一种输入排队交换结构的自适应包切分策略

高志江 曾华燊 申志军
(西南交通大学四川省网络通信技术重点实验室 成都 610031)

摘 要 针对传统的输入排队交换结构的数据包切分策略带宽利用率低、灵活性差等缺点,提出一种自适应包切分策略。新策略利用集中式调度的同步特性,在调度过程中通过输入端的队列状态来确定切分单元的大小,并动态调整算法匹配时间,有效地减少了填充字节和系统所需加速比。仿真分析表明,与现有的包切分策略相比,采用自适应策略的交换结构更能适应实际网络环境,且具有良好的时延性能。

关键词 包交换,输入排队,包切分
中图法分类号 TP393 **文献标识码** A

Daptive Packet Segmentation Scheme for Input Queued Switches

GAO Zhi-jiang ZENG Hua-xin SHEN Zhi-jun
(Sichuan Network Communication Technology Key Laboratory, Southwest Jiaotong University, Chengdu 610031, China)

Abstract Focusing on the problem of low bandwidth utilization and poor flexibility of traditional packet segmentation scheme for input queued switches, an adaptive segmentation scheme was proposed. Because of the synchronous properties of centralized scheduling, the new scheme adjusts the length of segmentation unit and the matching time during the scheduling by using the status of input ports, which can reduce the padding data and the speed up that the switch needs. Finally, the simulations demonstrate that the proposed scheme exhibits good delay performance under different traffic models and is well applicable to real network.

Keywords Packet switching, Input queued, Segmentation

1 引言

输入排队(Input Queued, IQ)系统是一种以 Crossbar 交换结构为中心并在其输入端组织分组数据排队的交换结构,如图1所示。由于其具有结构简单且严格无阻塞的特性,因此它广泛应用于目前的各种路由器和交换机中。变长帧的交换处理是输入排队交换结构需要解决的重要问题。由于该结构采用集中式的调度算法[1-3],在调度过程中需要统一执行端口间的匹配和 crossbar 的配置操作,因此所有被算法匹配的端口应在固定的时间内同步进行数据的转发处理。传统的输入排队交换结构是以固定长度的数据为基本单元来进行交换处理的,一般称该基本单元为信元(cell)。而现网中的数据通常以变长帧的方式进行传递,如作为 Internet 主要接入方式的以太网就是一种典型的采用变长帧(64Bytes-1518Bytes)来传送数据的网络。这就要求变长的数据帧在进入输入排队交换结构之前,需要将其按照一定的策略切分为定长的数据单元,然后进行交换处理,最后在交换结构的输出端再将其重组为原始的数据帧进行发送,这一过程称为包切分重组操作。

包切分重组操作的主要目的是使交换结构能够处理变长数据帧,以提高分组的交换效率,其中包的切分策略和传输方式是研究的重点。传统的包切分策略较为简单,但是会带来严重的带宽损失,并需要较大的加速比来保证系统的稳定性。文献[4,5]分别提出了 envelopes 和 cell-merging 的包切分策略,这两种方法是将相邻的数据帧合并起来进行统一的切分处理。这类策略能够有效地提高系统的带宽利用率,减少交换结构所需的加速比,但是对于切分单元的大小都没有作进一步的讨论。事实上,这种切分单元固定的方法灵活性较差,不同的切分大小对交换的性能有着显著的影响。

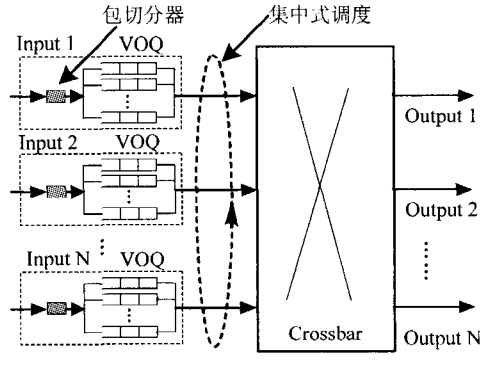


图1 输入排队交换结构

到稿日期:2011-11-13 返修日期:2012-03-19 本文受国家自然科学基金资助项目(60773102),中国工程科技中长期发展战略研究联合基金(U0970122)资助。

高志江(1985—),男,博士生,主要研究方向为交换体系结构,E-mail:santi7009@gmail.com;曾华燊(1945—),男,博士,教授,博士生导师,主要研究方向为网络体系结构、路由器测试技术;申志军(1976—),男,博士生,讲师,主要研究方向为网络与通信技术。

本文提出了一种新的自适应包切分策略。新策略根据输入端的队列长度动态地调整切分单元的大小，能够适应实时变化的业务流量以及不同的分组大小，最大限度地避免了填充字节的产生，有效地提高了系统的带宽利用率，对变化的网络流量具有良好的自适应能力。

2 包切分原理

假设数据包的长度为一个变量 L，切分单元长度为常量 C，称为信元。对于不断变化的 L 值，由于不存在一个有效的固定值 C（C=1Byte 时没有实际意义）总能整除 L，因此在切分过程中往往会出现最后一个切分单元小于 C 的情况，这时需要加入填充字节将其补齐至 C。另外，切分后的信元都要加入信元头(cell header)，以标示该信元在数据帧中的位置等信息[6]。填充字节和信元头的引入，增加了交换结构需要处理的数据量。为了使交换结构能够处理线速到达的数据、消除包切分处理带来的影响，交换结构内部需要一定的加速比，以提高处理能力。假设某数据帧长 L 等于 $C+1$，则该数据帧将被切分为两个长度为 C 的信元，需要 $C-1$ 的填充字节，系统的带宽损失将近 50%。因此，在最坏情况下交换结构需要大约 2 倍的加速比，以处理填充字节对结构的影响，这对交换系统的实现造成了很大压力。

由于输入排队交换结构采用集中式的同步调度算法以及业务数据的流量分布与数据包长都存在着随机性等特点，因此填充字节的出现是无法避免的。如何减少填充字节和系统所需加速比是包切分策略研究的重点。

假设数据帧的到达率服从参数为 λ 的 Poisson 分布，数据帧的长度 L 服从参数为 σ 的指数分布。以 c 为信元长度对数据帧按照传统方法进行简单的切分，则每次到达的数据帧可以被切分为 ceil(L/c)个信元。

设 $f(x)$为数据帧长 L 分布的概率密度函数，交换结构采用传统的、简单的包切分策略，即使用 ceil(L/c)函数的方式对每次进入交换结构的数据帧进行切分操作，若帧长不能被 c 整除，则最后一个切分单元需要用填充字节补齐。设每次切分的信元个数为一随机变量 Y，则有

$$P(N=k)=P((k-1)c<X\leqslant kc)$$
$$=F_X(kc)-F_X[(k-1)c],k\geqslant 1 \quad (1)$$

假设数据帧的长度服从参数为 σ 的指数分布，即

$$f(x)=\begin{cases}\sigma e^{-\sigma x}, & x>0\\ 0, & x\leqslant 0\end{cases}$$

则 Y 的分布律为

$$P(N=k)=1-e^{-\sigma kc}-(1-e^{-\sigma(k-1)c})$$
$$=e^{-\sigma(k-1)c}(1-e^{-\sigma c}) \quad (2)$$

设随机变量$\hat{Y}=Y-1$，则有

$$P(\hat{Y}=k)=P(Y=k+1)=e^{-\sigma kc}(1-e^{-\sigma c}) \quad (3)$$

可以看出，$\hat{Y}$服从几何分布。因此可以得到 Y 的数学期望为

$$E(Y)=E(\hat{Y})+1=\frac{e^{-\sigma c}}{1-e^{-\sigma c}}+1=\frac{1}{1-e^{-\sigma c}} \quad (4)$$

根据期望值可以进一步求出在经过包切分操作后系统为了线速转发数据而在交换结构内部需要的加速比：

$$s=\frac{E(Y)(c+ch)}{E(X)}=\frac{\sigma(c+ch)}{1-e^{-\sigma c}} \quad (5)$$

式中，ch 表示信元头的长度。

表 1 为当业务流数据按不同均值长度均匀到达时，经过简单包切分的交换结构所需的加速比与带宽利用率（c=64Bytes，ch=2Bytes）。可以看出，当平均帧长较小时，系统的带宽利用率很低，因此需要较大的加速比，这是由于此时的切分单元中的有效数据较短，而填充数据占用了很大一部分带宽。随着数据帧长的不断增加，填充数据所占比例将越来越小，对系统的影响也逐渐变小，因此只需较小的加速比就可以以线速率转发数据。

表 1 简单切分策略下系统带宽利用率对比

E(X)	E(Y)	加速比	带宽利用率
64	1.5820	1.6314	61.3%
128	2.5415	1.3105	76.3%
256	4.5208	1.1655	85.8%
512	8.5104	1.0970	91.2%
1024	16.505	1.0638	94.0%

3 自适应包切分策略

3.1 调度过程分析

观察一段输入排队交换的连续 N 次调度，将其记为序列 P，$P=\{P(i)|i=1,\cdots,N\}$。其中，一个完整的调度过程$P(i)$可以分为 3 个阶段：调度仲裁阶段、配置 crossbar 阶段和数据转发阶段，如图 2 所示。将每个阶段花费的时间分别记为 $Ta(i)$、$Tc(i)$和 $Ts(i)$。其中 $Ta(i)$和 $Tc(i)$是由调度算法及其交换结构决定的，对于相同的算法，有

$$\begin{cases}Ta(i)=Ta(j)=Ta\\ Tc(i)=Tc(j)=Tc\end{cases},1\leqslant i,j\leqslant N$$

$Ts(i)$为算法完成第 i 次匹配后的执行时间，反映了交换结构每次从单个输入端发往输出端的数据量。对于传统的以太网包切分策略，$Ts(i)$为交换结构转发 64Bytes 数据所需的时间，对于 cell-merging 切分策略，$Ts(i)$也为一个固定的值。

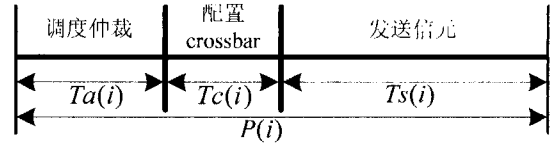


图 2 一个完整的交换调度过程

定理 1 $Ts(i)=Ts(j),1\leqslant i,j\leqslant N$ 是输入排队交换结构运行集中式调度算法的充分不必要条件。

证明：集中式的调度算法需要统一进行端口匹配、配置 crossbar 和数据的传输等。因此各个端口在整个交换调度过程中以同步的方式进行工作。

当 $Ts(i)=Ts(j),1\leqslant i,j\leqslant N$ 时，$P(i)$与 $P(j)$所花费的时间相同，交换结构在每个调度周期内都以同样大小的数据单元(cell)交换数据，交换结构的各个端口同步进行数据的仲裁和交换操作，因此满足集中式调度算法运行的条件。

当 $Ts(i)\neq Ts(j),1\leqslant i,j\leqslant N$ 时，交换结构在 $P(i)$和 $P(j)$这两个调度周期所花费的时间不同，因此在调度过程中所交换的数据单元大小也不相同。但是对于各个端口而言，在同一个调度周期 $P(i)$或 $P(j)$内，被算法匹配的端口之间仍以相同的时间 $Ts(i)$或 $Ts(j)$同步传送数据，因此各个端口在

单个调度周期内传输的数据单元大小相同，整个系统仍然能够以同步的方式进行工作。

证毕。

3.2 自适应包切分策略

目前的包切分策略都采用定长的切分单元，即 $Ts(i)=Ts(j)$。由于进入交换节点的数据包长度并不规律，因此很难找到一个固定大小的切分单元来适应动态变化的流量。切分单元太小会增加交换调度频率，高负载下带宽利用率下降严重；而切分单元太大会造成在处理小数据帧和低负载条件下的系统带宽利用率低下、灵活性较差，后面的仿真结果也说明了这一点。

针对上述问题，本节设计了一种切分单元变长的、能够自适应流量的切分策略。该策略同样采用 cell-merging 的方式将相邻数据帧合并起来进行处理，并以 VOQ 队列的状态信息为基础，动态地调整切分单元的大小，使其能够适应实时变化的业务流和数据帧，最大限度地提高系统的带宽利用率。

自适应包切分策略的核心思想是，在调度过程中，对于端口间已经建立的连接，在保证系统带宽利用率的前提下，尽可能多地执行数据转发操作。具体地，对于一个 N 端口规模的交换结构，假设在某一时刻开始一轮新的调度，在经过算法匹配后最终有 $M(M<N)$ 个 VOQ 队列与输出端口建立了交换通路，其 VOQ 队列表示为 $L^n=(l_1^n,\cdots,l_m^n)$，则该向量中的最小元素 $\min(L^n)$ 即为本次调度的数据单元。这样，被匹配的 M 个 VOQ 同时以 $\min(L^n)$ 的长度作为切分单元的大小，对数据帧进行切分操作并将其转发至输出端。$\min(L^n)$ 的大小与当前网络环境和调度算法有着密切的关系，当 $\min(L^n)$ 过小时，系统需要频繁地进行调度与配置操作，交换效率不高，而切分单元过大将会降低电路利用度并引发公平性问题。

下面讨论切分单元的有效范围。根据文献[7]中关于网络资源利用效率的属性分析，由于电路利用度随着分组长度的增加而降低，而交换效率随着分组长度的增加而提高，因此存在一个数据的分组长度置信区间，也就是自适应切分单元的有效范围使得整个系统保持较高的资源利用率。自适应切分单元的有效范围可以由以下联立方程确定：

$$\begin{cases} P_{loss}\approx(1-\rho)\rho^K \\ T_{\max}=LK/R \\ f=(L-ch)/L \\ F=\rho f \end{cases} \tag{6}$$

式中，P_{loss} 为分组丢失率，K 为输入队列容量，$T_{\max}$ 为最大传递时延，R 为传输速率，f 为有效数据率，ch 为信元头长度，F 为系统资源利用效率，L 则为切分单元长度。

以处理以太网帧为例，设定 $P_{loss}=1\times10^{-3}$，$R=1$Gbps，$T_{\max}=5$ms，假设每个切分单元有 $ch=2$Bytes 的信元头，根据式(6)可以得到切分单元的有效范围，如图 3 所示。

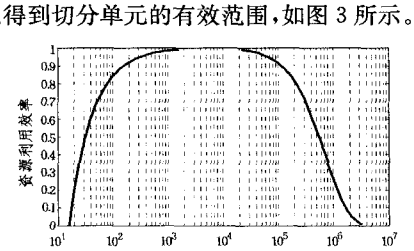


图 3 切分单元与资源利用效率的关系

从图 3 可以看出，在这种情况下，当切分单元在 200bits 至 10^5bits 之间时，系统能够保持 90%以上的资源利用效率。考虑到以太网中的最小帧长为 64Bytes，且其占有较大份额，结合上述的分析，可以将自适应切分单元的有效范围设定为 64Bytes 到 12500Bytes。

4 仿真实验

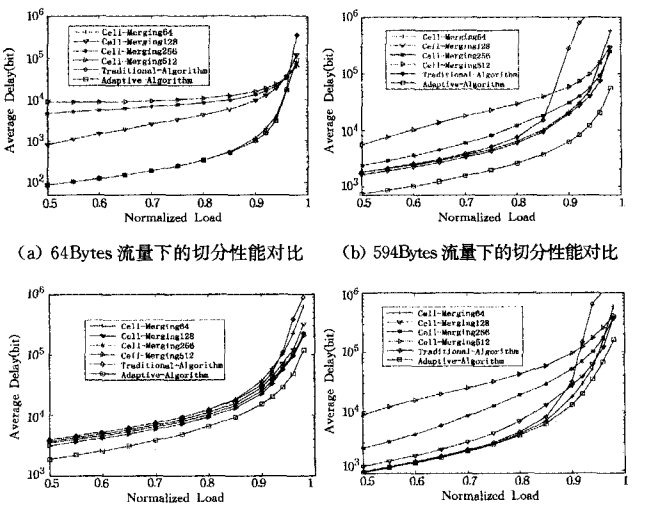
4.1 仿真环境

本节基于 OPNET 仿真软件建立的输入排队交换结构模型，对传统包切分策略、cell-merging 切分策略以及本文提出的自适应包切分策略进行了仿真实验，并对它们的性能进行了比较、分析。实验采用 32×32 的输入排队交换结构，运行 5 次迭代的 iSLIP[3] 调度算法，实验假设每个切分单元引入 2Bytes 的信元头，并以 IMIX 流量[8] 以及组成 IMIX 的 3 种分量为数据源，分别对以上几种切分策略进行仿真实验。

IMIX 流量也叫做 Internet 混合报文，是一种用来模拟经过 Internet 中继设备流量的模型，通常在实验中用来模拟真实的 Internet 流量。IMIX 流量由占有不同比例的 3 种数据帧组成，分别为 64Bytes(58.33%)、594Bytes(33.33%) 和 1518Bytes(8.33%)。对于传统切分策略设定其信元长度为 64Bytes，cell-merging 策略分别采用 64Bytes、128Bytes、256Bytes 和 512Bytes 4 种不同长度的切分单元，自适应策略的切分单元的有效范围设定为 64Bytes 到 12500Bytes 之间。

4.2 仿真结果

图 4 所示为在 4 种不同的流量模型下，采用不同包切分策略的交换结构所表现出的平均时延性能。可以看出，对于传统的切分策略，只有在处理与自己切分单元相同的 64Bytes 流量时，由于在实际上不存在切分的过程，因此其性能良好；而在其余的几种流量模型下性能均表现不佳，尤其当负载较高时，其时延性能急速恶化，这是由于它没有任何减少填充字节的措施，因此系统需要处理很多的信元头和填充数据，浪费了较多的带宽。



(a) 64Bytes 流量下的切分性能对比　(b) 594Bytes 流量下的切分性能对比

(c) 1518Bytes 流量下的切分性能对比　(d) IMIX 流量下的切分性能对比

图 4 不同流量模型下的包切分策略性能对比

采用了 4 种不同切分单元大小的 cell-merging 策略在不同的流量模型下的性能表现各不相同。如图 4(a)所示，改进

策略在处理64Bytes的流量时使用64Bytes的切分单元其性能最佳，而使用其他3种切分单元时，在低负载下其时延较大。这是因为对于这种小数据帧流量，在低负载条件下，由于数据量和数据帧长都较小，采用大的切分单元必定会引入较大的填充字节，从而导致带宽的浪费；只有在高负载下，当输入端产生一定的数据堆积后才会表现出合帧切分的优势。图4(b)所示为当输入流量的帧长提高到594Bytes时的情况，可以看出，采用128Bytes的切分单元性能最优；而在图4(c)中，对于1518Bytes的流量则采用512Bytes的切分单元有较好的性能。因此采用固定切分单元的cell-merging策略，无法选择一个固定的切分单元来适应不断变化的业务流量，而本文提出的自适应策略，则能够根据输入端流量动态地选择合适的切分单元，在各种流量下都能保持良好的性能。图4(d)为模拟真实Internet流量的IMIX模型下各种切分策略的性能对比。可以看出，自适应算法的性能仍然最佳。

结束语 本文首先介绍了输入排队结构的包切分原理，分析了传统切分策略的不足；然后针对传统包切分策略带宽利用率低且需要较大加速比以及cell-merging策略存在的切分单元灵活性较差等问题，提出了一种自适应包切分策略。该策略利用交换结构输入端的队列状态调整在每个调度周期中的切分单元大小，有效地提高了系统的带宽利用率。仿真实验表明，采用自适应切分策略的交换结构能够适应各种输入业务流，可显著提高数据帧的切分效率，且具有良好的时延性能。

参 考 文 献

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

[2] Beldianu S F, Rojas-Cessa R, Oki E. Scheduling for Input-Queued Packet Switches by a Re-configurable Parallel Match Evaluator[J]. IEEE communications on letters, 2010, 14(4): 357-359

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

[4] Kar K, Lakshman T V, Stiliadis D, et al. Reduced Complexity Input Buffered Switches[C]// Proc. HOT Interconnects VIII. Stanford University, Stanford, CA, August 2000

[5] Christensen K, Yoshigoe K, Roginsky A, et al. Performance Evaluation of Packet-to-Cell Segmentation Schemes in Input Buffered Packet Switches[C]// Proc. IEEE Int. Conf. On Communications (ICC'2004). Paris, France, 2004: 1097-1102

[6] Zoran M C. FPGA Implementation of IP Packet Segmentation and Reassembly in Internet Router[J]. SERBIAN Journal of electrical engineering, 2009, 6(3): 399-407

[7] 孙玉. 电信网络总体概念讨论[M]. 北京：人民邮电出版社，2007

[8] Agilent Technologies. JTC 003: Mixed Packet Size Throughput [EB/OL]. http://advanced. comms. agilent. com/n2x/doces/journal/JTC-003. html

(上接第96页)

结束语 本文对移动Ad hoc网络中的现有组播路由协议进行比较和分析后，选择控制开销小、有效性高、扩展性好的ADMR协议作为深入研究的对象。针对ADMR协议对网络移动性适应能力差的问题，引入了节点相对移动性概念，提出一种改进的基于节点相对移动性的自适应组播路由协RMNAM。RMNAM协议完全继承了ADMR协议的按需特性，并引入节点相对移动性度量作为组播转发树路径选择和源节点传输方式切换的重要因素，以满足移动Ad hoc网络对路由协议的高适应性要求。仿真实验结果证明，RMNAM协议在分组递交率与端到端时延方面优于ADMR，且与ODMRP相比，能更好地适应组播规模的扩展和网络负载的增加。

参 考 文 献

[1] Perkins C E. Ad Hoc Networking [M]. London: Addison Wesley, 2001: 12-16

[2] Toh C K. Ad Hoc Mobile Wireless Networks [M]. New Jersey: Prentice Hall PTR, 2002: 6-19

[3] The Wireless Networks Laboratory at Cornell University [OL]. http://people. ecc. connell. edu/-haas/wnl/

[4] The UCLA Wireless Adaptive Mobility Laboratory [OL]. http://www. cs. ucla. edu/NRL/Wireless/

[5] Jorjeta A, Jetcheva G, Johnson D B[OL]. http://tools. ietf. org/id/draft-jetcheva-manet-admr-00. txt

[6] Jetcheva J G, Johnson D B. Adaptive Demand-Driven Multicast Routing in Multi-hop Wireless Ad Hoc Networks[C]// ACM MobiHoc 2001. Long Beach, CA, USA, 2001

[7] Shurdi O, Miho R, Kamo B, et al. Performance Analysis of Multicast Routing Protocols MAODV, ODMRP and ADMR for MANETs[C]// 2011 International Conference on Network-Based Information Systems. 2011: 596-601

[8] Rezaei B A, Sarshar N, Roychowdhury V P. Distributed resource sharing in low-latency wireless ad hoc networks [J]. IEEE/ACM Transactions on Networking, 2010, 1(18): 190-201

[9] Shpungin H, Segal M. Near Optimal Multicriteria Spanner Constructions in Wireless Ad hoc Networks [C]// IEEE INFOCOM 2009. 2009: 163-171

[10] Mohamed Y A, Abdullah A B. Immune inspired framework for ad hoc network security [C]// Control and Automation, IEEE International Conference. 2009: 297-302

[11] Kong Ruo-shan. The Simulation for Network Mobility Based on NS2 [C]// International Conference on Computer Science and Software Engineering. 2008: 1070-1074