

文章编号:1001-9081(2010)05-1188-03

一种改进的拥塞控制算法

陈飞龙, 舒勤, 苏静

(四川大学电气信息学院, 成都 610065)

(chen85728fl@126.com)

摘要:为控制网络拥塞,提出一种改进的随机早期检测(RED)算法——基于加权随机早期检测(WHS-RED)算法。算法的基本思想是利用网络中队列长度及其变化率的加权,动态地控制网络丢包率。建立了数学模型,并给出了计算机仿真分析结果。实验证明改进算法减小了网络丢包率和带宽延迟,在维持网络稳定性和减小队列的波动性方面优于原RED算法。

关键词:队列;随机早期检测;拥塞控制;指数加权滑动平均;平均队列长度;标记概率

中图分类号: TP393 **文献标志码:** A

Improved congestion control algorithm

CHEN Fei-long, SHU Qin, SU Jing

(School of Electrical Engineering and Information, Sichuan University, Chengdu Sichuan 610065, China)

Abstract: A Weight Sum-Based Random Early Detection (WHS-RED) algorithm was proposed to deal with network congestion. WHS-RED algorithm used the rate of change of queue lengths in networks to drop network packets. The mathematical model was established. Simulation results prove that the proposed algorithm reduces packet loss rate and bandwidth-delay, while maintaining network stability and reducing the jitter of the queue, which is better than that of the original RED algorithm.

Key words: queue; Random Early Detection (RED); congestion control; Exponentially Weighted Moving Average (EWMA); average queue length; marking probability

随着数据通信的迅速发展,网络流量急剧增长,网络的突发性越来越严重,网络流量平均值和峰值之比达到数十倍,这种突发性使得网络拥塞问题越来越得到重视。拥塞控制就是要在网络节点处采取措施来避免拥塞发生和对网络拥塞作出反应。其中,随机早期检测(Random Early Detection, RED)算法^[1]是一种重要的拥塞控制算法。该算法利用早期检测平均队列长度来丢包,以达到降低延迟和提高吞吐量的目的。

1 RED算法的数学模型

1.1 RED算法描述

RED算法的主要思想通过指数加权滑动平均(Exponentially Weighted Moving Average, EWMA)的低通滤波器来计算平均队列长度,每接收到一个数据包就重新计算一次平均值,并通过该平均值与最小门限和最大门限来计算丢包概率。再由此概率计算每个数据包到达路由器时是否会引起网络拥塞,并在平均队列长度超过一定阈值后以一定的概率随机丢弃或者标记到达的数据包,以达到拥塞控制的目的。RED算法主要有以下两个部分。

1) 计算平均队列长度。RED算法采用指数加权滑动平均的低通滤波器来计算平均队列长度。计算公式如下:

$$avg_{(t)} = (1 - w_q) avg_{(t-1)} + w_q \times q_{(t)} \quad (1)$$

其中: $avg_{(t)}$ 表示平均队列长度; w_q 表示低通滤波器时间长度; $q_{(t)}$ 表示瞬时队列长度。每接收到一个分组计算一次平均队列长度。由EWMA模型可以使突发流量不会短时间内使平均队列长度大幅度增加。

2) 计算丢包概率。丢包概率决定网关丢包的概率大小,能反映网关当前拥塞的程度。RED算法设置了两个门限值, th_{min} 和 th_{max} 。算法根据平均队列长度 avg 与门限值的比值来决定是否丢弃到达的分组。计算公式如下:

$$p_b \leftarrow p_{max} \frac{avg - th_{min}}{th_{max} - th_{min}} \quad (2)$$

1.2 RED算法优缺点分析

优点 目前路由器广泛采用包调度的“丢尾”队列管理策略,而“丢尾”算法也比较简单。与传统的丢尾算法相比,RED算法不易引起全局同步或产生队列溢出现象;并且RED算法的丢包概率与带宽成正比,可以在各个流间保持较好的公平性,减小排队延迟等。

缺点 RED算法严重依赖参数的设置,当参数设置不合理时,性能会大大降低。RED算法是针对某个网络设计的,当网络情况发生变化时,其性能会受很大的影响。另外,由于RED算法采取队列长度丢包,对网络的突发性不敏感。因此,RED算法在稳定性和鲁棒性方面还需要改进^[2-4]。

2 改进的RED算法

2.1 改进算法要解决的问题

判断网络拥塞的依据一是平均队列的长度,二是平均队列的持续时间。如果平均队列很长,并且持续很久,则网络拥塞;但是当平均队列很长,而持续时间不久,则网络并不拥塞。RED算法以及其他主动队列管理算法都只是依据平均队列的长度来判断网络的拥塞程度,这种判断依据是不全面的。这也是RED算法太依赖于参数设置的原因。

收稿日期:2009-10-15;修回日期:2009-12-21。

作者简介: 陈飞龙(1985-),男,四川眉山人,硕士研究生,主要研究方面:计算机网络、现代信号处理; 舒勤(1958-),男,重庆人,教授,博士,主要研究方向:计算机网络、现代信号处理; 苏静(1986-),女,湖南邵阳人,硕士研究生,主要研究方向:计算机网络、现代信号处理。

RED 算法的丢包率与平均队列长度呈线性关系,当平均队列长度 $avg_{(t)}$ 在最大阈值 th_{max} 与最小阈值 th_{min} 之间时,丢包率 p_b 随平均队列 $avg_{(t)}$ 增长而线性均匀增长。在实际网络中,当网络不太拥塞时,丢包率应尽量小,当网络拥塞时,丢包率应偏大,由于 RED 算法拥塞判断的片面性,导致了不公平的丢包。

2.2 WHS-RED 算法描述

本文提出一种改进 RED 算法——基于加权随机早期检测 WHS-RED (Weight Sum-RED)。由于计算机网络是一个动态系统,路由器队列长度的变化可由以下公式^[5]得出,

$$avg_{(t)} = avg_{(t-1)} + y_t - C \quad (3)$$

其中: y_t 为 $t-1$ 时刻到 t 时刻到达的包数目,由于网络传送一个文件往往需要一个相对较长的时间周期,之后又会经历一个较长的空闲时间,故 y_t 会是一个大体呈波浪形的变化轨迹,当然由于网络的突发性, y_t 也会产生突变; C 为交换节点容量,为一固定常数。由此可知网络拥塞程度不仅与当前队列长度有关,与路由器处理速率和包到达率也有很大关系,且具有很强的突发性。由式(1)可以推出:

$$q_{(t)} - avg_{(t-1)} = (avg_{(t)} - avg_{(t-1)})/w_q \quad (4)$$

设 $avg_{(t)}$ 为队列的变化速率,则:

$$avg_{(t)}' = (avg_{(t)} - avg_{(t-1)})/w_q$$

基于预测的 RED 算法描述如下:

If $avg \leq th_{min}$, $p_b \leftarrow 0$;

If $th_{min} \leq avg \leq th_{max}$

$$p_b \leftarrow \left(\frac{avg_{(t)} - th_{min}}{th_{max} - th_{min}} + \beta \times \frac{avg_{(t)}'}{C} \right) p_{max}$$

If $th_{max} \leq avg$, $p_b \leftarrow 1$

其中 β 是比例系数。

由于改进算法利用了队列长度的变化速率控制丢包,与自适应 RED 算法有一定的相似之处,相比 RED 算法,改进算法能减小队列的抖动,提高网络的稳定性。又因为改进算法利用了平均队列长度控制丢包概率,相比其他只利用速率控制丢包的改进 RED 算法^[5],WHS-RED 能有效减小队列长度和网络延迟,更适应网络的变化。WHS-RED 算法相对于 RED 算法的复杂度上只是增加了一个存储 $avg_{(t-1)}$ 的存储单元,对算法复杂度并无多大影响。

3 仿真分析

3.1 仿真建模

本文用网络模拟器 NS2 对改进算法进行仿真分析。所用的网络拓扑如图 1 所示:节点 0、1 为发送端;节点 3 为接收端;节点 2 和 3 之间为瓶颈链路。节点 0、1 与中间节点 2 之间的链路带宽和延迟分别为 10 Mbps 和 2 ms;节点 2、3 之间的链路带宽和延迟分别为 2 Mbps 和 10 ms;节点 0 为 TCP 链接,用 ftp 流;节点 1 为 UDP 链接,用 CBR 流。分别在链路上用原 RED 算法和改进的 RED 算法进行拥塞控制。原 RED 算法参数值为: $th_{min} = 5$, $th_{max} = 15$; $p_{max} = 0.1$, $w_q = 0.002$ 。

改进算法参数 β 取 0.6,其余参数与原算法相同。

3.2 算法性能比较分析

3.2.1 队列比较

如图 2、3 分别是平均队列长度和瞬时队列长度对比图。可以看出,改进算法 WHS-RED 的队列抖动较小,其平均长度也小于 RED 算法。由于 WHS-RED 算法采用了预计队列长度丢包,加强了算法的鲁棒性和自适应性,故使队列的平均值

和抖动较小。

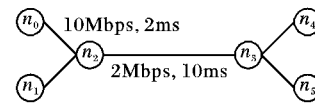


图1 网络拓扑

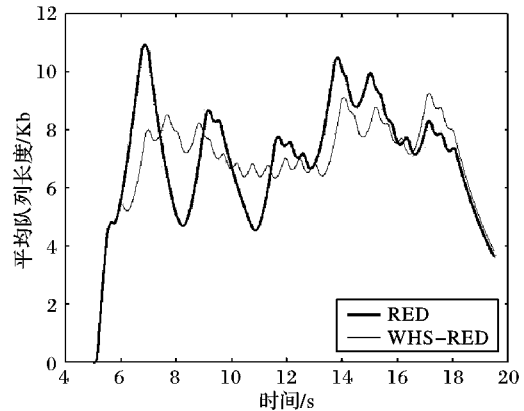


图2 平均队列长度比较

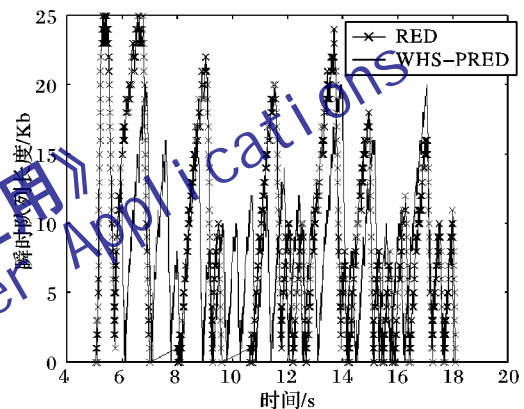


图3 瞬时队列长度比较

3.2.2 延迟比较

图 4 给出了 RED 算法和 WHS-RED 算法的队列延迟比较。可以看出, WHS-RED 算法的延迟和波动性均小于 RED 算法,反应了 WHS-RED 算法的稳定性和低延迟。

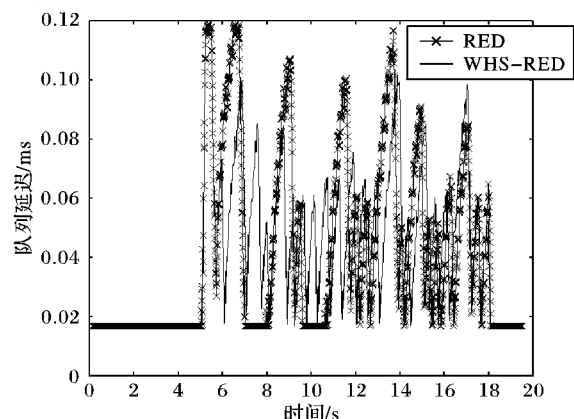


图4 延迟比较

3.2.3 丢包率比较

图 5 给出了 WHS-RED 算法和 RED 算法的丢包率对比。可以看出, WHS-RED 算法在大部分时间里,其丢包率都小于 RED 算法,从表 1 可以看出,在相同条件下, WHS-RED 算法的丢包率比 RED 算法少了 1.1 个百分点,表明了 WHS-RED 算法的优越性。

由分析结果可知,由于 WHS-RED 算法采用队列长度和速

率值加权和求概率丢包,使其减少了平均队列长度和丢包率,并减少了网络延迟及波动性,加强了网络的稳定性和鲁棒性。

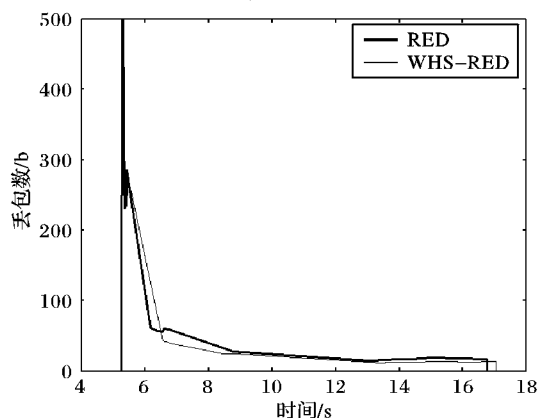


图5 丢包率比较

表1 丢包率数据对比

| 算法 | 发送包数 | 丢失包数 | 丢包率/% |
|---------|------|------|-------|
| RED | 4899 | 349 | 7.12 |
| WHS-RED | 4899 | 295 | 6.02 |

4 结语

本文介绍了随机早期检测(RED)的原理和实现方式,分析了 RED 算法的优缺点,并针对其参数设置难和维持网络稳定性方面的问题,提出了一种基于加权和的改进 RED 算法。

(上接第 1155 页)

参考文献:

- [1] FLOYD S, JACOBSON V. Random early detection gateways for congestion avoidance [J]. *IEEE/ACM Transactions on Networking*, 1993, 1(4): 397-413.
- [2] FENG WU-CHANG, KANDLUR D D, SAHA D, *et al.* A self-configuring RED gateway [C]// *INFOCOM '99: Proceedings of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies*. Washington, DC: IEEE Communications Society, 1999, 3: 1328-1346.
- [3] QUE D, CHEN ZHI-XIANG, CHEN BI. An improvement algorithm based on RED and its performance analysis [C]// *ICSP 2008: 9th International Conference on Signal Processing*. Washington, DC: IEEE Press, 2008: 2005-2008.
- [4] OTT T J, LAKESMAN T V, WONG L H. SRED: Stabilized RED [C]// *INFOCOM '99: Proceedings of Eighteenth Annual Joint Conference on the IEEE and Computer and Communications Societies*. Washington, DC: IEEE Communications Society, 1999, 3: 1346-1355.
- [5] 张明亮,叶澄清.一种基于速率的 RED 增强算法[J]. *浙江大学学报:工学版*, 2004, 38(2): 159-162.
- [6] BONNET P, GEHRKE J E, SESHADRI P. Towards sensor database systems [C]// *Proceedings of the 2nd International Conference on Mobile Data Management, LNCS 1987*. Berlin: Springer-Verlag, 2001: 3-14.
- [7] YAO Y, GEHRKE J. Query processing in sensor networks [C]// *CIDR 2003: Proceedings of the First Biennial Conference on Innovative Data Systems Research*. Washington, DC: IEEE Computer Society, 2003: 233-244.
- [8] INTANAGONWIWAT C, ESTRIN D, GOVINDAN R, *et al.* Impact of network density on data aggregation in wireless sensor networks [C]// *ICDCS '02: Proceedings of the International Conference on 22th Distributed Computing Systems*. Washington, DC: IEEE Computer Society, 2002: 457-458.
- [9] BIAN F, RANGWALA S, GOVINDAN R. Quasi-static centralized rate allocation for sensor networks [C]// *SECON '07: Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh, and Ad-Hoc Communications and Networks*. Washington, DC: IEEE Press, 2007: 361-370.
- [10] ZHAO J, GOVINDAN R, ESTRIN D. Computing aggregates for monitoring wireless sensor networks [C]// *Proceedings of First IEEE Workshop on Sensor Network Protocols and Applications*. Washington, DC: IEEE Press, 2003: 139-148.
- [11] VINH T-Q, TAKUMI M. A novel gossip-based sensing coverage algorithm for dense wireless sensor networks [J]. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 2009, 53(13): 2275-2287.
- [12] DESHPANDE A, GUESTRIN C, MADDEN S R, *et al.* Model-driven data acquisition in sensor networks [C]// *Proceedings of the Thirtieth International Conference on Very Large Data Bases*. [S. l.]: VLDB Endowment, 2004: 588-599.
- [13] DELIGIANNAKIS A, KOTIDIS Y, ROUSSOPOULOS N. Bandwidth-constrained queries in sensor networks [J]. *The International Journal on Very Large Data Bases*, 2007, 17(3): 443-467.
- [14] MADDEN S, FRANKLIN M J, HELLERSTEIN J M. TinyDB: An acquisitional query processing system for sensor networks [J]. *ACM Transactions on Database Systems*, 2005, 30(1): 122-173.
- [15] MADDEN S, FRANKLIN M J, HELLERSTEIN J M, *et al.* TAG: A tiny aggregation service for ad-hoc sensor networks [C]// *OSDI '02: Proceedings of the Fifth Symposium on Operating Systems Design and Implementation*. New York: ACM Press, 2002: 131-146.
- [16] KAMRA A, MISRA V, RUBENSTEIN D. CountTorrent: Ubiquitous access to query aggregates in dynamic and mobile sensor networks [C]// *SenSys'07: Proceedings of the 5th International Conference on Embedded Networked Sensor Systems*. Sydney, Australia: ACM Press, 2007: 43-57.
- [17] CHEN J-Y, PANDURANGAN G, XU DONGYAN. Robust computation of aggregates in wireless sensor networks: Distributed randomized algorithms and analysis [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2006, 17(9): 987-1000.
- [18] MANJHI A, NATH S, GIBBONS P B. Tributaries and deltas: Efficient and robust aggregation in sensor network streams [C]// *Proceedings of the 2005 ACM SIGMOD international conference on Management of Data*. New York: ACM Press, 2005: 287-298.
- [19] ROY S, CONTI M, SETIA S. Securely computing an approximate median in wireless sensor networks [C]// *Proceedings of the 4th International Conference on Security and Privacy in Communication Networks*. New York: ACM press, 2008: 6.
- [20] NATH S, GIBBONS P B, SESHAN S, *et al.* Synopsis diffusion for robust aggregation in sensor networks [J]. *ACM Transactions on Sensor Networks (TOSN)*, 2008, 4(2): 7.
- [21] CONSIDINE J, HADJIELEFTHARIOU M, LI FEIFEI, *et al.* Robust approximate aggregation in sensor data management systems [J]. *ACM Transactions on Database Systems (TODS)*, 2009, 34(1): 6.