩展性，特别是它的天然组播能力，一直被认为是最合适的交换互连架构[1]。大量的研究工作基于输入排队(Input Queued, IQ)的交叉开关交换结构展开[2,3]。不同于单播流量，组播信元的目的地址是一个输出端口集合。对于一个 $M\times N$ (M 和 N 分别代表交换结构的输入端口和输出端口数目)交换结构，每个组播信元的目的端口集合的大小可能在 1 到 N 之间，通常用扇出集合(fanout)来描述。IQ 交换机的问题在于输入端口上为解决队头阻塞(Head-Of-Line, HOL)问题而设置的 FIFO 排队数目受到限制[4]，同时调度上又继承了源于交叉开关结构集中控制导致的复杂性。

在交叉开关的每个交叉节点上添加有限数量的缓存，可以用带缓存的交叉开关交换结构

2011-03-21 收到，2011-10-09 改回
国家自然科学基金(60903184, 60173167, 60773150)和国家 863 计划项目(2008AA01A324, 2008AA01A323)资助课题
*通信作者：周婷　zhout06@mails.tsinghua.edu.cn

(Combined Input and Crosspoint Queued, CICQ)[5]解决这种集中的调度复杂性。在 CICQ 中，交叉开关上复杂的集中式二分图匹配调度转化成为分布式多队列调度问题。CICQ 吸引了大量的对组播流量支持的研究[6−9]。在组播流量下，CICQ 的性能优于 IQ，但是随着交换规模的变大，由于输入排队的结构问题，CICQ 同样会遭遇性能的下降[8,10]。

另外，在交换结构内部完成信元的复制将会减少可用的带宽，因此通常需要配合一定的加速比才能获得较好的吞吐率性能表现。交叉开关中基于最大匹配算法构建的组播调度法，至少需要 $O(\lg N / \lg\lg N)$ 的加速比才能保证可允许流量下 100%的吞吐率[11]，N 是交换结构的端口数。对于 CICQ 的组播调度，如果交叉节点的缓存数目固定，那么 $O(\lg\lg N / \lg\lg\lg N)$倍加速比[8]是必须的；当加速比不变的时候，交叉节点缓存则将随着端口数目的增加以 $\lg N$ 数量级增加。最近的研究显示，基于交叉开关的交换结构在组播流量下吞吐率不可能随着端口数的增加达到 100%[10]。由此，促使我们寻找新的交换结构和调度算法以更加有效的支持组播流量。

受到经典的负载均衡两级交换结构[12]启发，本文提出了一种全新的两级组播交换结构(TSMS)。该结构继承了两级负载均衡结构实现复杂度低的优点。TSMS两级结构的第1级是个组播到单播的交换结构(Multicast To Unicast, MTU)，完成信元的复制工作并且将复制后的信元均匀地放置到中间缓存排队；第2级则是一块联合输入和输出排队(CIOQ)交换，完成信元的转发工作，并且设计了合适的调度算法LFCNF-UMBA。理论分析发现只需采用2倍加速比就可以保证在任意允许(admissible)组播输入流量下达到吞吐率100%，同时证明了TSMS实现上述100%吞吐的必要条件是加速比不能低于 $2-2/(N+1)$。因此作者认为TSMS配合2倍加速比是一个较好的选择，并通过仿真实验证明了2倍加速比下TSMS在吞吐率性能和延时性能上都具有良好的表现，很好地印证了理论分析的结论。

2　交换体系结构与流量模型

2.1 单级组播结构和流量模型

组播有两种不同的调度策略：扇出分裂或者扇出不分裂。扇出不分裂是指一个组播信元在一个时间片内同时发送到所有的目的端口。而组播信元在扇出分裂策略下可以在多个时间片内发送。这是一种尽职工作(work-conserving)策略，即调度服务器在输出队列非空时可以持续处于工作状态。显然扇出策略能显著地提高组播交换系统的吞吐率性能，但是由于一个组播信元可能要在多个时间片内才能完成传送，这也增加了输入端的负载。为了减少这种性能损失，本文暂时只考虑扇出不分裂情况。

组播的可允许流量有两个特点。(1)与空间特性相关。在一个时槽内，来自于一个输入端口的信元可能直接占满了所有的输出端口(假定在广播情况下，信元要同时发送到所有的输出端口)。而在单播情况下，只有所有的输入端口同时达到了它们的最大的负荷才可能占满整个交换机带宽。这种空间的负载不均衡可能导致输入负载集中在某几个输入端口。(2)与时间相关。在一些时间片内，到达的组播信元拷贝会超出交换机的输出容量(对一个 $M\times N$ 的交换机，输出容量是 N)。而在单播情况下，任何时间片内，输入流量都不会超过输出容量。这种时间上的负载不均衡致使输出负载集中在某些时间段上。

考虑一个 $N\times N$ 的交换结构，输入端口和输出端口的数目均为 N，并以相同的线速处理数据包。可允许流量模式是指输入和输出都在负载范围以内。令输入端口 $i(1\le i\le N)$ 的信元到达是一个离散时间随机过程 $A_i(t)$。在一个时隙 t 内，最多有一个信元到达一个输入端口。定义 λ_{ij} 为输入端口 i 到输出端口 j 的到达速率，到达过程的集合 $A(t)=\{A_i(t), 1\le i\le N\}$。

对于单播来说，若输入和输出都在负载范围以内，即

$$\lambda_i=\sum_{j=1}^{N}\lambda_{i,j}\le 1,\ \ \lambda_j=\sum_{i=1}^{N}\lambda_{i,j}\le 1,\ \ \lambda_{i,j}\ge 0 \quad (1)$$

成立，则 $A(t)$ 被认为是容许的，否则就是非容许的。

在组播情况下，定义 $A_{iF}(t)$ 为输入 i 到输出 j 的集合 F 的到达过程，其到达速率为 λ_{iF}，到达过程的集合 $A(t)=\{A_i(t),1\le i\le N\}$。组播流量是可允许的，当且仅当，

$$\lambda_i=\sum_{F\subseteq U}\lambda_{i,F}\le 1,\ \lambda_j=\sum_{i=1}\sum_{j\in F,F\subseteq U}\lambda_{i,j}\le 1,\ \lambda_{i,F}\ge 0 \quad (2)$$

定义 L 为中间缓存(Middle Buffer)的平均负载，等于中间缓存在一个时间片内接收到的信元拷贝的平均数目。

从式(2)可以得出，

$$L=\sum_{j=1}^{N}\lambda_j=\sum_{i=1}^{N}\sum_{F\subseteq U}|F|\lambda_{i,j}\le N \quad (3)$$

式(3)表明，在可允许的组播流量模式下，中间缓存的平均负载不能超过 N。

2.2 两级组播交换结构

本文提出了一个两级的组播结构模型(Two-Stage Multicast Switch, TSMS)，如图 1 所示。该模型由一个组播到单播转换模块(Multicast to

Unicast, MTU) 和一个联合输入和输出排队(Combined Input and Output Queued, CIOQ)交换结构组成。在 MTU 模块中输入缓存(input buffer)中的组播信元拷贝成中间缓存(middle buffer)中的多个单播信元。由于交叉开关的天然组播能力，这个复制工作可以在一个时间片内完成。而在第 2 级 CIOQ 交换结构，中间缓存中的单播信元根据它们的目的地址发送到各自相应的输出缓存(output buffer)中。下面我们从排队和调度算法这两个方面来描述 TSMS。

(1)排队：由于包含两个交叉开关结构，TSMS 因此分为三阶段进行缓存：输入缓存(input buffer)，中间缓存(middle buffer)以及输出缓存(output buffer)。输入缓存在第 1 块交叉开关之前，用来临时地存储到达的信元。在这个阶段，我们采用了扇出数排队(Fanout Cardinal Number Queueing, FCNQ)策略，将输入缓存根据一个组播信元的 FCN 分成 N 条排队。例如，如图 1 所示，假定一个信元的扇出集合是 $\{2,4\}$，那么将它插入第 2 条 FCNQ，因为这个信元的 FCN 是 2。中间缓存位于两块交叉开关结构之间，接受来自第 1 块交叉开关的信元，并将信元转发到第 2 块交叉开关上。中间缓存同样根据输出端口数分成 N 条排队，每条排队按照虚拟输出排队(VOQ) 组织，将单播信元按照它们的目的输出端口缓存。输出缓存位于第 2 块交叉开关之后，存储那些不能立即转发到外部链路的信元。输出缓存的设置有两方面的原因。第一，作为重新排序缓存，用来解决由于多路径带来的乱序问题；第二，处理由于加速比带来的时间超载问题。

如图 1 所示， MTU 交换部分，包含输入缓存和第 1 块交叉开关结构，用来将组播流量模式转换为单播流量模式。后面的 CIOQ 交换部分，包含中间缓存、第 2 块交叉开关结构以及输出缓存，则作为一个常规的点到点(point-to-point)交换机。

(2)调度：TSMS 在两个阶段分别使用了不同的调度策略：LFCNF-UMBA 调度算法和最大匹配调度算法(MMS)。在 MTU 上应用的 LFCNF-UMBA 调度算法分两个阶段，第 1 阶段是最大 FCN 优先(LFCNF)策略，用以确定输入缓存中先复制哪个信元；第 2 阶段是均匀分配中间缓存(Uniform Middle Buffer Allocation, UMBA)策略，确定将信元拷贝到哪一个中间缓存单元中去。

MTU 上的调度首先要解决输入冲突的问题。在一个时间片内，一个输入端口可能有不止一个信元被拷贝复制。其次，需要解决的是中间缓存的超载，即在一个时间片内，复制过来的信元总数超过了中间缓存的容量(也是 MTU 交换的输出能力)。

LFCNF 策略的核心思想是给予大的 FCN 更高的调度权限。一个时间片中可能有不止一轮调度，这取决于加速比。一次调度中有 N 轮迭代(iteration)。在每轮调度中，一个入口只能发一个包，一个出口只能收一个包。如表 1 所示，在第 i 轮迭代中，调度算法检查所有的 FCN 为 $(N-i+1)$ 的 FCNQs，并且尽可能挑选出那些没有输入冲突的信元，它们的 FCN 都是 $(N-i+1)$。

UMBA 策略如表 2 所描述。当一个 FCN 为 k 的信元请求中间的时候， UMBA 策略分配 k 个连续的中间缓存单元给这个信元。UMBA 的目标是将组播信元的拷贝均匀地分散到中间缓存。LFCNF 和 UMBA 一起构成第 1 阶段完整的调度算法。

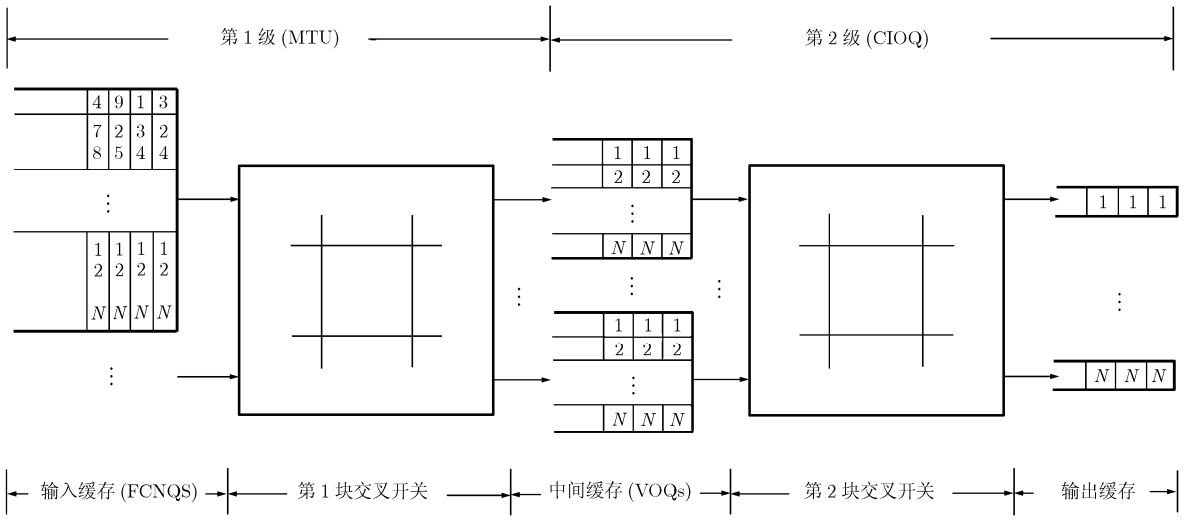


图 1 两级组播交换结构

表 1 LFCNF 策略

Largest Fanout Cardinal Number (LFCNF) First policy
Let Iidle be the set of idle inputs, and |Oidle| be the number of idle outputs.
(1) **for** $i = 1$ to N **do**
(2)　**for** $in = 1$ to N **do**
(3)　　**if** $in \in$ Iidle and the $(N-i+1)$ th queue at input port in is not empty and |Oidle| >= $(N-i+1)$ **then**
(4)　　The cell at the head of $(N-i+1)$th queue at input port in is selected to be scheduled
(5)　　Iidle←Iidle\\{in}
(6)　　|Oidle|← |Oidle|− k
(7)　　**end if**
(8)　**end for**
(9) **end for**

表2 UMBA策略

Uniform Middle Buffer Allocation(UMBA) policy
Let Allocation Iterator (AI) to indicate the last Middle Buffer to which
the latest cell is copied. When a cell with FCN of k requests Middle Buffers,
(1)Copy the cell to the following Middle Buffers: AI%N +1,(AI +1)%N +1, ⋯ ,(AI+k−1)%N+1
(2)AI ← (AI+k−1)%N+1

图2中描述了一个 2×4 的MTU交换机中的LFCNF-UMBA 调度过程。在调度之前，输入缓存的状态如图2(a)所示，中间缓存为空。观察两个连续调度周期，并且在此期间没有信元到达。假定加速比为2，故每个时间片内有两轮调度。每个调度又包含4轮迭代。

第 1 阶段调度：根据LFCNF，由于两个输入端口的缓存中都不存在FCN为 4 的FCNQ，故第1轮迭代轮空。在第2次迭代的时候输入端口1的第3个FCNQ被选中，再由UMBA，信元{1,3,4}复制到中间缓存的3个连续单元。到第3，第4次迭代时由于端口冲突，没有信元可以调度，轮空。

第2阶段调度：第1，第2轮迭代轮空。在第3轮迭代的时候，根据LFCNF，端口 1 和端口2的第2个FCNQ同时被选中。UMBA将中间缓存的第4和第1单元分配给信元{1,2}；将第2第3单元分配给信元{1,3}。假定中间缓存的第1块和第 N 块(这个例子中是第4块)单元相连接。

通过两轮的调度周期，中间缓存的状态如图2所示(为了简单表示，中间缓存没有以虚拟输出排队(VOQ)的形式来表示)。在CIOQ交换中，应用任何极大匹配调度算法，例如iSLIP, PIM都可以保证一个很高的吞吐率。整个组播调度过程如下：当一个输入端口有一个组播信元到达，首先进入到输入缓存中相应的FCNQ中排队。接着根据LFCNF-UMBA调度算法，通过第1级交叉开关复制到中间缓存。最后，CIOQ采用极大匹配调度算法将信元调度出去。

3　稳定性分析

定义 1（稳定性）　如果在一个 $M \times N(M > 1)$ 的MTU构中，排队里的任何信元都可以在有限时间内被调度，那么这个交换结构是稳定的。

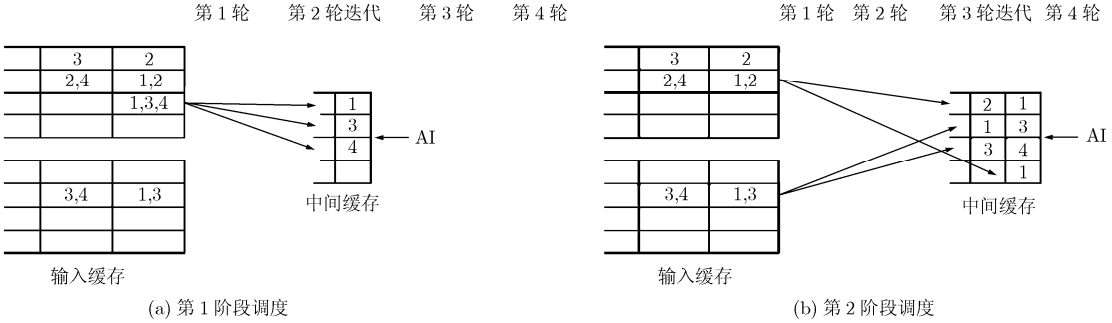
系统的稳定性，意味着交换结构获得100%的吞吐率。简言之，当系统的负载在可以承受的范围之内，每个端口上的队列长度有界，那么这个调度是稳定的。

在这一部分，我们首先分析MTU稳定性，然后证明当加速比为2的时候可以保证TSMS在任何可允许流量下是稳定的。

定理 1（必要性）　在一个 $M \times N(M > 1)$ 的MTU 分裂调度最少需要加速比为 $2 - 2/(N+1)$，可以保证在任何可允许组播流量模式稳定。

证明　首先设定一种特殊的两端口可允许组播流量模式，证明在这种特定的流量模式下，没有调度算法能在加速比少于 $2 - 2/(N+1)$ 的情况下保证MTU 结构的稳定性。

两端口(Two Port, TP)流量模式，只有两个活



(a) 第 1 阶段调度　　(b) 第 2 阶段调度

图 2 LFCNF-UMBA 调度过程

动的输入端口：

(1)第 1 个输入端口在每 $(N+1)$ 个时间片内接收 N 个单播信元，每个信元的 FCN 都是 1。

(2)第 2 个输入端口在每 $(N+1)$ 个时间片内接收 N 个组播信元，每个信元的 FCN 都是 N 。

TP 流量是可允许的组播流量模式。因为每两个输入端口的负载是 $N/(N+1)$，该负载小于 1；并且每个输出负载是 $(N+N\times N)/(N+1)=N$，可以看出，无论是输入还是输出负载都没有超载。

由于输入冲突的存在，在一个调度周期内最多只能调度一个单播或者一个广播信元。而且因为中间缓存超载，一个组播信元和一个单播信元也不能在一个调度周期内同时调度。由于，一个调度周期内只有一个信元可以被调度，所以，在每 $(N+1)$ 个时间片内，MTU 需要 $2N$ 个调度周期来调度这 $2N$ 个信元，也就是说要求的加速比为 $2N/(N+1)=2-2/(N+1)$。 证毕

定理 2 (充分性) 在一个加速比为 2 的 $M\times N$ $(M>1)$ 的 MTU 交换中，通过 LFCNF 策略可以保证在任何可允许组播流量模式稳定。

证明 假定一个信元 a，它的扇出集合为 $F_a(|F_a|=k)$ 在 t_{start} 时刻到达输入端口 i。令 I_i 为输入端口 i 上的信元集合，M 为输入端口 i 上 FCN 不小于 k 的信元集合。即 $M=\{c=|F_c|\geq k, c\in I_i\}$。

因为有 2 倍的加速比，所以在一个时槽内有两个调度周期。令 $L_t(L_s)$ 为一个时槽内中间缓存接收到的平均拷贝信元数。

假定信元 a 不能在有限的时间内被调度。相应的，存在 $M\neq\varnothing$，因为有 $a\in M$ 。

(1)如果 $k\geq\lceil(N+1)/2\rceil$ 在 LFCNF 调度策略下，具有最大 FCN 的信元会首先被调度。因为信元 a 没有被调度，那么一定有一个 FCN 不小于 k 的信元在这两个调度周期中被调度了 $(L_s\geq k)$。因此有 $L_t=2L_s\geq 2k\geq N+1>N$。

(2)如果 $k<\lceil(N+1)/2\rceil$，在每个时槽，又有以下 3 种情况：

(a)$|M|$ 增加 1 (time-slot A) 在这种情况下，一个信元 c $(|F_c|\geq k)$ 到达输入端口 i，M 中没有信元被调度。在每一个调度周期里，复制的信元都不少于 $N-k+1$，否则，根据 LFCNF 调度策略，输入端口 i 上 FCN 为 i 的信元将会同时被调度。因而，有 $L_t=2L_s\geq 2(N-k+1)$。

(b)$|M|$ 减少 1 或者 2 (time-slot B) 因为信元 a 没有被调度，根据 LFCNF，则必有一个 FCN 不小于 k 的信元在每个调度周期里被调度 $(L_s\geq k)$，由此，有 $L_t=2L_s\geq 2k$。

(c)$|M|$ 保持不变 (time-slot C):

(i)一个信元 c $(|F_c|\geq k)$ 到达输入端口 i，同时 M 中有一个信元被调度。

在一个调度周期里，由于 M 中有一个信元被调度，所以 $L_{s1}\geq k$ 。而在另一个调度周期里，输入端口 i 上没有 FCN 为 k 的信元被调度，所以 $L_{s2}\geq N-k+1$。由此，有 $L_t=L_{s1}+L_{s2}\geq N+1$。

(ii)没有信元 c $(|F_c|\geq k)$ 到达输入端口 i，同时 M 中没有信元被调度。

输入端口 i 上没有 FCN 为 k 的信元被调度，所以 $L_s\geq N-k+1$。由于 $k<\lceil(N+1)/2\rceil$，从而得到 $L_t=2L_s\geq 2(N-k+1)>N+1$。

令 $(\text{FCN}=N)$ 为 M 在 t_{start} 时刻的状态，假定 T 时槽从 t_{start} 开始，包括 t_A 时槽，t_B 时槽，t_C 时槽。因为 $M\neq\varnothing$，故 $t_A+|M_{t_{\text{start}}}|>t_B$

从(a), (b), (c)的讨论可知，中间缓存的平均负载是

$$L_t\geq\frac{t_A\times 2(n-k+1)+t_B\times 2k+t_C\times(N+1)}{t_A+t_B+t_C}$$
$$>(N+1)-\frac{|M_{t_{\text{start}}}|(N+1-2k)}{T}$$

因此可知，当 $T>|M_{t_{\text{start}}}|(N+1-2k)$ 时，$L_t>N$。

从(1), (2)可知，如果一个信元不能在有限的时间内调度出去，则说明中间缓存的负载超过了 N，这与前面的假设组播流量是可允许的相矛盾(由式(3)可知，可允许的组播流量模式下中间缓存的负载不会超过 N)。 证毕

引理 1 如果输入流量模式是可允许的组播流量模式的话，运用 UMBA 策略的 MTU 交换的输出流量模式是可允许的单播流量。

证明 令 $\lambda'_{i,j}$ 为从中间缓存 i 到输出缓存 j 的单播流量，$\lambda_{i,F}$ 是输入端口 i 到 F 中的输出的组播流量。假定输入流量 $\lambda_{i,F}$ 是可允许的，并且 $\lambda'_{i,j}$ 是 MTU 交换的输出流量。根据式(2)，有

$$\lambda'_j=\sum_{i=1}^{N}\lambda'_{i,j}=\left(\sum_{i=1}^{N}\sum_{j\in F,F\subseteq U}\lambda_{i,F}\right)\leq 1$$

然后通过 UMBA 对流量的均匀化，可以得到

$$\lambda'_j=\sum_{j=1}^{N}\lambda'_{i,j}=\left(\sum_{i=1}^{N}\sum_{F\subseteq U}|F|\lambda_{i,F}\right)\Big/N\leq 1$$

因此，$\lambda'_{i,j}$ 是可允许的单播流量。 证毕

定理 3 加速比为 2 的 TSMS 在任何可允许的组播流量模式下都是稳定的。

证明 证明分两步。

(1)在任何可允许组播流量模式下，MTU 在 2

倍加速比下都可以保证稳定(根据定理 2，并且输出可允许的单播流量(根据引理 1)。

(2)2 倍加速比的 CIOQ 交换结构应用最大匹配调度算法可以在任何可允许的单播流量模式下保持稳定[8]。

因此，整个 TSMS 当加速比为 2 时可以在任何可允许的组播流量模式下保持稳定。　　证毕

4　仿真结果

评估组播调度算法的一个重要指标就是吞吐性能和延时性能。通常情况下，吞吐率高的算法在相同的输入负载下平均延时也较低。本文重点评估算法在饱和吞吐的情况下延时的性能。饱和吞吐是指在保证加于输出端口的目标负载为100%的情况下，实际测定的输出端口吞吐率。所用的流量到达模型为均匀流量模型和特定的两端口流量模型。

均匀流量(uniform traffic)模型，信元以相同的概率 ρ 到达每一个输入端口，平均的FCN为 $N/2$，N 为端口数，系统有效的负载为 $\rho N/2$。

两端口流量(two port traffic)模型，在一个时槽内，单播信元(FCN = 1) 以 $\rho N/(N+1)$ 的概率到达输入端口1，而一个广播信元 $\rho N/(N+1)$ 以 $\rho N/(N+1)$ 的概率到达输入端口2。其他输入端口为空(ρ 为输出负载)。

从图 3 和图 4 显示了TSMS在两种流量条件下的延时性能，并和输出排队(OQ)做了对比。2 倍加速比的时候，不论是均匀流量还是两端口流量，TSMS都可以获得100%的吞吐率性能。而在1 倍加速比情况下，对于均匀流量可以得到90%的吞吐率，而TP流量下仅仅只有54%的吞吐率。与OQ相比，两级的结构存在使得系统有一个大于 1 的初始延时。从图中可以看出，延时的变化十分平稳，几乎没有抖动。TSMS在 2 倍加速比时，TP流量下的延时曲线几乎平行于OQ。

TSMS的优势不仅仅在吞吐率上，它的可扩展性也很好。当交换结构的规模从16×16上升到128×128的时候，均匀流量下系统的性能几乎没有什么变化，而在TP流量下，系统延时性能会更好一些，如图 5 所示。因为在TP流量模式下，输入端只有两端口活动，那么规模越大，活动窗口所占的比例越小，造成HOL拥塞的可能性也就越小。

只要在输入端输入的是可允许流量，那么通过TSMS的两级调度，总可以在有限时间内将信元调度出去，图 6 中显示，第 2 阶段调度分别为极大匹配调度和轮询调度的情况下，系统的延时虽然在后者有所增加，但是仍然能保证100%的吞吐率。

5　结束语

针对传统路由器交换结构在组播功能支持上的不足，本文提出了一种全新的组播交换结构TSMS。它是在经典的两级负载均衡交换结构基础上针对组播功能做出的创新，因此继承了两级负载均衡结构实现复杂度低的优点。TSMS在两级结构的第 1 级完成组播分裂，第 2 级实现分裂后的分组交换。交换结构内部完成组播分裂的实现方式造成了TSMS需要配合一定的加速比才能获得较好的吞吐率性能表现。本文证明了TSMS实现100%吞吐的必要条件是加速比不能低于2−2/(N+1)，同时理论和仿真实验都表明在调度算法LFCNF-UMBA下，TSMS只需采用2倍加速比就可以保证在任意允许(admissible)组播输入流量下达到吞吐率100%。当然，作为第 1 个以两级交换为基础的组播交换结构实现方式，TSMS还存在一定的不足，例如第 1 级的组播分裂存在一定的公平性问题，这些都将作为本文的后续研究工作。

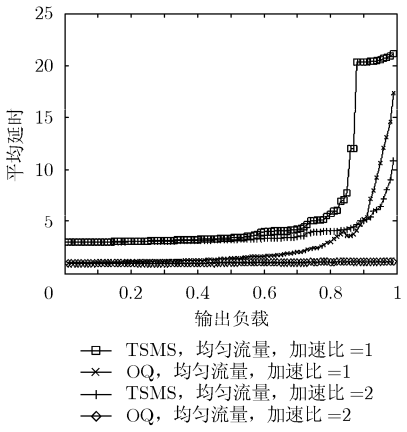


图3 均匀流量条件下TSMS延时性能

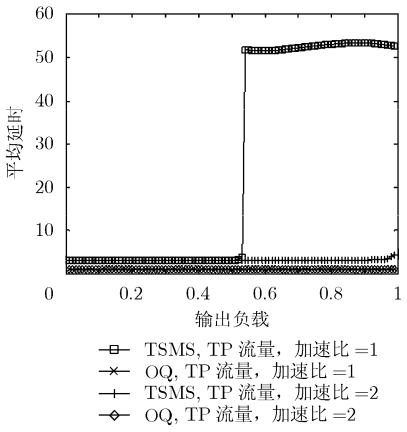


图4 两端口流量条件下TSMS延时性能

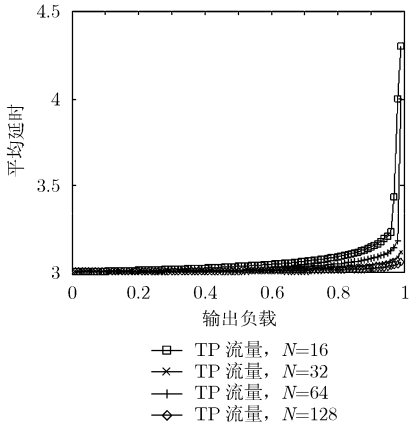


图5 TP流量下不同规模的TSMS延时性能

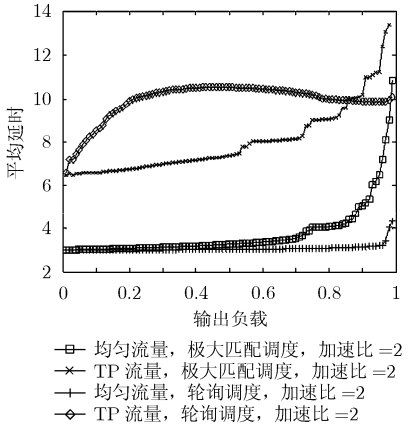


图6 TSMS中第2阶段不同调度方法对延时的影响

参 考 文 献

[1] Mhamdi L, Gaydadjiev G, and Vassiliadis S. Efficient multicast support in high-speed packet switches. *Journal of Networks*, 2007, 2(3): 28–35.

[2] Marsan M A, *et al.*. Multicast traffic in input-queued switches: optimal scheduling and maximum throughput. *IEEE/ACM Transactions on Networking*, 2003, 11(3): 465–477.

[3] Bianco A, *et al.*. Practical algorithms for multicast support in input queued switches. Workshop on High Performance Switching and Routing, Poznan, Poland, 2006: 187–192.

[4] Kanizo Y, Hay D, and Keslassy I. The crosspoint-queued switch. The 28th Conference on Computer Communications, Rio De Janeiro, Brazil, 2009: 729–737.

[5] Nabeshima M, *et al.*. Performance evaluation of a combined input-and crosspoint-queued switch. *IEICE Transactions on Communications*, 2000, 83(3): 737–741.

[6] Chang C S, Hsu Y H, Cheng J, *et al.*. A dynamic frame sizing algorithm for CICQ switches with 100% throughput. The 28th Conference on Computer Communications, Rio De Janeiro, Brazil, 2009: 747–755.

[7] Sun S, He S, Zheng Y, *et al.*. Multicast scheduling in buffered crossbar switches with multiple input queues. Workshop on High Performance Switching and Routing, Hong Kong, 2005: 73–77.

[8] Giaccone P and Leonardi E. Asymptotic performance limits of switches with buffered crossbars supporting multicast traffic. *IEEE Transactions on Information Theory*, 2008, 54(2): 595–607.

[9] Senin I V, Mhamdi L, and Goossens K. Efficient multicast support in buffered crossbars using networks on chip. Proceedings of the 28th IEEE Conference on Global Telecommunications, Honolulu, HI, USA, 2009: 1–7.

[10] Mhamdi L. On the integration of unicast and multicast cell scheduling in buffered crossbar switches. *IEEE Transactions on Parallel and Distributed Systems,* 2009, 20(6): 818–830.

[11] Koksal C E. On the speedup required to achieve 100% throughput for multicast over crossbar switches. 16th International Workshop on Quality of Service, Enschede, Netherlands, 2008: 60–64.

[12] Chang C S, Lee D S, and Jou Y S. Load balanced Birkhoff-von Neumann switches, Part I: one-stage buffering. *Computer Communications*, 2002, 25(6): 611–622.

周 婷： 女，1974 年生，博士生，研究方向为为可扩展路由器体系结构、交换网络、调度算法.

赵有健： 男，1969 年生，博士，教授，博士生导师，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、调度算法、混洗交换高速背板.

王瑞生： 男，1983 年生，博士生，研究方向为可扩展路由器体系结构、交换网络、调度算法.