# 基于建模的 TCP拥塞控制机制的研究

吉根云，蔡 勇
（江南大学 信息工程学院，江苏 无锡 214122）

摘 要：为了对以往 TCP拥塞控制机制的发展和完善，提出一种基于建模的 TCP拥塞控制机制的半马尔可夫新模型，并做了进一步的研究，能够有效地加深理解 TCP拥塞控制机制，对探索改进空间具有指导意义。

关键词：建模；TCP拥塞控制机制；半马尔可夫模型；随机点过程
中图分类号：TP309.3 文献标识码：A 文章编号：1009－7961(2008)01－0051－05

## Study of TCP Congestion Control Mechanisms Based on Mathematical Model

JI Gen-yun, CAI Yong
(Southern Yangtze University, Wuxi Jiangsu 214122, China)

Abstract: This paper further discusses TCP congestion control modeling in the Model mechanism with Semi-Markov Process, which could increase the understanding of this mechanism and provide the possibilities of exploring the improved space.

Key words: mathematical model; TCP; congestion control mechanisms; semi-Markov process; random point process

## 0 引 言

Internet中的网络资源分配主要是网络带宽的分配，在 Internet体系结构中，网络本身并不提供带宽资源分配的机制，这一功能是由网络中每台主机的 TCP协议实现的。如果某台主机在自身数据传输的过程中发现丢包，那么说明网络中某一交换节点出现拥塞，根据 TCP协议中的拥塞控制机制，该主机这时应降低自身的传输速率；反之如果一段时间内没有丢包，该主机将提高自身的传输速率。这种机制使得主机不能任意提高自身的传输速率，防止了大规模网络拥塞的出现，实现了 Internet中的网络资源分配。目前，90%以上的网络流量使用 TCP协议进行数据传输。对 TCP拥塞控制机制的协议的研究是长期的、多方面的。

## 1 目前的 TCP拥塞控制机制的模型现状

作为 Internet中资源分配的重要机制，对于 TCP拥塞控制机制的研究绝不仅仅限于该机制本身。研究人员对于如何改进现有网络，进一步提高资源分配的效率、公平性和多样性进行了不懈的研究。

在以上的研究中，仿真实验和网络实测是两种主要的研究方法，数学建模所起的作用并不突出。目前，这个领域的数学建模主要可分为两个方向：

a. TCP拥塞控制机制的建模：该模型主要用于分析网络环境对 TCP协议性能的影响。模型使用丢包率和延迟这两个参数来描述网络环境，用平均发送速率反映 TCP协议的性能。模型将这三个参数联系起来，分析丢包率和延迟（RTT)对 TCP发送速率的影响。模型并不关心网络丢包、延迟是如何产生的，因此它实际只考虑了网络对 TCP的影响，而没有考虑 TCP对网络影响。

b. 网络拥塞控制系统的建模：与上面模型不同，这种模型不仅考虑网络对 TCP的影响，同时也考虑 TCP输出对网络的影响。模型使用反馈控制系统来描述这两者之间的相互作用：TCP发送速率的高低

收稿日期：2008－01－01

作者简介：吉根云（1970－），男，江苏无锡人，硕士，研究方向：网络应用与分布式计算。

影响网络的丢包和延迟,它们则反过来作用于 TCP,影响 TCP的数据发送。这种模型对于理解 Internet拥塞控制的工作机理很有帮助,但是目前这种模型仅仅能对非常简单的网络环境进行建模。

## 2 TCP拥塞控机制模型

TCP拥塞控制机制的工作原理中,发挥主要作用的是 TCP拥塞控制窗口(cwnd)。该窗口的值决定了一条 TCP连接某一时刻网络中最大允许的数据包数量。数据发送端根据网络的拥塞程度调节该窗口的大小,从而调节网络中的数据包数量。当网络不拥塞的时候,TCP增加这个窗口的大小;当网络发生拥塞的时候,TCP减少这个窗口的大小。因此,对于一条 TCP连接,该窗口值可以看成是一个随时间变化的函数 W(t)。函数 W(t)不仅与 TCP拥塞控制机制有关,也与网络丢包和延迟有关。由于网络丢包和延迟的随机性,W(t)的变化也是随机的,W(t)可以看作一个随机过程。图1表示 W(t)过程的一条样本轨道。

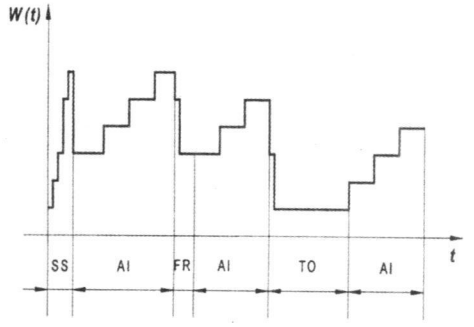


图 1 TCP拥塞控制窗口 W(t)

定义:TCP窗口变化过程 W(t)是一条 TCP连接在网络中的数据包数量(W)随时间变化的过程。

用两种不同的方法对过程进行数学建模,研究当时,过程 W(t)的稳态性质,窗口 W的概率分布。并根据排队论中的 Little s Law,得到过程平均窗口大小和 TCP平均发送速率的关系。

TCP的半马尔可夫模型:理论上讲,TCP拥塞控制机制可以看作一有限状态自动机,数据发送端根据当前的状态和收到的确认包类型(是否确认了新的数据包,是否是重复确认等等)来决定输出(重传旧包或者发送新包)。因此可以用马尔可夫过程来描述 TCP拥塞控制机制中的状态转移,但是由于该机制有很多具体的细节,相应的马尔可夫链也会比较复杂。在这里,可忽略 TCP拥塞控制机制中的慢启动和快速重传算法,研究一个只包含超时重传的理想线增倍减拥塞控制算法。图 2表示这种算法的 cwnd变化曲线 W(t)。

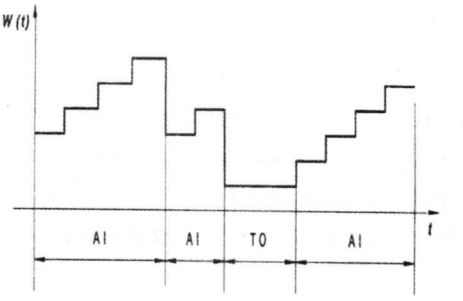


图 2 包含超时重传的理想线增倍减拥塞控制算法

下面分两个步骤考虑这个连续时间离散状态的随机过程 W(t):

首先考虑 W(t)中 W的跳变规律:令 $W_n$ 表示 W(t)过程第 n次状态转移后的窗口大小,根据 TCP拥塞控制机制,cwnd的转移规律可以概括为下面三条:

a:如果没有丢包发生,那么 $W_{n+1} = W_n + 1$

b:如果丢包情况不严重,TCP快速重传丢包,那么 $W_{n+1} = [W_n/2]$

c:如果丢包情况严重,TCP进入超时重传,对于任何 $W_n$, $W_{n+1} = 1$

对于这三条状态转移规则,将来 $W_{n+1}$ 状态仅与当前状态 $W_n$ 有关,所以 $\{W_n\}$ 是过程 W(t)的嵌入式马尔可夫链。图 3表示了该马尔可夫链的状态转移图,包含线增倍减和超时重传两种状态。线增倍减的状态用 cwnd窗口大小的值来表示;超时重传状态的 cwnd都等于 1,但是它们的 t_rtxsh ift(重传移位计数器)不同 2。图中虚线表示由于丢包引起的状态转移,实线表示当 TCP收到足够的确认包后窗口增加的状态转移。

其次考虑 TCP在每个状态下的停留时间。假设 $\tau_n$ 是 TCP在第 n个状态下的停留时间。根据 TCP的协议,对应于上面三种不同的状态转移,$\tau_n$ 也分为三种情况:

a:对应于第一种情况:此时 TCP处于线性增加阶段,$\tau_n$ 约等于 b× RTT,其中,参数 b由 TCP接收端是否使用延迟确认机制确定。如果不使用,b = 1;反之,b= 2。$\tau_n$ 的取值取决于当前的 RTT。

b:对应于第二种情况:在区间 [0, b× RTT] 均匀取值。

c:如果发生超时重传,那么 $\tau_n$ 的大小应等于

重传定时器的预先设定值。该值由两个参数决定, RTO(重传超时时间)和 t_rtxshift(重传移位计数器), $\tau_n = RTO \times t\_rtxshift$

可以看出, $\tau_n$的取值与只与 EMC链中当前状态和下一个状态的值, 以及 RTT和 RTO有关, 而 RTT和 RTO对于所有的 $\tau_n$又是共同的。因此, 根据附录 A, W(t)是一半马尔可夫过程 (SMP)。

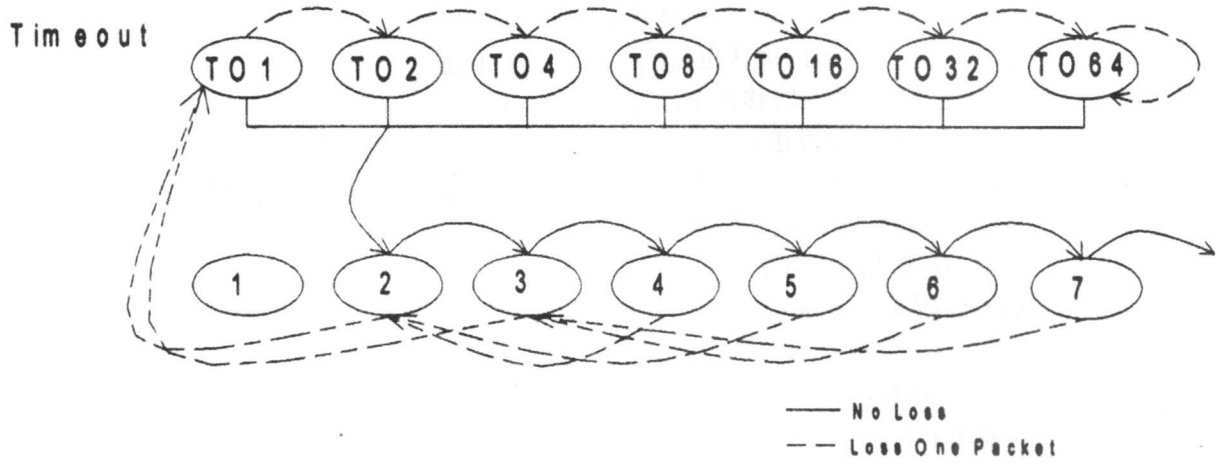


图 3　马尔可夫链状态转移图

## 3　模型的计算

模型的计算分为两步: 第一步计算嵌入式马尔可夫模型中的 $\{\pi_j\}$, 第二步计算得到 SMP模型的 $\{p_j\}$。

### 3.1　EMC的计算

首先计算计算 EMC的状态转移矩阵 P:

在 P矩阵中, 每个状态一般对应三种状态转移 $(i, i) \to (i, i+1)$, $(i, i) \to (i, [i/2])$ 和 $(i, i) \to (i, 1)$, 它们分别代表窗口增加、减半和超时这三种状态转移, 分别对应的 $P(i, i+1)$, $P(i, [i/2])$ 和 $P(i, 1)$这三个转移概率。显然, 这三个概率的和应等于 1。$P(i, j)$的取值应与 TCP协议和丢包模型都有关系, 不同丢包模型下 $P(i, j)$的计算方法是不同的。目前, 模型只能分析无记忆的丢包模型或者丢包率只与窗口大小相关的丢包模型。其它有记忆的丢包模型将使得模型的马尔可夫假设不成立。下面将计算独立等概率 (IID)丢包模型假设下的矩阵 P, 如图 4。

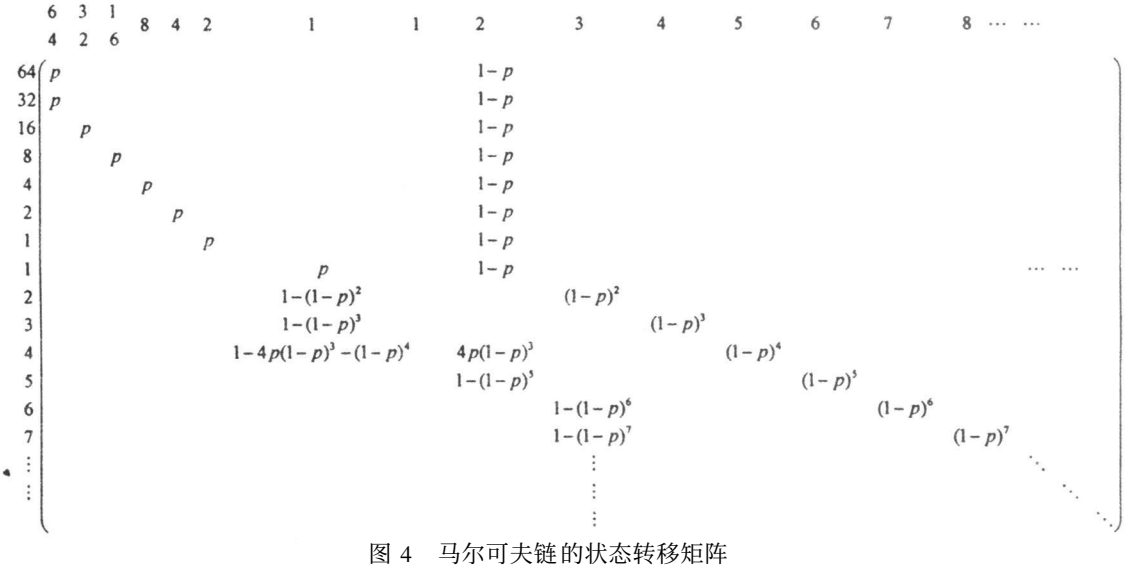


图 4　马尔可夫链的状态转移矩阵

a: 超时重传状态 (图 3中的 TO1到 TO64): 如果重传成功, TCP脱离超时重传状态, 概率值为; 如

果重传失败，TCP将超时定时器翻倍，再重传该数据包，概率为 $p$。

b窗口增加的状态转移：指 TCP状态从 $W_n$ 转移到 $W_n+1$。由于只考虑窗口线性增加阶段，那么 TCP从 $W_n = w$ 转移到 $W_{n+1} = w+1$ 的条件是收到大约 $w$ 个新的确认包。如果收端不使用延迟确认算法，每一个数据包都产生一个对应的确认包。在不考虑确认包丢失的情况下，该状态转移概率应为 $(1-p)^w$，这即是发端成功向收端发送 $w$ 个数据包的概率。如果收端使用延迟确认算法，那么大约每两个数据包产生一个确认包，所以状态转移概率是 $(1-p)^{2w}$。这即是发端成功向收端发送 $2w$ 个数据包的概率。归纳起来，$P(W_{n+1} = w+1 \mid W_n = w) = (1-p)^{bw}$。

c窗口减少的状态转移：对于不同的丢包程度，窗口减少分为减半和进入超时重传两种状态转移。当丢包严重或者窗口较小时，TCP超时重传；反之，TCP窗口减半并进入快速重传。下面我们根据窗口大小分情况讨论：

1）当 $W_n$ 等于 1，2，3时，一旦丢包 TCP就会进入 TO状态，所以，

$P(W_{n+1} = TO \mid W_n) = 1 - P(W_{n+1} = W_n + 1 \mid W_n)$，当 $W_n = 1, 2, 3$

2）当 $W_n = 4$ 时，一个丢包将使 TCP窗口减半，当出现两个以上的丢包时，TCP进入 TO状态。所以，

$P(W_{n+1} = 2 \mid W_n = 4) = 4p(1-p)^3$

$P(W_{n+1} = TO \mid W_n = 4) = 1 - P(W_{n+1} = 2 \mid W_n = 4) - P(W_{n+1} = 5 \mid W_n = 4)$

3）当 $W_n > 4$ 时，对于假设的 I.I.D丢包过程，一轮中同时丢失 3个或者更多数据包的概率是很小的，所以这里忽略 TCP发生超时重传的可能，只考虑 $W_n \rightarrow W_n/2$ 状态转移的发生，得到，

$P(W_{n+1} = [w/2] \mid W_n = w) = 1 - (1-p)^{bw}$。

## 3.2 SMP的计算

计算出 EMC的 $\{\pi_j\}$ 之后，根据附录的公式，现在计算 TCP在每个状态 止的平均停留时间：

a线性增加状态的平均停留时间 $a_j$，$a_j$ 的表达式在前面已经提到（$a_j = b \times RTT$）

b超时重传状态的平均停留时间：$a_{TO_j} = RTO \times t_{rtxshift}$ 通常情况下 $RTO$ 可以用 4 $RTT$ 来近似 [40]。

有了 $\{\pi_j\}$，根据附录可以得到 $W(t)$ 的时间平均 $\overline{W}$。

## 3.3 TCP平均发送速率与 Little's Law

前面得到了 $W(t)$ 的时间平均 $\overline{W}$，下面根据 Little's Law得到它和平均发送速率 $T$ 的关系。

Little's Law表示了一个排队系统中顾客到达率，顾客排队长度的时间平均和顾客平均等待时间这三者之间的关系。对于这条定理，系统的选取决定了上面三个量的物理意义。在这里，我们将每个数据包看作一个顾客，将整个网络看作一个服务员，将每个数据包进入网络一直到它被确认离开网络的这一段过程看作网络对数据包的服务过程。这样一个服务系统同样满足 Little's Law。

在这个系统中，TCP的平均发送速率就是顾客到达率，TCP连接在网络中数据包数量的时间平均就是顾客排队长度的时间平均，而每个数据包在网络中的停留时间就是顾客的平均等待时间。对这三者运用 Little's Law，就得到

$$T = \overline{W}/\overline{}, \quad (1)$$

其中 $T$ 是 TCP连接的平均发送速率，表示过程 $W(t)$ 的时间平均，表示数据包的在网络中的平均停留时间。

这个公式中 $T$ 和 $\overline{W}$ 的意义都很明确，需要说明的是 这个量。一个数据包在网络中的平均停留时间在这里不是指从它从发送端发出到它被接收端收到为止的这一段时间，而是指从它从发送端发出到发送端已经确认它已离开网络为止的这一段时间。这段时间可称为数据包的生命期（lifetime）。发送端确认一个数据包已离开网络有两种方式：

a如果该包没有丢失，它被成功确认就标志着它已经离开了网络。

b如果该包丢失，那么发送端重传该包的时候就标志着发送端已经确认原来那个数据包离开了网络。

对于没有丢失的数据包，生命期等于 $RTT$；对于丢失但被快速重传的数据包，生命期也近似等于 $RTT$，只有对于那些超时重传的数据包来说，它们的生命期才远大于 $RTT$。

这个等式将平均发送速率，窗口大小的时间平均以及数据包生命期三者联系起来，使我们能够通过计算 TCP的发送速率。

以下是数据包生命期的计算方法。需要说明的在丢包率不大的情况下 [$p < 0.1$]，和 $RTT$ 是非常接近的。通常，网络丢包率都在这个范围之内，所以并不需要下面的计算方法，而直接用 $RTT$ 代替 就可以了。

根据前面的分析, 可以将数据包依据其结束方式的不同分为三类, 用 $CI_i$表示第 $i$类的平均生命期, 用 $CP_i$表示第 $i$类数据包占总数据包的百分比。

a: 被成功确认的数据包: 那些没有丢失而且被收端及时确认的数据包。这类数据包的平均生命期为 RTT, 所占比例应为 $1-p$, $CI_1 = RTT$, $CP_1 = 1-p$。

b: 丢失但被快速重传的数据包: 它们的平均生命期大约为 $CI_2 = RTT \times 3/2$[4]。$CP_2$一般不能直接计算出来。考虑到丢失的数据包或者被快速重传, 或者超时重传, 所以 $CP_2 = p - CP_{timeout}$, 其中 $CP_{timeout}$是超时重传数据包的比例。

c: 超时重传的数据包: 超时重传的数据包生命期等于 $RTO \times 2^{backoff}$, backoff是超时重传系数。因此, 可将超时重传的数据包根据其 backoff系数分类, 在每类中,

$$CI_{3,\ backoff} = RTO \times 2^{backoff}$$

同时考虑 EMC模型, TCP进入一次 TO状态意味着有一个数据包的生命期为 $RTO \times x$时间; 而 TCP每进入一次线性增加状态, 意味着它发送出 $(i+1)$个数据包, 所以,

$$CP_{3,\ BACKOFF} = \pi_{TO,\ backoff} / \left(\sum (i+1) \times \pi_{Ai} + \sum \pi_{TO,\ backoff}\right),$$

$$\sum \pi_{Ai} + \sum \pi_{TO,\ backoff} = 1,$$

$$CP_{timeout} = \sum CP_{3,\ backoff}$$

三种情况讨论完毕, 使用下式计算所有数据包的平均生命期,

$$l = \sum_i CI_i \times CP_i$$

## 3.4 讨　论

以上较为详细地讨论了 SMP模型中 $\{\pi_i\}$, $\{p_i\}$, $\overline{W}$, 平均发送速率 T以及数据包平均生命期 l的计算方法。事实上, 模型中 $P_i$的确定方法可以根据需要做一定的修改。例如, 如果不考虑超时重传对 TCP的影响, 我们可以将 EMC状态转移矩阵中与超时重传状态相关的项全部去掉; 再如, 在带宽较小的链路上, TCP的窗口增加使得链路上数据包增多, 由此引起的排队延迟使 RTT随 W增加而增大, 实际上也就是 $a_i$随 W的增加而增大。这种情况可以用下面的等式来表达 $a_i = b \times RTT(W)$, 函数 RTT(W)可以通过实测 RTT与 W的相关性来得到。这就是所谓的 TCP窗口准线性 (sub— linearly) 增长; 另外, 窗口大小限制的影响也可以通过修改 $P_i$来解决。

总之以上给出的是一种通用计算方法, 具体细节可以根据需要或者丢包模型的不同而作相应调整。

# 4 结　论

TCP建模的目的是将 TCP的平均发送速率与网络丢包、延迟联系起来。本文则使用半马尔可夫模型对 TCP连接在网络中报文数量的变化过程 W(t)进行了系统的分析, 使用该模型系统地研究 TCP连接在网络中报文数量变化的过程, 用数值计算的方法得到 TCP窗口大小的概率分布。讨论了如何使用 Little's Law得到 TCP平均发送速率。得到了过程 W(t)的稳态概率分布, 平均窗口大小和平均发送速率, 同时模型还能够精确反映接收窗口对 TCP平均发送速率的影响。

参考文献:

[1] 吕国晗, 李星. TCP窗口变化过程的嵌入式马尔可夫模型[J]. 东南大学学报: 自然科学版, 2002, 增刊.

[2] 杜可亮, 李星, 吕国晗. 安全组播会议系统的设计与实现[J]. 计算机工程, 2002, 28(S1): 190—193.

[3] Abouzeid A A, S Roy and M Azizoglu. Stochastic Modeling of TCP over Lossy Links[A]. in INFOCOM[C], 2000.

[4] Altman M, H Balakrishnan and S Floyd. Enhancing TCP's Loss Recovery Using Limited Transmit[S]. RFC3042, 2001.

[5] Altman E, K Avrachenkov and C Barakat. A stochastic model of TCP/IP with stationary random losses[A]. in ACM SIGCOMM[C], 2000.

[6] Robust to Packet Reordering[J]. Comp Commun Rev, 2002, 32(1).

(责任编辑: 吴延东)