·计算机工程与应用·

通用高速分组交换调度算法

王俊芳，张思东

(北京交通大学电子信息工程学院 北京 海淀区 100044)

【摘要】在iSLIP算法的基础上，应用二部图匹配中对角线数据无竞争的数学原理，采用关联指针的处理方法，提出了一种基于虚拟输出排队(VOQ)缓冲模式下的高速交换调度算法——迭代的关联指针轮转(i-CPRR)算法。该算法简化了指针的轮转方式，降低了设计难度。仿真表明，该算法减少了调度过程中的迭代次数，提高了算法在高负载条件下的带宽利用率，从而降低了交换系统的数据延时和VOQ队列深度，在高速交换系统中具有广泛的应用价值。

关 键 词 i-CPRR算法; iSLIP; 匹配; 分组交换; 调度算法; 虚拟输出排队

中图分类号 TP393.07 文献标识码 A doi:10.3969/j.issn.1001-0548.2010.01.017

High-Speed Packet Switching Scheduling Algorithm

WANG Jun-fang and ZHANG Si-dong

(School of Electronics and Information Engineering, Beijing Jiaotong University Haidian Beijing 100044)

Abstract On the basis of iSLIP (iterative slip) algorithm, a VOQ (virtual output queuing) based high speed switching scheduling algorithm, i-CPRR (iterative-correlatived pointer round-robin) algorithm is presented. In this algorithm, the math principle of uncontested diagonal data in bipartite graphs matching is utilized and the correlative pointer processing method is adopted. This algorithm simplifies the round-robin mode of the pointer and reduces the design difficulty. The simulation results show that the algorithm decreases the iterative times in the scheduling procedure, improves the bandwidth utilization under heavy load, reduces the time delay and the depth of VOQ queue in the switching system. It has wide application prospective in high speed switching systems.

Key words i-CPRR algorithm; iSLIP; matching; packet switching; scheduling algorithm; virtual output queuing

随着Internet的高速发展，高速交换系统中服务质量已成为网络发展的核心技术和热点问题之一。服务质量控制是指网络能够提供有保证的、可控制的、可预测的数据传输服务，满足不同用户的应用需求。对服务质量的要求不仅体现在民用市场，在一些特殊应用场合，如军事、航天等领域的表现更加突出。

分组调度是实现网络服务质量控制的基础，通过控制不同类型的分组对链路带宽的使用，可以使不同的数据流得到不同等级的服务。在IETF提出的综合服务(Int-serv)框架中，保证服务可为单个流提供有严格端到端时延和低分组丢失率的电路型服务，这种服务需要在路由器中实现基于流的加权服务公平调度。目前常用的调度方法多是基于VOQ输入排队的调度算法，交换架构采用Crossbar的交换架构或多平面级联(MPMS)的交换结构[1]。MPMS可以很好地解决交换容量的扩展问题，但其调度问题非常复杂，对于高速率和多端口的MPMS是难以实现的[2]。调度CRRD算法不需要内部加速即可在均匀流量下获得100%的吞吐率，但在非均匀流量下仅能获得63%的吞吐率[3-4]。基于Crossbar交换架构的调度算法很多，以指针滑动多次迭代循环优先级匹配(iSLIP)算法应用最为广泛，但它在应用中仍有许多需改进之处。

基于VOQ输入缓冲的交换调度算法—迭代的关联指针轮转(i-CPRR)算法，属于无权重二部图匹配算法，它充分应用了二部图匹配中对角线上数据无竞争的数学原理，采用一组关联指针对输入输出端

收稿日期：2009－09－12； 修回日期：2009－12－15
基金项目：国家863计划(2007AA701210)
作者简介：王俊芳(1963－)，男，研究员，主要从事通信系统与网络总体以及路由交换技术等方面的研究.

口的授权与接受处理进行整体控制，提高了调度算法的吞吐率，减少了调度所需的时间；并应用Round-Robin的机制，提高了VOQ数据被调度的公平性，使得该调度算法带宽利用率更高、公平性更好、时延特性更小。

1 i-CPRR算法描述

1.1 交换调度算法的数学模型

交换调度算法可以看作是一个二部图匹配的问题[5]，如图1所示。调度算法就是要寻找二部图[*X*,*Y*,*E*]或加权二部图[*X*,*Y*,*E*,*W*]的一个匹配*M*。其中*X*、*Y*分别代表输入输出端口的集合，*E*代表*X*、*Y*之间边的集合，*W*为边权值的集合。

对于二部图[*X*,*Y*,*E*]，若队列Q_{ij}的长度$L_{ij}(n)>0$，则边$e_{ij}\in E$。对于加权二部图[*X*,*Y*,*E*,*W*]，若$L_{ij}(n)>0$，则边$e_{ij}\in E$，其权值为W_{ij}，权值越大，说明其获得服务的优先级越高。选择W_{ij}的主要方法有：$W_{ij}=L_{ij}(n)$，即对应VOQ队列的长度；$W_{ij}=d_{ij}(n)$，即对应VOQ队列的HOL信元的等待时间。

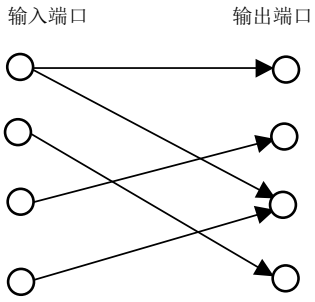

图1 二部图匹配

关于二部图最大匹配问题已有很多的算法，目前最有效的为文献[6]描述的算法，其复杂度为*O*(*n*5/2)。文献[7]提供的仿真试验表明，在每个端口数据包均匀独立到达时，最大匹配算法可以实现100%的吞吐率。虽然该算法能够保证找到一个最大匹配，但运算时间太长，且难以实现。在允许的流量范围内，如果各端口流量不均，最大匹配可能引起不稳定和不公平。在非允许的流量时，可能导致某些端口“饿死”(starvation)，即有的输出端口长时间空闲。

为此，在实际应用中一般采用启发式算法简化计算复杂度，即采用并行处理，多次迭代找到极大匹配；也可以采用串行处理，每次匹配一个端口，多次处理实现。根据处理方式的不同可以分为并行迭代算法与非迭代算法。非迭代算法一般采用串行调度的处理方法，对每个端口进行轮流调度，典型的如ISP算法及其改进算法[8]。并行迭代算法一般采用请求、授权、接受3个步骤来进行调度，通过多步迭代，在输入端口和输出端口之间匹配尽可能多的发送通道。根据3个步骤中采取策略的不同，算法的吞吐率、复杂度、收敛速度也不尽相同，典型算法有如RRM算法[9]、iSLIP算法[9]以及基于iSLIP算法的改进算法如ESLIP算法[5]、LP-iSLIP算法[10]等。其中iSLIP算法最具有代表性。

1.2 iSLIP算法

iSLIP算法即指针滑动多次迭代循环优先级匹配算法，属于迭代算法的一种，是为了克服RRM算法的输出仲裁同步问题而改进的轮转算法，它按照请求、授权、接受3个步骤进行，克服了输出仲裁同步问题，避免了“饿死”现象，能够保证每个端口都可以得到服务。文献[5,9]指出iSLIP算法具有以下特点：

(1) 对于独立同分布业务，iSLIP只需一次迭代就可产生100%的吞吐率。

(2) 每个连接均可得到服务。在第一次迭代中匹配出的连接，在下一个时隙中优先级最低。

(3) 迭代一次的iSLIP算法在大业务负荷之下，发往同一个输出端口的各个输入缓存队列具有相同的吞吐量。

(4) 算法至多迭代*N*次就可以达到收敛。所谓算法收敛，就是在某一轮迭代中，没有匹配出新的连接。计算机仿真表明，该算法在$\log_2 N$次内迭代收敛。

但是iSLIP算法也会由于Round-Robin指针同步导致在高负载的条件下带宽利用率下降[9]，本文对这种特性进行了分析仿真，发现由于iSLIP算法的2*N*个指针变化是互相独立的，因此调度过程中出现多个输出端口的授权指针同时指向同一个输入端口的情况，也存在多个输入接受指针指向同一个输出端口的情况。因此，收敛迭代次数增加，带宽利用率下降。

本文通过对二部图匹配过程中对角线数据无竞争数学原理的研究，找到了一种关联指针处理的模式。在该模式下，所有的指针关联在一起，形成一个关联指针。在每次关联指针变换时，各端口指针同时变化，从而解决了Round-Robin指针同步的问题，即是i-CPRR算法。

1.3 i-CPRR算法定义

i-CPRR算法是一种基于VOQ输入缓冲的并行匹配算法，是迭代的关联指针轮转算法，属于无权重二部图匹配算法。它在iSLIP算法的基础上，利用二部图匹配时对角线上数据无冲突的数学原理，通

过特殊的指针关联技术，对iSLIP算法中的各个端口指针进行优化处理，解决了Round-Robin指针同步的问题。其具体操作流程如下：对于一次匹配，在一个新时隙开始时，所有的输入和输出端口都没有匹配；各个端口的处理优先级与指针通过一种特定的算法关联，在每次匹配之后，只有尚未匹配的端口有资格参加下一轮的匹配，传送通道一旦建立，就不会在后面的迭代中被取消，即使重新匹配能够得到更多的匹配数目。该算法在迭代收敛时或迭代次数达到$N/2$次后停止，迭代完成后，指针自动更新。其操作分以下3个步骤：

(1) 请求。在每个输入端口存在多个缓冲队列，在请求阶段，每个输入端口向其非空的VOQ队列对应的输出端口发送请求，如式(1)的$\boldsymbol{R}$矩阵所示：

$$\boldsymbol{R}=\begin{bmatrix}0&1&1&0\\1&1&1&0\\1&0&1&0\\1&1&0&1\end{bmatrix} \tag{1}$$

其请求被优先级化，输入端口向输出端口上发送具有最高优先级的信元。

(2) 授权。无论输出端口接收到多少请求，它只能选择一个。首先，它根据预先设定的优先级顺序，从输入端口中选择一个端口进行授权，如式(2)的$\boldsymbol{G}$矩阵所示：

$$\boldsymbol{G}=\begin{bmatrix}0&0&0&0\\1&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \tag{2}$$

(3) 接受。如果一个未匹配的输入端口收到多个输出授权，它只能接收一个，并通知这个输出端口，如式(3)的$\boldsymbol{A}$矩阵所示：

$$\boldsymbol{A}=\begin{bmatrix}0&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \tag{3}$$

在步骤(2)中，各个输出端口独立的仲裁器在发出请求的输入端口中有规则地选择一个给予授权，可以达到以下的效果。

(1) 在每一步迭代中，该算法至少匹配2个端口，因此，该算法能够在$N/2$次迭代内收敛到最大匹配结果。统计结果表明，实际收敛速度远小于$\log_2 N$。

(2) 可以确保所有的发送请求最终必然得到授权，因此，没有端口会由于得不到发送机会而阻塞。

(3) 该算法在时间上不是独立的，每N次仲裁各个端口的选择优先级循环一次，因此虽然在每次的调度中，各个端口得到处理的机会是不公平的，但是在连续N次的调度中却是非常公平的，这就减少了由于随机造成的不稳定性，对于输入端业务缓冲区的设置是非常有益的。

2　算法仿真结果与讨论

在仿真时，开关调度算法的分析和比较均采用两种流量模型[8,11]，一种是独立同分布的贝努里过程，一种是burst过程。在该算法仿真时采用了独立同分布的贝努里到达过程，该过程的参数为端口输入负载λ，即每个时间槽信元到达该端口的概率为λ，信元到达各目的输出端口的概率相同。在该条件下，在100万个信元时隙中，对算法的各个特性如迭代次数、迭代收敛条件下，不同业务流量的端口匹配特性、业务带宽利用率、数据时延特性以及VOQ队列的最大值等进行了统计。

2.1　迭代次数

迭代次数是衡量一个算法处理速度的关键指标，迭代次数越少，每次迭代匹配的端口越多，处理速度越快。

i-CPRR算法由于采用了输入输出指针关联的技术，在请求矩阵$\boldsymbol{R}$中的任何一个值，其所在的输入端口与输出端口的优先级次序是相同的，同时任意一个输入端口的请求在所有的输出端口上，其被输出端口处理的优先级各不相同。同样，任意一个输出端口的授权在所有的输入端口上，其被处理的优先级也不相同。因此，该算法迭代次数大大降低，其理论最大迭代次数为$N/2$，比iSLIP算法的N次快了一倍。统计表明在均匀业务下，iSLIP算法最大迭代次数统计平均值小于$\log_2 N$，而i-CPRR算法在不到$\log_2 N$次的一半时就已经收敛。图2给出了i-CPRR算法在不同输入负载条件下，算法收敛时的迭代次数。

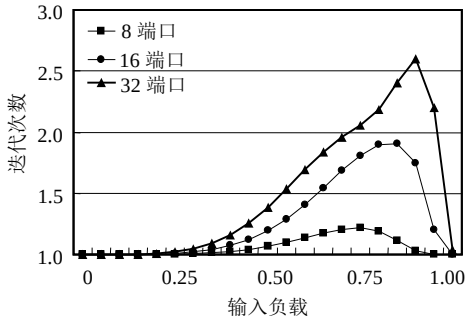

图2　迭代次数与输入负载关系

2.2 带宽利用率

图3给出了带宽利用率和输入负载的关系曲线。结果表明：i-CPRR算法在输入流量为均匀业务且端口负载小于95%时，带宽利用率与输入带宽基本一致；当负载超过95%，即在高负载时也具有较高的带宽匹配特性。这与文献[5,8]的研究相符，即当端口负载较低且输入流量为均匀业务时，所有的调度算法都具有几乎相同的带宽利用率(带宽利用率与输入带宽基本一致)；当端口负载较高时，不同调度算法才具有不同的带宽利用率。而本文的i-CPRR算法在负载超过95%时，带宽利用率也很高。

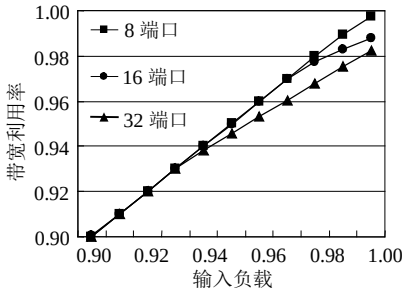

图3 带宽利用率与输入负载关系

2.3 数据时延特性

输入缓冲的交换开关模型中，调度算法主要影响信元在VOQ中的等待延时。由于i-CPRR算法无法精确控制单个信元的等待延时，只能采用统计方法对一段时间内的平均时延特性或最大时延特性进行分析。

i-CPRR算法的公平性比iSLIP算法有了极大的变化，算法保证在N个信元时隙内，每个输入端口的每个VOQ队列的HOL信元最大时延为N，而iSLIP算法为N^2个信元时隙，因此，其时延特性也有极大的改善。

如图4所示，当负载小于75%时，所有输入端口的最大时延只有10个信元时间。在重负载时仍然具有很小的时延特性。因此，该数据表明i-CPRR算法的公平性非常好，时延特性得到有效的控制。

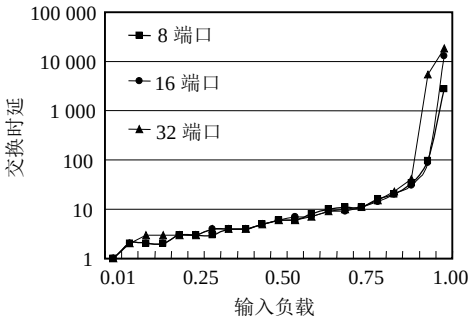

图4 时延特性与输入负载关系

2.4 VOQ队列处理

数据仿真表明，在业务流为均匀业务时，流量小于85%的条件下，VOQ队列最大值不会超过20，再次体现了i-CPRR算法良好的公平特性。图5给出了VOQ队列在各端口输入负载在70%～90%时，VOQ队列的最大深度。

VOQ队列的值为本文设计VOQ输入缓冲提供了统计参考，当采用固定缓冲时，理论上输入端口为每个输出队列分配不低于VOQ的最大值即可。考虑实际应用中有加速比P的设计，在ATM交换中P>1.2(即业务归一化流量最大为83%)，在IP的交换中取2<P<4，此时VOQ队列的最大值不会超过15。在实际应用中，如果再采用反压控制共享缓冲等一些工程应用技术，即使有业务突发，队列设计在缓冲区小于32时就能满足需求。

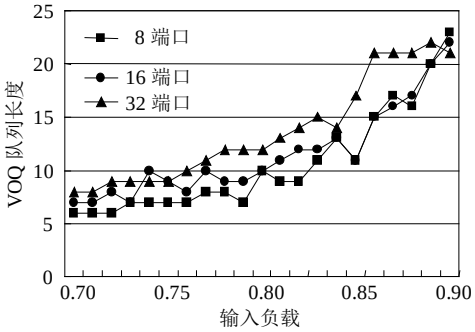

图5 VOQ队列长度与输入负载关系

2.5 算法扩展性

在实际应用中，交换调度必须和信元优先级、组播等诸多因素综合考虑。

在考虑优先级调度时，i-CPRR调度算法采用调度相关的优先级调度策略[5,12]，这样每个端口的VOQ队列就从原来的N个增加到NP个(N为端口数目，P为优先级数目)，并发送NP个请求。调度算法按不同优先级的顺序进行逐一调度。在调度过程中，充分利用该算法在输入端接受匹配后，就不会在后面的处理中更新的特点，采用简单的流水线处理，就能直接设计实现支持优先级的扩展的i-CPRR算法。

工程应用中，以目前主流的FPGA，在时钟频率80 MHz的条件下，采用3优先级、16×16交换矩阵、信元长度64 B，i-CPRR算法可提供40 Gb/s的交换调度，端口速率可达到2.5 Gb/s，满足目前大多数战术级交换设备的需要。

3 结 束 语

本文研究了目前基于VOQ虚拟输出排队调度的

各种算法，在iSLIP算法的基础上，采用关联指针处理技术，设计了一种高效、硬件实现方便的i-CPRR算法，并在均匀业务模型下对其性能进行了仿真。仿真结果表明，该算法解决了高负载条件下iSLIP算法带宽利用率降低的问题，算法迭代次数明显减少，处理能力明显提高，并有效地降低了交换系统的排队时延，具有广泛的应用前景。

参 考 文 献

[1] MA Xiang-jie, LAN Ju-long. Emulating output queueing with the central-stage buffered clos packet switching network[C]//IEEE Conference on High Performance Switching and Routing. [S.l.]: IEEE, 2008: 98-103.

[2] MEKKITTIKUL A, MCKEOWN N. A practical scheduling algorithm to achieve 100% throughput in input-queued switches[C]//Proceedings of the 17th Annual Joint Conference of the IEEE Computer and Communications Societies. San Francisco: IEEE, 1998: 792-799.

[3] CHIUSSI F, GERLA M, SIVARAMAN V. Traffic shaping for end-to-end delay guarantees with EDF scheduling [C]//The 8th International Workshop on Quality of Service. Pittsburgh: IEEE, 2000: 10-18.

[4] 马祥杰, 李秀芹, 兰巨龙, 等. 一种多级多平面分组交换结构中的带宽保证型调度算法[J]. 电子与信息学报, 2009, 31(6): 1475-1478.
MA Xiang-jie, LI Xiu-qin, LAN Ju-long, et al. A novel scheduling scheme with bandwidth guarantees in the multiple-plane and multiple-stage packet switching fabric[J]. Journal of Electronics & Information Technology, 2009, 31(6): 1475-1478.

[5] 朱培栋. 高性能路由器[M]. 北京: 人民邮电出版社, 2005: 55-57.
ZHU Pei-dong. High-performance router[M]. Beijing: The Publishing House of People Post, 2005: 55-57.

[6] HOPCROFT J E, KARP R M. An n5/2 algorithm for maximum matching in bipartite graphs[J]. SIAM Journal on Computing, 1983, 1(2): 225-231.

[7] MCKEOWN N, MEKKITTIKL A, ANANTHARAM V, et al. Achieving 100% throughput in an input-queued switch[J]. IEEE Transactions on Communications, 1999, 47(8): 1260-1267.

[8] 孙志刚. 路由器高速交换开关调度算法的研究与实现[D]. 长沙: 国防科技大学, 2000.
SUN Zhi-gang. Research and implementation of scheduling algorithms in high-speed router switches[D]. Changsha: National University of Defense Technology, 2000.

[9] NICK M. Scheduling algorithms for input-queued switches [D]. Berkeley, California: University of California, 1995.

[10] 李 秋, 戚宇林. 输入排队iSLIP算法的改进与比较[J]. 华北电力大学学报, 2009, 36(2): 106-109.
LI Qiu, QI Yu-lin. Improvement and comparison of input queue iSLIP algorithm[J]. Journal of North China Electric Power University, 2009, 36(2): 106-109.

[11] 马祥杰, 兰巨龙, 毛军鹏, 等. 输入排队Crossbar架构下的流量模型[J]. 电子学报, 2009, 37(1): 170-174.
MA Xiang-jie, LAN Ju-long, MAO Jun-peng, et al. Traffic model for input-queued crossbar fabric[J]. Acta Electronica Sinica, 2009, 37(1): 170-174.

[12] 张 怡, 周 诠, 黎 军. 星上交换系统输入缓存调度算法[J]. 电子与信息学报, 2009, 31(6): 1429-1432.
ZHANG Yi, ZHOU Quan, LI Jun. An input-buffer scheduling algorithm in satellite switching system[J]. Journal of Electronics & Information Technology, 2009, 31(6): 1429-1432.

编 辑 黄 莘