第7卷第2期　　空军工程大学学报(自然科学版)　　Vol.7 No.2
2006年4月　　JOURNAL OF AIR FORCE ENGINEERING UNIVERSITY(NATURAL SCIENCE EDITION)　　Apr.2006

# 一种支持包保序的并行交换调度算法

仇兴峰[1]，余美荣[1]，董雨果[1,2]
(1. 空军工程大学 电讯工程学院，陕西 西安 710077；
2. 信息工程大学 国家数字交换系统工程技术研究中心，河南 郑州 450002)

**摘　要**：由于并行交换结构的负载平衡特性和并行原理，到达同一目的输出端口的分组包被分散到了各个交换模块，当它们抵达输出端口时，其先后顺序无法得到保障。为解决该难题，文中提出了虚拟输入排队(VIQ)结构和包保序轮询(SKRR)算法，并且从理论上分析了这种新技术的吞吐率和时延性能。

**关键词**：交换；负载平衡；时延；调度
**中图分类号**：TP274　　**文献标识码**：A　　**文章编号**：1009-3516(2006)02-0043-05

移动网络与 Internet 的融合需要超高速的网络节点设备，例如 $T$ 比特甚至 $P$ 比特的交换机。并行交换(PS-Parallel Switching)结构[1-3]是高速交换领域的新兴技术，它通过现有的低速交换机来构造高速大容量的交换系统。包保序对 PS 结构是十分重要的，因为，如果到达的包通过不定长分组的形式进行交换，分组包的乱序将会导致当前版本 TCP 出错[4]；如果到达的包首先通过切片成为很多 Cell，Cell 的失序将为以后的 Cell 重组造成困难[3,5]。

在一个 PS 结构中，根据并行原理(Parallelism)，目的地为同一输出端口的高速包被分散到多个低速的交换模块中进行交换处理。由于各个交换模块的时延状态各异，当这些包被送到输出端口时，它们抵达 PS 结构的先后顺序无法得到保障。目前公开的文献中至少有两种办法预防包失序：一种办法是在低速交换模块中建立整形机制(Reshaping Mechanism)[3]，但这种办法不能解决包在输出端口失序的问题；另一种办法是在输出端口设置大容量的输出缓存，重新恢复包的先后顺序[2]，但是该方法可能导致 HOL 阻塞。另外，还存在其他的包保序技术，例如两级交换的 3DQ 缓存技术和 IPv6 路由器的 POKD 技术[6]，但是它们不适用于 PS 结构。

本文为 PS 结构提出一种预防包失序的新技术，包括虚拟输入排队(VIQ)结构和包保序轮询(SKRR)算法：Cell 在 VIQ 中按照它们的先后顺序重新排序，然后由 SKRR 算法进行调度，分配到目的输出端口。理论分析表明，PS 结构采用这种新技术后，拥有与输出排队(OQ-Output Queue)结构相同的吞吐率，并且其平均时延不大于 OQ 的平均时延加上一个常数。

## 1　PS 结构模型

一种典型的 PS 结构模型如图 1 所示，该 PS 结构为 $N \times N$，端口线速率为 $R$，每个输入端口包含一个分路器(Demultiplexer)，输入分路器中的缓存分为 $K$ 段 FIFO，每段 FIFO 存储对应的到中间各交换模块的业务[7]，每个输出端口包含一个合路器(Multiplexer)；交换部分由 $K$ 个 OQ 交换模块组成，每

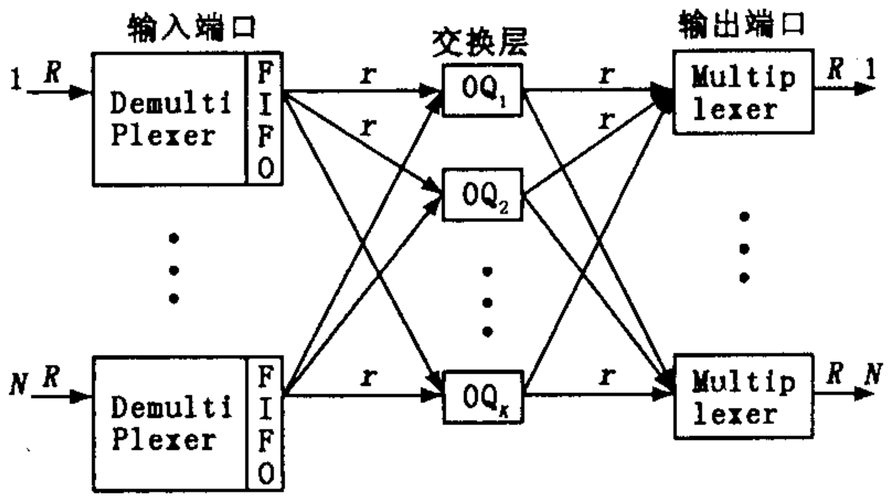


图 1　并行交换结构示意图

**收稿日期**：2005-06-22
**基金项目**：国家"863"课题基金资助项目(2001-AA-12-4-011)
**作者简介**：仇兴峰(1976-)，男，江苏徐州人，讲师，硕士生，主要从事通信技术研究.

个 OQ 交换模块为 $N \times N$,端口线速率为 $r$,内部加速比(Core Speedup)$S$ 定义为 $Kr/R$。从工程实现角度出发,本文假定 $S$ 为 1,PS 结构处理的单元为 Cell,每个输入分路器的调度判决都是独立执行的,各交换模块之间没有共享信息,PS 内部亦无反馈通信,并且 PS 结构满足尽职工作(Work - Conserving)[1-3]的条件。

PS 模型中包保序的要求是:对于任意输出端口,来自相同输入端口的 Cell 必须依照其进入 PS 结构的先后顺序来离开 PS 结构[2-3]。例如,任意一段时间内,目的输出端口为 $j$ 的所有 Cell 按照先后次序 Q 抵达输入端口 $i$,在 PS 结构中进行并行交换后,从输出端口 $j$ 发出,必须按照次序 Q 离开 PS 结构。然而,如下原因将导致 Cell 在 PS 结构中失序。

Cell 在输入缓存中失序:因为负载平衡的需要,到同一输出端口的 Cell 将被均分到各个交换层对应的 FIFO。由于各个 FIFO 的拥塞状况不尽相同,这些 Cell 在 FIFO 中的排队时延因而也不一样,它们抵达 PS 结构的先后次序可能因为排队时延不相同而被扰乱;当然,同一 FIFO 中的 Cell 是不会失序的。

Cell 在 OQ 交换模块中失序:与上面的分析同理,Cell 将因为在各个 OQ 交换模块中不同的排队时延而失序。

在本文中约定:Cell 由 IP 分组包切片后形成的等长数据包,长度一般为 512 byte 或 64 byte;时钟在线速率为 $R$ 的条件下,发送或接收一个 Cell 的时间;层:指中间的 OQ 交换模块,例如,第 $k$ 层即指第 $k$ 个 OQ 交换模块。

## 2 包保序的新技术

我们解决 PS 结构 Cell 失序问题的基本思路是:首先,在每一层 OQ 的输出端口根据 PS 结构的输入端口对 Cell 进行整形排序,纠正之前的失序问题,然后采用保序调度算法把重新排序后的 Cell 分配到 PS 结构的输出端口,从而确保 Cell 按照先后顺序离开输出端口。

### 2.1 虚拟输入排队

如图 2 所示,在每一层的输出端口,Cell 根据输入端口在 $VIQ_1$ 中排队,$VIQ_1$ 包含 $N$ 个 FIFO,对应 $N$ 个输入端口。例如,$FIFO_1$ 缓存输入端口 1 来的 Cell,$FIFO_2$ 缓存输入端口 2 来的 Cell,FIFO $N$ 缓存输入端口 $N$ 来的 Cell。可以看出 $VIQ_1$ 中任意 FIFO 里缓存的 Cell 具备如下性质:它们的输入端口和输出端口都是相同的,例如,从输入端口 2 来的、通过第 $k$ 层、目的地为输出端口 $j$ 的所有 Cell 都会在第 $k$ 层输出端口 $j$ 的 $VIQ_1$ 中的 $FIFO_2$ 中缓存排队。如第 2 节分析的那样,同一 FIFO 内的 Cell 不会失序,所以 $VIQ_1$ 能够确保 Cell 的先后顺序不被扰乱。以下用 $VIQ_1(i, k, j)$ 表示第 $k$ 层的第 $j$ 输出端口的 $FIFO_i$。

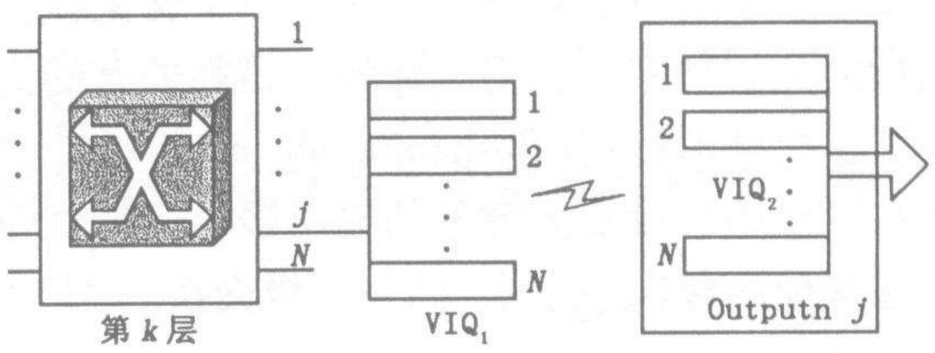


图 2 虚拟输入排队 $VIQ_1$ 和 $VIQ_2$ 图

$VIQ_2$ 也包含 $N$ 个 FIFO,到达同一输出端口的 Cell 根据输入端口在 $VIQ_2$ 中排队。$VIQ_2(i, j)$ 表示第 $j$ 输出端口的 $FIFO_i$。这样,$VIQ_2$ 中的每个 FIFO 实际上是缓存了来自相同输入端口、经过不同层的 Cell。基于 $VIQ_1$ 和 $VIQ_2$,这里要提出的问题是,要实施 Cell 保序,$VIQ_1$ 中的 Cell 应该怎样分配给 $VIQ_2$? 显然,需要合适的负载平衡调度算法(输入端口分路器执行的)和保序调度算法(输出端口合路器执行的)。

### 2.2 保序的轮询调度算法

文献[5]分析指出,最简单实用的负载平衡调度算法就是把目的地为同一输出端口的 Cell 按照轮询方式均匀的分配给每一层。本文采纳这种负载平衡算法,例如,如何调度输入端口 $i$ 上的要到输出端口 $j$ 的所有 Cell,第 1 个 Cell 分配给第 1 层,并在 $VIQ_1(i, 1, j)$ 中排队;第 2 个 Cell 分配给第 2 层,并在 $VIQ_1(i, 2, j)$ 中排队,第 $N$ 个 Cell 分配给第 $N$ 层,在 $VIQ_1(i, N, j)$ 中排队;第 $(N+1)$ 个 Cell 分配给第 1 层,在 $VIQ_1(i, 1, j)$ 中排队。显然,为了保序,$VIQ_2(i, j)$ 应该从 $VIQ_1$ 中按照如下方式收集 Cell:第 1 个 Cell 从 $VIQ_1(i, 1, j)$ 收集,第 2 个 Cell 从 $VIQ_1(i, 2, j)$ 收集,第 $N$ 个 Cell 从 $VIQ_1(i, N, j)$ 收集,然后,又开始从 $VIQ_1(i, 1, j)$ 收集。

在 SKRR(Sequence - Keeping Round - Robin)算法中,$VIQ_2(i, j)$ 用轮询指针 $p1(i, j)$ 来记住前一个 Cell 来自于哪个层或 $VIQ_1(i, k, j)$,输出端口 $j$ 用指针 $p2(j)$ 以轮询方式从 $VIQ_2$ 中收集 Cell。可以看出,指针 p1

$(i, j)$用来支持 Cell 保序,而指针 p2$(j)$用来保证尽职工作以及调度的公平性。这两种指针的主要区别是:如果 p1$(i, j)$指向一个空的 $VIQ_1$,那么它将一直指向该 $VIQ_1$ 项直至该 $VIQ_1$ 不为空;但是,如果 p2$(j)$指向的 $VIQ_2$ 为空,它将跳过空的 $VIQ_2$ 项直到指向一个非空的 $VIQ_2$ 项。SKRR 算法的形式化描述如下:

每一个 $VIQ_2$ 执行如下操作:

1)初始化:$p1(i, j) \leftarrow VIQ_1(i, 1, j)$;

2)$p1(i, j) \leftarrow VIQ_1(i, (k+1) \bmod K, j)$。

并且,每一个输出端口 j 执行如下操作:

If $VIQ_2(i, j) \neq 0$, then $p2(j) \leftarrow VIQ_2(i, j)$; else $p2(j) \leftarrow VIQ_2((i+1) \bmod N, j)$。

从以上分析描述可以看出,SKRR 算法简单、便于实现,不过该算法需要 $N^2K$ 个缓存,而相同交换能力的 OQ 仅仅需要 $N^2$ 个缓存,这还意味着 SKRR 算法需要复杂的缓存管理机制来处理大量的指针。

## 3　性能分析

本节用一个 OQ 结构作为参考交换结构来对比分析 PS 结构的性能,以下用 SKRR 来表示应用 VIQ 和 SKRR 的 PS 结构。在比较 SKRR 和 OQ 的平均时延性能和吞吐率之前,一些符号说明如下:$IQ_k^i(t)$:系统时间[0, $t$]内第 $i$ 输入端口中上与第 $k$ 层对应 FIFO 的队列长度;$A_{ij}(t)$:系统时间[0, $t$]内抵达第 $i$ 输入端口、目的地为第 $j$ 输出端口的 Cell 总和;$A_{ij}^k(t)$:$A_{ij}(t)$中被分配到第 $k$ 层的 Cell 总量;类似地, $B_{ij}(t)$和 $B_{ij}^k(t)$表示离开输入端口抵达交换层的 Cell;$C_{ij}(t)$和 $C_{ij}^k(t)$表示离开交换层抵达输出端口的 Cell;$D_{ij}(t)$表示离开输出端口的保序 Cell 总量。

### 3.1　平均排队时延

本文在分析时采用了与文献[7]相同的思想:首先在外来输入流量为 PS 结构输入端口流量的条件下,比较 SKRR 与 Delayed - OQ(见定理 1);然后在外来输入流量为 PS 结构交换层流量的条件下,比较 Delayed - OQ 与 OQ(见定理 2)。

定理 1:　SKRR 的平均时延不大于一个 Delayed - OQ 的平均时延加上一个常数 $NK$。

证明:根据 OQ 的性质[8],Delayed - OQ 具备性质

$$D_{ij}^{DOQ}(t+\lambda) \geqslant A_{ij}(t) \tag{1}$$

$\lambda$ 是平均时延, $D_{ij}^{DOQ}(t)$ 是输出端口的 Cell 数量。类似地,对于任意输入端口有 $B_{ij}^k(t + K \cdot IQ_i^k) \geqslant A_{ij}(t)$ 。故得

$$\sum_k B_{ij}^k(t + K \cdot IQ_i^k) \geqslant \sum_k A_{ij}(t) \tag{2}$$

以输入端口的视角来看,交换层和输出端口可以被认为是一个 Delayed - OQ。由式(1) 得

$$D_{ij}(t+\lambda) \geqslant \sum_k B_{ij}^k \tag{3}$$

由式(2) 和式(3) 可得 $D_{ij}(t + K \cdot IQ_i^k + \lambda) \geqslant \sum_k A_{ij}(t)$ 。因为 $IQ_i^k \leqslant N$ 以及 $D(t)$ 是非递减的函数,最后得到下式

$$D_{ij}(t + NK + \lambda) \geqslant D_{ij}^{DOQ}(t+\lambda) \tag{4}$$

结论得证。

定理 2:　SKRR 的平均时延不大于一个标准 OQ 的平均时延加上一个常数 $2NK$。

证明:首先比较标准 OQ 和 Ddelayed - OQ 的平均时延。根据其性质,OQ 结构满足 $D_{ij}^{DOQ}(t+\tau) \geqslant A_{ij}(t)$,其中 $\tau$ 为平均时延, $D_{ij}(t)$ 为输出端口的 Cell 数量。先计算 $VIQ_1$ 的平均时延,如第 2 节分析,离开 $VIQ_1$ 的 Cell 要保持它们抵达输入端口的先后顺序,所以 $VIQ_1$ 必须补偿输入端口排队导致的时延差异[3, 5]。因为输入端口的最大时延为 $NK$ 系统时钟[6],所以 $VIQ_1$ 的最大时延为 $NK$ 系统时钟,故可得

$$C_{ij}^k(t + NK) \geqslant B_{ij}^k(t) \tag{5}$$

另外,SKRR 的所有输出端口可被当作一个 OQ 结构,因此有:

$$D_{ij}(t+\tau) \geqslant \sum_k C_{ij}^k(t) \tag{6}$$

由式(5) 和式(6),并且把 $VIQ_1$ 和 $VIQ_2$ 当作 Delayed - OQ,可得到 $D_{ij}^{DOQ}(t + NK + \tau) \geqslant \sum_k B_{ij}^k(t)$。所以 OQ 和 Delayed - OQ 的关系为

$$D_{ij}^{DOQ}(t + NK + \tau) \geqslant D_{ij}^{OQ}(t + \tau) \tag{7}$$

由式(4) 和式(7),可以得出 SKRR 和 OQ 的平均时延差异的最大值为 $2NK$ 系统时钟,结论得证。

现在来估算在高速路由器中 SKRR 的时延代价。考虑一个 16 端口的 PS 交换机,端口线速率为 40 Gbps,内部加速比为 1,且 $N = K$,Cell 的长度取为 64 byte。根据定理 2 的结论,该 PS 交换机的平均时延比一个标准的 OQ 交换机最大高出 $2NK = 6.7\mu s$,该时延性能满足商业标准。

### 3.2 吞吐率

本文中,如果交换结构 $A$ 的总排队队长 $Q^A(t)$ 和交换结构 $B$ 的队长 $Q^B(t)$ 满足 $Q^B(t) \leqslant Q^A(t) \leqslant Q^B(t) + C$ ($C$ 是常数),可认为这两个交换结构的吞吐率相同[9],通过简单推导在定理 3 中给出一个宽松的界值。

定理 3: SKRR 和 OQ 具有相同的吞吐率。

证明:由定理 1 和定理 2,我们知道对于任意的 $i$, $j$ 和 $k$,有 $VIQ_1(i, k, j) \leqslant N$ 和 $IQ_k^i(t) \leqslant N$ 成立。因此我们得到

$$Q^{SKRR}(t) = \sum_i K \cdot IQ_i^k(t) + \sum_i \sum_j \sum_k VIQ_1(i,k,j) + Q^{OQ}(t) \leqslant KN^2 + KN^3 + Q^{OQ}(t)$$

又因为定理 2 有 $Q^{OQ}(t) \leqslant Q^{SKRR}(t)$,根据吞吐率的定义,结论得证。

因为 OQ 交换结构的吞吐率具备很多重要的性质,定理 3 非常有用,例如,OQ 在 Admissible Bernoulli i. i. d. 流量条件下吞吐率为 100%,所以由定理 3,SKRR 也有该性质。

### 3.3 性能对比

比较 OQ - PPS、两级交换和本文提出的 VIQ&SKRR3 种交换结构的性能,性能指标选取为是否支持包保序、系统所需缓存数量、相对时延(Relative Delay)和吞吐率。简便起见,我们默认 $K = N$,且加速比为 1。

如表 1 所示,VIQ&SKRR 比 OQ - PPS 多需要 $(N^3 - N^2)$ 个缓存,这些额外需要的缓存主要用来支持 Cell 保序,不过,缓存的增加并没有加大 VIQ&SKRR 的时延,它们具有相同的时延能力。此外,与两级交换相比,虽然它们都支持保序,但是 VIQ&SKRR 多需要 $N^2$ 个缓存,这些额外多出的缓存主要用来增加并行处理能力,所以 VIQ&SKRR 的时延比两级交换少 $2N^2$ 系统时钟。

表 1 交换结构的性能比较

| 交换结构 | 支持保序 | 缓存数量 | 相对时延 | 吞吐率 |
|---|---|---|---|---|
| OQ - PPS | 否 | $3N^2$ | $2N^2$ | 与 OQ 相同 |
| 两级交换 | 是 | $N^2 + N^3$ | $4N^2 - 2$ | 与 OQ 相同 |
| VIQ&SKRR | 是 | $2N^2 + N^3$ | $2N^2$ | 与 OQ 相同 |

## 4 结论

应用 PS 结构我们能够把多个低速低容量的交换模块组建成为高速大容量的交换系统,但是包保序问题严重阻碍着 PS 结构的工程实现,是一个未被很好解决的公认难题。本文提出一种新技术来解决这个问题,该新技术包含:对失序 Cell 进行重组的 VIQ 缓存结构,以及保证 Cell 保序和公平性的高效调度算法 SKRR。理论分析表明,应用该技术的 PS 结构和 OQ 结构具有相同的吞吐率,并且其平均时延最多比 OQ 结构高出 $2NK$ 系统时钟。这种新技术的代价是需要较大数量的缓存并且缓存管理机制较复杂,我们在下一步研究工作中考虑采用共享缓存,从而降低复杂度,提高此技术的工程应用性。

**参考文献:**

[1] Iyer S, Awadallah A, McKeown N. Analysis of a Packet Switch With the Memories Running Slower Than the Line Rate[J]. Proc of Infocom, 2000, (2): 529 - 37.

[2] Wang Walter, Dong Libin, Wayne Wolf. A Distributed Switch Architecture with Dynamic Load - balancing and Parallel Input - Queued Crossbars for Terabit Switch Fabrics[EB/OL]. http://www. ieee - infocom, 2002.

[3] Iyer S, McKeown N. Making Parallel Switches Practical[J]. Infocom, 2001, (3):1680 - 1687.

[4] Bennett J C R, Partridge C Shectman N. Packet Reordering is Not Pathological Network Ehavior[J]. IEEE/ACM Transactions on Networking, 1999, 7(6):789 - 798.

[5] Sundar Iyer, Nick McKeown. Analysis of the Parallel Packet Switch Architecture[EB/OL]. http://klamath. stanford. edu/ ~ sundaes/Papers/tonpps. pdf, 2003.

[6] Yue Chen, Yuguo Dong. A Packet - Order - Keeping - Demultiplexer in Parallel - Structure Router Based on Flow Classification [EB/OL]. http://www. cis. ohio - state. edu/ ~ iccnmc/, 2003.

[7] Cheng - Shang Chang, Duan - Shin Lee, Ching - Ming Lien . Load Balanced Birkhoff - von Neumann Switches, Part II: Multi - stage Buffering[J]. Computer Communications, 2002, 25:623 - 634.

[8] Indra Widjaja, Anwar I. Elwalid. Exploiting Parallelism to Boost Data - Path Rate in High - Speed IP/MPLS Networking[EB/OL]. http://www. ieee - infocom. org/, 2003.

[9] Leonardi E, Mellia M, Neri F, et al. On the Stability of Input - Queued Switches With Speed - up[J]. IEEE/ACM Transactions on Networking, 2001, 9(1):104 - 118.

(编辑:门向生)

# A Scheduling Algorithm for Maintaining Packet Order in Parallel Switches

QIU Xing - feng[1], YU Mei - rong[1], DONG Yu - guo[1,2]

(1. The Telecommunication Engineering Institute, Air Force Engineering University, Xi'an, Shaanxi 710077, China; 2. National Digital Switching System Engineering & Technology R&D Center, Zhengzhou, Henan 450002, China)

**Abstract**: Due to the load - balancing and parallelism of parallel switches, high - speed arrived packets (or cells) with same destination will be spread into many low speed switching fabrics for processing. When these packets are sent to the output, their sequence cannot be guaranteed. This paper proposes a structure of Virtual Input Queues (VIO) and a scheduling algorithm named Sequence Keeping Round - robin (SKRR), and simultaneously evaluates the throughput and the average delay performance for this technique in theory.

**Key words**: switch; load balancing; delay; scheduling