

异构网络环境中的拥塞丢包率仿真分析

唐子蛟, 蔡乐才, 符长友

(四川理工学院计算机学院, 四川 自贡 643000)

摘 要:针对 TCP Vegas 算法在异构网络环境中流量公平性导致的拥塞丢包问题,将 TCP Vegas 拥塞控制算法与主动队列控制策略来行结合分析,提出了将网络模型中不同层次的拥塞控制机制进行结合的算法。网络仿真表明,结合算法能有效避免拥塞丢包。该算法对随机早期检测算法进行了改进,使其能够区分突发流量,从而降低拥塞丢包率,这也将为高可靠性网络的发展提供一个优良的参考价值。

关键词:异构网络;TCP vegas;拥塞丢包率;仿真分析

中图分类号:TP391

文献标志码:A

异构网络是由不同硬件、运行在不同协议下支持不同作用和功能的一种复杂的网络环境。在这种网络环境中,多种原因都会引起数据丢包(包括算法公平性引起的丢包、误码率引起丢包、超时丢包、错误传输丢包、接收分组缓冲区大小、TTL 等)^[1],这将严重影响网络的稳定性。针对上述问题,本文通过分析 TCP Vegas 算法和 TCP Reno 算法在共存网络环境中流量的公平性^[2],发现前者的流量竞争力较低。分析发现,因 TCP 系列算法不能智能区分丢包原因,所以会造成拥塞避免算法被“误调”,进而导致拥塞丢包率增大、网络性能降低。为了更好的研究和学习异构网络环境中流量公平性对拥塞丢包率的影响问题,本文提出了结合网络模型中不同层次的的拥塞控制算法来降低拥塞丢包率和改善网络性能的方法^[3-4]。

1 TCP Vegas 算法描述

TCP Vegas 算法是 TCP 系列中算法与众不同的一个,它相对以前出现的各版本做了较大改进,包括对慢启动、快速重传、拥塞避免 3 个阶段都进行了改进。其最大的优点在于具有高带宽利用率和稳定的吞吐量;最

大的特色是拥塞状况的判断是通过数据往返时间估算来预测(相比以前版本的“亡羊补牢”式的避免方式具有质的飞跃);最致命的缺陷是公平性问题。该算法的基本思想是:当期望的吞吐量与实际的吞吐量之差超过指定阈值时,就认为网络拥塞程度严重,应该减小发送窗口;而当两者之差小于指定阈值时,则认为连接没有完全有效的利用带宽,应该要增大发送窗口。Vegas 在拥塞避免阶段的具体算法如下^[5]:

(1) diff 估算

$$\text{diff} = \text{expected} - \text{actual} = \text{cwnd}/\text{basertt} - \text{cwnd}/\text{minrtt}$$

其中, *basertt* 代表当路由器缓存中无数据包时的 *RTT* 值。*minrtt* 为上一个 *RTT* 的估算值,由于此时路由器中已有数据包,需要排队,故 *minrtt* 大于 *basertt*。*expected = cwnd/basertt* 为期望的吞吐量,是理想情况下的吞吐量。*actual = cwnd/minrtt* 为实际的吞吐量。

(2) 计算路由器中累积的数据包个数

$dc = \text{Diff} * \text{basertt}$, 表示吞吐量的差值与链路时延的乘积。

收稿日期:2013-05-22

基金项目:四川省教育厅项目(13zb0136);四川理工学院科研项目(2012ky10)

作者简介:唐子蛟(1982-),男,四川广安人,实验师,主要从事计算机网络通信与信息处理技术方面的研究,(E-mail)756050627@qq.com

(3) 窗口调整

Vegas 算法在每一个 RTT 末根据 dc 进行窗口调整。

$$cwnd(n + 1) =$$

$$\begin{cases} cwnd(n) + 1 & dc(n) < upnum \\ cwnd(n) & donum \leq dc(n) \leq upnum \\ cwnd(n) - 1 & dc(n) > upnum \end{cases}$$

以上分段表达式中, $dc(n)$ 代表第 n 个 RTT 时,估算出的路由器中累积的数据包个数。 $cwnd(n)$ 代表第 n 个 RTT 时的拥塞窗口。

Vegas 算法由于不能有效区分突发数据流和拥塞,甚至在新的数据流加入时也被认为发生了拥塞,导致其通过牺牲自己的流量让出带宽,这样,在复杂的异构网络中就会使网络性能严重下降,影响流量的公平性。由于 TCP Vegas 存在的问题,没有像 TCP Reno 那样被广泛使用,但将它独特的优势和 IP 拥塞控制策略相结合来达到提高网络性能的目的值得我们去研究和探索。

2 拥塞丢包过程仿真

2.1 异构网络环境

仿真模型如图 1 所示,在节点 0 到节点 4、节点 1 到节点 5 分别应用 TCP Vegas 和 TCP Reno,节点 2 到节点 3 为瓶颈链路,带宽为 5 Mbps,延时为 10 ms,两侧链路带宽均为 10 Mbps,延时为 20 ms,路由器缓存为 100,包大小为 552 字节,节点 0 在 0 s 开始,节点 1 在 2 s 时开始加入,模拟时间持续 8 s。

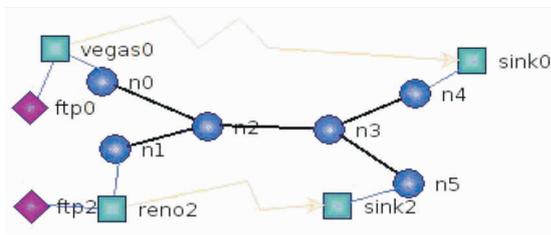


图 1 仿真环境拓扑图

2.2 TCP Vegas 结合主动队列算法的丢包率仿真

将图 1 所示的拓扑结构进行动画仿真,效果如图 2 所示,其中,数字代表节点,节点间通过有线连接,主动队列管理算法采用随机早期预测检测算法,红色和蓝色分别代表 ftp0 和 ftp1 两种数据流,数据流通过节点 2 和节点 3 这条瓶颈链路到达指定目的节点,同时,可以观察到不同分组被对方接收后都会返回 ACK 帧.在某个时刻节点 2 处有向下的蓝色加粗箭头,它代表此刻在这

个节点有数据包丢失的情况发生。

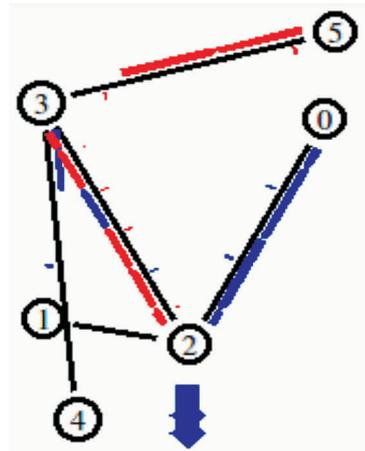


图 2 丢包仿真

本文通过在瓶颈链路上设置相对小的带宽以及结合 RED 主动队列管理算法来观察两种数据分组在节点 2 处的丢包率和吞吐量,进而分析丢包原因。

图 3 中,水平轴和垂直轴分别代表时间(s)和丢包率。由图 3 显示的结果可以观察到:红色代表的分组在 0 s 开始,但是在 2.3 s 时分组的丢包率陡然上升,之后逐渐降低的过程;同时,绿色代表的分组丢包率呈快速增高的趋势,直到第 6 s 分组传输结束而开始缓慢降低。

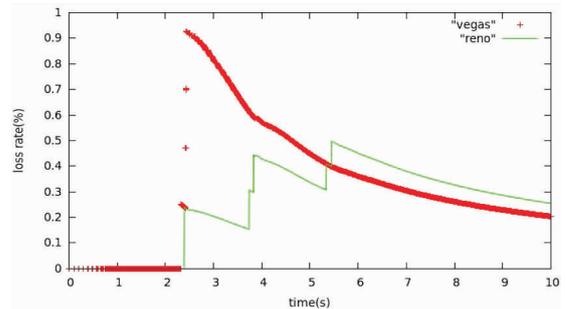


图 3 丢包率对比分析

图 4 为节点 2 处的吞吐量,由它可以观察到:0 s 时节点 0 和节点 4 之间的通信开始,中间节点 2 的吞吐量开始增大,直到 2 s 时,第二个数据流加入,吞吐量才呈现减少的过程。在 4 s 时出现转折,绿色代表的分组数据量开始平缓增加,红色代表的分组数据量逐渐减少。

随机早期预测检测算法使用指数加权平均算法来计算平均队列长度,并用作判断拥塞的依据,将此算法运用在路由器中可以减小丢包率、提高吞吐量,能较好地提高网络稳定性和分组传输效率^[6]。图 4 中,在 2 s

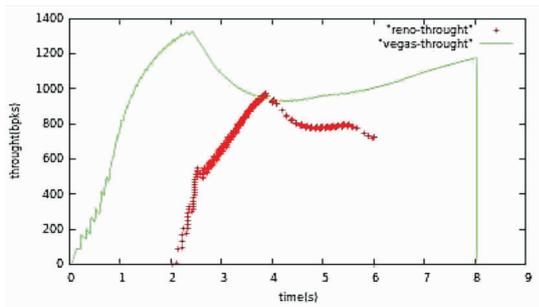


图 4 吞吐量对比分析

时加入的分组数据导致了先前分组数据量的减少,二者共享瓶颈链路时,不同的分组数据都在竞争带宽。由于 TCP Vegas 算法是以数据报的往返时间来标识网络拥塞状态,当有其他数据报进来,计算所得的 RTT 值会增加,从而导致在阈值上限时其拥塞窗口减小,降低了网络性能。在二者共存的环境中, TCP Vegas 不具备区分拥塞的功能,其他突发数据报的加入都被认为产生了拥塞事件,此时会错误调用拥塞避免算法,牺牲自己的吞吐量,降低数据发送速率来避免拥塞发生,导致了其公平性较 Reno 差。这种公平性的优劣取决于算法,改进 TCP Vegas 算法让其与 RED 主动队列管理算法更好地协调工作来降低丢包率,从而实现提高网络性能的目的。

3 改进后的算法描述

为了更好地改善网络性能,降低拥塞丢包率,需要改进 TCP Vegas 算法的公平性以及更好地协调网络层上 RED 算法进行工作。通过分析得知: TCP Vegas 算法中,拥塞窗口大小的调整与给定的门限值 $donum$ 和 $upnum$ 及路由器缓存中数据包的数量相关,并以此判断拥塞是否发生。同样,在网络层运行的 RED 算法中,作为判断拥塞状况的平均队列长度也与队列长度的门限值相关。所以,能够通过合理调整两个算法的门限值来改善网络性能^[7]。

改进的算法思想描述如下:调整队列长度上下门限值。对于突发数据分组,通过改进 RED 算法来区分突发分组数据量,从而降低丢包的发生。拥塞丢包的发生是全部分组的速率超过瓶颈带宽,平均队列越长,丢包概率越大,而路由缓存的队列长度超过最大门限值,导致丢包概率增加。丢包概率值分三种情况:(1)平均队列长度小于队列最小门限值时,丢包概率为 0;(2)平均队列长度介于队列最小门限值和最大门限之间时, $p =$

$p/(1 - count * p)$; (3)其他情况为 1。改进第二种情况的边界丢包概率值的计算公式为 $p = (e^{mqm} - e^{mqmin}) / (e^{mq} - e^{mqmin})$, m 代表缓存中平均队列增长频率, qm 、 $qmin$ 、 q 分别表示平均队列长度、队列最小门限、物理缓存。对边界丢包概率值的改进可以对这种特殊情况下 TCP Vegas 的公平性有利,从而能够缓解丢包。改进算法的丢包率情况如图 5 所示。

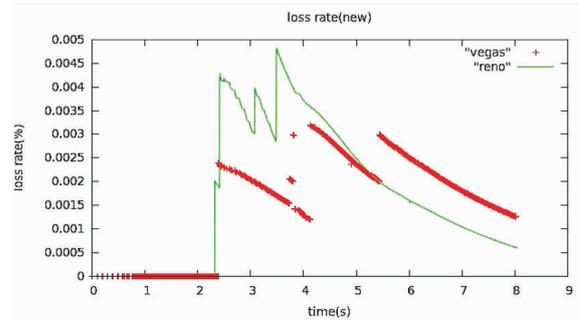


图 5 改进后算法的丢包率

对比图 3 和图 5 的丢包率,我们可以看到改进后的算法 TCP Vegas 丢包率明显降低了, Vegas 算法控制的流量丢包没有连续的朝着一个趋势变化,而是在适当的时候,丢包率起伏变化,在 4 s 时,丢包率跳高上升说明了在此刻 Reno 算法丢包率上升对其存在一定的影响,但由于在改进的 RED 算法协调作用,在瓶颈带宽一定时,丢包率增大,数据流量出现轮询发送的状态,保证了较好的网络性能。在 6 s 时丢包率有一个升高,但之后快速降低。总的来说,改进的 RED 算法能够区分突发流量引起的拥塞丢包,从而降低数据丢包率。

4 结论

通过对异构网络环境中拥塞丢包率的分析,说明了如何改进 RED 算法中的丢包概率边界值,来区分突发数据流,进而减少拥塞丢包,保证在这种网络环境中 TCP Vegas 和 RED 算法更好的协调工作。实验表明:

- (1)合理调整 TCP Vegas 算法中的拥塞窗口上下门限值能够减少拥塞丢包。
- (2)通过与改进的 RED 算法结合, TCP Vegas 算法能够减少公平性差带来的拥塞丢包。

参考文献:

- [1] 胡 晗.基于误码丢包率监测的无线 TCP 改进[J].计算机应用,2011,31(10):2657-2659.

- [2] 周铁军,寇小文,李阳.TCP Vegas 和 TCP Reno 的兼容性问题的解决办法[J].湘潭大学学报,2008,30(4):130-134.
- [3] 朱春,王塞云.改进的网络拥塞控制策略算法研究[J].计算机仿真,2011,28(12):141-143.
- [4] 闫二辉,朱敏,丁青,等.一种基于比例因子的 TCP Vegas 慢启动策略[J].计算机应用研究,2011,28(1): 253-255.
- [5] 王斌,陈元琰,胡愚,等.TCP Vegas 拥塞避免机制的改进算法[J].计算机应用,2010,20(9):2486-2487.
- [6] 李涛.改进 TCP 拥塞控制算法的仿真及应用研究[J].计算机仿真,2011,28(6):181-184.
- [7] 高鹏,戴旭初.基于 NS 的主动队列管理算法的仿真与分析[J].计算机仿真,2006,23(5):98-100.

Simulation Analysis of Loss Tolerance of Congestion in a Heterogeneous Network Environment

TANG Zi-jiao, CAI Le-cai, FU Chang-you

(School of Computer Science, Sichuan University of Science & Engineering, Zigong 643000, China)

Abstract: In a heterogeneous network environment, in order to solve the loss tolerance problem of congestion caused by the flow fairness of TCP Vegas algorithm, the TCP Vegas congestion control algorithm and active queue control strategy are combined to carry on the analysis. Then, a kind of algorithm that combines congestion mechanism of different levels in the network model is proposed. Network simulation shows, the algorithm can avoid congestion lost packet effectively. In the network environment, the random early detection algorithm is improved, so that the new algorithm can distinguish the bursty traffic, thereby the congestion loss rate is reduced, which will also provide a good reference value of the development of high reliability network.

Key words: heterogeneous network; TCP Vegas; loss tolerance of congestion; simulation analysis