

编码 TCP 的建模与性能分析

张三峰^{1,2}, 李爽^{1,2}, 蓝翔^{1,2}, 胡永东^{1,2}

(1. 东南大学 计算机科学与工程学院, 江苏 南京 211189; 2. 东南大学 计算机网络和信息集成教育部重点实验室, 江苏 南京 211189)

摘要:研究了无线网络端到端分组丢失和编码纠错对 TCP 协议工作过程的影响机制, 基于三维 Markov 链建模描述 TCP 拥塞窗口和可用窗口的变迁过程, 在忽略慢启动阶段的条件下, 以最大窗口尺寸、端到端分组丢失率和编码冗余系数为输入参数, 通过数值计算求解编码 TCP 的吞吐量, 据此定量分析影响编码 TCP 性能的因素。基于 NS2 的模拟实验结果表明基于 Markov 链的数值计算结果具有较高的准确度。

关键词:无线 mesh 网; 传输控制; 网络编码; Markov 链

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2013)11-0112-09

Modeling and analyzing for network coded TCP

ZHANG San-feng^{1,2}, LI Shuang^{1,2}, LAN Xiang^{1,2}, HU Yong-dong^{1,2}

(1. School of Computer Science and Engineering, Southeast University, Nanjing 211189, China;

2. Key Laboratory of Computer Network and Information Integration, Ministry of Education, Southeast University, Nanjing 211189, China)

Abstract: The evolutionary process of network coded TCP against lossy links and error correcting coding were analyzed. Ignoring the slow start phase, the state transition process of congestion window and available window was described based on a three-dimension Markov chain model. With maximum window size, end-to-end packet loss rate and redundancy factor as input parameters, throughput of network coded TCP was calculated to carry quantitative analysis of factors affecting the network coded TCP performance in the numerical analysis model. Simulation results based on the NS2 simulator show that the numerical analysis model achieves fairly high accuracy.

Key words: wireless mesh network; transmit control; network coding; Markov chain

1 引言

无线 mesh 网(WMN, wireless mesh network)因其低成本和部署灵活性等内在优势, 被视作提供“最后一公里”接入的潜在解决方案, 得到了学术界和产业界的广泛关注。WMN 中越来越多的业务需要 TCP 协议的支持, 但 TCP 协议在 WMN 环境中的性能较差^[1], 原因在于一方面无线链路竞争和冲突导致更多的随机分组丢失, 另一方面 TCP 协议错误地将无线链路误码导致的随机分组丢失识别成拥塞信号, 并降低发送窗口, 使得 TCP 协议无法保持最大吞吐量对应的发送速率。

利用网络编码的纠错特性可以改善无线链路

随机分组丢失对 TCP 的影响。例如, 文献[2]提出的网络编码方法不仅能够通过一次传输发送多个分组来降低发送的次数, 而且可以通过降低信道竞争减少冲突分组丢失; 文献[3]利用网络编码技术挖掘无线路径的分集优势, 设计速率控制和拥塞避免方法, 通过估算分组丢失率, 发送冗余编码报文, 实现了较低的解码延迟并能防止 TCP 超时和重传, 提高了 TCP 的吞吐性能。这些网络编码技术的问题在于一方面实现起来较为复杂, 另一方面没有解决网络编码的成批处理特性和 TCP 的流式传输之间的矛盾。

针对该问题, Sundararajan^[4]提出了一种称为 TCP-NC(network coded TCP)的编码传输方法, 以一种简单而巧妙的方式将网络编码技术应用到现有

收稿日期: 2013-06-16; 修回日期: 2013-10-09

基金项目: 国家自然科学基金资助项目(61300200); 江苏省自然科学基金资助项目(BK2011335)

Foundation Items: The National Natural Science Foundation of China (61300200); The Natural Science Foundation of Jiangsu Province (BK2011335)

的协议栈中。TCP-NC 在传统的 TCP/IP 协议栈的 TCP 层和网络层之间插入了一个编码层（如图 1 所示）。编码层的发送端对拥塞窗口内的 TCP 报文进行随机线性编码，并按照一定的系数 R 添加冗余报文。接收端每收到一个线性无关的编码报文即回复一个 TCP 确认，而不是等到能够解码一批才回复，这样发送端的 TCP 可以正常执行滑动窗口操作。Sundararajan^[5]介绍了在 TCP-Reno 协议中实现 TCP-NC 的细节。TCP-NC 解决了 TCP 窗口滑动机制和成批编解码之间的矛盾，提高了 WMN 中 TCP 的吞吐性能。

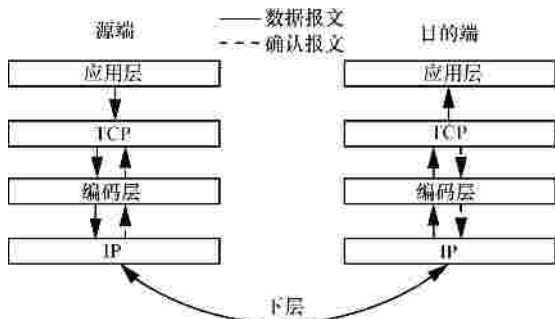


图 1 TCP-NC 协议栈

TCP-NC 在无线链路质量下降的时候，通过发送冗余纠错报文可以降低 TCP 的可见分组丢失率，提高端到端的吞吐性能，但这种做法改变了 TCP 流在拥塞控制、资源调度、公平性等方面的行为特征，需要建立分析模型，描述 TCP-NC 协议的动态性质，为相关的性能优化工作提供理论基础。

本文基于三维 Markov 链建模 TCP-NC 的工作过程，该模型在忽略慢启动过程的简化条件下，描述拥塞窗口和可用窗口的变化过程，建立端到端分组丢失率 p 、编码冗余系数 R 、最大窗口尺寸 cw_{max} 、超时周期长度、吞吐率之间的定量关系。基于本文成果，可以在给定 p 、 R 和 cw_{max} 条件下描述吞吐率、拥塞窗口变化的过程，进而为相关的性能优化工作提供理论基础，比如通过冗余系数、最大窗口尺寸等参数控制编码业务流的发送速率，使其与非编码业务流的发送速率满足公平性比例限制。

2 相关工作

TCP 协议在多种网络应用系统中都有重要作用，而且 TCP 协议的工作机制相对清晰，有利于进行形式化描述。因此，在有线或者无线网络环境下，对多种版本的 TCP 协议进行建模描述，理解 TCP

协议和网络环境之间的交互机制，为性能优化工作提供理论基础一直是研究者重点关注的问题。

Casetti 等使用马尔科夫模型建模描述单个 TCP 流的性能，然后结合标准队列分析方法建模多个 TCP 流的性能，在吞吐率、排队延迟和分组丢失率之间建立了定量关系，是较早进行有线网络 TCP 建模的工作之一^[6]。与之相比，Sikdar 等较全面地建模描述有线网络中 Tahoe、Reno 和 SACK 3 个版本的 TCP 协议的延迟和稳态吞吐性能^[7]，对超时、慢启动阶段和延迟 ACK 定时器的行为进行了更加细致的描述，因而获得了更好的近似精度。文献[8]在 TCP-Reno 建模的基础上对 TCP-NewReno 建模，考虑了快速恢复和超时行为建模问题。可以看出，有线网络中的 TCP 建模工作是逐步细化、逐步扩展到涵盖各种版本的多种特性的，其中有些工作得到了数值解，有些只有解析解，数值解的计算复杂度普遍较高，而解析解的近似精度较低。这些工作基本没有涉及到链路的随机分组丢失问题，但它们的分析方法基本被无线网络的相关研究借鉴。

Abouzeid 等较早地研究随机分组丢失环境下的 TCP 吞吐性能建模方法^[9]。更进一步地，Baccelli 在链路错误导致的随机分组丢失和拥塞导致的队列分组丢失 2 种原因并存情况下建模多个 TCP 的联合吞吐率演化过程^[10]。而文献[11]使用 Markov 更新过程建模无线网络中的 TCP-Reno 和 TCP-NewReno，重点考虑了 802.11MAC 和 DSR 路由协议对 TCP 吞吐性能的影响。无线网络中的 TCP 建模工作考虑了随机分组丢失、信道竞争、队列分组丢失等更加复杂的因素，根据已有的建模分析工作可以得到这些 TCP 协议在无线网络中的最优工作参数，但这些工作的研究对象仍然是 TCP-Reno 和 TCP-NewReno 等常用版本的 TCP 协议。如果用这些建模工作直接去预测 TCP-NC 的性能会因为协议工作机制差异导致较大的偏差。

使用网络编码技术改善无线网络中 TCP 协议的吞吐性能是最近几年的热点。但网络编码也进一步增加了 TCP 协议工作过程的复杂性，对 TCP-NC 的建模工作还比较少。Kim 等^[12,13]基于网络编码完全屏蔽随机分组丢失的假设，提出了一种简单的分析模型，在冗余系数足以弥补链路分组丢失的情况下建模 TCP-NC 的吞吐性能和最大窗口之间的关系。但在实际条件下，TCP-NC 可能需要和其他 TCP 流并存，需

要考虑各种冗余系数和拥塞控制问题,这种情况下它们的模型就显得过于简单。与之相比,本文工作可以分析各种冗余系数的情况,并能量化拥塞窗口的演化过程,这对于 TCP-NC 的速率控制、公平性保证、流间友好性优化等工作具有更重要的作用。

3 TCP-NC 的工作过程分析

3.1 TCP-NC 的编码发送和确认机制

假设端到端往返时间 RTT 远大于源端拥塞窗口内报文的发送处理时间,则可以认为 TCP-NC 按“轮 (round)”发送报文,每轮的起始时刻连续发送报文直到待确认报文数等于拥塞窗口大小,然后等待确认返回,每轮时长为 RTT 时间。TCP 每向拥塞窗口注入一个原始报文,编码层即发送 R 个(编码系数 $R - 1$,其小数部分累积至整数发送)编码报文,其中,每个编码报文是当前拥塞窗口内所有原始报文的线性组合;接收端每收到一个编码报文即检查是否可以将一个原始报文表示为已经接收到的编码报文和未确认报文的线性组合,若可以即认为可以“seen”^[4]该原始报文,并对其回复一个确认报文;发送端收到确认报文后执行拥塞窗口滑动操作。

图 2 描述编码冗余系数为 1 时, TCP-NC 的编码发送和确认过程:接收端收到 q_1 ,即回复对 p_1 的确认;而当收到 q_4 时,可得 $p_2 = (q_4 - q_1 - p_3 - 2p_4)/3$,回复对 p_2 的确认。可以看出:

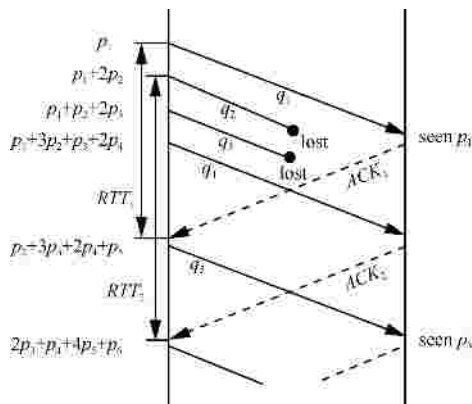


图 2 TCP-NC 的编码和确认

- 1) TCP-NC 中 TCP 所看到的 RTT 可能大于实际的端到端往返时间;
- 2) 本轮丢失的报文可以延迟到下一轮确认,而下一轮中可以立刻得到确认的报文数目会少于接收端收到的编码报文数目;
- 3) 若某一轮发出的报文全部丢失则出现超时。

3.2 TCP-NC 的窗口变化过程

TCP 协议的工作过程一般包括慢启动、拥塞避免、超时重传等几个阶段。在慢启动阶段, TCP 每收到一个确认报文,按照 $CW = CW + 1$ 方式增加拥塞窗口 CW ; 当 CW 超过慢启动门限 $Ssthresh$ 的时候,进入拥塞避免阶段,按照 $CW = CW + 1 / CW$ 方式增加; 当有报文在重传超时时间(RTO)内未被确认,则进入超时重传阶段,同时 CW 置为 1, $Ssthresh$ 置为 2。在慢启动阶段,每过一轮,拥塞窗口翻倍;在拥塞避免阶段,每过一轮拥塞窗口加 1。慢启动阶段经过有限的轮次即进入拥塞避免阶段,因此在 TCP 建模的相关工作中通常忽略慢启动阶段。

笔者同样从拥塞避免阶段开始分析 TCP-NC 的拥塞窗口变化。以图 3 为例分析变化过程,每一轮上方的数字表示 TCP 在该轮次注入拥塞窗口的报文数目。TCP-NC 的拥塞窗口可能经历如下的变化过程: 设第 j 轮的拥塞窗口为 4,在报文全部确认之后,第 $j+1$ 轮的拥塞窗口变为 5; 设第 $j+1$ 轮发送的报文中任一个丢失,编码作用使得前 4 个报文得到确认,而最后一个报文不能在本轮被确认(不能在本轮得到确认的报文数目定义为累计未确认报文数 LS); 第 $j+1$ 轮返回的 4 个确认使得第 $j+2$ 轮的拥塞窗口值 CW 增至 $5 \frac{4}{5}$,而最多未确认报

个数 $\lfloor CW \rfloor$ 为 $\lfloor 5 \frac{4}{5} \rfloor = 5$ (符号 $\lfloor x \rfloor$ 表示取不超过 x 的最大整数); 由于在第 $j+2$ 轮开始时刻,累计未确认报文数 $LS=1$,第 $j+2$ 轮 TCP 仅能向拥塞窗口注入 $\lfloor CW \rfloor - LS = 5 - 1 = 4$ 个新的报文(定义 $EW = \lfloor CW \rfloor - LS$ 为有效窗口); 若第 $j+2$ 轮发送的 4 个报文也有 1 个丢失,则第 $j+3$ 轮的拥塞窗口 CW 变为 $6 \frac{2}{6}$,累积未确认报文数 $LS=2$,有效窗口 EW 为 $4 - 2 = 2$; 随着轮次的推进,有效窗口可能逐渐减小,若第 i 轮有效窗口内的报文全部丢失,则出现超时; 超时之后,将重新进入拥塞避免阶段,拥塞窗口和有效窗口重置为 1。

另外,注意到在编码的作用下,接收端收到任何“新的”编码报文后均会回复序列号递增的确认报文,因此不会有乱序到达的报文以致出现快速重传或快速恢复的情况。

为描述 TCP-NC 拥塞窗口的周期性变化规律,本文定义了一个 TCP-NC 周期为从第 1 轮开始到 TCP

超时重传计时器时为止，它经历了拥塞窗口逐渐增大，但有效窗口从 1 开始先增加后降为 0 的过程。

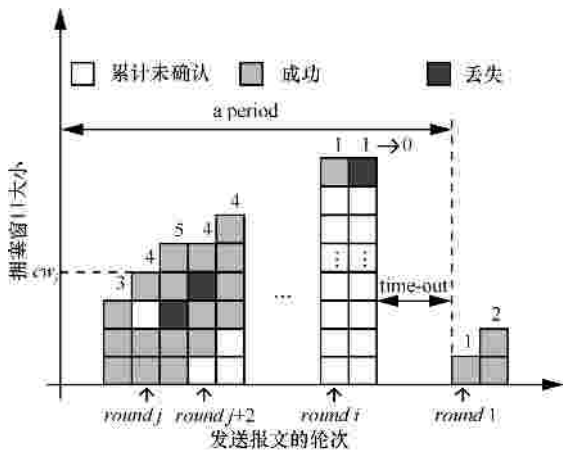


图 3 TCP-NC 窗口变化(R=1)

若编码系数大于 1，则可能减少累积分组丢失数。如图 4 所示，第 j+2 轮 TCP 向拥塞窗口注入 4 个新的原始报文，但编码层可能发出 5 个线性无关的编码报文（第 j+1 轮的最后一个报文未被确认，仍然在缓存区中参与编码）；虽然第 j+2 轮丢失了一个编码报文，但成功发送了 4 个编码报文，累积未确认数 LS 仍然为 1；若第 j+2 轮发送的 5 个编码报文全部成功到达，则累积未确认数 LS 降为 0；也就是说，若编码系数足以弥补分组丢失，则 TCP 拥塞窗口将逐步增加到窗口上限。

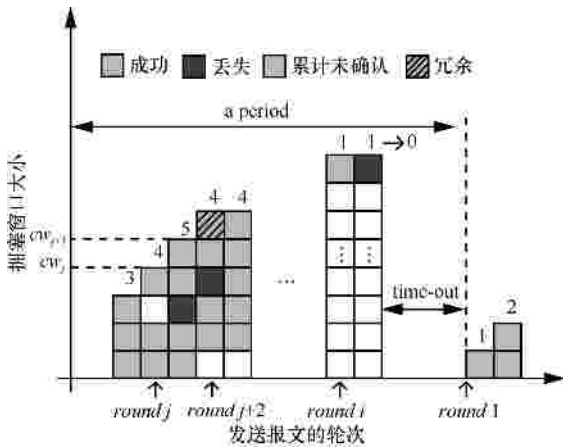


图 4 TCP-NC 窗口变化(R>1)

4 TCP-NC 的端到端吞吐率分析

4.1 描述 TCP-NC 窗口变化的 Markov 模型

假设 TCP-NC 的端到端路径上的往返分组丢失率为 p，分组丢失的原因可能是因信道误码、报文

碰撞或者队列溢出导致的数据报文丢失。可以用三维矢量表示 t 轮时 TCP-NC 的状态

$$S_t = \{(m_t, n_t, EW_t) | t \in N^+, m_t \in N^+, n_t, EW_t \in N^0, n_t < m_t, EW_t \leq m_t\} \quad (1)$$

其中， m_t 和 n_t 分别表示 t 轮时拥塞窗口的整数部分和小数部分； $CW_t = m_t \frac{n_t}{m_t}$ 表示 t 轮内发送端的拥塞窗口； EW_t 则表示 t 轮时的有效窗口；考虑到 TCP 协议发送报文时仅用到拥塞窗口的整数部分，t 轮时累积未确认的报文数量可以表示为 $LS_t = m_t - EW_t$ 。

图 5 表示 S_t 的变化过程（其中，纵轴为拥塞窗口 $CW = (m, n)$ ，横轴为有效窗口 EW ）：在第 t 轮，状态为 $S_t = (m_t, n_t, EW_t)$ 的发送端可以发送 EW_t 个报文，可能返回介于 $[0, EW_t]$ 之间的确认报文，根据返回确认报文个数的不同，发送端在 t+1 轮进入不同的状态。比如在 t=1 的起始时刻， $S_1 = (1, 0, 1)$ ，发送一个编码报文，若该报文对应的 ACK 未返回（概率为 p）则进入状态 $(1, 0, 0)$ ，若该报文对应的 ACK 返回（概率为 1-p），则进入状态 $(2, 0, 2)$ 。由于返回确认报文的个数仅和当前状态及当前时刻的链路质量有关，而和历史状态无关，因此该随机过程具有 Markov 性。

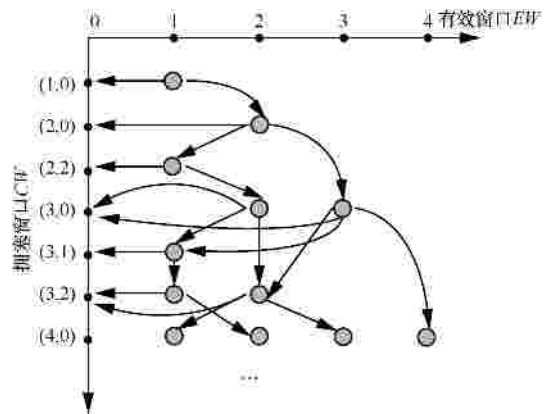


图 5 TCP-NC 的 Markov 状态转移图

其中， CW 轴上所有状态 $(m,n,0)$ 的 LS 取值不小于 1，而 EW 取值为 0，因此这些状态不可能有确认报文返回，也不可能迁移到其他状态，可以认为以概率 1 转移到自身，即为吸收状态。到达 CW 轴上的吸收状态后，源端将不能再发送任何报文，而是等待超时后将 CW 和 EW 重置为 1，从未确认的报文开始重新传输。从起始状态 $(1,0,1)$ 出发，到 CW 轴上的某个吸收状态为止，就是拥塞窗口的一个变化周期。

一个周期的时长为从起始状态 $(1,0,1)$ 出发

到达任一吸收状态加上随后等待重传的超时时间 (RTO)。每步转移对应一轮的持续时间,即为链路层的端到端往返时间 RTT 。从 $(m,n,k), k \geq 0$ 出发,发送 k 个报文后,根据失败报文数目 $j(0 \leq j \leq k)$,转移至不同的状态 (a,b,c) ,其中, $b=(n+k-j)\%m, a=m+\lfloor(n+k-j)/m\rfloor, c=a-(m-k)-j$,其对应的单步转移概率为

$$q_{m,n,k;a,b,c} = C_k^j p^j (1-p)^{k-j} \quad (2)$$

除了到这 $k+1$ 种状态的转移之外,到其他状态的转移概率为 0。吸收状态以概率 1 转移到自身,即 $q_{m,n,0;m,n,0} = 1$ 。据此可以确定转移概率矩阵 Q 的每个元素。

4.2 计算 TCP-NC 的期望吞吐率

长期来看,从起始状态 $(1,0,1)$ 出发,将以不同的概率经过不同的转移路径到达 CW 轴上的某个吸收状态。在编码冗余系数为 1,无发送端最大拥塞窗口限制的条件下,TCP-NC 单位时间内的期望吞吐率可以表示为

$$E(T) = \sum_{\forall s=(m,n,0) \in S} \sum_{path_i \in P(s)} (P_{s,path_i} \frac{N_{s,path_i}}{D_{s,path_i} + RTO_{s,path_i}} \cdot PSize) \quad (3)$$

其中, $s=(m,n,0)$ 表示 CW 轴上的某个吸收状态, $path_i \in P(s)$ 表示从起始状态到达吸收状态 s 的一条转移路径, $P_{s,path_i}$ 表示从起始状态沿 $path_i$ 到达 s 的概率, $N_{s,path_i}$ 表示这条路径上成功发送的报文数目, $D_{s,path_i}$ 表示该转移路径持续的时间, $RTO_{s,path_i}$ 表示沿 $path_i$ 到达 s 后的超时时间。

$P_{s,path_i}$ 取值为 $path_i$ 上所有转移概率的乘积

$$P_{s,path_i} = \prod_{x=1}^{|path_i|} q_{s_x;s_{x+1}} \quad (4)$$

根据拥塞窗口的变化过程可知

$$\begin{aligned} N_{s,path_i} &= 1 + 2 + 3 + \dots + (m-1) + n \\ &= m(m-1)/2 + n \end{aligned} \quad (5)$$

$$D_{s,path_i} = |path_i| RTT \quad (6)$$

其中, $|path_i|$ 表示路径上的转移次数。

如前所述,TCP-NC 中 TCP 看到的往返时间不同于链路层的 RTT 根据 TCP 的 RTO 估算方法 RTO 将随着 TCP 看到的往返时间的变化而变化,这里笔者用 $m \cdot RTT$ 近似超时时间 RTO 即 $RTO_{s,path_i} \approx m \cdot RTT$, m 是吸收状态拥塞窗口值的整数部分。

对于在编码冗余系数 R 为 1,拥塞窗口限制为

CW_{max} 的情况,作者在图 5 中不限制 m 的取值,但修正有效窗口的计算方法为 $EW = \min(m, CW_{max}) - LS$,即当 m 超过 CW_{max} 的时候根据 CW_{max} 计算有效窗口。

对于 $R > 1$ 且冗余报文不足以完全弥补随机分组丢失的情况,可以将其简化为 $R=1$ 且 TCP 可见的端到端分组丢失率 $p' = 1 - R(1-p)$ 的情况。若冗余报文足以弥补随机分组丢失时,则按照文献[12]的方法计算。

类似地,可得 $t \rightarrow \infty$ 时,所有超时时刻对应拥塞窗口值的期望为

$$E(CW) = \sum_{\forall s=(m,n,0) \in S} \min(m \frac{n}{m}, CW_{max}) \cdot \sum_{path_i \in P(s)} P_{s,path_i} \quad (7)$$

以设置最大拥塞窗口限制 CW_{max} , $R=1$ 为例,TCP-NC 的期望吞吐率和超时时刻的期望窗口可按算法 1 迭代计算。

算法 1 迭代计算 TCP-NC 期望吞吐率

输入:最大拥塞窗口 cw_{max} ,端到端随机分组丢失率 p 、 m 的上限 M 。

输出:期望吞吐率 $E(T)$,期望最大窗口 $E(CW)$ 。

- 1) 根据 cw_{max} 和 M 初始化 Markov 链 S 的状态空间及其初始分布 P
- 2) 初始化转移概率矩阵 Q ,吸收状态的初始分布 P' 以及 $E(T)$ 、 $E(cw)$
- 3) $cnt \leftarrow 0$
- 4) $cnt \leftarrow \frac{M(M-1)}{2} + 1$ //最大转移次数
- 5) while $cnt < \overline{cnt}$ do
- 6) $P \leftarrow PQ$
- 7) for $s = (m,n,0)$ in S
- 8) if $P'(s) \neq P(s)$
- 9) $E(T) += \frac{m(m-1)/2 + n}{count + \min(m, cw_{max})} (P_s - P'_s)$
- 10) $P'(s) \leftarrow P(s)$
- 11) endif
- 12) $cnt ++$
- 13) endfor
- 14) endwhile
- 15) for $s = (m,n,0)$ in S
- 16) $E(cw) += \min(m \frac{n}{m}, cw_{max}) P'(s)$
- 17) endfor
- 18) return

4.3 计算精度和复杂度分析

状态空间 S 可能包括无限个非吸收状态和吸收

状态，这导致算法 1 的计算复杂度很高，考虑到一定的端到端分组丢失率条件下，吸收状态的吸收概率在 m 大于一定取值后随 m 的增加而降低，因此，算法 1 设置 m 的上限为 M ，忽略 m 值较大情况，而且根据实验结果设置满足复杂度和精度要求的 M 。在此条件下，三维 Markov 链的所有可能状态个数 $|S|$ 的上限为 $M \cdot M \cdot CW_{\max}$ 。算法 1 的迭代计算使用到了两层嵌套的循环以及一个 $|S| \times |S|$ 的转移概率矩阵。算法的时间、空间复杂度均为 $O(M^4 \cdot CW_{\max}^2)$ ，其中，转移概率矩阵为稀疏矩阵。可以看出 M 对算法 1 的时空复杂度具有很大影响， M 越大算法的复杂度越高，但精度也越高。这里使用吸收状态的累计概率 $\underline{P} = \sum_{m=0}^M P_{(m,n,0)}$ 表示算法 1 的精度， \underline{P} 越接近 1，算法的精度越高。

5 模拟实验和模型验证

5.1 实验配置

笔者在 NS-2.34^[14] 上模拟实现 TCP-NC 协议，然后在不同的网络拓扑和网络参数条件下进行模拟实验，并统计 TCP-NC 和平台自带的 TCP-Reno 协议的吞吐性能。再将模拟结果和根据本文建模计算所得的 TCP-NC 吞吐率、以及根据文献 [6,12] 2 种相关模型计算所得的吞吐率结果进行对比，验证本文建模工作的有效性，并分析 TCP-NC 的性能特点。

首先在 NS-2.34 平台中 TCP-Reno 协议的基础上模拟实现了 TCP-NC 协议，具体做法是在 NS-2.34 的 TCP-Reno 协议代码的 `recv` 函数和 `send` 函数的处理过程中调用网络编码有关的处理函数：在发送端 TCP 层截获发往网络层的数据报文；对拥塞窗口内的所有未确认报文进行随机线性编码，然后按照冗余系数调整编码报文发送数目；在接收端截获网络层交付的编码报文，判断“Seen”状态的报文序号，然后伪造并回送确认报文到发送端。

在网络编码的实现方面，笔者选用有限域 (GF256) 作为数域，并预先计算有限域上的加法表、乘法表与逆表以便通过查表的方式进行编解码运算。编码时从有限域上随机生成非零系数，再对窗口内的报文进行线性混合。解码时先求系数逆矩阵，再与编码报文矩阵相乘。在模拟时候常常不需要对实际数据进行编解码，这时只需要通过高斯消元法求出秩，再根据秩与编码报文数目比较判断编

码报文是否线性无关即可。

实验拓扑为如图 6 所示的 4 跳链式拓扑以及两条流交叉的拓扑。在源节点和目的节点之间部署一条 FTP 业务流，FTP 负载饱和，即业务流的发送速率仅由 TCP 的拥塞窗口限制。传输协议使用 NS-2.34 自带的 TCP-Reno 或者 TCP-NC，TCP-NC 的中间节点不编码。路由协议使用手工配置的转发表，MAC 层使用 IEEE 802.11a 协议，每条链路的带宽设为 1 Mbit/s，关闭 RTS/CTS 以及链路层的自动重传。报文大小设置为 1 KB。另外，如未额外声明，实验均有 $CW_{\max} = 50$ 。

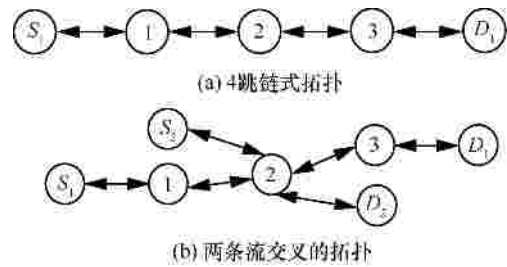


图 6 实验的网络拓扑

5.2 编码冗余系数对模型精度的影响

将实际链路层端到端分组丢失率设为 p' ， $R' > 1$ 的情况转化为 $R=1$ 且 TCP 可见的端到端分组丢失率为 $p = 1 - R'(1 - p')$ 的情况，再利用分析模型的式 (3) 计算其吞吐率。图 7 验证这种转化方法对精度的影响。图 7 中， $R=1$ 对应的曲线为根据数值分析模型式 (3) 计算的不同端到端分组丢失率 p 对应的吞吐率； $R=1.1$ 和 $R=1.2$ 对应的曲线表示根据 NS2 模拟的统计结果所得的 TCP 可见分组丢失率和吞吐率的关系，其中，TCP 可见分组丢失率按照 $p = 1 - R'(1 - p')$ 转换。

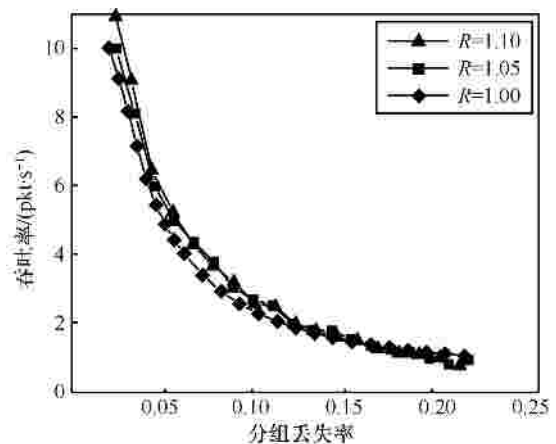


图 7 TCP-NC 的编码冗余系数对精度的影响

从图 7 中可以看出,这种简化在较高的分组丢失率条件下对精度的影响很小,但在较低的端到端分组丢失率(2%)的时候,NS2 模拟所得的实际吞吐率比数值分析结果要高 10%左右,这是因为冗余编码虽然使得平均分组丢失率和 $R=1$ 对应的分组丢失率相当,但根据文献[4]实现的 TCP-NC 是在冗余系数的小数部分累积为整数之后才发送冗余报文,因此,实际上 $R>1$ 的每个报文的分组丢失率是大于 $R=1$ 的情况的,这使得 TCP-NC 可以较早地结束发送速率较低的阶段。

5.3 计算复杂度和精度的关系

因为数值分析模型计算各个吸收状态对应吞吐率的概率平均值,因此,式(3)统计的吸收状态越多,吞吐率的计算精度越高,但计算复杂度也越高。图 8 表示 m 的上限值 M 和精度的关系。 M 为限制计算复杂度的指标,累计吸收概率 P 则衡量了吞吐率的精度。可以看出,对每种分组丢失率 M ,超过一定界限之后 P 都是接近于 1 的,超过此界限之后提高 M 对改善精度没有太大的贡献。同时,分组丢失率越低这个界限值越大,实现给定精度所需的计算开销也越高。因此,一般选取累计吸收概率大于 0.95 的 M 作为上限值。

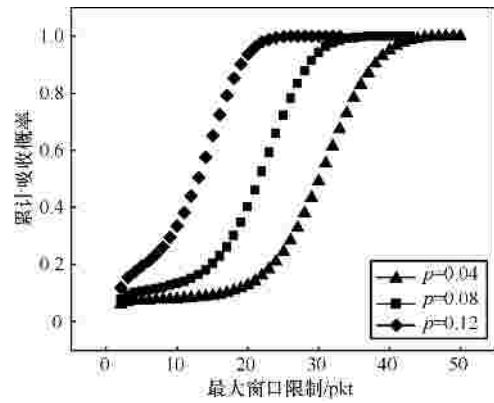


图 8 计算复杂度和精度的关系

5.4 模型计算和模拟结果的对比较证

图 9(a)在图 6(a)的拓扑中设置了不同的端到端分组丢失率,对比根据式(7)计算的平均拥塞窗口和根据模拟实验统计结果所得的平均拥塞窗口。图 9(b)在上述条件下直接对比吞吐性能。图 9(c)在图 6(a)的拓扑中设置不同的最大拥塞窗口限制,然后对比根据式(7)计算的吞吐率和根据模拟实验统计结果所得的吞吐率。图 10 在图 6(b)的拓扑中存在两条 TCP 流的情况下重复上述对比实验。其中,图 10(a)因为两条 TCP 流的端到端分组丢失率相同,计算公式也相同,曲线重合,因此只列出一条流的曲线。

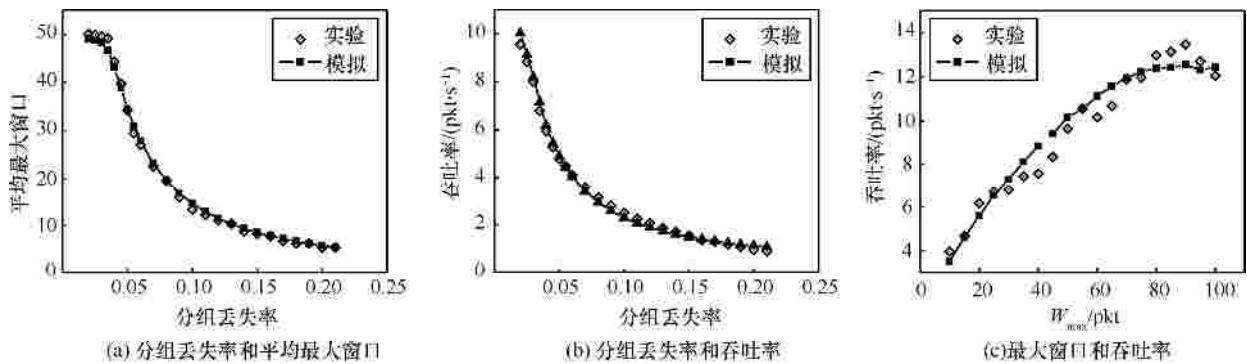


图 9 图 6(a)的拓扑中不同的分组丢失率和最大窗口限制条件下的对比

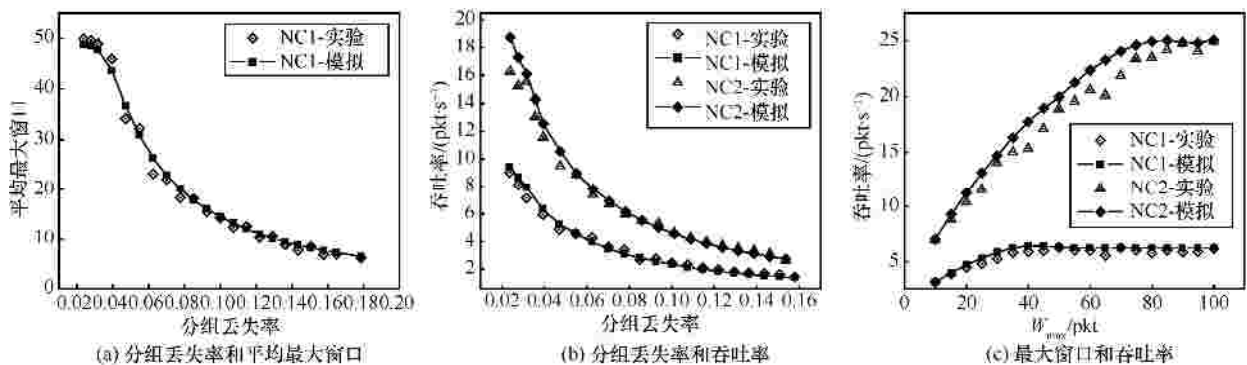


图 10 图 6(b)的拓扑中不同的分组丢失率和最大窗口限制条件下的对比

可以看出数值分析模型对拥塞窗口期望的预测和模拟实验的统计结果吻合(误差在 2% 以内),表明本文模型的预测精度比较高。而且作者发现,吻合程度随着分组丢失率的增加而上升,这是因为较低分组丢失率时, TCP-NC 所能达到的拥塞窗口最大值较大,但数值分析模型受计算复杂度的限制忽略了超过上限 M 的吸收状态对期望吞吐率的影响。

5.5 与相关工作的对比

考虑到直接相关的工作较少,作者选取 2 个相近的工作进行对比。一个是 Abouzeid 等提出的无线网络中考虑端到端随机分组丢失的 TCP Reno 的建模工作^[6],一个是 Kim 提出的假设编码完全掩盖分组丢失的 TCP-NC 建模^[12]。由于两者都只能计算吞吐率,这里只在拓扑(a)中对比各模型计算结果和模拟结果的偏差程度。图 11 为在各种编码系数和分

组丢失率的组合情况下的实验结果。其中, Kim 的模型和最大窗口有关,在每个窗口都是同样的曲线。可以看出: Abouzeid 等的模型相当于用 TCP Reno 模型来预测 TCP-NC 协议的吞吐率,因此在编码系数大于 1 时的偏差较大;而 Kim 的模型实际上是一条稳定状态下(维持在最大拥塞窗口限制水平上)的直线,在每个图中都是当分组丢失率低于某个上限的时候,也就是编码可以掩盖分组丢失的时候和模拟结果吻合,但若不能掩盖分组丢失则偏差很多;而本文的模型在不同的冗余系数下,在较大分组丢失率的范围内都较为准确地预测了 TCP-NC 流的吞吐率。

5.6 基于不同模型的 TCP-NC 的友好性对比

根据本文的模型可以得出更具流间友好性的编码冗余系数 R 。图 12 所示为带宽 0.05 Mbit/s,端到端分组丢失率为 0.06,在图 6(a)的拓扑中运行同

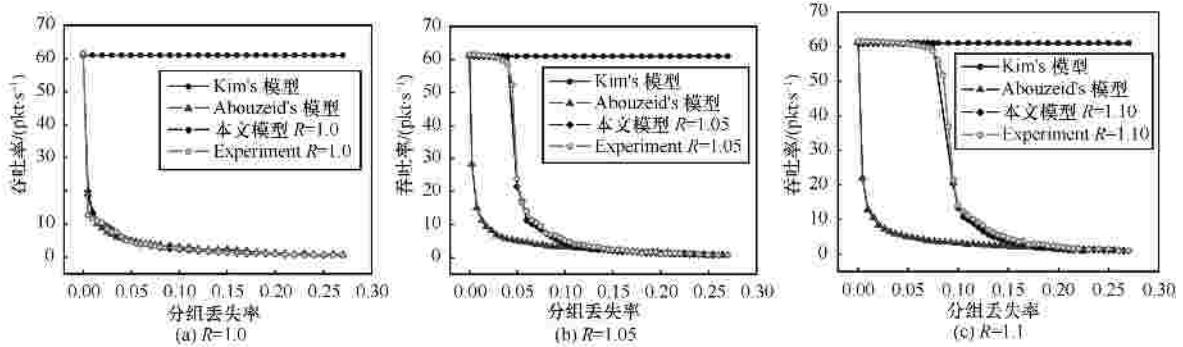


图 11 不同分组丢失率情况下和相关工作的对比

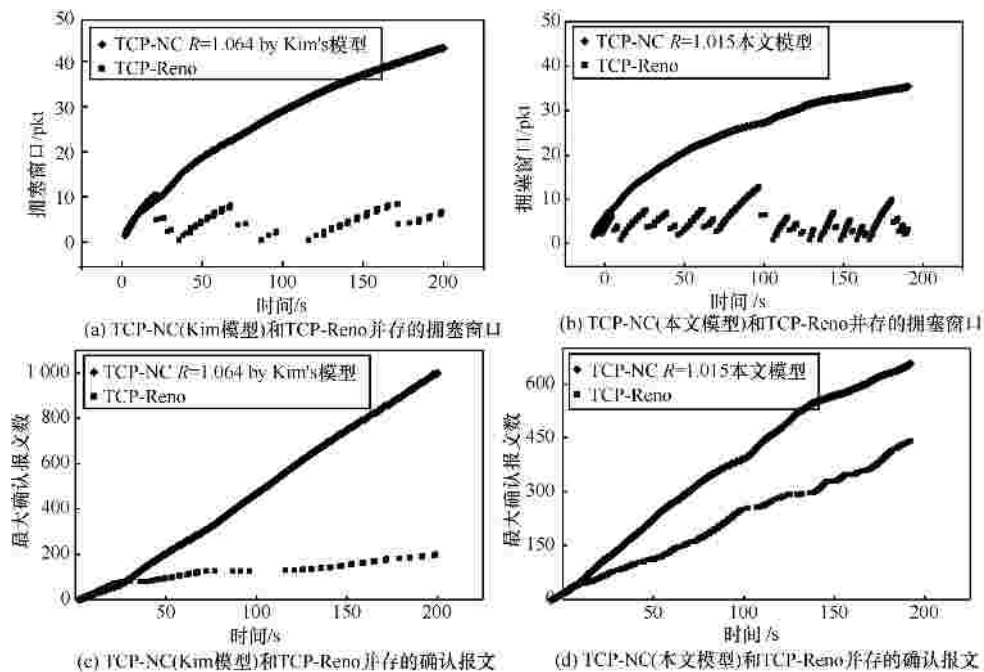


图 12 TCP-NC 和 TCP-Reno 共存时的友好性比较

源同宿的 TCP-NC 和 TCP-Reno 的两条流的情况下的性能对比结果。其中,图 12(a)和图 12(b)对比拥塞窗口大小,图 12(c)和图 12(d)对比确认报文序号变化规律,图 12(a)和图 12(c)中根据 Kim 的模型计算得到 TCP-NC 流的冗余系数为 1.064,图 12(b)和图 12(d)中根据本文的模型计算得到 TCP-NC 流的冗余系数为 1.015。从图 12 中可以看出,当带宽不足的时候,在 2 种不同的冗余系数下,TCP-NC 和 TCP-Reno 两条流之间的吞吐率的比值分别为 4.963 和 1.415,本文模型计算出的编码冗余系数使得 TCP-NC 的流间友好性更好,较好地控制了两条流占用带宽资源的差别。

6 结束语

本文提出了一种给定端到端分组丢失率和最大拥塞窗口计算 TCP-NC 吞吐率的数值分析模型,模型可以较精确地计算 $R=1$ 时 TCP-NC 的平均最大窗口、平均吞吐率等性能,而对于 $R > 1$ 的情况则通过转换 TCP 可见分组丢失率来归为 $R=1$ 的情况。数值分析模型在较高的端到端分组丢失率情况下具有较好的计算精度和较低的计算开销。获取三维 Markov 模型的解析解和降低计算复杂度是下一步需要努力的方向。

参考文献:

- [1] C'ACERES R, IFTODE L. TCP performance in IEEE-802.11-based ad hoc networks with multiple wireless lossy links[J]. IEEE Journal on Selected Areas in Communications, 2007,13(5):1329-1342.
- [2] HUANG Y, GHADERI M, TOWSLEY D, *et al.* TCP performance in coded wireless mesh networks[A]. Proceedings of IEEE SECON[C]. 2008. 179-187.
- [3] GHEORGHU S, LOPEZ A, RODRIGUEZ T P. Multipath TCP with network coding for wireless mesh networks[A]. Proceedings of IEEE ICC 2010[C]. Cape Town, South Africa, 2010.1-5.
- [4] SUNDARARAJAN J K, SHAH D, MEDRAD M, *et al.* Network coding meets TCP[A]. Proceedings of IEEE INFOCOM[C]. Rio de Janeiro, Brazil, 2009.280-288.
- [5] SUNDARARAJAN J K, SHAH D, M'EDARD M, *et al.* Network coding meets TCP theory and implementation[J]. Proceedings of IEEE,2011, 99(3):490-512.
- [6] CASETTI C, MEO M. A new approach to model the stationary behavior of TCP connections[A]. Proceedings of IEEE INFOCOM[C]. Tel Aviv, Israel, 2000. 367-375.

- [7] SIKDAR B, KALYANARAMAN S, VASTOLA K S. Analytic models for the latency and steady-state throughput of TCP Tahoe, Reno, and SACK[J]. IEEE/ACM Transactions on Networking, 2003, 11(6): 959-971.
- [8] PARVEZ N, MAHANTI A, WILLIAMSON C. An analytic throughput model for TCP newReno[J]. IEEE ACM Transactions on Networking, 2010,18(2):448-461.
- [9] ABOUZEID A A, ROY S, AZIZOGLU M. Stochastic modeling of TCP over lossy links[A]. Proc of IEEE INFOCOM[C]. Tel Aviv, Israel, 2000.1724-1733.
- [10] BACCELLI F, KIM K B. TCP throughput analysis under transmission error and congestion losses[A]. Proc of IEEE INFOCOM[C]. Hong Kong, China, 2004.2833-2843.
- [11] LI X, KONG P Y, CHUA K C. TCP performance in IEEE 802.11-based ad-hoc networks with multiple wireless lossy links[J]. IE Transactions on Mobile Computing, 2007,6(2):1329-1342.
- [12] KIM M, M'EDARD M, BARROS J. Modeling network coded TCP throughput: a simple model and its validation[A]. Proceedings of IEEE ICST2011[C]. Berlin, Germany, 2011. 131-140.
- [13] KIM M, KLEIN T, SOLJANIN E, *et al.* Modeling Network Coded TCP: Analysis of Throughput and Energy Cost[R]. MIT Technical Report, 2012.
- [14] NS2[EB/OL]. <http://www.isi.edu/nsnam/ns/>, 2013.

作者简介:



张三峰(1979-),男,山东金乡人,东南大学讲师,主要研究方向为网络编码、无线 mesh 网络、机会网络等。



李爽(1991-),女,河南新蔡人,东南大学学生,主要研究方向为网络编码、无线 mesh 网络等。

蓝翔(1993-),男,江苏扬州人,东南大学学生,主要研究方向为网络编码、无线 mesh 网络等。

胡永东(1973-),男,安徽宿松人,东南大学博士生,主要研究方向为无线网络、网络服务质量保证、网络性能分析等。