计算机科学 2002Vol. 29№. 8

ATM 交换机输入缓存队列 HOL 阻塞研究*)

On the HOL Blocking of ATM Switches Based FIFO Queues Mechanism

余 英 李 晶 张玉琢

(云南师范大学计算机科学系 昆明 650092)

Abstract The HOL blocking exists in the ATM switches based on FIFO queues mechanism. Under certain conditions, the throughput can be shown to be limited to approximately 58%. In order to satisfy the QoS of ATM services and obtain better effects, this paper proposes improvements by using priority mechanisms and PIM algorithm.

Keywords ATM switch, Priority queues, HOL blocking, PIM algorithm

1. 概述

在 ATM 交换机制中,输入缓存无阻塞交换技术成为提高交换效率的一个重要因素。输入缓存交换网的主要问题就是存在 HOL(head of line,队头)阻塞,即位于输入缓冲区 HOL 之后的信元,由于 FIFO(first in,first out,先进先出)缓冲区 HOL 信元的阻塞,既使当前时隙该信元指向的输出端口处于空闲状态,也无法在此时隙参与交换的现象。为了解决这一问题,已经提出了许多排队方法及其调度规则,大致可概括为如下几类:(1)滑动窗口法:在一个信元交换时隙内,依秩观测并处理一个 FIFO 队列中前 K(K>1)个信元,从中选取一个满足交换条件的信元参与交换连接,以此消除 HOL 阻塞[1];(2)在每个输入端口设置与网络输出端口数相同的多输入缓冲队列,以此消除 HOL 阻塞[2];(3)使用不同连接调度算法,如神经网络或迭代算法,以使吞吐率达到最大[3]。本文研究输入端口具有多 FIFO 队列、内部无阻塞的 N×N 交换网络,采用一种合理的机制,以求消除 HOL 阻塞,提高交换效率。

2. 优先级输入缓存模型的设置

优先级排队机制在通信、计算机,尤其综合业务数字网中有广泛的应用前景。不同的业务有不同的服务质量需求,ATM 要为各种类型的业务提供服务,而不同的业务对信元丢失率、时延与时延抖动的要求各不相同,所以 ATM 必须对分属不同业务的信元区别对待,为不同业务提供不同的服务质量。为了适应宽带业务的不同传输要求,从 1993 年起,ATM 论坛从流量控制的角度出发区分了 CBR、rt-VBR、nrt-VBR、ABR、UBR 五类业务。综合各类业务特性,CBR 与 rt-VBR 有最高的时延与时延抖动要求,需赋予较高的优先级,在保证以上两种业务的相应要求后可进一步服务 nrt-VBR 业务(此业务虽不需实时通信,但有带宽要求),进而是 ABR 业务(不需实时通信,带宽要求也可根据网络拥塞情况动态调配)。对 UBR 业务可不做任何承诺,因此,UBR 业务只能在以上业务仍有剩余带宽的性况下才可传送。以上分析将作为优先级输入、输出缓存设置的依据。

根据以上分析,要有效传输各类 ATM 信元,首先应考虑的是有关时间透明性问题,也就是保证时延敏感业务的相应特性。为此,我们在交换机每个输入端口处设置一个优先级仲裁(依据每个 ATM 信头的 VCI 标志),根据五类服务类型特性做四个优先级设定(图 1)。从输入优先级缓存的设置可看出,由于每个输入端口的缓存队列由原来无优先级设置的一个增至 4 个,使每个输入端口的 HOL 信元增至 4 个,以后再由不同的输出策略来裁决待输出的最终 HOL。由以下输出策略分析我们将看到:优先级缓存的设置除满足了不同信元对时延特性的需求,即满足各类信元对特定带宽的需求之外,对缓解输入队头阻塞,提高交换效率也打下了基础。

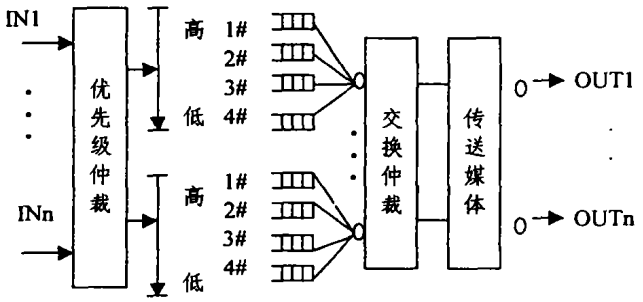

图 1 优先级输入缓存排队模型

3. 优先级输入队列并行迭代匹配算法 Pr—PIM

3.1 Pr-PIM(priority-parallel iterative matching,优先级并行迭代匹配)算法[1]

假设交换机为一 N×N 交换结构。且每一输入端口均有多个输入队列(本文为 4 优先级队列)。算法中"匹配"指的是在输入与输出之间寻找一对无冲突的端口,并建立端口间的连接以进行信元交换;而"并行迭代"是指端口间为实现最大匹配而在算法中重复执行的基本操作次数。在一个信元交换时隙内可进行多次迭代。Pr-PIM 算法的实现分以下几个步骤:

(1)request(申请)阶段。由于每个输入端口均有 4 个优先级队列,故有 4 个 HOL 信元分别要发送至指定的输出端口。为避免由多队列带来的更大范围的竞争冲突,本算法规定第一次参加迭代匹配的各端口信元只能是当前最高优先级队列的 HOL 信元,只有该信元竞争失败,该端口次优先级队列的 HOL 才可以参与下一轮迭代竞争。各输入端口在每一次迭代中均只允许有一个信元向所需输出端口提出发送申请,由图 2 可见,由于各输入 HOL 信元目的输出端口的随机性,在发

*)云南省教委青年科学基金资助(项目编号 9941025)。余 英 副教授,从事计算机网络通信领域研究。

送的过程中出现了多输入端口竞争同一输出端口的现象。

(2)Grant(允许)阶段。每个输出端口根据接受申请的个数进行裁决,若只有一个输入端口提出申请,则可直接向该端口发出允许信息。若申请数超过一个,则按各申请信元的优先级级别加以选择,让高优先级信元的申请先得到响应;若存在同优先级信元申请竞争,则按随机原则选出一个,确认后向输入端发出 Grant 信息。

(3)以上两步完成后若仍有输出端口未得到匹配,则重复步骤(1)至(2)进行第二轮迭代匹配,从第二轮迭代开始,仅有在前一轮竞争中失败的端口才可以参与匹配,此时对应端口的队头信元应由前一轮优先级的下一级优先级队列队头信元充当。如此迭代匹配,直到所有输出端口都得到匹配或迭代到 4 级优先级队列 HOL 信元均参与完匹配为止。

(4)send(发送)阶段。每一输入端口在迭代匹配结束后同时向对应输出端口发送指定队头信元。

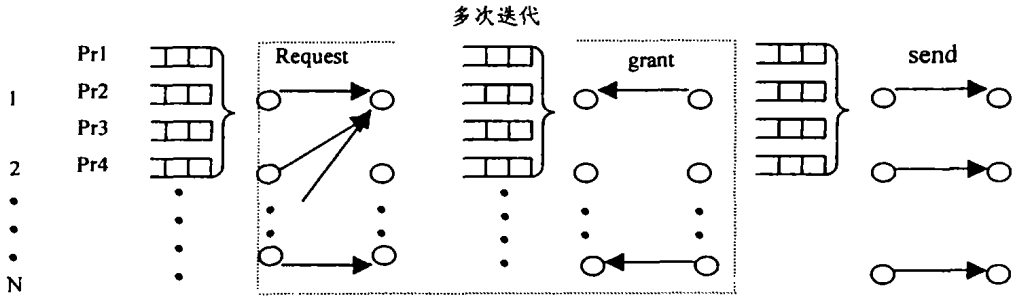

图 2 优先级输入队列并行迭代匹配

以上四个步骤的操作应在信元交换允许的一个时隙范围内完成。由于 ATM 交换时间的限制,PIM 算法的实现应由高速硬件完成。

3.2 计算机仿真

在输入缓存设置了四优先级队列后,经过多次叠代,HOL 阻塞的改善情况如何?为此,我们借助计算机仿真实验得到如下特性曲线(图 3)。

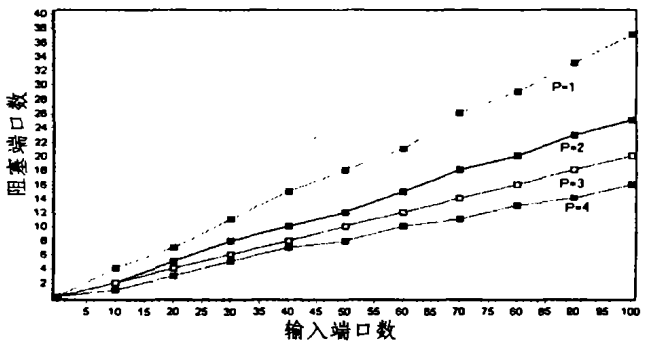

图 3 端口数与阻塞率关系

性能分析(1):参照图 3 可以看到,在四级优先级输入缓存的设置下,采用 PIM 算法后,HOL 阻塞现象得到了极大的改善:对于端口数较少的 ATM 交换机,经最大 4 级迭代后,绝大部分端口都得到了匹配。HOL 阻塞现象有了极大的缓解。但随着交换机端口数量的增加,经 4 级迭代后,仍有相当一部分输出端口未得到匹配。图 4 主要说明在一定端口限定下,随着到达率的增加,输入端口的阻塞情况。

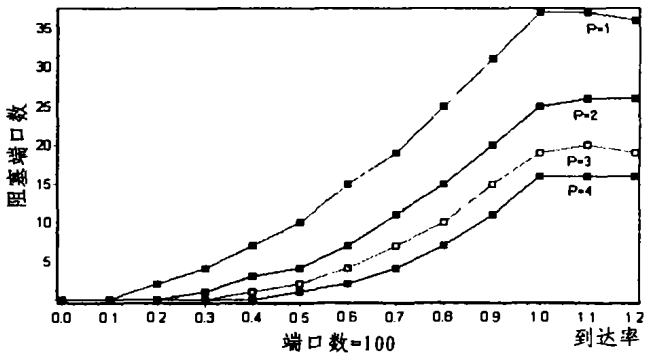

图 4 到达率与阻塞率关系

4. Pr-PIM 算法优化方案

4.1 基于轮询的优化方案

在 PIM 算法中,接收端对到来的多个 Request 的选择采用的是先按优先级顺序,当存在同级选择时随机选取的原则。由于各输入端口信元到达的随机性以及匹配迭代过程中策略的随机性,导致某些输入端口可能发生"饥饿"现象(即在某一时段内一直得不到服务),故应对上述 Pr-PIM 算法进一步改进,靠一定的策略来达到均衡各路匹配的目的。

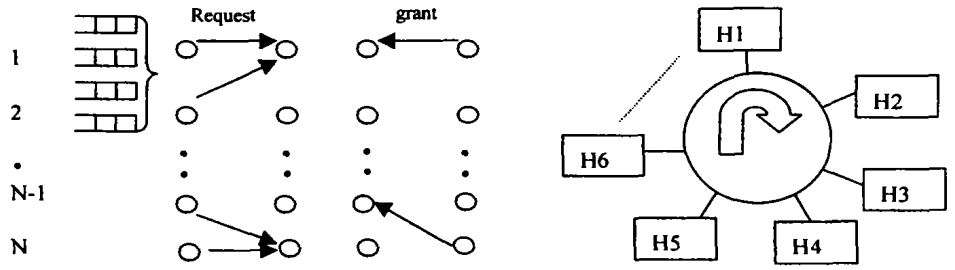

图 5 优先级轮询方案

在文[5]的基础上基于轮询的优化方案是对 Pr-PIM 算

(下转第 100 页)

但对自适应均衡来说，由于实际中信号特别是信道的变化较大，不易事先确定输入信号的特征及阈值，M 的结构也就难以确定，因此上述算法在实际中不便于使用。同时，其计算量还可以进一步降低，这就是以下我们改进后得到的算法。

3 改进的小波变换域中的自适应均衡算法

首先根据式(9)来估计 $\hat{R}_r(n)$：

$$\hat{R}_r(n)=\beta\hat{R}_r(n-1)+(1-\beta)r(n)r^T(n),0<\beta<1 \quad (9)$$

根据文[3]的结论：相关矩阵中元素的大小随着其与对角线距离的增加而呈指数下降，于是在迭代估计到 L 次时，取远小于1的一个值 ε，将 $\hat{R}_r(n)$对角线上最大绝对值的 ε 倍作为阈值，令 $\hat{R}_r(n)$中非角线上绝对值小于阈值的元素置零即估计得 $\hat{R}_r(n)$的稀疏结构。在 L 次后，仍按式(9)估计 $\hat{R}_r(n)$，其稀疏结构保持不变。

而对于方程组(8)，我们采用预条件共轭梯度法[7,8]来求解。由于共轭梯度法的收敛速度主要依赖于系数矩阵的条件数，因此需对 $\hat{R}_r(n)$作预处理。我们取 $A=diag(1/\sqrt{\hat{r}_{11}},1/\sqrt{\hat{r}_{22}},\cdots,1/\sqrt{\hat{r}_{NN}})$，其中 $\hat{r}_{ii}(i=1\sim N)$是 $\hat{R}_r(n)$中对角线上的元素。由于 $\hat{R}_r(n)$中元素的大小随着其与对角线距离的增加而呈指数下降[3]，因此 $A\hat{R}_rA=\tilde{R}_r$ 是近似于单位阵的矩阵，其条件数也近似较小。可见 A 选作为预处理矩阵是较为方便和可行的，这样我们就可以将方程组(8)转变为方程组(10)：

$$\tilde{R}_r(n)\hat{h}(n)=-2e(n)\hat{r}(n) \quad (10)$$

其中 $\hat{h}(n)=A^{-1}\hat{g}(n),\hat{r}(n)=Ar(n)$ (11)

即下降方向 $\hat{g}(n)$为：$\hat{g}(n)=A\hat{h}(n)$ (12)

由于 $\tilde{R}_r$ 由于是条件数较小的稀疏阵，因此用共轭梯度法求解的迭代次数为 $O(\sqrt{N})$数量级；而 $\hat{R}_r(n)$每行约有 $O(\log N)$个非零元素[3]，L 很小，则用(9)式估计时约需 $O(N\log N)$次运算，于是整个自适应均衡算法的计算量约为 $O(N\log N)$数量级。根据以上讨论，我们可以将新算法的步骤归结如下：

1)置初值 $\hat{R}_r(0)=0$，设定 ε 和 L；

2)计算式(1)、(2)、(3)及式(9)；

(下转第114页)

(上接第53页)

法的改进(图5)，具体体现在 Pr-PIM 算法的 grant 阶段，也就是当多个输入端口竞争同一输出端口时，输出端口采用一种什么样的策略来决定优胜者，以达到对各输入端口的请求较为公平的效果。以输出端口1为例，grant 部分的裁决按如下步骤执行：

(1)设初值1=＞ H1，相当于初始化时，将逻辑令牌先赋予第一个端口。

(2)判断竞争者的优先级别，若级别最高信元只有一个时，可直接向该输入端口发出 grant 信息，转(5)。

(3)若级别最高信元为多端口，则判断竞争端口中有无H1号端口，若有，则向 H1号输入端口发送 grant 信息；若无，则让1+H1 mod N ＝＞ H1，相当于逻辑令牌按序下传，而后继续寻找有无与新的 H1值匹配的竞争输入端口，按此法，直到找到为止。

(4)逻辑令牌轮询至下一端口：1+H1 mod N ＝＞ H1 。

(5)结束该轮裁决。当下一时隙到来时转(2)。

从以上算法可以看出，当出现多输入端口同一优先级信元产生竞争时，竞争优胜者的获胜优先级马上由最高降为最低，同时把下一轮的获胜最高优先权让给下一个端口，在这一问题的处理上，该策略充分体现了一种公平的原则，也从一定程度缓解了"饥饿"现象的发生。同时从硬件实现的角度考虑，由于消除竞争采用的是逻辑令牌轮询的预定方式，从硬件实现上比 PIM 采用的随机方式要简便，在简化硬件的前提下提高了竞争仲裁速度。

4.2 动态优先级权重参数的设置

当 ATM 交换机各输入端口的四优先级队列有相同的到达率，即各业务为均匀分布时，以上算法的处理方法是合理的，但若 ATM 各类业务出现了非均匀性，如果还沿用原来的算法，势必会降低相关业务的 QoS 特性。具体解释是：由以上分析我们知道，各端口优先级队列的划分依据是各类业务的时延及时延抖动特性，在均匀业务的前提下，各端口同一优先级队列的到达率是一致的，所以在出现同优先级信元竞争时采用的是随机的(Pr-PIM)或兼顾公平原则的基于轮询的优化算法。但在非均匀业务的前提下，各端口同一优先级队列的到达率不相等时，高到达率队列可能由于没有得到及时的服务产生信元丢失。为此，对同一优先级的信元进行竞争仲裁时应加一权重因子(≥1)，该权重因子的获取应依据当前队长是否处于限定队长边界而定。以上 Pr-PIM 及基于轮询的优化算法对同优先级 HOL 信元的处理方法相当于各队列权重因子皆为1的特例。

4.3 基于滑动窗口的动态迭代轮数调整

参照图3我们知道：随着交换机端口数的增加，经一轮(4次)迭代后，仍有一部分端口未得到匹配；同时由图4可以看到，要让所有端口均得到匹配，应进一步增加迭代次数。为此，我们采用滑动窗口法，突破 HOL 信元的限制，完成第一轮迭代并发现有未匹配端口后，随即对 HOL 后的信元进行下一轮迭代，直至完成所有端口的匹配。

结束语 HOL 阻塞严重影响了输入缓存 ATM 交换机的吞吐率性能，本文在 ATM 交换机四优先级输入缓存的前提下，运用综合治理的方法消除队头阻塞，进一步满足 ATM 交换机的 QOS 需求。

参 考 文 献

1 Chen M, Georganas N D, Yang O W W. Fast algorithm for multi-channel/port traffic assignment. IEEE ICC'94, 1994. 96～100

2 McKeown N, Anantharam V, Walrand J. Achieving 100% throughput in an inputqueued switch. IEEE INFOCOM'96, 1996. 296～302

3 Brown T X, Lin K H. Neural network design of a banyan network controller. IEEE J Sel Areas Commun, 1990, 8:1289～1298

4 Nong Ge, et al. Analysis of Nonblocking ATM Switches with Multiple Input Queues [J]. IEEE/ACM Transactions on Networking, 1999,7(1): 60～63

5 McKeown N. iSLIP: A Scheduling Algorithm for Input-Queued Switches[J]. IEEE Transactions on Networking. 1999,7(2)