一种基于缓存状态的 CICQ 交换机

郑　敏[①]　　郑竹林[①]　　王　斌[②]
[①](河南科技学院机电学院　新乡　453005)
[②](中科院微系统研究所上海无线研究中心　上海　200050)

摘　要：CICQ (Combined Input and Cross-point-Queued switch)交换机是一种在交叉点缓存加入少量缓存的交换结构，是当前研究的一个热点。该文研究了基于交叉点缓存的各种调度算法和基于 CICQ 的交换结构，提出了 LFF–LBF 算法，运用通畅度和拥塞度两个概念，保证了最急迫的信元被最先服务。仿真分析表明该算法在均匀分布和突发业务源的情况下具有良好的时延性能和稳定性能。

关键词：时延；组合输入交叉点排队；吞吐率
中图分类号：TN916.4　　**文献标识码**：A　　**文章编号**：1009-5896(2007)12-2978-03

A New CICQ Fabric Based on State of Cross-point-Buffer

Zheng Min[①]　　Zheng Zhu-lin[①]　　Wang Bin [②]
[①](*School of Mechanics and Electronics*, *Henan Institute of Science and Technology*, *Xinxiang* 453005, *China*)
[②](*Shanghai Institute of Microsystem and Information Technology*, *China Academy of Sciences*, *Shanghai* 200050, *China*)

Abstract: CICQ (Combined Input and Cross-point-Queued switch) is a fabric system with buffers in cross-points, which is a hotspot in switching area. In this paper, we analyze the current situation, and presented the LEF-LBF scheduling algorithm based on the two conceptions of congestion degree and unblocking degree to serve the urgent cells first. The simulation results show that LFF–LBF can offer very good performance on average delay and stability for uniform traffic with Bernoulli and burst arrivals.

Key words: Delay; CICQ; Throughput

1　引言

Internet 同时面临两个问题：更快的交换速度和服务质量(QoS)保障。近年来，交叉点缓存交换机(Combined Input and Cross-point-Queued switch, CICQ)被认为是一种解决这两个问题的理想架构。通过在交叉点加少量的缓存，各个输入端口和输出端口的调度器可以相互独立地工作，大大简化了交换机的调度算法，这种分布式的调度机制非常有助于实现支持 QOS 的高速交换机。很多实用性架构已经被提出来，如文献[1−3]。

但是众所周知，交叉点缓存块的数量同交叉点数量是一个数量级关系(O(N^2)，N是端口数)。这决定着大容量交换机使用大容量交叉点缓存是不现实的。本文在前人的基础之上，提出了一种基于交叉点缓存状态的调度算法，确保“最紧迫”的信元首先得到服务。计算机仿真分析比较表明，这种算法对UBP和UIBP数据流具有非常好的适应性。

2　交换机的结构

如图 1 所示，CICQ 模型是一个 $N\times N$ 的交换结构，它的每个输入端口上都有 N 个 VOQ 队列，分别对应一个输出端口，例如VOQ_{ij}用于存储第i个输入端口到第j个输出端口的信元。本文定义具有相同输入端口和输出端口的一系列信元为一个连接，如f_j表示输入端口i到输出端口j的连接。显然连接f_{ij}和队列VOQ_{ij}一一对应，为了描述方便，本节中经常混合使用连接f_{ij}与队列VOQ_{ij}两种说明方法，并不加以区分。

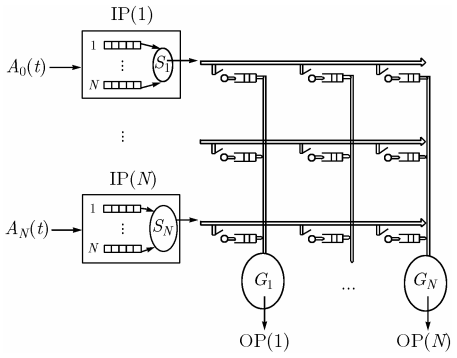

图 1　CICQ 交换的结构

本文用IP(i)和OP(j)(i,j=1,…,N)分别表示第i个输入端口和第j个输出端口。B_{ij}表示连接输入端口X_i和输出端口Y_j的交叉点缓存(只能存放一个信元)。L 表示交换结构采用的定长信元的长度。R 表示输入和输出端口的最大链路速率。λ_{ij}

2006-05-17 收到，2006-11-09 改回

表示连接f_{ij}的平均速率。$Y_{ij}(t)$和$X_{ij}(t)$表示第t时隙B_{ij}缓存的状态，当B_{ij}为非空时，则$Y_{ij}=1$，$X_{ij}=0$否则$Y_{ij}=0, X_{ij}=1$。$L_{ij}(t)$是以信元为单位的第t时隙VOQ_{ij}队列长度。

输入端口$i(1\le i\le N)$的信元到达是一个离散时间的随机过程$A_i(t)$。在一个时间片t内，最多有一个信元到达一个输入端口。到达输入i且输出是j的信元放入队列VOQ_{ij}中。在第t个时隙时队列VOQ_{ij}的长度表示为$L_{ij}(t)$。定义$A_{ij}(t)$为输入i到输出j的到达过程，到达速率为λ_{ij}，到达过程的集合$A(t)=\{A_i(t),1\le i\le N\}$必须是可容许的，即满足：

$$\sum_{i=1}^{N}\lambda_{ij}<1,\forall j,\quad \sum_{j=1}^{N}\lambda_{ij}<1,\quad \forall i \tag{1}$$

3　两阶段的调度算法

在 CIOQ 交换机中，本文将采用 LFF-LBF 算法，与 LQF/RR 不同的是，调度算法中用的比较量是交叉点缓存的状态。在介绍算法之前，先介绍描述交叉点缓存状态的两个概念：

通畅度：通畅度是输出端口通过交叉点缓存接收信元的难易程度，去往输出端口j的通畅度为：$P(j)=\sum_{i=1}^{N}X_{ij}$，其中，当$B_{ij}$为空时，$X_{ij}=1$，反之，$X_{ij}=0$。

拥塞度：拥塞度指输入端口向输出端口发送信元时被阻塞的程度，输入端口i的拥塞度为：$Q(i)=\sum_{j=1}^{N}Y_{ij}$，其中，当$B_{ij}$为非空时，$Y_{ij}=1$；反之，$Y_{ij}=0$。

下面，用这两个概念来阐述调度算法 LFF-LBF，算法的核心思想是：选择最迫切需要转发的信元进行服务。对于输入端口而言，选择对应输出端口通畅度最高的的交叉点缓存发送信元。对于输出端口而言，选取对应输入端口阻塞程度最高的交叉点缓存中的信元进行服务。算法的具体描述如下：

(1)输入端口调度　对于输入端口i，以U_r为调度依据，$U_r=\max_j\{U_j\}, j=1,\cdots,N$，然后输入端口$i$向交叉点缓存$\mathrm{B}_{ir}$发送分组。其中$U_j$的表达式为

$U_j=X_{ij}*P(j)=X_{ij}\sum_{i=1}^{N}X_{ij}$，其中当$B_{ij}$为空时，$X_{ij}=1$；反之，$X_{ij}=0$。

(2)输出端口调度　对于输出端口j，以W_s为调度依据，$W_s=\max_i\{W_i\}, i=1,\cdots,N$，然后输出端口$j$从交叉点缓存$B_{sj}$中取出分组。其中$W_i$的表达式为

$W_i=Y_{ij}*Q(i)=Y_{ij}\sum_{j=1}^{N}Y_{ij}$，其中，当$B_{ij}$为非空时，$Y_{ij}=1$；反之，$Y_{ij}=0$。

4　计算机仿真分析

本节采用计算机仿真的方法对 LFF-LBF 调度算法的性能进行分析。计算机仿真主要集中在算法的信元平均时延和稳定性两方面的性能上。因为，时延是 QoS 最主要的指标，而系统的稳定程度直接说明了系统的队列是否会无限增长。仿真选择交换结构的规模为 16×16。各 VOQ 的长度不做限制。仿真的时长是 500000 个时隙(时隙为发送一个信元需要的时间)，样本取样从第 50000 时隙才开始，平均时延是在[50000,500000]期间计算出来的。符合可容许条件的业务流采用均匀分布的贝努利业务流(UBP)和均匀分布的突发业务流(UIBP)两种。突发业务的突发长度选取 10，50 和 100 个信元 3 种情况进行。

稳定性的研究采用稳定参数进行评估。根据文献[4]，我们可以知道，当稳定参数$L(n)$满足$E(L(n))<\infty$时，调度算法是稳定的。$L(n)$的表达式是：

$$L(n)=\big(\mathrm{VOQ}_{11}(n)^2+\cdots+\mathrm{VOQ}_{1N}(n)^2+\cdots+\mathrm{VOQ}_{N1}(n)^2+\cdots+\mathrm{VOQ}_{NN}(n)^2\big)^{1/2} \tag{2}$$

$$L=E(L(n)) \tag{3}$$

其中$E(L(n))$是$L(n)$的算术平均。

在本节中，我们不打算从理论上证明 $L<\infty$，仅仅依靠仿真来比较 LQF-RR、OCF-OCF 和 LFF-LBF 等调度算法的时延和稳定性性能。下面，分两种进行讨论。

4.1 均匀分布的贝努利业务流

设各个输入端口信元到达过程满足独立同分布的贝努利过程，并且为均匀分布，输入和输出均是时隙同步，时隙的宽度等与单位信元的传输时间。图 2 是 LQF-RR、OCF-OCF 和 LFF-LBF 在均匀业务源下输入负载与平均时延的关系。从图中可以看出，当负荷低于 0.9 时，3 种调度算法的的性能差别不大。当负荷介于 0.9~0.95 时，OCF-OCF 和 LFF-LBF 差别不大，LQF-RR 性能稍差。当负荷超过 0.95 时，LQF-RR 性能最优。

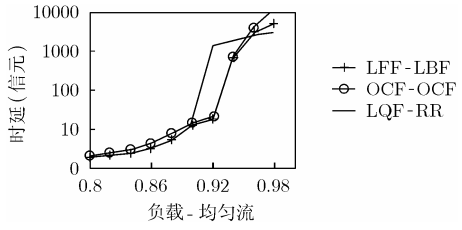

图 2　输入负载(UBP)与时延的关系

4.2 突发业务源

本节采用 ON/OF 业务模型，突发长度分别选择 10，50 和 100 个信元，一个突发周期内的信元指向同一输出端口，图 3 和 4 是在突发长度为 100 的情况下，LQF-RR、OCF-OCF 和 LFF-LBF 3 种调度算法的延时和稳定性的比较。图 3 和图 4 是进一步研究突发长度对延时和稳定性的影响。

在图 3 中，当负荷低于 0.65 时，3 种调度算法的性能差别不大。当负荷介于 0.65~0.9 时，3 种算法中 LFF-LBF 的

延时性能最好。当负荷超过 0.9 时，LFF–LBF 的延时性能远远好于其它两种算法。

在图 4 中，当负荷低于 0.85 时，3 种调度算法的稳定性能差别不大。当负荷介于 0.85~1.0 时，3 种算法中 LFF–LBF 的延时性能最好。当负荷大于 0.85 时， LFF–LBF 的稳定性能远远好于其它两种算法。换句话说， LFF–LBF 在高负载时，表现出极好的稳定性。

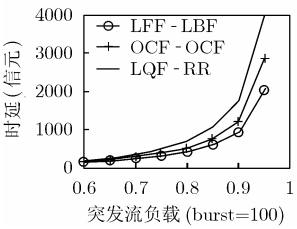

图 3 输入负载(UIBP)与时延的关系

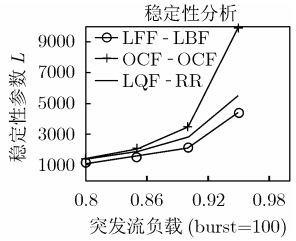

图 4 输入负载(UIBP)与稳定参数的关系

图 5 和图 6 显示了突发长度对延时和稳定性的影响。研究表明在 3 种不同突发长度的情况下，LFF–LBF 算法仍然表现出最优的性能，这进一步说明 LFF–LBF 算法对突发性业务有良好的适应性。

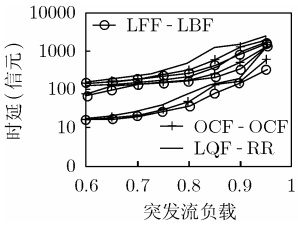

图 5 突发长度与时延的关系

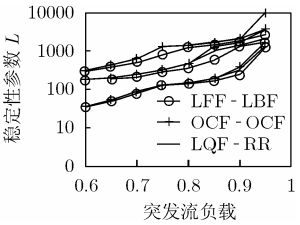

图 6 突发长度与稳定参数的关系

5 结束语

本文中对 CICQ 交换机提出了一种基于交叉点缓存状态的 LFF–BF 算法，它的核心思想是：对输入端口而言，选择对应输出端口通畅度最高的的交叉点缓存发送信元；对输出端口而言，选取对应输入端口阻塞程度最高的交叉点缓存中的信元进行服务。仿真研究表明 LFF–LBF 算法在 ON/OF 流的情况下，具有最优的稳定性和最低的时延特点，性能超越了 LQF-RR 和 OCF-OCF。

参 考 文 献

[1] Nabeshima M. Performance evaluation of a combined input and crosspoint queued switch. *IEICE Trans. Commun*, 2000, E83-B (3): 737–741.

[2] Javidi T, Magill R, and Hrabik T. A high-throughput scheduling algorithm for a buffered crossbar switch fabric. Internation Conference on Communication, Neuvo Y, 2001, vol.5: 1586–1591.

[3] Rojas-Cessa R, *et al.*. CIXB-1: Combined input-one-cell-crosspoint buffered switch. Proc. IEEE HPSR, Dallas, USA, 2001: 324–329.

[4] Rojas-Cessa R, Oki E, and Jonathan Chao H. CIXOB-k: Combined input-crosspoint-output buffered switch. IEEE GLOBECOM, San Antonio,USA, 2001, Vol. 4: 2654–2660.

郑 敏： 男，1974 年生，博士，研究方向为交换技术，Ad hoc 网络、传感器网络.

郑竹林： 男，1962 年生，硕士，副教授，研究方向为计算机图形学、机电一体技术.

王 斌： 男，1970 年生，博士，研究方向为交换技术和调度算法、MPLS, GMPLS，3GPP LTE 和 WCDMA.