输入排队 Crossbar 架构下的矩阵模型及 MM-LQF 调度策略

马祥杰, 毛军鹏, 兰巨龙, 张百生
(解放军信息工程大学信息工程学院, 河南郑州 450002; 国家数字交换系统工程技术研究中心, 河南郑州 450002)

摘 要: 输入排队 Crossbar 交换是高性能交换设备最为常用而关键的技术之一. 本文建立了 IQ-Crossbar 架构下的矩阵模型, 给出了 IQ-Crossbar 的状态矩阵、队长矩阵、到达矩阵和匹配矩阵的数学定义, 并通过分析 IQ-Crossbar 的信元排队机理, 提出和证明了队长矩阵迭代定理和状态矩阵迭代定理. 该矩阵模型为分析 IQ-Crossbar 架构下的调度算法提供了理论依据. 基于所建立的矩阵模型, 在分析现有 LQF 调度算法优缺点的基础上, 本文提出了一种新的调度策略 MM-LQF, 该策略的运算效率是 LQF 的 3.72 倍, 支持的端口门限速率是 LQF 的 2.35 倍, 在贝努利均匀流量重载条件下平均时延是 LQF 的 1/2; 在贝努利 Diagonal 流量条件下吞吐率为 100%.

关键词: 输入排队交叉开关; 矩阵模型; 队长矩阵; 调度策略; 最长队列优先
中图分类号: TN919.21 文献标识码: A 文章编号: 0372-2112 (2008) 01-0009-08

Matrix Model for Input-queued Crossbar Fabric and MM-LQF Scheduling Scheme

MA Xiang-jie, MAO Jun-peng, LAN Ju-long, ZHANG Bai-sheng
(*Information Engineering Institute, PLA Information Engineering University, Zhengzhou, Henan 450002, China;*
National Digital Switching System Engineering & Technological R&D Center, Zhengzhou, Henan 450002, China)

Abstract: The input-queued Crossbar Switching is one of the most popular and crucial technologies of the high-performance switching systems. The matrix model for IQ-Crossbar fabric is given in this paper, which has provided and well-defined the precise concepts of IQ-Crossbar fabric, such as the state matrix, the queueing length matrix, the arriving matrix, and the matching matrix. Based on analyzing the mechanism of the cell's queueing in the IQ-Crossbar, two matrix theorems of queueing length iteration as well as the state iteration are discussed and proved. The matrix model given in this paper provides the theoretical reference to IQ-Crossbar scheduling algorithms. Based on the matrix model set up in this paper and the analysis of the advantages and disadvantages of LQF algorithm, a new scheduling scheme of MM-LQF is provided, which has 3.72 times of operational efficiency, 2.35 times of port gate rates, 0.5 times of cell delay under heavy Bernoulli uniform load, 100% throughput under Bernoulli diagonal load of the LQF algorithm.

Key words: input queued crossbar; matrix model; queueing length matrix; scheduling scheme; longest queue first

1 引言

构建未来宽带信息网络基础设施是世界各国面临的一项极具挑战性的系统工程. 为了构建我国的信息高速公路, 国家科技部率先推出了 863 信息领域"高性能宽带信息网络(3TNet)"重大专项. 研制具有自主知识产权的高性能 IPv4/v6 双栈核心路由器(Core Router, CR)则是该专项的关键所在, 这需要研究大容量交换结构、高速转发引擎、多队列并行调度等多项核心技术. 交换结构是限制传统路由器交换容量提升的主要因素, 研究更为先进的包交换架构与调度策略以支撑 CR 庞大的交换容量一直是学术界广泛关注的焦点.

Crossbar 交换架构突破了传统交换系统的容量瓶颈, 解决了共享总线和共享处理器交换结构的局限性, 通过为输入端口和输出端口提供独立通道来保证端口吞吐量, 良好的可扩展性可以满足对系统总容量的需求. Crossbar 交换架构具备无阻塞交换结构, 它在实现系统吞吐量的同时, 也满足了最小包时延、最小丢包率等系统关键指标. 调度器(Scheduler)是 Crossbar 结构转发信元(Cell)的指挥中心, 它通过闭合或开启交换矩阵(i, j)处的 *Crosspoint*, 在输入端口 i 和输出端口 j 间建立或拆除信元通道. 输入端口可能因为竞争输出端口而发生

收稿日期: 2006-03-28; 修回日期: 2007-11-21
基金项目: 国家 863 信息技术领域重大专项项目(No. 2005AA121210); 国家 973 重点基础研究发展计划(No. 2007CB307102)

冲突,因此必须设置缓存队列以防止信元丢失. 根据缓存队列位置不同, Crossbar 交换结构可以分为输入排队(Input Queueing, IQ),输出排队(Output Queueing, OQ),组合输入输出排队(Combined Input Output Queueing, CIOQ)以及组合输入 Crosspiont 排队(Combined Input-Crosspoint Queueing, CICQ)等多种排队机制. OQ 排队机制在端口中引入了加速因子(Speedup Factor),从而 信元可以加速交换直接进入输出端口队列,避免了因端口竞争在输入端口的滞留,所以 OQ 排队机制对均匀的流量和突发流量均可获得高达 100% 的吞吐量. 文献[1, 2]中对基于 OQ 队列的调度算法进行了研究. 但存储器件技术发展的相对滞后阻碍了需要 N 倍加速比的 OQ 交换架构在 CR 中的应用,所以需要较多端口数量和较高端口速率的核心节点难于采用 OQ-Crossbar 交换架构. 输入排队 Crossbar(IQ-Crossbar)交换架构信元交换在输入端口设置缓存队列,因而不需要交换加速,在 CR 中得到了广泛的研究和应用. 文献[3~ 14]对基于 IQ-Crossbar 架构及其调度算法进行了深入的研究. CIOQ 和 CICQ 则是 IQ-Crossbar 的扩展形式,试图采用线速或者较小的加速因子获得或逼近 OQ 的性能[15~ 17]. 由此可见,基于 IQ-Crossbar 的交换成为 CR 工程实现中最为常用而关键的技术.

输入端口信元最简单的排队方式就是采用先入先出方式(First In First Out, FIFO),先到达的信元先接受服务,经过 Scheduler 调度穿越并离开 Crossbar fabric. 但排头信元发生竞争冲突时会阻碍后续信元的正常调度,发生“排头阻塞”(Head Of Line, HOL). VOQ 机制[17, 18]根据信元的目的端口不同分别排队,发往同一输出端口的信元单独排队. 这样,对于一个 $N \times N$ 的交换网络来讲,在输入端口中建立 N^2 个虚拟输出子队列(Virtual Output sub-Queue, VOQ). IQ-VOQ 排队机制有效地解决了 HOL 阻塞问题,在大容量交换设备中得到了广泛的应用. 然而对 IQ-VOQ 架构的信元排队机制的数学模型的研究很少,本文从 IQ-VOQ 排队策略出发,提出了 IQ-Crossbar 架构下的矩阵模型,建立了状态矩阵、队长矩阵、到达矩阵和匹配矩阵,给出了队长矩阵迭代定理和状态矩阵迭代定理. 基于该模型提出了 LQF 的改进调度策略 MM-LQF,该策略的运算效率是 LQF 的 3.72 倍,支持的端口门限速率是 LQF 的 2.35 倍,在重载情况下信元时延是 LQF 的 1/2.

本文以下内容组织方式为:第 2 部分给出了 IQ-Crossbar 架构的矩阵模型,第 3 部分建立了基于矩阵模型的 MM-LQF 调度策略,第 4 部分总结全文.

我们在本文中约定,交换网络: $N \times N$ 的 IQ-VOQ Crossbar 交换结构;信元:由 IP 分组切片后形成的等长数据包,长度 $L(Cell)$ 一般为 512 比特或 64 字节;时隙 $T(Cell)$:在线速率为 R 的条件下,发送或接收一个信元的时间; IP_i:交换网络中第 i 个输入端口; EP_j:交换网络中第 j 个输出端口; q_{ij}:输入端口为 IP_i,输出端口为 EP_j 的 VOQ 队列; $L(q_{ij})$:VOQ 队列 q_{ij} 的队长.

2 IQ-Crossbar 架构的矩阵模型

图 1 给出了基于 IQ-VOQ 排队机制的 3 × 3 Crossbar 交换结构. 从图中可以看出, IQ-Crossbar 架构包括四个组成部分:输入端口 IP、输出端口 EP、Crossbar 矩阵和调度单元 Scheduler. 输入端口根据到达信元目的端口不同,把每个信元缓存到 VOQ 队列中. IQ-Crossbar 架构无内部加速,所以输出端口信元到达速率不会超过线速 R 也就不需要缓存队列,信元直接由输出端口离开交换结构. Crossbar 矩阵由 N^2 个可控 Crosspoints 组成,闭合或断开对应端口之间的通路. 调度单元根据缓存队列状态信息确定一组输入输出端口匹配关系,在时隙结束前配置 Crossbar 矩阵.

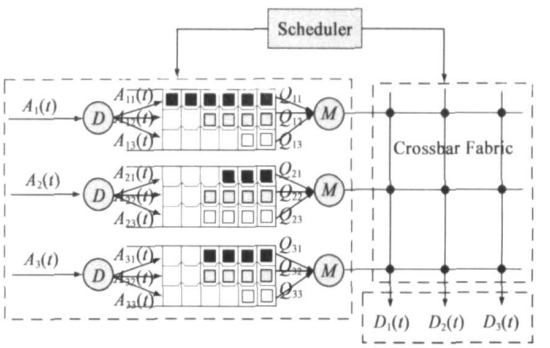

图 1 基于 IQ 的 3×3 Crossbar 交换架构

IQ-VOQ 中信元的排队状态是 Scheduler 决策各时隙中端口调度的依据. 在每个时隙起始时刻, VOQ 队列发送请求 Request 到对应输出端口,传递当前 IQ-VOQ 队列状态给 Scheduler 用作调度决策. 根据不同的衡量角度,通常把队列状态分为信元排队延迟(Queuing Delay)、信元排队长度(Queuing Length)和信元排队有无(Queuing Existence)三类. 排队延迟与排队长度分别是从最大信元排队时延和信元数量角度描述队列状态的,基于这两种队列状态的调度算法称为最大权重调度算法,如 OCF(Oldest Cell First)和 LQF(Longest Queue First)等. 文献[19, 20]证明了最大权重调度算法有很好的信元排队时延性能(如 OCF 算法)和较低的信元丢包率(如 LQF 算法),但是需要在每个时隙起始时刻向 Scheduler 传递信元队列的时延和队长参数,这往往需要更多的参数编码比特数和更大的时间复杂度,由于现有电子器件很难在几十甚至几纳秒内为 G 比特级端口速率的 Scheduler 做出调度决策,从而限制了其在 CR 中的应用. 基于信元排队有无队列状态仅需要 1 比特的参数编码因而在高速率、大容量路由器的交换结构中得到了广泛应

用. 下边定义1~ 4分别给出了IQ-VOQ中的基于时隙的状态矩阵 $\boldsymbol{Q}(n)$、队长矩阵 $\boldsymbol{L}(n)$、到达矩阵 $\boldsymbol{A}(n)$和匹配矩阵 $\boldsymbol{M}(n)$[21~26].

2.1 IQ-VOQ 状态矩阵 $\boldsymbol{Q}(n)$

定义1　IQ-VOQ 状态矩阵 $\boldsymbol{Q}(n)$　假设矩阵 $\boldsymbol{Q}(n)$为 $N\times N$ 矩阵, 其元素 $q_{ij}(n)$ ($i=1,\ \cdots,N;j=1,\ \cdots,N$)表示第 n 时隙输入端口为IP_i($i=1,\ \cdots,N$)输出端口为 EP_j($j=1,\ \cdots,N$)的VOQ队列 q_{ij}中信元的有无. 元素为"1"表示对应VOQ队列中有信元排队; 元素为"0"表示对应VOQ队列为空. $\boldsymbol{Q}(n)$矩阵的列向量 $\boldsymbol{Q}_j(n)$为输出端口 EP_j 对应的VOQ队列, 则称矩阵 $\boldsymbol{Q}(n)$为交换网络的IQ-VOQ状态矩阵, $\boldsymbol{Q}(n)$矩阵的定义式如下:

$$\boldsymbol{Q}(n)=\begin{bmatrix} q_{11}(n) & q_{12}(n) & \cdots & q_{1N}(n)\\ q_{21}(n) & q_{22}(n) & \cdots & q_{2N}(n)\\ \cdots & \cdots & \cdots & \cdots\\ q_{N1}(n) & q_{N2}(n) & \cdots & q_{NN}(n)\end{bmatrix}=\begin{bmatrix}\boldsymbol{Q}_1(n) & \boldsymbol{Q}_2(n) & \cdots & \boldsymbol{Q}_n(n)\end{bmatrix} \tag{1}$$

$$q_{ij}(n)=\begin{cases}0, & n\text{ 时隙}q_{ij}\text{无 Cell 排队}\\ 1, & n\text{ 时隙}q_{ij}\text{有 Cell 排队}\end{cases}\qquad \forall i,j\in\{1,2,\ \cdots,N\} \tag{2}$$

由定义式(2)可知, 每个元素 $q_{ij}(n)$有两种可能的取值, 所以对于 $N\times N$ 交换结构共有 2^{N^2}种不同的 $\boldsymbol{Q}(n)$矩阵组合, 形成集合 $\overline{\boldsymbol{Q}}(n)=\{\boldsymbol{Q}^1(n),\ \boldsymbol{Q}^2(n),\ \cdots,\ \boldsymbol{Q}^{2^{N^2}}(n),\}$, 则称集合 $\boldsymbol{Q}(n)$为VOQ矩阵空间. 为了便于进一步讨论, 我们给出基于时隙的IQ-VOQ队长矩阵 $\boldsymbol{L}(n)$的定义.

2.2 IQ-VOQ 队长矩阵 $\boldsymbol{L}(n)$及其衍生列向量 $\overrightarrow{\boldsymbol{L}}(n)$

定义2　IQ-VOQ 队长矩阵 $\boldsymbol{L}(n)$ 及其衍生列向量 $\overrightarrow{\boldsymbol{L}}(n)$　假设矩阵 $\boldsymbol{L}(n)$为 $N\times N$ 矩阵, 其元素 $l_{ij}(n)$ ($i=1,\ \cdots,N;j=1,\ \cdots,N$)表示第 n 时隙输入端口为IP_i($i=1,\ \cdots,N$)输出端口为 EP_j($j=1,\ \cdots,N$)的VOQ队列队长. 行向量 $\overrightarrow{\boldsymbol{L}_i}(n)$和列向量 $\overrightarrow{\boldsymbol{L}_j}(n)$分别为输入端口 IP_i 和输出端口EP_j 的队长向量, 列向量 $\overrightarrow{\boldsymbol{L}}(n)$为矩阵 $\boldsymbol{L}(n)$所有行向量 $\overrightarrow{\boldsymbol{L}_i}(n)$取模后所得的列向量. 则称 $\boldsymbol{L}(n)$为交换网络的IQ-VOQ队长矩阵, 称 $\overrightarrow{\boldsymbol{L}}(n)$为队长矩阵的衍生列向量, 队长矩阵 $\boldsymbol{L}(n)$及其衍生列向量 $\overrightarrow{\boldsymbol{L}}(n)$的定义式如下:

$$\boldsymbol{L}(n)=\begin{bmatrix} l_{11}(n) & l_{12}(n) & \cdots & l_{1N}(n)\\ l_{21}(n) & l_{22}(n) & \cdots & l_{2N}(n)\\ \cdots & \cdots & \cdots & \cdots\\ l_{N1}(n) & l_{N2}(n) & \cdots & l_{NN}(n)\end{bmatrix}=\begin{bmatrix}\overrightarrow{\boldsymbol{L}_1}(n)\\ \overrightarrow{\boldsymbol{L}_2}(n)\\ \cdots\\ \overrightarrow{\boldsymbol{L}_N}(n)\end{bmatrix}=\begin{bmatrix}\overrightarrow{\boldsymbol{L}_1}(n) & \overrightarrow{\boldsymbol{L}_2}(n) & \cdots & \overrightarrow{\boldsymbol{L}_N}(n)\end{bmatrix},$$

$$l_{ij}(n)=L(q_{ij}),\ l_{ij}(n)\geqslant 0,\ \forall i,j=1,2,\ \cdots,N \tag{3}$$

$$\overrightarrow{\boldsymbol{L}}(n)=\begin{bmatrix}l_1(n)\\ l_2(n)\\ \cdots\\ l_N(n)\end{bmatrix}=\begin{bmatrix}|\ \overrightarrow{\boldsymbol{L}_1}(n)\ |\\ |\ \overrightarrow{\boldsymbol{L}_2}(n)\ |\\ \cdots\\ |\ \overrightarrow{\boldsymbol{L}_N}(n)\ |\end{bmatrix}=\begin{bmatrix}\sum_{j=1}^{N} l_{1j}(n)\\ \sum_{j=1}^{N} l_{2j}(n)\\ \cdots\\ \sum_{j=1}^{N} l_{Nj}(n)\end{bmatrix},$$

$$l_i(n)=\sum_{j=1}^{N} l_{ij}(n),\ \forall i,j\in\{1,2,\ \cdots,N\} \tag{4}$$

由定义1和定义2, 可知:

$$q_{ij}(n)=\begin{cases}0, & l_{ij}(n)=0\\ 1, & l_{ij}(n)>0\end{cases}\qquad \forall i,j\in\{1,2,\ \cdots,N\} \tag{5}$$

2.3 IQ-VOQ 到达矩阵 $\boldsymbol{A}(n)$及其衍生列向量 $\overrightarrow{\boldsymbol{A}}(n)$

在容许流量(Admissible Traffic, AT)到达条件下, VOQ队列 q_{ij}的到达矩阵$\boldsymbol{A}(n)$定义如下:

定义3　IQ-VOQ 到达矩阵 $\boldsymbol{A}(n)$ 与衍生列向量 $\overrightarrow{\boldsymbol{A}}(n)$　假设矩阵 $\boldsymbol{A}(n)$为 $N\times N$ 矩阵, 其元素 $a_{ij}(n)$($i=1,\ \cdots,N;j=1,\ \cdots,N$)表示第 n 时隙输入端口IP_i($i=1,\ \cdots,N$)输出端口为 EP_j($j=1,\ \cdots,N$)的VOQ队列 q_{ij}的信元到达过程. 元素为"1"表示VOQ队列 q_{ij}有信元到达; 元素为"0"表示VOQ队列 q_{ij}无信元到达. 行向量 $\overrightarrow{\boldsymbol{A}_i}(n)$为输入端口 IP_i($i=1,\ \cdots,N$) N 个虚拟队列($q_{i1},\ q_{i2},\ \cdots,\ q_{iN}$)的到达向量, 列向量$\overrightarrow{\boldsymbol{A}}_{(}n)$为矩阵$\boldsymbol{A}(n)$所有到达向量 $\overrightarrow{\boldsymbol{A}_i}(n)$取模后所得的列向量. 则称 $\boldsymbol{A}(n)$为交换网络的IQ-VOQ到达矩阵 $\boldsymbol{A}(n)$, 称 $\overrightarrow{\boldsymbol{A}}(n)$为矩阵 $\boldsymbol{A}(n)$的衍生列向量, 到达矩阵 $\boldsymbol{A}(n)$及其衍生列向量 $\overrightarrow{\boldsymbol{A}}(n)$的定义式如下:

$$\boldsymbol{A}(n)=\begin{bmatrix} a_{11}(n) & a_{12}(n) & \cdots & a_{1N}(n)\\ a_{21}(n) & a_{22}(n) & \cdots & a_{2N}(n)\\ \cdots & \cdots & \cdots & \cdots\\ a_{N1}(n) & a_{N2}(n) & \cdots & a_{NN}(n)\end{bmatrix}=\begin{bmatrix}\overrightarrow{\boldsymbol{A}_1}(n)\\ \overrightarrow{\boldsymbol{A}_2}(n)\\ \cdots\\ \overrightarrow{\boldsymbol{A}_N}(n)\end{bmatrix} \tag{6}$$

$$a_{ij}(n)=\begin{cases}0, & n\text{ 时隙}q_{ij}\text{无 Cell 到达}\\ 1, & n\text{ 时隙}q_{ij}\text{有 Cell 到达}\end{cases}\qquad \forall i,j\in\{1,2,\ \cdots,N\} \tag{7}$$

$$\overrightarrow{\boldsymbol{A}}(n)=\begin{bmatrix}a_1(n)\\ a_2(n)\\ \cdots\\ a_N(n)\end{bmatrix}=\begin{bmatrix}|\ \overrightarrow{\boldsymbol{A}_1}(n)\ |\\ |\ \overrightarrow{\boldsymbol{A}_2}(n)\ |\\ \cdots\\ |\ \overrightarrow{\boldsymbol{A}_N}(n)\ |\end{bmatrix}=\begin{bmatrix}\sum_{j=1}^{N} a_{1j}(n)\\ \sum_{j=1}^{N} a_{2j}(n)\\ \cdots\\ \sum_{j=1}^{N} a_{Nj}(n)\end{bmatrix}$$

$$a_i(n)=\sum_{j=1}^{N} a_{ij}(n)\leqslant 1,\ \forall i\in\{1,2,\ \cdots,N\} \tag{8}$$

2.4 IQ-Crossbar 匹配矩阵 $M(n)$

在时隙结束前, Scheduler 根据调度算法决定一组输入端口 IP_i 与输出端口 EP_j 连接映射, 并用该映射配置 Crossbar 交叉开关, 建立信元通道完成该时隙的交换. 用来配置 Crossbar 交叉开关的连接映射是一个定义如下的匹配矩阵 $M(n)$:

定义 4 IQ-Crossbar 匹配矩阵 $M(n)$ 假设矩阵 $M(n)$ 为 $N\times N$ 矩阵, 其元素 $m_{ij}(n)$ ($i=1, \cdots, N; j=1, \cdots, N$) 表示第 n 时隙 Crossbar 结构的对应 IP_i 和 EP_j 的 Crosspoint 的状态. 元素为"1"表示对应的 Crosspoint 闭合; 元素为"0"表示对应的 Crosspoint 开启. 则称矩阵 $\boldsymbol{M}(n)$ 为 IQ-Crossbar 结构的匹配矩阵, 匹配矩阵 $\boldsymbol{M}(n)$ 的定义式如下:

$$\boldsymbol{M}(n)=\begin{bmatrix} m_{11}(n) & m_{12}(n) & \cdots & m_{1N}(n) \\ m_{21}(n) & m_{22}(n) & \cdots & m_{2N}(n) \\ \cdots & \cdots & \cdots & \cdots \\ m_{N1}(n) & m_{N2}(n) & \cdots & m_{NN}(n) \end{bmatrix} \tag{9}$$

$$m_{ij}(n)=\begin{cases} 0, & n \text{ 时隙 Crosspoint } k_{ij} \text{开启} \\ 1, & n \text{ 时隙 Crosspoint } k_{ij} \text{闭合} \end{cases}$$

$$\forall i,j\in\{1,2,\cdots,N\} \tag{10}$$

$$\sum_{j=1}^{N} m_{ij}(n)\leqslant 1,\ \sum_{i=1}^{N} m_{ij}(n)\leqslant 1,\ \forall i,j\in\{1,2,\cdots,N\} \tag{11}$$

根据以上对队长矩阵 $\boldsymbol{L}(n)$、到达矩阵 $\boldsymbol{A}(n)$ 和匹配矩阵 $\boldsymbol{M}(n)$ 的描述, 可以推导出队长迭代矩阵定理:

2.5 队长矩阵迭代定理与求值引理

定理 1 队长矩阵迭代定理 假设交换网络在时隙 n 的队长矩阵、到达矩阵和匹配矩阵分别为 $\boldsymbol{L}(n)$、$\boldsymbol{A}(n)$ 和 $\boldsymbol{M}(n)$, 那么下一时隙的队长矩阵满足 $\boldsymbol{L}(n+1)=\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)$.

证明 假设对于 $N\times N$ 的交换网络, $\forall i,j\in\{1,2,\cdots,N\}$, n 时隙的队长为 $l_{ij}(n)$, 离开信元数为 $m_{ij}(n)$, $(n+1)$ 时隙到达信元数为 $a_{ij}(n+1)$, 那么 $(n+1)$ 时隙的队长为 $l_{ij}(n+1)=l_{ij}(n)-m_{ij}(n)+a_{ij}(n+1)$.

根据定义 2、定义 3 和定义 4 可知 $\boldsymbol{L}(n+1)$.

$$\boldsymbol{L}(n+1)=\begin{bmatrix} l_{11}(n+1) & l_{12}(n+1) & \cdots & l_{1N}(n+1) \\ l_{21}(n+1) & l_{22}(n+1) & \cdots & l_{2N}(n+1) \\ \cdots & \cdots & \cdots & \cdots \\ l_{N1}(n+1) & l_{N2}(n+1) & \cdots & l_{NN}(n+1) \end{bmatrix}$$

$$=\begin{bmatrix} L_{11}(n)-m_{11}(n)+a_{11}(n+1) & L_{12}(n)-m_{12}(n)+a_{12}(n+1) & \cdots & L_{1N}(n)-m_{1N}(n)+a_{1N}(n+1) \\ L_{21}(n)-m_{21}(n)+a_{21}(n+1) & L_{22}(n)-m_{22}(n)+a_{22}(n+1) & \cdots & L_{2N}(n)-m_{2N}(n)+a_{2N}(n+1) \\ \cdots & \cdots & \cdots & \cdots \\ L_{N1}(n)-m_{N1}(n)+a_{N1}(n+1) & L_{N2}(n)-m_{N2}(n)+a_{N2}(n+1) & \cdots & L_{NN}(n)-m_{NN}(n)+a_{NN}(n+1) \end{bmatrix}$$

$$=\begin{bmatrix} l_{11}(n) & l_{12}(n) & \cdots & l_{1N}(n) \\ l_{21}(n) & l_{22}(n) & \cdots & l_{2N}(n) \\ \cdots & \cdots & \cdots & \cdots \\ l_{N1}(n) & l_{N2}(n) & \cdots & l_{NN}(n) \end{bmatrix}-\begin{bmatrix} m_{11}(n) & m_{12}(n) & \cdots & m_{1N}(n) \\ m_{21}(n) & m_{22}(n) & \cdots & m_{2N}(n) \\ \cdots & \cdots & \cdots & \cdots \\ m_{N1}(n) & m_{N2}(n) & \cdots & m_{NN}(n) \end{bmatrix}$$

$$+\begin{bmatrix} a_{11}(n+1) & a_{12}(n+1) & \cdots & a_{1N}(n+1) \\ a_{21}(n+1) & a_{22}(n+1) & \cdots & a_{2N}(n+1) \\ \cdots & \cdots & \cdots & \cdots \\ a_{N1}(n+1) & a_{N2}(n+1) & \cdots & a_{NN}(n+1) \end{bmatrix}=\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)$$

证毕.

引理 1 队长矩阵求值引理 假设交换网络在时隙 n 的队长矩阵为 $\boldsymbol{L}(n)$, 对于任意 k 时隙的到达矩阵和匹配矩阵分别为 $\boldsymbol{A}(k)$ 和 $\boldsymbol{M}(k)$ $k\in\{0,1,\cdots,n\}$, 那么队长矩阵 $\boldsymbol{L}(n)$ 满足:

$$\boldsymbol{L}(n)=\sum_{k=1}^{n-1}\{\boldsymbol{A}(k)-\boldsymbol{M}(k)\}+\boldsymbol{A}(n) \tag{12}$$

证明 由定理 1 可知,

$$\begin{aligned}\boldsymbol{L}(n)&=\boldsymbol{L}(n-1)-\boldsymbol{M}(n-1)+\boldsymbol{A}(n)\\&=\boldsymbol{L}(n-2)-\boldsymbol{M}(n-2)+\boldsymbol{A}(n-1)-\boldsymbol{M}(n-1)+\boldsymbol{A}(n)\\&=\boldsymbol{L}(0)-\boldsymbol{M}(0)+\boldsymbol{A}(1)-\cdots-\boldsymbol{M}(n-2)+\boldsymbol{A}(n-1)\\&\quad-\boldsymbol{M}(n-1)+\boldsymbol{A}(n)\\&=\boldsymbol{L}(0)-\boldsymbol{M}(0)+\sum_{k=1}^{n-1}\{\boldsymbol{A}(k)-\boldsymbol{M}(k)\}+\boldsymbol{A}(n)\end{aligned}$$

由于假设交换网络信元从时隙 1 开始计算, 从而 $\boldsymbol{L}(0)=0$, $\boldsymbol{M}(0)=0$. 所以,

$$\boldsymbol{L}(n)=\sum_{k=1}^{n-1}\{\boldsymbol{A}(k)-\boldsymbol{M}(k)\}+\boldsymbol{A}(n)$$

证毕.

2.6 状态矩阵迭代定理

由公式 4 可知, IQ-VOQ 队列状态与队长直接相关, 为此, 我们定义状态函数 $S(x)$ 和矩阵状态函数 $\boldsymbol{S}(\boldsymbol{A})$

如下:

如果函数 $S(x)$ 满足如下关系式, 则称函数 $S(x)$ 为定义在实数空间上的状态函数:

$$S(x)=[x./x]=\begin{cases}0, & x=0\\ 1, & x\neq 0\end{cases},\ \forall x\in\mathbf{R} \qquad (13)$$

如果对于任意实矩阵 $\boldsymbol{A}=\begin{bmatrix}a_{11} & a_{12} & \cdots & a_{1N}\\ a_{21} & a_{22} & \cdots & a_{2N}\\ \cdots & \cdots & \cdots & \cdots\\ a_{N1} & a_{N2} & \cdots & a_{NN}\end{bmatrix}$, 满足:

$$\boldsymbol{S(A)}=\begin{bmatrix}S(a_{11}) & S(a_{12}) & \cdots & S(a_{1N})\\ S(a_{21}) & S(a_{22}) & \cdots & S(a_{2N})\\ \cdots & \cdots & \cdots & \cdots\\ S(a_{N1}) & S(a_{N2}) & \cdots & S(a_{NN})\end{bmatrix}=\begin{bmatrix}[a_{11}./a_{11}] & [a_{12}./a_{12}] & \cdots & [a_{1N}./a_{1N}]\\ [a_{21}./a_{21}] & [a_{22}./a_{22}] & \cdots & [a_{2N}./a_{2N}]\\ \cdots & \cdots & \cdots & \cdots\\ [a_{N1}./a_{N1}] & [a_{N2}./a_{N2}] & \cdots & [a_{NN}./a_{NN}]\end{bmatrix} \qquad (14)$$

矩阵函数 $\boldsymbol{S(A)}$ 满足如下关系式, 则称矩阵函数 $\boldsymbol{S(A)}$ 为定义在实矩阵空间上的矩阵状态函数, 并记作:

$$\boldsymbol{S(A)=[A./A]} \qquad (15)$$

定理 2　状态矩阵迭代定理　假设交换网络在时隙 n 的队长矩阵、到达矩阵和匹配矩阵分别为 $\boldsymbol{L}(n)$、$\boldsymbol{A}(n)$ 和 $\boldsymbol{M}(n)$, 那么下一时隙的队列状态矩阵满足 $\boldsymbol{Q}(n+1)=S[\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)]=\{[\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)]./[\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)]\}$.

证明　由公式 4 可得,

$$q_{ij}(n+1)=\begin{cases}0, & l_{ij}(n+1)=0\\ 1, & l_{ij}(n+1)>0\end{cases}\quad \forall i,j\in\{1,2,\cdots,N\}$$

由状态函数 $S(x)$ 的定义可得,

$$q_{ij}(n+1)=S[l_{ij}(n+1)],\ \forall i,j\in\{1,2,\cdots,N\},$$

由定理 1 可知, $l_{ij}(n+1)=l_{ij}(n)-m_{ij}(n)+a_{ij}(n+1)$,

所以, $q_{ij}(n+1)=S[l_{ij}(n)-m_{ij}(n)+a_{ij}(n+1)]$, $\forall i,j\in\{1,2,\cdots,N\}$

由定义 1 可知 $\boldsymbol{Q}(n+1)$.

$$\boldsymbol{Q}(n+1)=\begin{bmatrix}q_{11}(n+1) & q_{12}(n+1) & \cdots & q_{1N}(n+1)\\ q_{21}(n+1) & q_{22}(n+1) & \cdots & q_{2N}(n+1)\\ \cdots & \cdots & \cdots & \cdots\\ q_{N1}(n+1) & q_{N2}(n+1) & \cdots & q_{NN}(n+1)\end{bmatrix}=$$

$$\begin{bmatrix}S[l_{11}(n)-m_{11}(n)+a_{11}(n+1)] & S[l_{12}(n)-m_{12}(n)+a_{12}(n+1)] & \cdots & S[l_{1N}(n)-m_{1N}(n)+a_{1N}(n+1)]\\ S[l_{21}(n)-m_{21}(n)+a_{21}(n+1)] & S[l_{22}(n)-m_{22}(n)+a_{22}(n+1)] & \cdots & S[l_{2N}(n)-m_{2N}(n)+a_{2N}(n+1)]\\ \cdots & \cdots & \cdots & \cdots\\ S[l_{N1}(n)-m_{N1}(n)+a_{N1}(n+1)] & S[l_{N2}(n)-m_{N2}(n)+a_{N2}(n+1)] & \cdots & S[l_{NN}(n)-m_{NN}(n)+a_{NN}(n+1)]\end{bmatrix}$$

$$=S\begin{bmatrix}l_{11}(n)-m_{11}(n)+a_{11}(n+1) & l_{12}(n)-m_{12}(n)+a_{12}(n+1) & \cdots & l_{1N}(n)-m_{1N}(n)+a_{1N}(n+1)\\ l_{21}(n)-m_{21}(n)+a_{21}(n+1) & l_{22}(n)-m_{22}(n)+a_{22}(n+1) & \cdots & l_{2N}(n)-m_{2N}(n)+a_{2N}(n+1)\\ \cdots & \cdots & \cdots & \cdots\\ l_{N1}(n)-m_{N1}(n)+a_{N1}(n+1) & l_{N2}(n)-m_{N2}(n)+a_{N2}(n+1) & \cdots & l_{NN}(n)-m_{NN}(n)+a_{NN}(n+1)\end{bmatrix}$$

$$=S\left\{\begin{bmatrix}l_{11}(n) & l_{12}(n) & \cdots l_{1N}(n)\\ l_{21}(n) & l_{22}(n) & \cdots l_{2N}(n)\\ \cdots & \cdots & \cdots \cdots\\ l_{N1}(n) & l_{N2}(n) & \cdots l_{NN}(n)\end{bmatrix}-\begin{bmatrix}m_{11}(n) & m_{12}(n) & \cdots m_{1N}(n)\\ m_{21}(n) & m_{22}(n) & \cdots m_{2N}(n)\\ \cdots & \cdots & \cdots \cdots\\ m_{N1}(n) & m_{N2}(n) & \cdots m_{NN}(n)\end{bmatrix}+\begin{bmatrix}a_{11}(n+1) & a_{12}(n+1) & \cdots a_{1N}(n+1)\\ a_{21}(n+1) & a_{22}(n+1) & \cdots a_{2N}(n+1)\\ \cdots & \cdots & \cdots \cdots\\ a_{N1}(n+1) & a_{N2}(n+1) & \cdots a_{NN}(n+1)\end{bmatrix}\right\}$$

$$=S\{\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)\}=\{[\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)]/[\boldsymbol{L}(n)-\boldsymbol{M}(n)+\boldsymbol{A}(n+1)]\}$$

证毕.

在上边提出的 IQ-Crossbar 矩阵模型中, 到达矩阵 $\boldsymbol{A}(n)$ 描述了基于时隙的 IQ-VOQ 虚拟输出队列的信元到达过程, 是队长递增的因素; 匹配矩阵 $\boldsymbol{M}(n)$ 描述了基于时隙的 IQ-Crossbar 的匹配过程, 匹配成功的 IQ-VOQ 队列在该时隙送出一个信元, 在这个意义上, 匹配矩阵 $\boldsymbol{M}(n)$ 是队长递减的因素; 队长矩阵 $\boldsymbol{L}(n)$ 的定义描述了基于时隙的 IQ-VOQ 队列中排队信元的数量信息, 它是 LQF 类调度算法的基础; 状态矩阵 $\boldsymbol{Q}(n)$ 描述了基于时隙的 IQ-VOQ 队列状态信息, 它是基于三步迭代极大匹配调度算法的基础; 定理 1 从时隙迭代的角度描述了 IQ-VOQ 队长矩阵 $\boldsymbol{L}(n)$ 基于矩阵模型的迭代关系, 引理 1 给出了 $\boldsymbol{L}(n)$ 的计算方法, 这两个定理是本文分析基于队长信息调度算法的依据; 定理 2 从时隙迭代的角度描述了 IQ-VOQ 状态矩阵 $\boldsymbol{Q}(n)$ 基于矩阵模型的迭代关系, 该定理是本文分析极大匹配调度算法的依据. 图 2 中描述了 IQ-Crossbar 矩阵模型的时隙关系图.

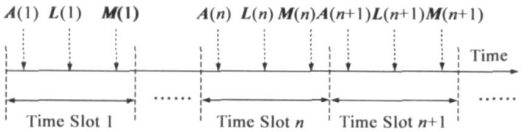

图 2 *IQ*-Crossbar矩阵模型的时隙关系图

3 基于 IQ-Crossbar 矩阵模型的 MM-LQF 调度策略

3.1 MM-LQF 调度策略算法描述

最长队列优先 LQF 算法是 IQ 架构中采用多比特信息的基于权重的信元调度策略[19], 其基本思想是优先服务最长的输入队列, 从而使每次端口连接匹配 M 的总权重L 最大化, 如下式所示:

$$L = \max_{\forall M}\left[\sum_{i,j\in M} l_{ij}(n)\right] \quad (16)$$

LQF 算法在均匀流量的情况下, 各输入端口负载相同, 平均到达速率一致, 不同队列间不存在差别, 因此其平均性能和最大匹配类算法基本相同; 但在突发流量情况下, 特别是存在未激活流量时, 最大匹配类算法和LQF 算法性能差别较大, LQF 算法更加稳定, 理论证明 LQF 算法对于所有容许流量都是稳定的[19, 20].

但是 LQF 算法在硬件实现上相当复杂且运行时间太长, 其复杂度为 $O(N^2\log_d N)$[19], 限制了 LQF 算法在 CR 中的应用. 本文第二部分建立的矩阵模型采用矩阵形式描述了 IQ-VOQ 队列队长参数, 定理 1 给出了利用矩阵迭代求解队长参数的简化方法. 矩阵迭代队长参数求解方法首先省去了输入队列中对 N^2 个队长参数的复杂运算过程, 只需要简单的把 1 比特信元到达信息传递给 Scheduler 即可; 其次由于采用了矩阵迭代求解方法, 可以方便的解决列向量求解后在行向量元素间产生的冲突问题, 从而省去了对行向量进行二次求解的运算量, 进一步简化队长参数的运算复杂度. 图 3 给出了基于 IQ-Crossbar 矩阵模型的 3×3 交换架构 MM-LQF 调度策略工作原理图, 输入端口 IP_1、IP_2、IP_3 简单的把信元到达信息向量 $\vec{A}_1(n)$、$\vec{A}_2(n)$、$\vec{A}_3(n)$ 传递给 Scheduler, Scheduler 完成队长矩阵参数的迭代和匹配矩阵的求解过程, 最后 $\boldsymbol{M}(n)$ 配置 Crossbar Fabric 的对应交叉点, 完成信元交换.

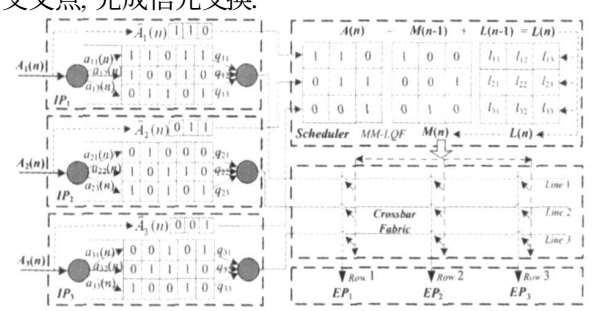

图 3 基于 IQ-Crossbar矩阵模型的 3×3 交换架构 MM-LQF 调度策略

MM-LQF 调度算法描述如下:

Step 1: 令 $n=1, K=1, I=\{1, 2, \cdots, N\}$, $\boldsymbol{L}(0)=\boldsymbol{M}(0)=\boldsymbol{A}(0)=\vec{\boldsymbol{0}}$;

Step 2: 在时隙 n 中有信元到达的每个队列 $q_{ij}(i, j=1, 2, \cdots, N)$ 向 Scheduler 发送请求(Request), 输入端口 $IP_i(i=1, 2, \cdots, N)$ 中的 N 个队列 $q_{ij}(j=1, 2, \cdots, N)$ 的请求形成到达行向量 $\vec{\boldsymbol{A}}_i(n)(i=1, 2, \cdots, N)$;

Step 3: Scheduler 将来自输入端口的 N 个到达行向量 $\vec{\boldsymbol{A}}_i(n)(i=1, 2, \cdots, N)$ 送入到达矩阵 $\boldsymbol{A}(n)$, 然后根据队长矩阵迭代定理 $\boldsymbol{L}(n)=\boldsymbol{L}(n-1)-\boldsymbol{M}(n-1)+\boldsymbol{A}(n)$ 计算当前时隙队长矩阵 $\boldsymbol{L}(n)$;

Step 4: Scheduler 选取队长矩阵 $\boldsymbol{L}(n)$ 中优先级指针 K 指向的列向量中权重最大(即队列最长) 的元素 i_1, 再选取$\{(K+1) \bmod N\}$ 指向的列向量的子集$\{I/i_1\}$ 中权重最大的元素 i_2, 再选取$\{(K+2) \bmod N\}$ 指向的列向量的子集$\{I/(i_1, i_2)\}$ 中权重最大的元素 i_3, 依此类推, 直到队长矩阵中的所有列向量都选出权重最大的元素为止; 这些选出的元素的位置即为 IQ-Crossbar 匹配矩阵 $\boldsymbol{M}(n)$ 非零元素的位置, 用 $\boldsymbol{M}(n)$ 配置 Crossbar 的 Cross-point 对应的开关实现该时隙的信元交换; MM-LQF 算法中由队长矩阵计算匹配矩阵的方法如图 4 所示;

Step 5: 令 $n=n+1, K=(K+1) \bmod N$, 转向(2).

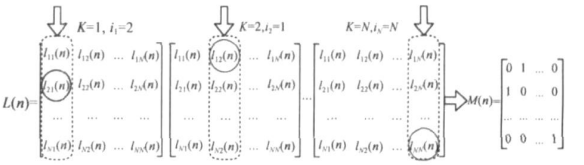

图 4 MM-LQF 调度算法中由队长矩阵计算匹配矩阵

MM-LQF 开销和代价分析: 与 LQF 调度算法相比, MM-LQF 调度算法需要增加一组 VOQ 队长计数器、一组列向量优先级指针 K 状态变量和一个匹配矩阵状态变量. VOQ 队长计数器用来计算迭代后队长矩阵变量, 列向量优先级指针 K 是 MM-LQF 中当前列向量标识变量, 匹配矩阵状态变量保存配置交叉开关的 MM-LQF 迭代结果. 这些计数器和状态变量在现有硬件中实现是非常方便和简单的.

3.2 MM-LQF 调度策略仿真实验结果与分析

我们通过仿真实验对 MM-LQF 算法和 LQF 算法的运算复杂度和平均包延迟等性能进行了分析比较. 实验假设输入端口到达流量分别为贝努利业务源均匀分布、ON-OFF 突发业务源(突发长度 $l=10$) 和贝努利业务源 Diagonal 分布, 输入队列最大缓存深度为 500Cells, 实验结果如下.

图 5 给出了 LQF 与 MM-LQF 算法的运算复杂度比较曲线图, 当交换结构端口数 N 依次从 2 增加到 20 时, LQF 的运算复杂度曲线呈类指数迅速增加, 而 MM-

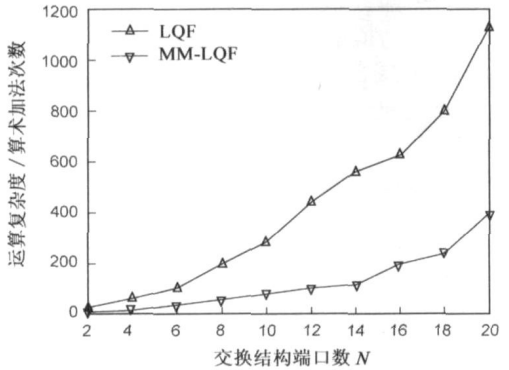

图 5　MM-LQF 与 LQF 在不同端口数 N 下的运算复杂度

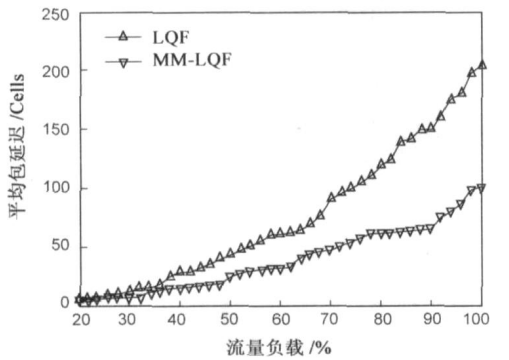

图 6　MM-LQF 与 LQF 贝努利业务源均匀分布的平均包延迟

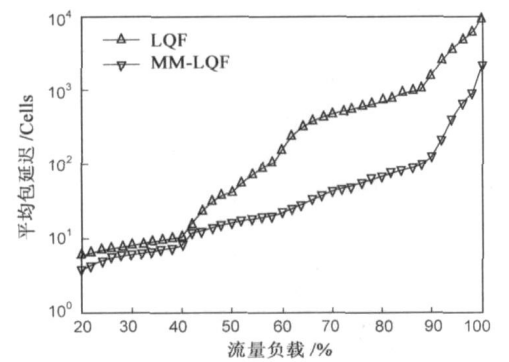

图 7　MM-LQF 与 LQF ON-OFF 突发业务源的平均包延迟 (突发长度为 10)

LQF 的曲线上升非常缓慢. MM-LQF 算法的平均运算效率是 LQF 的 3.72 倍, 并且当信元速率增加到 LQF 门限速率 R 的 2.35 倍时 MM-LQF 算法仍然是稳定的. 这是因为 MM-LQF 省去了 N^2 个队长参数的复杂运算和队长矩阵 N 个行向量的迭代运算, 从而运算效率大大提高, 支持更高的端口门限速率. 图 6、图 7 和图 8 给出了 LQF 和 MM-LQF 不同流量下的平均包延迟对比曲线, 实验采用 16 × 16 的交换结构, 信元到达速率 $r = 0.99R$. 在贝努利业务源均匀分布流量下, MM-LQF 包延迟性能一直优于 LQF; 当流量为大于 80% 重载时 MM-LQF 平均延迟仅为 LQF 的 1/2. 在 ON-OFF 突发业务源情况下, MM-LQF 平均包延迟增加缓慢, 流量负载超过 40% 时性能明显优于 LQF. 对于贝努利业务源 Diagonal 的流量分布, LQF 最大吞吐率仅为 80%, 在流量负载 0.8 至 1.0 时平均包延迟不再收敛; 而 MM-LQF 的最大吞吐率可以达到 100%. 这是因为 MM-LQF 算法中充分考虑了信元到达信息, 因而具有更好的流量适应性, 同时高效率的运算使其可以在一次 LQF 调度的时间内完成至少 2 次的信元交换.

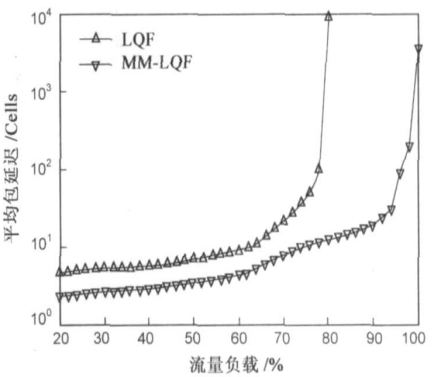

图 8　MM-LQF 与 LQF 贝努利业务源 Diagonal 分布平均包延迟

与 LQF 算法相比, MM-LQF 调度策略的优势在于:

(1) 节省了 N^2 个队长参数的运算量, 只需要通过发送请求(Request), 简单的把 1 比特信元到达信息传递给 Scheduler 即可;

(2) 解决了列向量求解后在行向量元素间产生的冲突问题, 节省了对行向量进行二次求解的运算量;

(3) 其运算效率是 LQF 的 3.72 倍, 支持的端口门限速率是 LQF 的 2.35 倍;

(4) 在贝努利均匀流量重载条件下平均时延仅为 LQF 的 1/2; 在贝努利 Diagonal 流量条件下吞吐率为 100%.

4　结论

本文在分析 IQ-Crossbar 架构特点的基础上构建了矩阵模型, 该模型精确定义了 IQ-VOQ 状态矩阵 $\boldsymbol{Q}(n)$、队长矩阵 $\boldsymbol{L}(n)$ 及其衍生列向量 $\vec{\boldsymbol{L}}(n)$、到达矩阵 $\boldsymbol{A}(n)$ 及其衍生列向量 $\vec{\boldsymbol{A}}(n)$、IQ-Crossbar 匹配矩阵 $\boldsymbol{M}(n)$ 等概念, 通过分析该矩阵模型的工作机理和信元排队策略, 提出并证明了 IQ-Crossbar 架构矩阵模型的两个定理: 队长矩阵迭代定理和状态矩阵迭代定理. 该矩阵模型为分析基于 IQ-Crossbar 的调度算法的工作机理和系统性能提供了理论依据. 本文在分析 LQF 调度算法优缺点的基础上, 提出了基于矩阵模型的 MM-LQF 调度策略, 实验结果表明, MM-LQF 调度策略的运算效率是 LQF 的 3.72 倍, 支持的端口门限速率是 LQF 的 2.35 倍, 在贝努利均匀流量重载条件下平均时延是 LQF 的 1/2; 在贝努利 Diagonal 流量条件下吞吐率为 100%.

参考文献:

[1] M A Bonuccelli, A Urpi. A multicast FCFS output queued switch without speedup[A]. In Second IFIP conference in Networking[C]. Pisa, Italy, 2002. 1057- 1068.

[2] Amit Prakash, Sadia Sharif, Adnan Aziz. An O(log2N) parallel algorithm for output queuing[A]. In Proceedings IEEE INFOCOM[C]. New York, USA, 2002. 127- 136.

[3] N McKeown. Scheduling Algorithms for Input-Queued Cell Switches[D]. PhD Thesis, University of California, Berkeley, California, USA, 1995.

[4] N McKeown, A Mekkittikui, V Anantharam, J Walrand. Achieving 100% throughput in an input-queued switch[J]. in IEEE Transactions on Communication, 1999, 47(8): 1260- 1267.

[5] N McKeown. The iSLIP scheduling algorithm for input-queued switches[J]. IEEE/ACM Transactions on Networking, 1999, 7(2): 188- 201.

[6] N McKeown, V Anantharam, J Walrand. Achieving 100% throughput in an input-queued switch[A]. In IEEE Proceedings of INFOCOM[C]. San Francisco, USA, 1996. 296- 302.

[7] N McKeown, Anderson TE. A quantitative comparison of scheduling algorithms for input-queued switches[J]. Computer Networks and ISDN Systems, 1998, 30(24): 2309- 2326.

[8] A Mekkittikul, N McKeown. A practical scheduling algorithm to achieve 100% throughput in input-queued switches[A]. In IEEE Proceedings of INFOCOM[C]. San Francisco, USA, 1998, 2: 792- 799.

[9] E Altman, Z Liu, R Righter. Scheduling of an input queued switch to achieve maximal throughput[J]. Probability in the Engineering and Informational Sciences, 2000, 14: 327- 334.

[10] C S Chang, W J Chen, H Y Huang. On service guarantees for input buffered crossbar switches: A capacity decomposition approach by Birkhoff and von Neumann[A]. In IEEE IWQoS'99[C]. London, UK, 1999. 79- 86.

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

[12] C S Chang, D S Lee, C M Lien. Load balanced Birkhoff-von Neumann switches Part II: Multi-Stage buffering[J]. Computer Communications, 2002, 25(6): 623- 634.

[13] T Anderson, S Owicki, J Saxe, C Thacker. High speed switch scheduling for local area networks[J]. ACM Trans on Computer Systems, 1993. 319- 352.

[14] J Dai, B Prabhakar. The throughput of data switches with and without speedup[A]. in IEEE Proceedings of INFOCOM[C]. Tel Aviv, Israel, 2000, 556- 564.

[15] I Stoica, H Zhang. Exact emulation of an output queuing switching by a combined input output queuing switch[A]. in 6th IEEE/IFIP IWQoS[C]. Napa, California, 1998. 218- 224.

[16] Shang Tse Chuang, Ashish Goel, Nick McKeown, Balaji Prabhakar. Matching output queueing with a combined input output queued switch[A]. in IEEE Proceedings of INFOCOM[C]. San Francisco, USA, 1998, 17(6): 1030- 1039.

[17] K J Christensen. Design and evaluation of a parallel-polled virtual output queued switch[A]. in Proceedings of the IEEE International Conference on Communications[C]. Helsinki, Finland, 2001. 112- 116.

[18] A Mekkittikul, N McKeown. Scheduling VOQ switches under norr uniform traffic[J]. CSL Technical Report, Stanford University, 1997. 97- 747.

[19] N McKeown. Scheduling algorithms for input-queued cell switches[D]. PhD Thesis, University of California at Berkeley, 1995.

[20] N McKeown, Thomas E Anderson. A quantitative comparison of scheduling algorithms for input-queued switches[J]. Computer Networks and ISDN Systems, 1998, 11: 319- 352.

[21] Rajendra Bhatia. Matrix Analysis[M]. Springer, Verlag, 1996.

[22] Roger A. Hron. Matrix Analysis[M]. Cambridge University Press, 1990.

[23] Gene H Golub, Charles F, Van Loan. Matrix Computations[M]. Johns Hopkins University Press, 1997.

[24] Dario Bini, Victor Y. Pan. Polynomial and Matrix Computations[M]. Birkhauser, 1992.

[25] Richard A Brualdi, Herbert J Ryser. Combinatorial Matrix Theory[M]. Cambridge University Press, 1991.

[26] J S Przemieniecki. Theory of Matrix Structural Analysis[M]. Courier Dover Publications, 1985.

作者简介:

马祥杰 男, 1977 年生于河北省滦南县, 解放军信息工程大学信息工程学院博士研究生. 主要研究方向为高速网络交换结构与调度算法. E-mail: maxiangjie100@163.com

毛军鹏 男, 1975 年生于陕西省咸阳市, 解放军信息工程大学信息工程学院博士研究生. 主要研究方向为网络交换结构拓扑结构、P2P 应用技术.

兰巨龙 男, 1962 年生于河北省张北县, 解放军信息工程大学信息工程学院教授, 博士生导师. 主要研究方向为网络路由理论与技术、交换结构与调度技术.

张百生 男, 1969 年生于河北省迁西县, 解放军 93735 部队高级工程师. 主要研究方向为交换结构与调度技术、软交换技术.