具有最小缓存复杂度的负载均衡交换方法

李 挥[1,2],林良敏[1],黄佳庆[3],王 蔚[3],安辉耀[1,2],伊 鹏[4],汪斌强[4]
(1. 北京大学深圳研究生院集成微系统重点实验室,广东深圳 518055;2. 上海北京大学微电子研究院,上海 201203;
3. 华中科技大学电子与信息工程系,湖北武汉 430074;4. 信息工程大学,国家数字交换系统工程技术研究中心,河南郑州 450002)

摘 要: 对两级自路由交换结构,提出了一种新的分割聚合流的负载均衡方法. 该方法通过群组集线器对输入输出线进行分组以获得统计复用的优点并减少接入控制计算的复杂度,并对输入输出及中间端口进行缓存结构的优化设计以实现分组线速转发并降低缓存的复杂度. 理论分析和仿真结果表明,对于任意允许的流量模式,可以达到 100 %的吞吐率. 与其它负载均衡交换方法相比,本方法具有最低的缓存复杂度 $O(N)$,很小的固定排队延迟 $O(1)$. 这些特性使之在下一代网络中更适合超大规模的分组交换结构.

关键词: 缓存; 负载均衡; 自路由; 大规模交换
中图分类号: TP393, TN256 **文献标识码**: A **文章编号**: 0372-2112 (2009) 11-2367-06

A Load Balancing Scheme of Minimum Buffers for Scalable Switches

LI Hui[1,2], LIN Liang-min[1], HUANG Jia-qing[3], WANG Wei[3], AN Hui-yao[1,2], Yi Peng[4], WANG Bin-qiang[4]
(1. *Key Lab of Integrated Microsystems, Shenzhen Graduate School, Peking University, Shenzhen, Guangdong* 518055, *China*;
2. *Shanghai Research Institute of Microelectronics, Peking University, Shanghai* 201203, *China*;
3. *Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan, Hubei* 430074, *China*;
4. *University of Information Engineering, National Digital Switching Center, Zhengzhou, Henan* 450002, *China*)

Abstract: Load-balanced Split Aggregated Flow (LB-SAF) scheme was proposed for two-stage load-balanced self-routing switching structures. By grouping the signal lines with self-routing concentrators, this scheme obtained statistical multiplex gain and reduced the computing complexity for call admission control; moreover, optimal buffer design at line group of each stage achieved wire-speed forwarding and reduced the buffer complexity. Mathematical analysis and simulations show that it can guarantee 100 % throughput for any admissible traffic pattern. Compared with other previous proposed schemes, LB-SFA has properties such as lower buffer complexity of O(N) and small constant queuing delay of O(1). These advantages make it more suitable for very large scale switching structures in Next Generation Network (NGN).

Key words: buffer; load balancing; self-routing; large scale switch

1 引言

由于内存存取速度的限制,目前大部分分组交换结构只使用输入缓存或结合一些其它结构,比如输出缓存、交叉点缓存. 对于大规模的分组交换,如何有效的调度数据分组以达到高吞吐率、低延迟的目的,成为最近几年研究的主要课题之一. 已提出的集中式分组调度算法 iSLIP[1] 和 DRRM[2],它们本身就是系统规模扩展瓶颈. 因此,Chang 等[3,4]提出了一种不要分组调度器的负载均衡 Birkhoff-von Neumann (LB-BvN) 交换结构,其在所有允许的流量分布模式下都能够达到 100 %的吞吐率. 但是由于其采用 Crossbar 作为基本交换结构,硬件复杂度是 $O(N^2)$. 此外,该交换结构分解算法的复杂度高达 $O(N^{4.5})$,无论是硬件实现或者是计算要求都难以适用于超大规模交换结构.

Tsai 等人[5]提出了一种基于 Banyan 网络[6]的负载均衡方法(LB-BY),采用固定的循环连接配置模式来替代 LB-BvN 方法中的速率分解算法,极大的降低配置和计算复杂度,其在线的交换调度算法的复杂度仅为 $O(1)$. 但是,该方法会导致大最大 $O(N)$ 时隙的排队延迟,且其交换结构的硬件复杂度也是 $O(N^2)$. 类似的还有文献[7,8]中提出的(LB-OP). 文献[9]提出了一种缓存复杂度只为 $(N^{1.5})$ 的动态邮箱共享方法(LB-DMS),由于它采用 Crossbar 结构,故时延和元件复杂度问题仍然存在.

收稿日期:2008-10-18;修回日期:2008-12-05
基金项目:国家 863 计划(No. 2007AA01Z218,2008AA01Z214);国家自然科学基金(No. NSFC:60872010,60872005);上海市重大科技攻关项目(No. 08DZ150010D);广东自然科学基金(No. 8251805704000001)

本文作者提出了一种基于排序集线器的多路径自路由交换结构(MP-SR)[10]具有完全分布式和自路由的特性,不需要 *I/O* 匹配调度算法,没有内部缓存,高度的模块化和及良好的可扩展性. 其缺点是即使在相互独立同分布(i.i.d.)的允许流量模式下,仍然会发生阻塞,不能达到 100 %的吞吐率. 结合两级负载均衡的概念,本文作者在[11]中首先提出级联两级 MP-SR 结构来构造一个负载均衡交换结构,如图 1 所示,但没有解决如何在输入输出及中间各个重要端口实现分组高效缓存的方法.

基于上述研究[10,11],本文提出了一个新的分割聚合流的负载均衡方法(LB-SAF),以及数据缓存机制来处理自路由群组交换的数据缓存转发问题. 在任意允许的流量模式下,跟其它负载均衡方法相比,该方法具有最小的缓存复杂度 $O(N)$,很小的固定排队延迟 $O(1)$,并且不会出现阻塞.

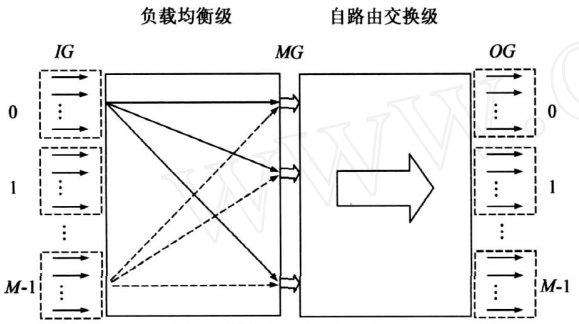

图1 群组交换结构示意图

2 交换结构

交换结构如图 1 所示,其特点是将 N 个输入按顺序均分成 M 个线路的群组,每个群组有 G 根线路,即 $N = M \times G$($N = 2^n$, $M = 2^m$, $G = 2^g$, $M < G$, n, m, g 为正整数),同样地,输出也是采取同样的做法. 为方便起见,让 IG_i(OG_j)代表相应的输入(输出)群组,而 MG_k 表示的是在两级交换结构之间的线路群组(i, j, $k = 0, 1, \cdots, M-1$). 每一级交换结构都是使用排序集线器通过可自路由网络互连而成的. 第一级的功能是将任意允许流量模式变成在第二级 MP-SR 输入处完全均匀的流量,那么这些流量就可以无阻塞的在第二级自路由到目的输出了.

图2 2G-to-G集线器示意图

每两个群组的 2G 根线路如图 2 接入到一个排序集线器中. 排序集线器的作用是可以把 2G 个输入信号中最大(最小)的 G 个信号自路由到具有最大(最小)输出群组地址"1"("0")的 G 个输出端口,并且阻塞掉多于 G 个信号的部分,其详情参考文献[10].

3 负载均衡结构及其算法

假设本交换结构按时隙为单位进行数据交换处理,并且在所有的输入端口数据分组是同步到达的,以使得分组可以在一个时隙内传输. 在本结构中,分组以 G 条线路群组为单位进行交换的,而不是以具体的每条线路,这样可以增大交换的粒度,获得统计复用增益及降低复杂度. 下面介绍相关基本定义并提出分割聚合流的负载均衡方法(LB-SAF).

3.1 可变长度分组负载均衡

定义 1 从同一个输入群组 IG 进来要到同一个输出群组 OG 去的所有分组的集合称为聚合流(AF:Aggregated Flow).

定义 2 以固定的分割长度 Ls 分割聚合流得到的数据再加上标签信息称为交换结构的信元,如图 3(a)所示.

定义 3 将一个信元数据等长度分割成 M 份,信元数据片(Cell Slice)就是其中任意的一份再加上标签信息,如图 3(b)所示.

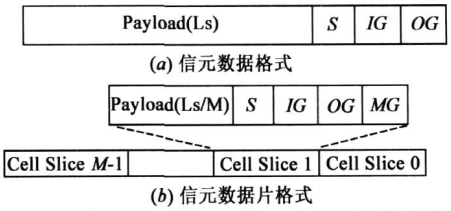

图3 信元和信元数据片以及自路由和顺序号标签格式

本结构到达输入端口的分组长度可以随应用而变. 考虑如图 4 中一个输入群组 IG,在 G 个输入线路要进入交换结构之前,有一个分组聚合分割器(PAS:Pack-

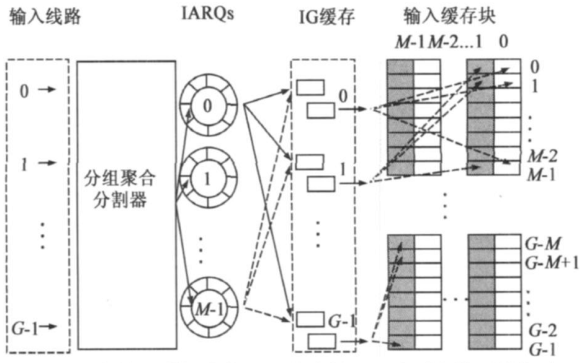

图4 一个输入群组的数据聚合分割环形队列和信元数据片存储示意图(同种填充方式的 M 个小缓存块保存同个信元的数据片)

et Aggregated Splitter)根据当前到达分组的目标输出端口 OG,将 G 个分组分成 M 个聚合流,存到相应的输入聚合环形队列(IARQ:Input Aggregating Ring Queues)中.根据后面将引入的算法,PAS 将切割聚合流,再贴上 IG、OG 和分割顺序号 S 之后,将信元存在相应输入线上的 IG 缓存块中,以便再分成 M 个信元数据片,同时加上 MG 标签信息以传输到相应中间群组,如图 4 所示.在分割后,一个信元将会通过 M 个并行的线路分发,信元数据片就是该交换结构中的自路由单元.在实现时这 G 个输入(出)线路通常就构成了一个输入(出)线卡.

如图 5 所示,在中间级,去往同一个 OG 信元数据片存进同一个 FIFO 队列中,以确保在每个数据片的时间里面,每个中间级群组中只有不超过 G/M 个数据片被并行地传输到第二级的任意一个输出群组,这样就可以保证在第二级 MP-SR 中是无阻塞的.

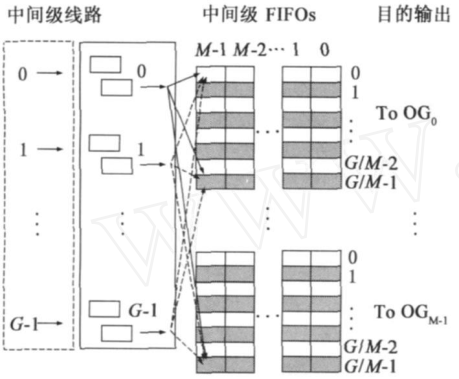

图5　中间级的信元数据片重组装示意图(同种填充方式的 M 个小缓存块组成一个FIFO队列)

在输出群组处,如图 6 所示,所有的数据首先存进缓存模块中,然后信元组装发送器(CAS:Cell Assembly Sender)计算来自同个 IG 的信元数目.根据算法,CAS 把属于同个聚合流的数据存到输出组装环形队列(OARQ:Output Assembly Ring Queues)中的相应位置.在分组完整性检查之后,把分组放到相应输出线的 OG 缓存块中以便发送.

图 4 中的 IG 和图 6 中的 OG 缓存块,是用来准备数据以便在同个时隙将它们同步发送出去的.每个 IG 和 OG 缓存块的大小是 $2Ls$.其中一半即大小 Ls 的存储

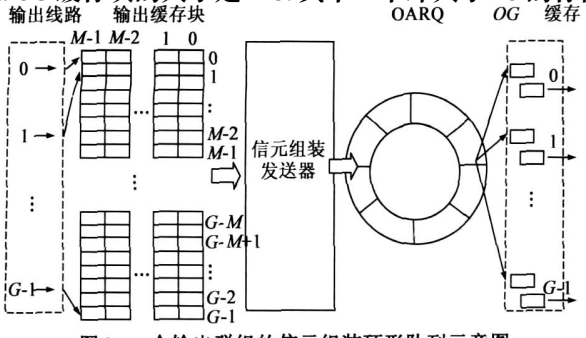

图6　一个输出群组的信元组装环形队列示意图

用于发送数据的时候,另外 Ls 可以缓存到达的数据.在下个时隙,它们的功能互换.这里时隙是指图 3(b)中一个信元的持续时间,由 M 个顺序的数据片时间单元组成.该时间单元就是交换结构的基本自路由数据单元的长度.

总的来说,分组从输入到输出,经以下几个阶段:

(1)到达阶段(arrival phase):新的分组到达 IG_i(i = 0,1,…,M - 1).

(2)聚合分割阶段(aggregating split phase):在每个输入群组 IG_i 的分组聚合分割器 PAS,将对分组进行检查,确定其 OG_j,按照 M 个聚合流 $AF(IG_i, OG_j)$ 来区分,将分组存放到相应的 IARQ 中.在以 Ls 长度分割聚合流之后,信元以轮询循环(Round-Robin)的方式存到 IG 缓存块中,然后 PAS 再切割信元,并将信元数据片并行地存到输入缓存模块中,如图 4 所示.每个 PAS 算法的功能如下:分割顺序标记算法(算法 1)用来确定其 S(用于在输出重组分组);为实现负载均衡,信元切割算法(算法 2)将生成 MG 端口号,用作第一级结构自路由的标签信息.当信元被放到 IARQ 的时候,顺序号 S 和 $IG(OG)$ 标签将被加上.而 MG 标签信息会在信元数据片存到输入缓存模块时加上.其数据结构如图 3 所示.

(3)均衡阶段(balancing phase):根据 MG,信元通过第一级交换结构,被送往相应的中间级群组.

(4)数据片重组装阶段(slices assembling phase):在这一阶段,所有到相同 OG 的信元数据片将被放到 G/M 个相应的 FIFO 队列中并行传输,如图 5.

(5)交换阶段(switching phase):根据 OG_j,信元通过第二级的自路由交换结构,把信元送到目的输出群组.

(6)重组阶段(reassembly phase):基于 IG 和 S,队列存储算法(算法 3)保存信元到 OARQ 中相应的位置,然后 CAS 以轮询的方式,移动完整的分组到相应的 OG 缓存块,以等待下个时隙传输.

(7)离开阶段(departure phase):分组离开 OG_j(j = 0,1,…,M - 1).

分组聚合分割器功能:假设在某一时隙,从输入群组 IG_i 进入交换结构的 G 个分组,有 a_j 个分组要到 OG_j(j = 0,1,…,M - 1).经过分组聚合分割器 PAS,把到同一个 OG_j 的分组,存往相应的输入环形队列 IARQ,然后根据算法 1,PAS 以固定的长度 Ls 分割队列中的数据,如图 7,并计算出 S 标签信息.在加上 S、IG 和 OG 的信息后,信元以轮询的方式被移动到相应的 IG 缓存块中.接着,执行算法 2.

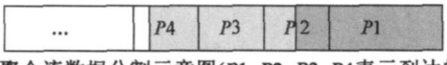

图7　聚合流数据分割示意图($P1$, $P2$, $P3$, $P4$表示到达的长度不同的多个分组,而同种填充方式的部分是长度为Ls的数据块)

信元重组发送器功能:假设在某时隙 t,从输出群组 OG_j 出来的信元数为 G,进入信元重组发送器 CAS,首先,计算每个输入群组 IG_i 来的信元数据片数,根据算法 3,把属于同个聚合流的数据存到相邻的队列位置中,最后去掉所有标签信息,把完整分组存到相应的 OG 缓存块,以便离开.

算法 1 确定从一个聚合流中分割出来的信元序号的,以便在输出重组时使用. 在初始时,$S=0$,在每次从输入环形队列中分割出长度 Ls 的数据后,将 S 标签信息,连同 OG_j 和 IG_i 一起加到 Ls 数据前面,如图 3(a),然后 $S=(S+1) \bmod 2G$,也就是说 S 是一个长度是 $(g+1)$ bits 的数据(因为在重组阶段,输出环形队列是 $2C_S$, $G=2^q$).

算法 2 确定中间级标签信息 MG,以实现负载均衡的功能. 随着信元被切割成 M 个数据片,每个数据片将分别地被顺序加上 0,1, …, $M-1$ 作为 MG 标签. 然后,所有属于同个信元的数据片并行地被存到 M 个具有相同的填充方式的小缓存块中,如图 4 所示.

算法 3 该算法用于根据聚合流的不同,来重组从交换结构出来的数据,以便发送. 假设在某时隙 t,从输出群组 OG_j 出来的信元数据片数为 $G\times M$,其中每个输入群组 IG_i 来的信元数据片数为 a_i(记为 $IG_i(S, MG)$),其中 S 和 MG 是它们对应的标签),根据 IG_i 为索引来确定不同的聚合流 AF,并以顺时针方向,分别地为 $AF(IG_0)$, $AF(IG_1)$, …, $AF(IG_{M-1})$ 在输出环形队列里分配大小为 $(a_i\times L_S)/M$ 的存储空间. 对于某输入群组 IG_i,如果第一个进来的是 $IG_i(S, MG)$,那么以分配空间的首地址开始,将其存放到(S-Smin+MG)处. 接着进来的信元都按照顺序存储,以便进行完整性校验. 如果分组完整,以轮询方式存到相应的 OG 缓存块,以便在下个时隙送走. 否则,就丢弃相应的分组.

3.2 环形队列以及分割长度

输入聚合环形队列(IARQ: Input Aggregating Ring Queue)

在每个输入群组,有 M 个输入环形队列 IARQ,对应 M 个聚合流. 本结构所有的环形队列,包括输出环形队列,都是逻辑上的,以便当前面到达的数据在进行分割处理时,后到达的数据可以循环地存进该队列中. 因而,每个输入环形队列的长度便是一个需要确定的问题,以保证缓存数据的要求,而不会有溢出. 如图 4,可以看到每个输入环形队列与在文献[12]里的光突发交换之前设置的组装队列相似,只是本结构中的输入环形队列分为两半,一半用于数据分割处理,另一半用于数据缓存. 当数据分割处理部分完毕以后,就清除释放空间. 在 IARQ 中采用以下的聚合方法:

(1) 当分组来到时,按顺序往输入环形队列中存储,如果队列的长度超过 L_S(就是说组装阈值长度 $L_T=L_S$),那么立刻发送出去.

(2) 如果超过了预设的最大等待时间(就是组装阈值时间[12] $T_{out}=2T_{slot}-T_p$,其中,T_{slot} 是交换结构的时隙时间长度,T_p 是输入发送处理时间,一般比 T_{slot} 小),那么便把当前队列中的数据填充到长度为 L_S 发送出去.

上述聚合方法与文献[12]中的混合组装(Hybrid Assembly)相似. 其结论是,混合组装的队列长度会随着 T_{out} 的增大而增大,但是不会超过具有相同 L_T 的基于长度阈值的组装队列的最大长度. 当 $L_T/L_{max}=2$ 时(其中 L_{max} 是最大传输单元 MTU 的长度),队列长度在 $2L_{max}$ 以内. 在本结构中,$L_T=L_S$. 因此,IARQ 的长度可以设置为 $L_{IARQ}=nL_{max}>2L_{max}+r\cdot T_p$, 其中 r 是输入群组的数据速率,n 是一个正整数. 整个输入群组所需的缓存为 $M\times L_{IARQ}>M(2L_{max}+r\cdot T_p)$. 这么小容量的缓存(具体大小见后面表 2)便于超高速的实现.

输出组装环形队列(Output Assembly Ring Queue)

用于把属于同个聚合流的数据存到相邻的队列位置中,其大小应保证没有数据溢出. 在一个时隙内,从输出群组出来的全部数据大小至多为 C_S,所以只要将输出环形队列的长度设置为 $L_{OARQ}=kL_S>C_S+r'\cdot T_p'$,其中,$r$ 是输出群组的数据速率,T_p' 是输出发送处理时间,k 是一个正整数. 具体的存储方法如算法 3.

分割长度 L_S(Split Length)

在该方法中,所有的数据以长度为 L_S/M 的信元数据片在交换结构中传输. 因此,在定义 L_S 的具体长度时有两个主要的因素:延迟和增加的开销. 在提出的方法中,由缓存引起的延迟是固定的,并且仅仅是几个时隙的时间,用于接收和转存数据. 由于这个延迟很小,故几乎可以忽略不同 L_S 的影响,故本结构倾向于用大的 L_S 以获得低的开销. 特别地,如果 $L_S>2L_{max}$,那么在每个 L_S 中至少有一个完整的分组. 另外一个问题是,对于每个输出群组来说,其吞吐量不超过 C_S. 故在每个时隙,C_S 要不小于 $\alpha Thp(IG)$(其中 $\alpha>1$ 等效于提速,由增加的开销引起,$Thp(IG)$ 是 IG 的吞吐量). 实际上本方法中,L_S 通常都是不小于 L_{max} 的.

4 性能分析及仿真

4.1 吞吐量分析

定义 4 $A_{i,j}(n)$ 代表至第 n 个时隙,累计到达输入群组 IG_i 去往输出群组 OG_j 的分组个数,令 $A_{i,j}(0)=0$;这样到达过程即为 $\{A_{i,j}(\cdot), i,j=0,1,\cdots,M-1\}$. 同时假设任意时隙内到达输入群组 IG_i 去往输出群组 OG_j 的数据流量为 $\lambda_{i,j}$,若 $\lambda_{i,j}$ 满足以下两式,则称此到达的输入流量模式为允许的.

$$\sum_{j=0}^{M-1}\lambda_{i,j}\le C_S,\ j=0,1,\cdots,M-1 \qquad (1)$$

$$\sum_{i=0}^{M-1}\lambda_{i,j}\le C_S,\ i=0,1,\cdots,M-1 \qquad (2)$$

允许流量模式吞吐率分析

定理 1　对于任意允许流量模式,负载均衡的自路由群组交换可获得 100 %的吞吐率.

证明　假设所有的输入线路均处于满负荷状态,且满足式(1)、(2),到达过程用 $M\times M$ 的流量矩阵 $A(t)$ 表示,其元素 $\lambda_{i,j}$ 代表单位时间内从输入群组 IG_i 去往输出群组 OG_j 的数据流量;到达第 2 级交换模块输入端的流量分布用矩阵 $B(t)$ 表示,其元素 $\mu_{k,j}$ 代表单位时间内从中间群组 MG_k 去往输出群组 OG_j 的数据流量. 取合适的 L_S,使得

$$\sum_{j=0}^{M-1}\lambda_{i,j}=C_S$$

因为在任意 IG 的每个信元的 M 个数据片通过一个二叉树均匀地被分发到 M 个中间群组,而且输入流量是允许的,故对于第一级中任意的集线器要到群组地址"1"("0")的流量均不超过 C_S,这意味着没有超负载. 根据算法 2,经过负载均衡处程,如图 8,有

$$\sum_{j=0}^{M-1}\mu_{k,j}=\sum_{i=0}^{M-1}a_{i,k}=C_S$$

其中 $a_{i,k}$ 是输入群组 IG_i 分配到中间群组 MG_k 的信元数,故对于每个中间群组而言,都没有超负荷.

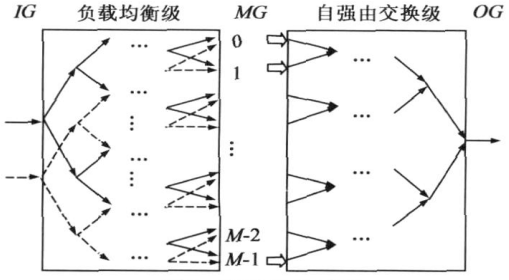

图8　流量分发过程

经过负载均衡级后,第二级交换结构的集线器的输入来自中间群组,具有以下性质:对于所有的 MG,具有相同自路由标签 OG 信元数据片自然地形成一个均匀分布的负载,如图 8 所示. 所以,所有流量可以自路由通过第二级而不会有阻塞. 因此,第二级的所有集线器均没有超负载. 也就是说,两级结构中的任意集线器都不会出现超负载,则整个结构可以达到 100 %的吞吐率.　　证毕.

为了验证定理,采用网络模拟器 NS2[13] 来进行吞吐率性能仿真. 取 $M=16$, $G=64$,在每个输入群组上使用流量生成器产生基于 UDP 协议的固定比特率数据流,此外,分组大小被设置为 $L_S=1.5$ KB. IARQ 长度设置为,$L_{IARQ}=2L_S+G\times L_S=(G+2)\times L_S=99$ KB. 输入群组总共需要的内存大小是 $M\times L_{IARQ}=M(G+2)\times L_S=(N+2M)L_S=1584$ KB. 另外,输入输出处两个缓存模块和中间级 FIFO 所需要的总内存大小是 $3C_S=288$ KB. 故输入群组所有存储(包括 FIFO)是 $(N+3G+2M)L_S=1872$ KB. OARQ 的长度设置为 $L_{OARQ}=3C_S=288$ KB. 故输出群组所有存储(包括 FIFO)是 $6C_S=576$ KB. 而整个交换结构包括输入输出中间级等所有存储器(包括 FIFO) TotalBuf:

$$\begin{aligned}TotalBuf&=M\times\{(N+3G+2M+6G+3G)L_S\}\\&=M\times\{(N+12G+2M)L_S\}\\&=\{(M+12)N+M^2\}L_S\end{aligned}$$

因为 $M<G$,故 $TotalBuf<(M+13)NL_S$,其复杂度就是 $O(N)$.

如图 9 给出了仿真结果. 仿真时间为 5s,总的交换能力固定为 160 Gbps. 从图 9 中可以看到,在可允许流量下,仿真结果与理论分析一致.

4.2　性能比较

与 LB-BvN-IQ[1]、LB-OP[7,8]、LB-BY[5] 和 LB-DMS[9] 相比,LB-SAF 的优势在于时延小和缓存复杂度低,如表 1 所示. 排队时延是端到端通信过程中随机性最大的参数之一,其它交换结构排队时延在很大的范围内变动;而本交换结构该参数几乎是固定的,这使得它更适合用于实时通讯.

表 1　可允许流量下的性能比较

	LB-BvN-IQ	LB-OP	LB-DMS *	LB-BY	LB-SAF
结构	Crossbar	Cross bar	Crossbar	Banyan	MP-SR
排队延迟	$O(N)$	$O(N)$	$O(N)$	$O(N)$	$O(1)$
硬件复杂度	$O(N^2)$	$O(N^2)$	$O(N^2)$	$O(N^2)$	$O(N\log_2 N)$
缓存大小	$O(N^2)$	$O(N^2)$	$O(N^{1.5})$	$O(N2^)$	$O(N)$

* (LB-DMS 的缓存大小是根据其仿真结果估算的

表 2 所示的是不同交换规模下的缓存大小和容量. 目前,路由器制造商通常使用的经验法则是[14]:路由器

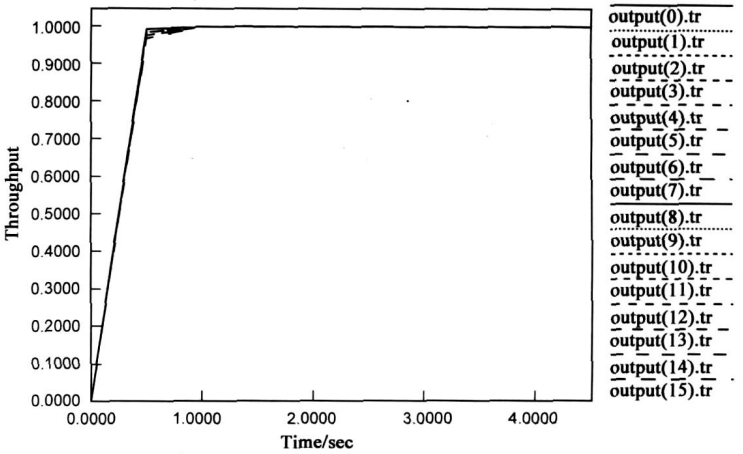

图9　可容许流量情况下G=64，M=16的吞吐量仿真结果

线卡应提供至少相当于一个来回程传送 250ms 时间的缓存. 对 10Gb/ s 线卡提供 250ms 缓存,就需要 2. 5Gbits 的缓存器,即 312. 5MBytes. 表 3 表明本结构比目前行业标准实际采用的缓存降低 1 到 2 个数量级,可以克服当前商用路由器中高速存储器和电源消耗的瓶颈.

表 2 不同交换规模下 LB-SAF 结构的缓存大小和容量

(*M*, *G*)	(16,64)	(32,128)	(64,256)
交换规模 $N = M \times G$	1024	4096	16384
群组吞吐量(Gbps)	10	20	40
交换容量(Gbps)	320	1280	5210
输入群组缓存大小(MB)	1. 872	6. 816	25. 92
中间群组缓存大小(KB)	288	576	1152
输出群组缓存大小(KB)	576	1152	2303
按行业传统规则计算每个线卡(群组)需要的缓存(MB)	2500	5000	10000

(设 $L_S = L_T = L_{max} = 1.5$KB, 交换结构和线路的时钟频率为 156. 25MHz, 一个时隙长度为 $T_{slot} = (8 \times 1.5K)/156.25M = 76.8\mu s$)

表 3 目前行业规则采用的线卡缓存与本结构线卡缓存的比较

(*M*, *G*)	(16,64)	(32,128)	(64,256)
(*M*, *G*)	(16,64)	(32,128)	(64,256)
交换规模 $N = M \times G$	1024	4096	16384
线卡(群组)吞吐量(Gbps)	10	20	40
交换容量(Gbps)	320	1280	5210
A = LB-SAF 结构线卡(群组)需要缓存(MB)	2. 736	8. 544	29. 375
B = 按行业传统规则计算每个线卡(群组)需要的缓存(MB)	312. 5	625	1250
B/ *A* 的缓存之比	114	73	42

5 结论

本研究对两级自路由交换结构,提出了一个新的分割聚合流的负载均衡方法,以及缓存优化设计. 通过排序集线器把信号线路分成群组,该结构增大了交换的粒度,可以获得统计复用的好处并且降低接入控制计算的复杂度. 理论分析和仿真表明,对于任意可允许流量,该方法可以达到 100 %的吞吐率. 与其它以 Crossbar 作为基本交换结构的负载均衡方法相比较,其明显的优点:最低的缓存复杂度 $O(N)$,很小的固定的排队延迟 $O(1)$. 这些特性使之在下一代网络(NGN)中更适合超大规模的交换结构.

参考文献:

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

[2] CHAO H J. Saturn: a terabit packet switch using dual round-robin[J]. IEEE Communications magazine, 2000, 38(12): 78 - 84.

[3] CHANG C S, LEE D, JOU Y. Load balanced birkhoff-von neumann switches. part I: one-stage buffering[J]. Computer Communications, 2002, 25(6): 611 - 622.

[4] CHANG C S, LEE D S, LIEN C M. Load balanced birkhoff-von neumann switches, part II: multi-stage buffering[J]. Computer Communications, 2002, 25(6): 623 - 634.

[5] TSAI Y R, LO C W. Banyan-based architecture for quasi-circuit switching [C]. IEEE ICNS2006, Silicon Valley, USA, 2006. 23 - 28.

[6] 任开新, 顾乃杰, 潘伟, 刘刚. 一种递归构造的合成 BANYAN 网络[J]. 电子学报, 2003, 32(2): 228 - 231.
Ren Kai xin, Gu Naijie, Pan Wei, Liu Gang. The recursively constructed composite banyan network [J]. Acta Electronica Sinica, 2003 32(2): 228 - 231. (in Chinese)

[7] LEE H I, LEE B C, SEO S W. A load balancing scheme for two-stage switches maintaining packet sequence [C]. IEEE ICC2006, Istanbul Turkey, 2006. 293 - 298.

[8] LEE H I, SEO S W. A load balancing scheme for birkhoff-von neumann input-queued switches [C]. IEEE ICC2008, Beijing 2008.

[9] CHENG H, JIN Y H, GAO Y, YU Y D, HU W S. Per-flow resequencing in load-balanced switches by using dynamic mailbox sharing[C]. IEEE ICC2008, Beijing, 2008.

[10] 李挥, 何伟, 伊鹏, 王秉睿, 雷凯, 安辉耀, 汪斌强. 排序集线器多级互连交换结构的多路径自路由模型[J]. 电子学报, 2008, 36(1): 1 - 8.
Li Hui, He Wei, Yi Peng, Wang Bing-rui, Lei Kai, An Hui-yao, Wang Bin-qiang. Modeling multi-path self-routing switching structure from multistage interconnection of sorting concentrators[J]. Acta Electronica Sinica, 2008, 36(1): 1 - 8. (in Chinese)

[11] HE W, LI H, WANG B R, CHEN Q S, YI Y, WANG B Q. Load-balanced multipath self-routing switching structure by concentrators[C]. IEEE ICC2008, Beijing, 2008.

[12] LU Z B, XU Z J, WAN B, ZHANG M, YE P D. Performance analysis of burst assembly under self-similar traffic with measured WAN packet size distribution [C]. IEEE ChinaCom2006, SHANG25-27 Oct. 2006.

[13] The network simulator-NS2 Available: http://www.isi.edu/nsnam/ns/.

[14] WISCHIK D, MCKEOWN N. Part I: buffer sizes for core routers[J]. ACM SIGCOMM Computer Communication Review, 2005, 35(2): 75 - 78.

作者简介:

李 挥 男, 1964 年生于广东汕头, 博士, 北京大学信息学院副教授, 博导. 深圳研究生院集成微系统学与工程重点实验室副主任, 先进网络技术实验室主任, 上海北京大学微电子研究院研究员, 主要研究方向为通信理论与系统, 流媒体理论与系统及其 VLSI 设计.

E-mail: huilihuge @yahoo. com. cn