基于输入排队的高速交换调度算法研究

张重洋,申金媛 ,刘润杰,张文英,穆维新
(郑州大学 信息工程学院,河南 郑州 450052)

摘 要:高速交换网络一般采用基于定长信元的交换结构,其性能决定于排队策略和信元调度算法.输入排队策略只有和一个有效的调度算法相结合,才能保证交换结构具有良好的吞吐率和时延等性能.主要阐述了基于 VOQ的最大数量匹配算法,最大权重匹配算法,稳定结合算法,神经网络算法等输入排队调度算法,分别从技术特点,性能指标和实现复杂度等多个方面进行比较和分析.分析了分布式和集中式两大类调度算法的工作方式,并根据各类算法的特点提出,神经网络算法可以通过定义其优先级函数实现其余各类算法.

关键词:输入排队;虚拟输出队列;二部图匹配;调度算法
中图分类号: TP18 **文献标识码**: A **文章编号**: 1673-4785(2008)03-0265-05

A study on scheduling algorithms for high-speed switching networks based on input-queuing

ZHANG Chong-yang, SHEN Jin-yuan, LIU Run-jie, ZHANG Wen-ying, MU Wei-xin
(Institute of Information Engineering, Zhengzhou University, Zhengzhou 450052, China)

Abstract: Most high-speed switching networks adopt a switching fabric with fixed-length cells, and their performance depends heavily upon queuing strategies and the cell scheduling algorithm. Only when the input-queuing strategy is combined with a proper switching algorithm can throughput and time delay for the switching fabric be optimized. This paper mainly discusses virtual output queuing (VOQ)-based algorithms, among them the maximum number matching algorithm, the maximum weight matching algorithm, the stable combination matching algorithm, and the Hopfield neural network (HNN) scheduling algorithm. The mechanisms, performance and implementational complexity of these algorithms are compared and analyzed. The working modes of distributed and centralized scheduling algorithms are analyzed. Finally, from the findings in our research, it is concluded that the HNN algorithm can realize other algorithms by defining a priority function.

Keywords: input-queuing; virtual output queuing; bipartite graph matching; scheduling algorithm

在高速信元交换中,为消除或减小队首(head of line, HOL)阻塞和解决信元对交换线路的竞争造成的丢失问题,必须对信元进行缓冲排队,在时间上将冲突信元分开.根据缓冲队列相对于交换结构的位置,调度算法可分为输入排队(input-queued, IQ)算法、输出排队(output-queued, OQ)算法以及组合输入输出(combined input/output-queued, CIOQ)排队算法三大类.目前研究的重点,应用最多的是输入排队算法.传统的输入排队采用简单的 FIFO(先进先出)机制,因为存在队首阻塞,其吞吐率只有 0.586[1].采用虚拟输出队列(virtual of queuing, VOQ)的输入排队方法不需要更高的加速比,可以消除队首阻塞,并使吞吐率达到 100%[2-3].目前,许多产品都采用基于 Crossbar交换结构的"输入排队+VOQ"排队系统,如 Tiny-Tera太比特路由器原型[4]和 Cisco GSR 12000系列的 IP路由器[5].

1 基本问题与算法

在信元交换中,每一个信元周期开始时,输入端口向调度器请求与相应的输出端口建立连接,调度器根据调度算法确定无冲突的端口配置方案.调度算法所寻求的结果就是在一个信元时隙内通过多次迭代达到输入和输出端口的最优匹配,从而使吞吐率接近 100%.因此调度问题可以看成解决二部图

收稿日期: 2007-11-15.
基金项目:河南省杰出青年基金资助项目(512000400),教育部留学回国人员科研启动基金资助项目.
通讯作者:申金媛. E-mail: jyshen@zzu.edu.cn.

匹配问题 (bipartite graph matching).

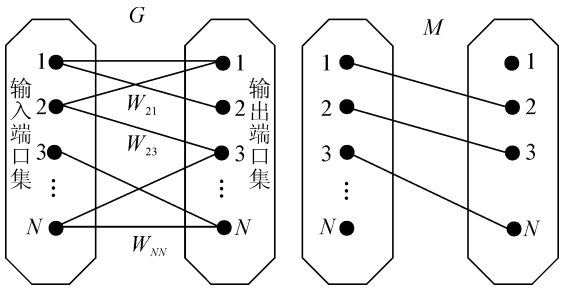


图 1 二部图 G及其匹配图 M

Fig 1 Bipartite graph G and its'matching graph M

如图 1所示, $G = [V, E]$, V为顶点集,包括 2个子集:左侧输入端口集和右侧输出端口集,分别代表输入和输出端口; E为边集,表示从输入端口到输出端口可能的连接边,其权重记为 W_{ij}. 例如:在 PM和 iSLIP中 $W_{ij} = 0$或 1,表示队列 Q_{ij}是否有信元传输;在 iLQF和 iOCF中, W_{ij}表示 Q_{ij}的长度或信元等待时间. 二部图 G的匹配图 M定义为边集 E的子集,且 M中没有 2条边有公共定点. 这就说明一个匹配图满足交换结构的传输限制条件,即在同一时隙内,输入端口传送的信元和输出端口接收的信元个数最多为一个.

实现二部图匹配的算法有最大数量匹配算法 (maximum size matching, MSM)[6-7],最大权重匹配算法 (maximum weight matching, MWM)[8-11],稳定结合算法[12],神经网络算法等. 设计调度算法的基本要求包括:公平性、稳定性、时延小、易于实现等[10].

1.1 最大数量匹配算法

最大数量匹配算法的目标是使匹配的边的数量达到最大. MSM采用 1位的请求信号长度作为边的权重,当某虚拟队列中有信元时,相应的边的权重为 1,否则为 0. 仿真实验[9]表明,在独立均匀和容许的信元流量时可以获得 100%的吞吐量,对于非均匀流量可能出现饿死现象,存在不稳定性和不公平性. 由于实现的复杂性,常用多次迭代的方法近似 MSM算法,其实现的迭代算法有 PM (parallel iterative matching)[2,6], iSLIP (iterative round robin matching with SLIP)[6-7]等.

PM是一种并行迭代匹配时序算法,在一个时隙内多次迭代产生不会冲突的匹配. 每次迭代包括 3个步骤: 1)请求 (Request):每个未匹配的输入端口向有输出要求的未匹配的输出端口发送请求信号; 2)响应 (Grant):一个未匹配的输出端口可能收到多个请求信号,随机地从中选择一个输入端口并向其发送响应信号; 3)接受 (Accept):一个未匹配的输入端口可能收到多个响应信号,随机地从中选择一个输出端口并向其发送接受信号. 每个信元时隙输入输出都重新开始并预设为未匹配,在某次迭代中完成的连接在后面的迭代中继续保持 (即使存在更多的匹配数). 一次迭代的 PM算法吞吐率只能达到 63%,多次迭代可达到 100%.

PM在硬件实现上复杂度比较高,对于一个 16×16的交换结构,要实现 PM算法,就需要将近 64 000个复杂的门器件.

iSLIP每次迭代也是请求、响应、接受 3个步骤. 不同的是,在第 2步的响应中,输出端口收到请求后,从它的固定循环列表里选择当前 Grant指针指向的输入端口,输出端口通知所有输入端口说明其请求是否被响应,当且仅当该响应在第 3步里得到确认接受后, Grant指针才增加 1(模 N)指向下一输入端口;在第 3步接受中,输入端口按照固定循环列表选择响应信号,当某个输出端口被接受时, Accept指针模 N加 1指向下一输出端口. 这种指针更新方法克服了 PM的不公平性,不仅可以预防端口出现饥饿现象,还使响应仲裁器趋向于非同步工作,可快速实现端口间的最大匹配.

实现 iSLIP算法的硬件需要包含 $2N$个仲裁器: N个用于输出端的响应仲裁器, N个用于输入端的接受仲裁器,每个仲裁器有 N个可编程的输入优先级解码器. 实现 iSLIP算法需要的门器件数量比 PM少了一半.

1.2 最大权重匹配算法

最大权重匹配算法的目标是使匹配边的总权重达到最大,除考虑输入队列是否为空外,还要考虑队列长度或排队时间等权重因素,因此请求信号长度由 1位变为多位. 在容许的信元流量下,不论是否均匀,最大权重匹配算法都可以达到 100%的吞吐量. 常用的迭代近似 MWM算法有:最长队列优先算法 (iterative long queue first, iLQF)、等待最久信元优先算法 (iterative old cell first, iOCF)[9-11]、最久端口优先算法 (iterative long port first, iLPF)[3, 6, 10-11]等.

iLQF是 LQF的迭代算法,也包括请求、响应、接受 3个迭代步骤,权重 $W_{ij}(n)$等于输入队列长度 $L_{ij}(n)$. 其输入端发送的请求信号为该队列的长度信息,该请求信号字宽 $2b$,其中 b由输入队列的最大长度 L_{max}决定,即 $2b \leq L_{max}$. iLQF算法有可能出现饿死现象. iOCF是 OCF的迭代算法,算法权重 $W_{ij}(n)$等于输入队列队首信元的等待时间 $T_{ij}(n)$. iOCF算法不会出现饿死现象,因为如果有队首信元得不到服务的话,那么它的权重 $T_{ij}(n)$会一直增加

到得到服务为止. iOCF其余性质和 iLQF一样.

iLPF是 LPF的迭代算法,其权值定义为

$$W_{ij}(n)=\begin{cases}R_i(n)+C_j(n), L_{ij}(n)>0,\\ 0, J_{ij}(n)=0.\end{cases}$$

式中:$L_{ij}(n)$是第 n时隙队列 Q_{ij}的长度. $R_{ij}(n)=\sum_{j=1}^{N}L_{ij}(n)$为输入占有,表示输入端口 i在第 n时隙所有缓冲队列总和; $C_{ij}(n)=\sum_{i=1}^{N}L_{ij}(n)$为输出占有,表示所有要发送到输出端口 j的信元缓冲队列长度总和, iLPF匹配总权重等于所有已匹配输入与输出的总和. iLPF迭代有 2个步骤:1)算法按输入与输出的占有率建立一个排序表; 2)从最大占有端口开始至最小占有端口,通过循环迭代,找到最大容量匹配. 由于在第一步中已经将请求信号按权重排序,因此在第 2步中无需对请求权重进行比较,使得算法容易由硬件实现.

表 1　MSM类与 MWM类算法的综合比较

Table1　The comprehensive comparison of MSM and MWM algorithms

算法	最大吞吐率%	权重	时间复杂度	稳定性	公平性
MSM	100	0/1	$O(N^{2.5})$	-	
PIM	100	0/1	$O(N*\log N)$	-	否
iSLIP	100	0/1	$O(i*N*\log N)$	-	是
MWM	100	W_{ij}	$O(N3*\log N)$	+	
iLQF	100	L_{ij}	$O(i*b*\log N)$	+	否
iOCF	100	T_{ij}	$O(i*b*\log N)$	+	是
iLPF	100	R_i+C_j	$O(N^2)$	+	是

MWM算法需要 $2N$个仲裁器和一个权重寄存器,每个响应仲裁器需要 $2N$个比较器,按信号权重信息给予优先匹配. MWM算法的硬件复杂度要比 iSLIP高.

1.3　稳定结合匹配算法

稳定结合问题[12] (stable marriage matching)最早由 Gale和 Shapley提出,因此该算法称为 GSA (Gale-Shapley algorithm), GSA算法利用输入输出端口定义的优先级清单寻找稳定的输入输出匹配. 如果在没有完成匹配的输入输出端口集合中不能发现一个端口,其优先级比已匹配的端口高,那么就称已完成匹配的输入输出端口是稳定的.

MUCFA (most urgent cell first algorithm)[13]算法利用 GSA和输入输出优先清单寻找稳定结合匹配. 输出端口 j根据队列 Q_{ij}队首信元的紧急值 (urgent value)为输入端口 i赋一个优先值并建立相应的优先清单. 信元的紧急值定义为在仿真的 FIFO-OQ队列中,排在该信元前面的信元的个数. 输入端口 i根据队首信元的紧急值为每个输出端口赋一个优先值并建立相应的优先清单.

在 JPM (joined preferred matching)[14]和 CCF (critical cell first)[15]算法中,输入优先清单的信元分别按等待时间和输出占有排列. 与 MUCFA算法一样, JPM和 CCF算法的输出优先清单的信元按紧急值排列.

LOOFA (lowest output occupancy first algorithm)[16]算法的输入优先清单与 CCF算法相同,输出优先清单按信元等待时间排列, LOOFA可以在传输流级限制每个分组的传输延迟.

表 2　稳定结合类算法综合比较

Table2　The comprehensive comparison of GSAs

算法	输入优先清单	输出优先清单	加速比	时间复杂度
MUCFA	紧急值	紧急值	4	$O(N^2)$
JPM	等待时间	紧急值	2	$O(N^2)$
CCF	输出占有	紧急值	2	$O(N^2)$
LOOFA	输出占有	等待时间	2	$O(N^2)$

表 2列出了稳定结合类算法的一些特性比较. 从表中可以看出,与最大匹配类算法相比,稳定结合算法除需要加速比外,还有以下特点:1)某些输入输出端口的优先清单可能是不完全的,结果有可能导致稳定结合数降低,算法性能下降; 2)在算法中,一个已建立的连接在随后的迭代中可能被拒绝,这是与最大匹配算法的显著区别; 3)目前没有证明稳定结合算法带宽的利用效率及是否会导致饿死现象,但算法会偏重于输入或输出的某一方; 4)稳定结合不一定有最大权重,而最大权重也不一定是稳定结合[17].

1.4　神经网络算法

人工神经网络具有高度的并行处理能力和快速收敛的特性,特别是 HNN (hopfield neural network)擅长解决优化问题,因此可以用 HNN实现信元调度器功能[18-20].

采用的 HNN包含 $N\times N$个神经元,分别对应于 VOQ输入端口 $N\times N$个缓冲器队首信元的服务状态,整个网络的状态可用 $N\times N$矩阵表示. HNN经过迭代收敛到稳定状态时,若矩阵元素 (i, j)为 1,则表示对应的神经元处于激发状态,相应的队首信元被送入交换结构,为 0则表示神经元处于抑制状态,相应的队首信元不被送入交换结构.

神经元的输入输出关系为

$$V = [1 + \exp(-gU)]^{-1}.$$

式中:U为输入,V为输出,g为增益系数.

为实现最优的无阻塞信元传输,确定 HNN在一个时隙内的调度规则为:1)每个输入端口最多发送一个信元;2)选出的信元必须有不同的目的地址;3)优先传送优先级别高的信元;4)使网络快速收敛.根据调度规则写出 HNN的能量函数:

$$E = \frac{A}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\Big(l_{ij} - \sum_{q=1}^{N}V_{iq}\Big)^2 + \frac{B}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1,p\neq i}^{N}\sum_{q=1}^{N}H_{ij,pq}V_{ij}V_{pq} - C\sum_{i=1}^{N}\sum_{j=1}^{N}P(Q_{ij})V_{ij} + D\sum_{i=1}^{N}\sum_{j=1}^{N}V_{ij}(1 - V_{ij}).$$

式中:A、B、C和 D是正常数;V_{ij}是神经元 (i, j) 的输出;I_{ij}是神经元 (i, j)的外部输入,若第 (i, j)个缓存器中有信元等待交换,其值为 1,否则为 0;$H_{ij,pq}$为 2个不同的神经元 (i, j)与 (p, q)之间的连接权重矩阵,若 2个神经元所对应的信元目的地址有冲突,其值为 1,否则为 0;$P(Q_{ij})$是缓存器中等待传输的信元优先级函数.可以推导出 $dE/dt \leq 0$,证明系统能量是随着时间收敛到稳定状态,此时神经元的状态就代表在一个时隙内传送的一组最优或接近最优的无阻塞信元集.

通过仿真实验[20]可以看到,与最大匹配算法性能相近,但神经网络算法硬件复杂度仅为 .用 CMOS VLSI技术实现的 HNN收敛时间只需 100 ns[21],并且处理速度基本不受网络规模扩展的影响.

2 算法比较与结合

MSM、MWM以及稳定结合算法属于分布式调度算法,这种算法的特点是在每个输入输出端口都存在一个仲裁器,由仲裁器决定本端口和哪个端口进行交换,并且各个端口的处理是独立的.比如 PM算法,一个未匹配的输出端口只响应它所接收到的请求信号而不管其他输出端口的响应情况.因此分布式算法的优点就是迭代次数少,计算速度快,但调度方案复杂,硬件实现难度大.分布式算法性能分析见表 1和表 2.

神经网络算法属于集中式调度算法,算法存在一个中央调度器,对所有端口进行集中处理,各端口的参数作为神经网络的输入,输出结果就是所要的最优或接近最优的配置矩阵.神经网络算法的优点是调度方案简单,易于硬件实现.而通过 CMOS VLSI技术实现的神经网络算法完全克服了迭代次数多,计算速度慢的缺点.

神经网络算法的迭代是通过能量函数完成的,能量函数的初始状态相当于分布式算法中的第一个步骤(请求);能量函数等式右端的第 1项和第 3项完成分布式算法中的响应步骤功能,即每个输入端口只传送一个信元,不同的是,分布式算法只响应优先级最高的请求,而神经网络算法从全局的角度选择;等式右端第 2项相当于接收步骤,即每个输出端口只接收一个信元;最后一项则加速了网络运算速度.其中优先级 $P(Q_{ij})$如果采用长队列优先原则,则可实现权重为队列长度 L_{ij}的 iLQF算法;如果采用信元等待时间原则,则可实现 iOCF算法;当然也可以采用多种优先级的有效融合原则.因此,具有灵活优先级定义的神经网络算法更能适应复杂的高速交换系统.

3 结束语

本文讨论了交换技术中调度算法的基本问题,介绍了当前比较重要的几类基于 Crossbar结构的输入排队调度算法,并分别从技术特点,性能指标和实现复杂度等多个方面进行比较和分析.尤其是 HNN算法具有大规模并行处理能力和快速收敛的特性,如果有效地定义其优先级参数,将非常适合大规模复杂高速交换系统.此外,在文献[22-23]中还提出了基于帧(frame)或包(envelope)的输入排队调度算法.其原理是将同一队列中相邻的若干信元组成帧或包,从而增加算法时间,使复杂调度算法的实现成为可能.总之,未来的调度算法既要适应网络带宽和网络规模的快速增长,具有良好的技术性能和扩展性,同时在延迟、公平性和服务多样性等方面也要有很好的保证.

参考文献:

[1] KAROL M, HLUCHYJ M, MORGAN S. Input versus output queuing on a space-division packet switch[J]. IEEE Trans on Communications, 987, 12(35): 1347-1356.

[2] ANDERSON T, OWICKI S, SAXES J, THACKER C. High speed switches scheduling for local area network[J]. ACM Transactions on Computer Systems, 1993, 11(4): 319-352.

[3] MEKKITTIKUL A, MCKEOWN N. A practical scheduling algorithm to achieve 100% throughout in input-queued switches[C]//IEEE INFOCOM'98. San Francisco, CA, 1998: 23-28.

[4] MCKEOWN N, IZZARD M, MEKKITTIKUL A, et al. The tiny-tera: a packet switches core [J]. IEEE Micro Magazine, 1997, 17(1): 26-33.

[5] Cisco Inc. Cisco 12000 series-Internet Router. Product Overview [EB/OL]. [2001-10-10]. http://www.Cisco.

com.

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

[7]MCKEOWN N. A fast switched backplane for a Gigabit switched router[J]. Business Communications Review, 1997, 27(12): 1-30.

[8]HOPCROFT J E, KARP R M. An O(n5/2) algorithm for maximum matching in bipartite graphs[J]. Society for Industrial and Applied Mathematics Comput, 1973(1): 225-231.

[9]MCKEOWN N, MEKKITTIKUL A, ANANTHARAM V, WALRAND J. Achieving 100% throughput in an input-queued switch[J]. IEEE Transaction Communication, 1999, 47(8): 1260-1267.

[10]MCKEOWN N. Scheduling algorithms for input-queued cell switch[D]. Berkeley: University of California at Berkeley, 1995.

[11] TARJAN R E. Data structures and network algorithms[C]//Conference on society for Industrial and Applied Mathematics, Pennsylvania, 1983: 105-109.

[12] GALE D, SHAPLEY L S. College admission and the stability of marriage[J]. American Mathematical Monthly, 1962, 69: 9-15.

[13] PRABHAKAR P, MCKEOWN N. On the speed-up required for combined input and output queued switching[R]. Stanford CSL-TR-97-738, 1997.

[14]STOICA I, ZHANG H. Exact emulation of an output queuing switch by a combined input and output queuing switch[C]//Proceedings of the IEEE IWQoS. Napa: IEEE Communications Society, 1998: 218-224.

[15]CHUANG S T, GOEL A, MCKEOWN N. Matching output queuing with a combined input/output-queued switch[J]. IEEE Journal on Selected Areas in Communications, 1999, 17(6): 1030-1039.

[16]KRISHNA P, PATEL N, CHARNY A, SIMCOE R. On the speed-up required for work-conserving crossbar switches[J]. IEEE Journal on Selected Areas in Communications, 1999, 17(6): 1057-1066.

[17]KAM A C, SIU K Y. Linear-Complexity algorithms for QoS support in input-queued switches with no speedup[J]. IEEE Journal on Selected Areas in Communications, 1999, 17(6): 1040-1056.

[18]ALI M. NGUYEN H. Neural network implementation of an input access scheme in a high speed packet switch[C]//IEEE GLOBECOM. Dallas, USA, 1989: 1192-1196.

[19]张便利,常胜江,李江卫,等. 实现虚拟输出队列调度的神经网络算法[J]. 光电子·激光, 2005, 16: 1316-1320.
ZHANG Bianli, CHANG Shengjiang, LI Jiangwei, et al. A neural network method achieving virtual output queuing scheduling[J]. Journal of Optoelectronics·Laser, 2005, 16: 1316-1320.

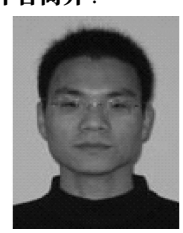
[20]SU X X, CHANG S J, MA T B. A neural network model for traffic prediction in ATM networks[J]. Journal of Optoelectronics·Laser, 2003, 14(8): 842-850.

[21]MARRAKCHI A, TROUDET T. A neural net arbitrator for large crossbar packet-switches[J]. IEEE Transactions on Circuits and Systems, 1989(7): 1039-1041.

[22]BIANCO A, FRANCESCHINIS M, GHISOLFI S, HILL A M, LEONARDI E, NERI F, WEBB R. Frame-based matching algorithms for input-queued switches[J]. IEEE Communications society, 2002, 2(2): 69-76.

[23]KAR K, LAKSHMAN T l. Reduced complexity input buffered switches[EB/OL]. 2007-07-11]. http://www.bell-labs.com/user/stiliadi/publications

作者简介:



张重洋,男,1984年生,硕士研究生,主要研究方向为人工神经网络与通信信息系统.

申金媛,女,1966年生,教授,博士,主要研究方向为人工神经网络、光电通信及信息处理、模式识别. 已在国内外核心学术期刊上发表论文 70余篇,其中被 SCI和 EI收录引用达 60多篇次.