区分服务中的分组标记策略研究*

马小骏, 严 俊, 顾冠群

(东南大学 计算机科学与工程系,江苏 南京 210096);
(教育部 计算机网络和信息集成支撑技术重点实验室,江苏 南京 210096)
E-mail: xjma@seu.edu.cn
http://www.seu.edu.cn

摘要: 区分服务是近两年的一个研究热点,其目的是为用户提供较大粒度的服务质量(QoS)保证. 与集成服务模型相比,区分服务不仅具有良好的可扩展性,而且更容易在传统IP分组交换网络上实现. 研究了区分服务的一个关键机制——分组标记策略,并提出了一个具有公平性的分组标记算法(fair marking packet algorithm,简称FMPA),通过仿真把该算法与现有的等比例的标记算法进行比较. 此外,还提出了一种分组再标记(remarking)算法,该算法可尽可能地维护分组原有的服务语义. 通过仿真对该算法进行了验证.

关键词: 区分服务;集成服务;分组标记;分组再标记;服务质量
中图法分类号: TP393 **文献标识码**: A

区分服务(DiffServ)[1,2]是近两年的一个研究热点,其目的是在传统的IP分组交换网络上为用户提供较大粒度的服务质量(QoS)保证. 与集成服务(IntServ)模型[3,4]相比,区分服务的最大优势在于其良好的可扩展性.

DiffServ的基本思想是把基于单个流(per-flow)或聚集流的管理工作分布到区分服务域(DiffServ domain,DS域)[1](DS域由一组相邻的属于同一行政管理域并能提供区分服务的路由器组成)边缘的入口/出口结点,由边缘结点根据一定的策略对流入/流出分组进行计量、监控和标记. 当DS域的内部路由器发生(或可能发生)拥塞时,内部路由器根据分组中的标记域内容选择优先丢弃哪些分组,标记域类似于ATM中的信元丢弃优先级(cell-loss priority,简称CLP). 因此,DS域的内部路由器无须维护基于per-flow的各种状态,保持了其实现的简单性,在维持较高吞吐率的同时又能提供不同类别的. 可见,与传统因特网仅提供BES(best-effort service)相比,DiffServ需要解决两个关键问题,即分组的标记策略和丢弃策略. 本文讨论分组标记策略问题.

本文第1节回顾当前已提出的一些分组标记策略. 第2节提出了具有公平性的新的标记算法. 第3节通过仿真验证并改进了该算法. 第4节讨论了中继网络的分组再标记问题,并提出了我们的解决方案. 最后为结论.

1 相关的工作

分组标记通常可以采用两种策略,一种是令牌桶的方法,入口/出口边界结点以固定的速率产

* 收稿日期: 2000-02-29; 修改日期: 2000-06-12
基金项目: 国家自然科学基金资助项目(69873008,69896249);东南大学计算机网络与信息集成教育部重点实验室资助项目

作者简介: 马小骏(1971-),男,江苏武进人,博士生,主要研究领域为高性能网络协议及性能分析;严俊(1976-),男,江苏江阴人,博士生,主要研究领域为高速、高性能网络及协议;顾冠群(1940-),男,江苏常州人,教授,博士生导师,中国工程院院士,主要研究领域为CIMS,高性能网络体系结构.

生令牌,对于每一个到达的分组,如果桶中有足够的令牌,则把该分组标记为"IN",否则标记为"OUT",其中"IN"和"OUT"指示了当链路发生拥塞时该分组丢弃的优先级高低,"IN"分组具有低的丢弃优先级,"OUT"分组具有高的丢弃优先级[5]. Stoica 等人[5]提出的 LIRA 策略以及 Nichols[7]提出的"Two-Bit"区分服务体系结构中都采用了这样一种策略.

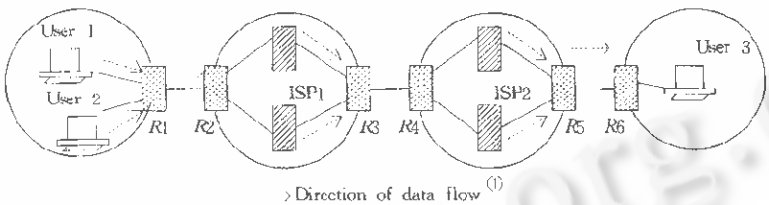
另一种方法则是基于时间滑动窗口机制来估计分组到达的速率.当该速率超过规定阈值时,边界结点按照某一概率把到达分组标记为"OUT",否则,标记为"IN". Clark 等人[5]提出的 TSW 分组标记算法以及 Feng 等人[7]提出的一种适应性的分组标记策略度属于这一类.

现有的分组标记算法的基本思想可以用下面的式(1)表示:

$$P_{OUT}(t)=(V_{AG}(t)-V_{RESV})/V_{AG}(t), \quad (1)$$

其中 $P_{OUT}(t)$表示时刻 t 分组被标记为"OUT"的概率,$V_{AG}(t)$为 t 时刻聚集流分组到达边界结点的平均速率,V_{RESV}为预留带宽(又称有效带宽或目标速率),即"IN"分组流入/流出速率的阈值.

上面的分组标记算法都是等比例地标记分组,即无论分组来自哪个数据源,只要分组到达速率高于规定的阈值,就都按照同一概率 $P_{OUT}(t)$把到达分组标记为"OUT".这样将使得某些具有带宽贪嗜的数据源可以得到更好的服务,而遵守规矩的数据源则无法得到应有的服务.如图 1 所示,假设路由器 $R1$ 的出口预留带宽为 1Mbps,用户 1 和用户 2 分别享用 0.5Mbps 的带宽,如果用户 1 和用户 2 的发送速率分别为 0.8Mbps 和 0.4Mbps,此时聚集速率为 1.2Mbps,大于预留带宽,用户 1 和用户 2 经过边界结点 $R1$ 的分组被标记为"OUT"的概率均为 0.167,显然,这对于用户 2 是不公平的,因为他的发送速率仅为 0.4Mbps,小于阈值 0.5Mbps.为了维持公平性*,我们需要有一种新的算法以解决这一问题.



①数据流传输方向.

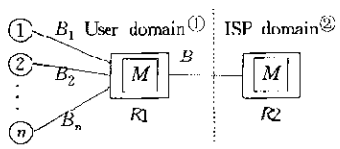
Fig. 1 Differentiated service model

图1 区分服务模型

2 新的分组标记算法

如图 2 所示,用户域向 ISP 申请的预留接入带宽为 B,M 为标记器.$R2$ 处的标记器的主要任务是对来自用户域的分组进行监控,以维持流入的"IN"分组符合约定.

用户域中有 n 个数据源(每个数据源也可以是一个子网),路由器 $R1$ 分配给各数据源的预留带宽为 B_i(预留带宽是指用于标记"IN"分组的带宽,实际使用带宽由于包括"OUT"分组,因此可以大于此值).各数据源在 t 时刻的实际发送速率



①用户域,②ISP域.

Fig. 2 Packet marking at edge node

图2 边界结点的分组标记

* "公平性"并不意味着"要求得越多,担当的丢弃风险就越要大".只有当数据源的速率超过预留带宽时,对于超出部分的分组,才会出现"超出越多,被丢弃得可能性就越大"的情况.

为 $V_i(t)$,$V_{AG}(t)$为经过 $R1$ 的聚集速率,于是有式(2)成立:

$$V_{AG}(t)=\sum_{i=1}^{n}V_i(t). \tag{2}$$

我们的标记算法见算法 1.

算法 1. 具有公平性的标记算法 FMPA(fair marking packet algorithm).

```
For 每个到达分组,计算聚集流平均速率 V_AG(t)
  If (V_AG(t)<B)
    标记该分组为 IN
  else
    计算该分组所属数据源的平均速率 V_i(t)
    if (V_i(t)>B_i)
      计算该分组被标记为 OUT 的概率 P_OUT_i(t)
      按概率 P_OUT_i(t)标记该分组为 OUT
      (该分组被标记为 IN 的概率为 1-P_OUT_i(t))
    else
      标记该分组为 IN
    endif
  endif
End
```

我们把这种具有公平性的标记算法简称为 FMPA. 在 FMPA 中,我们仍然采用滑动时间窗口来估计各数据源的发送速率 $V_i(t)$[5],$P_{OUT_i}(t)$由式(3)计算得出.

$$P_{OUT\_i}(t)=(V_i(t)-B_i)/V_i(t). \tag{3}$$

当数据源 i 的发送速率超过预留带宽并且聚集流速率超过阈值时,我们并没有立即以概率 1 把该数据源随后的分组标记为"OUT",而是以 $P_{OUT_i}(t)$的概率来标记随后的分组,这样可以防止震荡,即避免出现连续分组被标记为"OUT"的情况. 我们分别用 N_i 和 N 表示,在时间段 τ 内,数据源 i 发送分组以及聚集流分组中被标识为"OUT"的数目,N_i 和 N 可以用式(4)和式(5)来表示.

$$N_i(\tau)=\int_t^{t+\tau}P_{OUT\_i}(t)V_i(t)\mathrm{d}t, \tag{4}$$

$$N(\tau)=\sum_{i=1}^{n}N_i(\tau)=\int_t^{t+\tau}\mathrm{MAX}(V_{AG}(t)-B,0)\mathrm{d}t. \tag{5}$$

3 仿真及算法改进

为了验证我们的算法所能达到的公平性,我们在 NS[8]平台上进行了仿真实验,并与等比例的标记算法进行了比较. 图 3 为仿真拓扑,S_i 和 D_i 分别为数据源和目的端,路由器 $R1$ 与 $R2$ 之间的带宽为 33Mbps,但我们仅预留 8Mbps 用于标记"IN"分组,其余 25Mbps 用来传输"OUT"分组,$R2$ 与 $R3$ 之间的带宽为 10Mbps,$R1$ 分配给每个 S_i 的预留带宽为 2Mbps. 各路由器均采用 RIO[5]分组丢弃策略,"IN"分组的 RIO 参数为(60,100,0.02),"OUT"分组的 RIO 参数为(30,50,0.2). $S_1 \ldots S_4$ 分别以固定速率向 $D_1 \ldots D_4$ 发送数据,见表 1,我们选用了 3 组数据.

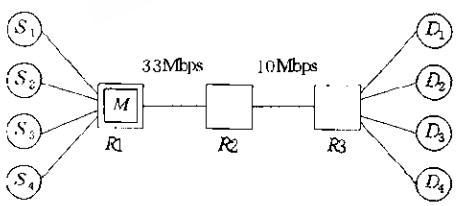


Fig. 3 Simulated topology
图3 仿真拓扑

图 4 显示了两种标记算法分别采用 3 组数据进行仿真所得到的结果,图中横坐标上的数字表示不同的

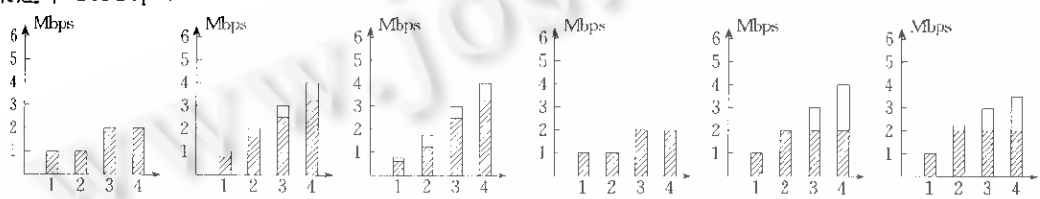
数据源,纵坐标表示数据源发送的分组中被标记为"IN"和"OUT"的平均吞吐率.从中可以看出,当聚集速率小于总预留带宽 B(8Mbps)时,两种标记算法的结果完全相同.当聚集速率大于总预留带宽 B 时(第 2 和第 3 组数据),尽管 S_1 的发送速率不超过 2Mbps,但在等比例标记算法中,仍有部分分组被标记为"OUT".而当采用了 FMPA 之后,只有超过阈值的数据源(如 S_3,S_4 等)才有部分分组被标记为"OUT".

Table 1 (Mbps)
表 1 (Mbps)

	S_1	S_2	S_3	S_4
Group 1①	1	1	2	2
Group 2②	1	2	3	4
Group 3③	1	3	5	7

①第 1 组数据,②第 2 组数据,③第 3 组数据.

可以看出,在使用第 1、2 组数据进行仿真时,无论采用哪种标记算法,均没有分组(包括"OUT"分组)被丢弃,这是由于聚集速率均没有超过链路带宽的缘故.当使用第 3 组数据仿真时,有部分分组被丢弃,因为网络中出现了瓶颈,$R2$ 和 $R3$ 之间的链路带宽为 10Mbps,小于数据源的聚集速率 16Mbps.



(a) Results for proportional packet marking algorithm
(a) 等比例标记算法仿真结果

(b) Simulation results for FMPA
(b) FMPA仿真结果

□ Out packet① ▨ In packet②

①Out分组,②In分组.

Fig. 4 Simulation results for two packet marking algorithms
图4 两种标记算法的仿真结果

我们从第 2 和第 3 组数据的仿真结果发现,采用 FMPA 之后,标记为"IN"分组的总吞吐率反而比采用等比例标记算法时要小.通过分析不难看出,在 FMPA 中,当聚集速率超过总预留带宽时,发送速率高于相应预留带宽的数据源的"IN"分组吞吐率均被限制为阈值 B_i(如第 2、3 组数据中的 S_3,S_4),而发送速率低于相应预留带宽的数据源的分组尽管全部被标记为"IN"(如第 2、3 组数据中的 S_1),但其剩余带宽没有被有效利用,从而导致整个聚集流被标记为"IN"的分组的总吞吐率小于阈值 B,并且如果低速的数据源越多,资源利用率越低.

我们可以改进 FMPA,在任一时刻 t,我们把所有数据源分为两类,一类为数据发送速率超过预留带宽 B_i 的数据源,另一类为数据发送速率未超过预留带宽 B_j 的数据源,$V_i(t)$,$U_j(t)$ 分别表示前后两类数据源的发送速率,K_1,K_2 分别表示两类数据源的个数.在后一类数据源中可以被继续利用的总带宽 B_{un} 为 $V_i(t)$.

$$B_{un}(t)=\sum_{j=1}^{k_2}(B_j-U_j(t)). \tag{6}$$

我们希望把 B_{un} 分配给前一类数据源,使得前一类数据源的"IN"分组数可以适当地超过相应的阈值.结合式(3),可以得到式(7).在 FMPA 中,我们用式(7)来标记分组得到改进后的 FMPA.

$$P_{OUT\_i}(t)=\mathrm{MAX}((V_i(t)-B_i-B_{un}(t)/K_1)/V_i(t),0). \tag{7}$$

我们同样用表 1 的 3 组数据对改进后的 FMPA 进行了仿真,结果如图 5 所示.可以看出,当使用第 2、3 组数据仿真时,数据源 S_3 和 S_4 被标记为"IN"的分组吞吐率大于各自的阈值(2Mbps),但聚集流的"IN"分组的总吞吐率并没有超过总阈值.

比较图5和图4(b)可以看出,FMPA经改进后,"IN"分组的总吞吐率明显提高,但改进后的FMPA仍不是最优的,我们可以通过一个例子说明.在第2组数据中,假设S_3的发送速率为2.1Mbps(而非3Mbps),其他数据源的发送速率不变,我们仍把B_{un}等比例地分摊给S_3和S_4,由于S_3的发送速率仅超过阈值B_3(2Mbps)0.1Mbps,而补充给它的份额0.5Mbps,将造成剩余份额(0.4Mbps)被浪费,因此我们可以继续改进算法,补充给S_3恰好的份额,而把剩余的份额都留给S_4.但是这将导致算法更复杂,影响标记效率,在此我们不再考虑.

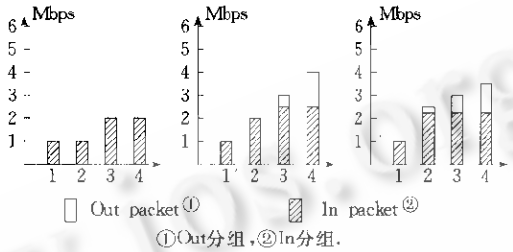


Fig. 5 Simulation results for improved FMPA
图5 改进后的FMPA仿真结果

4 分组标记的其他问题

标记工作通常可以在两个地方完成,一个是集成在用户端系统中,另一个是集成在边界路由器中,或者由专门的设备实现,如具有分组转发功能(仅需设定一条静态路由)的高性能的工作站或微机.Feng等人[7]分别称其为源集成的(或内部的)和源透明的(或外部的)标记策略.源集成的方式一般用于数据源端系统中,源透明的方式一般用于用户域边界和ISP网络边界处.

目前,许多应用都基于TCP/IP,而TCP协议本身并不能意识到预留带宽,其拥塞控制机制更不能保护预留带宽,任何一个"IN"/"OUT"分组的丢失都可能引起TCP进入慢启动(slow-start)或拥塞避免阶段,导致吞吐率大幅下降.为此,有人提出了一种"两窗口的TCP拥塞控制策略"[9],该策略把拥塞控制窗口cwnd分成rwnd和ewnd两部分,rwnd=RTT×r_{IN}(r_{IN}为"IN"分组的预留带宽),ewnd−RTT×r_{OUT}(r_{OUT}为额外获得的带宽).当"OUT"分组丢失时,cwnd−rwnd+ewnd/2,而只有当"IN"分组丢失时,才会有cwnd=cwnd/2,由于"IN"分组的丢失率远远低于"OUT",从而有效地提高了吞吐率.这种方式需要采用源集成的标记策略,因为端系统需要知道丢失的分组是"IN"分组还是"OUT"分组.然而,当中继网络对分组再标记*之后,会使分组的服务级别发生变化,原有的"IN"分组可能变成"OUT"分组,反之亦然.目前已提出的策略[5]是在ISP的边界入口/出口处估计分组的流入/流出平均速率,以此作为分组再标记的依据(而不管流入/流出分组原有的级别是"IN"还是"OUT"),见算法2.

算法2. 分组再标记算法.

For 在ISP网络的入口/出口处每个流入/流出分组,
  计算流入/流出分组的平均速率$V_{avg}(t)$
  If ($V_{avg}(t)>B$) /* B为预留带宽 */
    $P_{OUT}=(V_{avg}(t)-B)/V_{avg}(t)$,
    按照概率P_{OUT}标记该分组为OUT

* ISP网络的入口/出口处通常需要对分组进行再标记(remarking),以维护ISP与用户以及与其他ISP之间的服务合约(SLA).

else

$P_{IN}=MIN(B-V_{avg}(t))/V_{avg}(t),1)$,

按照概率 P_{IN} 把该分组标记为 IN

endif

End

下面,我们将提出一种稍加改进的算法,见算法 3.我们通过仿真实验对算法 2 和算法 3 进行了比较.图 6 为仿真拓扑,数据源 S_1 向目的端 D_1 发送"IN"分组,发送速率在 1.5Mbps～2.5Mbps 随机变化;数据源 S_2,S_3 分别向目的端 D_2,D_3 发送"OUT"分组,发送速率在 2Mbps～3Mbps 随机变化.路由 R 的预留带宽为 2Mbps(用于标记"IN"分组),路由器的处理能力为每秒 1 万分组(保证没有任何分组丢失).图 7 和图 8 分别为两个算法的仿真结果.从图 7 中可以看出,各个数据源被再标记的概率与各自的发送速率相关,数据源 S_1 有三分之二的分组被标记为"OUT".显然,大部分"IN"分组被再标记为"OUT".而算法 3 改变了这一情况,数据源 S_1 只有少量分组被再标记为"OUT".

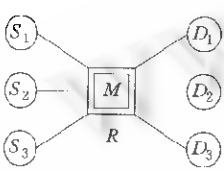
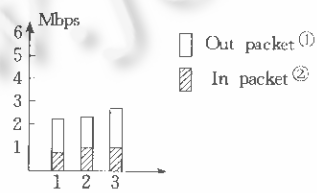
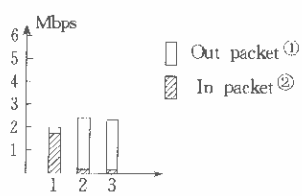


Fig. 6 Simulated topology
图6 仿真拓扑



①Out分组,②In分组.

Fig. 7 Simulation result for Algorithm 2
图7 算法2的仿真结果



①Out分组,②In分组.

Fig. 8 Simulation result for Algorithm 3
图8 算法3的仿真结果

算法 3. 改进的分组再标记算法 FMPA.

For ISP 网络的入口/出口处每个流入/流出分组,

If 为IN 分组

计算流入/流出 IN 分组的平均速率 $V_{avg\_IN}(t)$

If $V_{avg\_IN}(t)>B$ /* B 为预留带宽 */

$P_{OUT}=(V_{avg\_IN}(t)-B)/V_{avg\_IN}(t)$,

按照概率 P_{OUT} 标记该分组为 OUT

else

$P_{IN}=MIN((B-V_{avg\_IN}(t))/V_{avg\_IN}(t),1)$,

按照概率 P_{IN} 把下一个 OUT 分组标记为 IN

endif

End

算法3可以较好地与"两窗口的 TCP 拥塞控制策略"[9]相结合.另外,用户在管理策略上可能会保护某些重要的应用(把这些应用的分组都标记为"IN"),因此尽可能维持分组原来的服务语义是很有意义的.

5 结 论

本文讨论了区分服务中的分组标记问题,合理的分组标记策略将有助于实现服务区分并提高吞吐率.我们提出了具有公平性的标记算法 FMPA,该算法可按照分配给不同数据源的预留带宽公平地标记分组.由于该算法处理的对象是未标记过的分组,因而适合于数据分组的初始标记,但稍加修改也可用于中继网络边界处的分组再标记.

我们还讨论了分组的再标记问题,尽可能维护流入/流出分组原有的服务语义是很有意义的,为此,我们提出了一个可行的解决方案,并通过仿真进行了验证.

References:

[1] Blake, S., Black, D., Carlson, M., *et al*. An architecture for differentiated service. Technical Report, RFC 2475, 1998.

[2] Nichols, K., Jacobson, V., Zhang, L. A two bit differentiated services architecture for the Internet. Technical Report, RFC 2638, 1999.

[3] Braden, R., Clark, D., Shenker, S. Integrated services in the Internet architecture: an overview. Technical Report, RFC 1633, 1994.

[4] Braden, R., Ed., Zhang, L., Berson, S., *et al*. Resource reservation protocol (RSVP)——version 1 functional specification. Technical Report, RFC 2205, 1997.

[5] Clark, D. D., Fang, W. Explicit allocation of best-effort packet delivery service. IEEE/ACM Transactions on Networking, 1998,6(4):362~373.

[6] Stoica, I., Zhang, Hui. LIRA: a model for service differentiation in the internet. In: Proceedings of the NOSSDAV'98. Cambridge, England, 1998. 115~128.

[7] Feng, W., Kandlur, D. D., Saha, D., *et al*. Adaptive packet marking for maintaining end-to-end throughput in a differentiated-services internet. IEEE/ACM Transactions on Networking, 1999,7(4):685~697.

[8] Network simulator (NS). University of California at Berkeley, CA, 1997. http://www.nrg.ee.lbl.gov/NS.

[9] Feng, W., Kandlur, D. D., Sala, D., *et al*. Adaptive packet marking for providing differentiated services in the Internet. In: Proceedings of the ICNP'98. Austin, Texas, 1998. 108~117.

Packet Marking Algorithm Research in Differentiated Service Networks*

MA Xiao-jun, YAN Jun, GU Guan-qun

(*Department of Computer Science and Engineering, Southeast University, Nanjing* 210096, *China*);
(*Key Laboratory of Computer Network and Information Integration, Ministry of Education, Nanjing* 210096, *China*)
E-mail: xjma@seu.edu.cn
http://www.seu.edu.cn

Abstract: Differentiated service (DiffServ) architecture has become one of the research hotspots on computer networking in the last two years. Its intention is to provide quality of service for users at coarse granularity level. In contrast to integrated service architecture, DiffServ is not only more scalable, but also easier to be deployed in traditional packet-switching networks. In this paper, the packet-marking algorithm, one of the key mechanisms in DiffServ is studied. A new packet-marking algorithm called FMPA (fair marking packet algorithm) is presented. The new algorithm is compared with the existing proportional packet-marking algorithm by simulation. In addition, a packet remarking algorithm is proposed. By using it, the packet's original semantic can be kept to the largest degree. The simulation result shows it in this paper.

Key words: differentiated service (DiffServ); integrated service (IntServ); packet marking; packet remarking; quality of service (QoS)

* Received February 29, 2000; accepted June 12, 2000
Supported by the National Natural Science Foundation of China under Grant Nos. 69873008, 69896249; the Key Laboratory of Computer Network and Information Integration of Ministry of Education of China