

数据链路层停等ARQ协议的最佳帧长近似解

孙翔¹, 陈松明²

(1. 成都信息工程学院通信工程系 成都 610225; 2. 电子科技大学出版社 成都 610054)

【摘要】分析了通信信道中起伏噪声和突发噪声对数据传输的影响, 推导出了数据帧出错的概率即误帧率与信道误比特率的定量关系。为克服信道噪声引起的帧传送错误, 可在数据链路层采用面向连接的协议, 以保证向上层用户提供可靠交付服务。其中采用停止等待ARQ协议是最简单的, 它适用于许多特定场合。对于该协议, 该文详细推导了在信道误比特率较小的情况下的最佳帧长(使数据总传输时间最短)的计算方法及其近似表达式, 并分析讨论了该近似式的物理意义和适用范围。

关键词 误帧率; 最佳帧长; 帧头和超时开销; 停等ARQ协议

中图分类号 TP393.04

文献标识码 A

Approximate Solution of Optimal Frame Size in S-W ARQ Protocol of DLL

SUN Xiang¹, CHEN Song-ming²

(1. Dept. of communication, Chengdu University of Information and Technology Chengdu 610225;

2. Press, University of Electronic Science and Technology of China Chengdu 610054)

Abstract This paper considers the data transmission system with undulating noise and sudden noise. The quantitative relationship between the misuse frame rate and bit-error-ratio is derived. In this noisy channel, the connection-oriented protocol of data link layer is used to provide reliable delivery service for upper users. The calculation method and the approximate expression of optimal frame size in S-W ARQ protocol of data link layer are deduced in detail in the condition that the bit-error-ratio of channel is relatively small. Moreover, the physical meaning and the application area are also analyzed and discussed.

Key words frame error ratio; optimal frame size; overhead of frame head; stop-and-wait ARQ protocol and time out

在通信信道中起伏噪声和突发噪声将引起数据帧的传输错误^[1-3]。在数据链路层采用停止等待自动重发请求(Automatic Repeat reQuest, ARQ)协议, 向上层用户提供面向连接的服务^[4-6]。协议通过在收发双方之间对每帧的校验确认和出错重传机制, 实现数据的可靠交付。本文借鉴文献[7-10]的概念和方法, 提出了在已知两个节点之间的信道误比特率的情况下, 采用该协议实现数据传输最佳帧长(使数据总传输时间最短)的计算方法。

1 误帧率与误比特率的关系式

信道中的噪声引起数据的比特出错和帧出错(即1帧中至少有1个比特出错)。对于信道中常见的起伏噪声, 每个比特的出错概率可视为独立的。误比特率记为 P_b , 误帧率记为 P_f , 设数据帧长为 l_f (即数据部分加上控制信息), 则有:

$$P_f = 1 - (1 - P_b)^{l_f} = 1 - [1 - l_f P_b + P_b^2 l_f(l_f - 1)/2 - P_b^3 l_f(l_f - 1)(l_f - 2)/6 + \dots]$$

当 $l_f P_b \ll 1$ 时,

$$P_f \approx l_f P_b \quad (1)$$

实际上, 当 $l_f P_b = 0.1$ 时, 按式(1), 误差 0.5%; 当 $l_f P_b = 0.2$ 时, 按式(1), 误差 2%。

对于脉冲噪声或突发噪声引起的突发性误码, 当其发生时, 一连串的比特都要出错。文献[1]提出: 在产生突发性误码时, 误帧率与帧长成正比。

这样, 当差错率很低时, 式(1)可适用于信道中的起伏噪声和突发噪声。

2 停止等待ARQ协议的数学模型

本文采用如图1所示的文献[4]中停止等待ARQ协议的数据帧和确认帧的发送时间关系图, 作为求

解最佳帧长的数学模型。设结点A向结点B发送数据帧。结点B只发送确认帧而不发送否认帧,也不发送自己的数据帧。

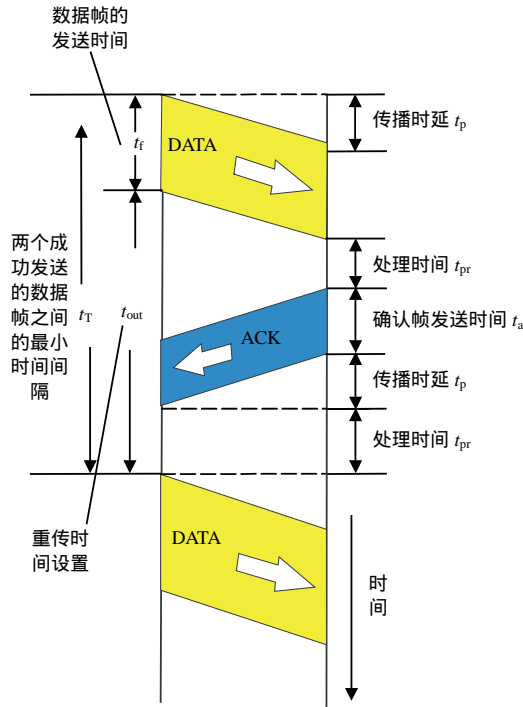


图1 停止等待协议中的数据帧

根据图1, 可分别求出数据帧的发送时间 t_f 和设置的重传时间 t_{out} 为:

$$t_f = l_f / R \quad (2)$$

$$t_{out} = t_p + t_{pr} + t_a + t_p + t_{pr} = t_a + 2t_{pr} + 2t_p \quad (3)$$

式中 l_f 为数据帧的长度(bit); R 为数据的发送速率(bit/s); t_p 为信道的传播时延; t_{pr} 为节点的处理时间; t_a 为确认帧的发送时间。两个发送成功的数据帧之间的最小时间间隔为:

$$t_T = t_f + t_{out} = t_f + t_a + 2t_{pr} + 2t_p \quad (4)$$

如发生差错, 则成功发送一个数据帧所需时间显然要超过 t_T 。

3 最佳帧长关系式的推导

设数据帧出现差错(包括帧丢失)的概率为 P_f , 由于确认帧很短, 出差错的概率相对小得多, 可忽略; 此外, 数据帧出错后允许多次重传。根据上述模型, 可以得到正确传送一个数据帧所需的平均时间 t_{av} 与 t_T 、 P_f 的关系式:

$$t_{av} = t_T / (1 - P_f) \quad (5)$$

不难看出, 当帧出错率增大时, t_{av} 也随之增大; 当信道无差错即 $P_f=0$ 时, $t_{av}=t_T$ 。

本文计算误码率为 P_b 的信道, 使用上述停止等

待ARQ协议的数据传送时间。设数据总长度为 N , 帧头的长度为 H , 帧的数据位长度为 n , 数据传送速率为 R 。由式(1)可得:

当 $l_f P_b \ll 1$ 时,

$$P_f = l_f P_b = (H+n)P_b \quad (6)$$

由式(4)~(6)可得:

$$t_{av} = (t_f + t_a + 2t_{pr} + 2t_p) / (1 - nP_b - HP_b) \quad (7)$$

由式(2)可得:

$$t_f = \frac{H+n}{R} \quad (8)$$

另外,

$$t_a = \frac{A}{R} \quad (9)$$

式中 A 为应答帧长。对于总长度为 N 的数据平均传输时间Delay, 有以下关系式:

$$\text{Delay} = t_{av} \frac{N}{n} \quad (10)$$

把式(7)~(9)代入式(10), 可得:

$$\begin{aligned} \text{Delay} &= \frac{N}{R} [n+H+A+2(t_{pr}+t_p)R] / n(1-nP_b-HP_b) = \\ &= \frac{N}{R} (n+C) / n(1-nP_b-HP_b) \end{aligned} \quad (11)$$

式中

$$C = H+A+2(t_{pr}+t_p)R = H+(t_a+2t_{pr}+2t_p)R = H+t_{out}R \quad (12)$$

是帧头和超时带来的开销位数。

帧太短, 需要多次的帧传送, 从而帧头开销增大; 帧太长, 由于信道误码率使重传次数增多, 使传送时间增长, 理论上存在最佳帧长。

利用数学上求极值的方法, 对Delay求关于 n 的偏导数, 令其为0, 可得到:

$$P_b n^2 + 2P_b C n - C(1-HP_b) = 0 \quad (13)$$

求解后可得:

$$n = \frac{(1-HP_b)/P_b}{1 + \sqrt{1 + \frac{1-HP_b}{P_b C}}} \quad (14)$$

本文对式(14)做进一步的近似和简化, 可推出最佳数据位长度:

$$n_{opt} \approx \sqrt{\frac{C}{P_b}} = \sqrt{\frac{H+A+2(t_p+t_{pr})R}{P_b}} \quad (15)$$

其成立的条件为:

$$P_b C = P_b [H+A+2(t_{pr}+t_p)R] \ll 1 \quad (16)$$

该条件满足式(1), 即 $l_f P_b = (H+n)P_b \ll 1$ 或 $HP_b \ll 1, nP_b \ll 1$ 。这里的传播时延 t_p 等于信道长度除以电磁波在信道中的传播速率; 应答帧一般无数

据位, 可使应答帧长 $A=H$ 。

把式(15)、(16)简化为每帧的最佳数据位长度:

$$n_{\text{opt}} \approx \sqrt{\frac{C}{P_b}} = \sqrt{\frac{2(H+t_p R+t_{pr} R)}{P_b}} \quad (17)$$

其条件是:

$$(H+t_p R+t_{pr} R)P_b \ll 1 \quad (18)$$

当数据位长度按式(17)取值时, 数据传送总时间取最小值:

$$\text{Delay}_{\min} \approx \frac{N}{R} \frac{1}{1 - \sqrt{2P_b(H+t_p R+t_{pr} R)}} \times [1 + \sqrt{2P_b(H+t_p R+t_{pr} R)}] \quad (19)$$

另外, 还可以求出停等协议的信道利用率(或协议效率), 它等于发送一帧数据的理想传输时间与正确传送一帧所需要的平均时间(包括开销、重发等)之比。

最佳利用率也就是数据位长度取最佳值时的利用率, 简单的可以由数据传输的理想时间 N/R 与 Delay_{\min} 的比值得到:

$$\varepsilon_{\text{opt}} = \frac{N/R}{\text{Delay}_{\min}} \approx \frac{1 - \sqrt{2P_b(H+t_p R+t_{pr} R)}}{1 + \sqrt{2P_b(H+t_p R+t_{pr} R)}} \quad (20)$$

4 最佳帧长近似解的说明

根据最佳数据位长度近似式(17)、(18)和最小传输时间、最佳利用率或效率表达式(19)、(20), 本文对最佳帧长作进一步的分析和说明:

(1) 每帧的最佳数据位长度与帧头和超时(协议设定的数据重传前的等待时间)带来的开销(位数)成正比。这是因为, 为了提高协议效率(或减小数据传输的实际时间)必须使数据帧中有效位数随着帧头开销和超时开销的位数成正比地变化。

(2) 每帧的最佳数据位长度与信道的误码率成反比。随着信道的误码率的增加, 每一帧出错的概率也随即增大, 使重传次数增多, 因而必须使帧中有效位数减小, 才能不使实际传送时间增大(或协议效率降低)。

(3) 最小传输时间、最佳利用率都与“帧头开销和超时开销的总位数与信道的误码率的乘积”直接相关。该乘积越小, 最小传输时间越小, 最佳利用率越大; 乘积越大, 最小传输时间越大, 最佳利用率越小。因为该乘积与误帧率成正比。

(4) 对于无差错信道, 即 $P_b=0$ 时, Delay_{\min} 、 ε_{opt} 可获得理想值, n_{opt} (数据总长度)可以取得最大值

(接近近似为 ∞), 也就是在理想情况($P_b=0$)下:

$$n_{\text{opt}}=N, \text{Delay}_{\min}=\frac{N}{R}, \varepsilon_{\text{opt}}=1$$

由于是无差错信道, 不需要重发, 且帧中数据位取最大值, 即数据总长度 N , 帧头开销与之相比很小, 可以忽略。因而可得到上述理想的数据传输时间与协议效率。

(5) 上述各近似式都是在帧头开销和超时开销的总位数与信道的误码率的乘积远小于1时(如 0.1)推导出来的, 这个条件也可作为采用停止等待ARQ协议的先决条件。这是因为, 帧头开销位数一般较小, 而超时开销与信道长度成正比。当信道长度增加(满足上述条件时), 最佳数据位长度也要增加; 但当信道长度进一步增加(不满足上述条件时), 为保证每次数据传送的信道利用率而使数据位长度也跟着增加, 这就造成误帧率增加, 因而重传次数增多而导致总的传送效率降低, 此时一般不采用停止等待ARQ协议而采用其他协议(如连续ARQ协议等)。

5 结束语

本文提出了在信道差错率很低时, 停止等待ARQ协议的最佳帧长的近似解, 在满足帧头开销和超时开销的总位数与信道的误码率的乘积远小于1的条件下, 每帧的最佳数据位长度与帧头和超时带来的开销(位数)成正比, 与信道的误码率成反比。该条件还可作为采用停止等待ARQ协议的先决条件。

参考文献

- [1] BURTON H O. Error and error control[J]. Proc. IEEE, 1972, 60(11): 1293-1301.
- [2] OSCAR Y T, OLIVER M C, PETER C M, et al. On the frame-error rate of concatenated turbo codes[J]. IEEE Transactions on Communications, 2001, 49(4): 602-608.
- [3] KRISHNAMOORTHY S, ROBERT M, SRIKANTESWARA S, et al. Channel frame error rate for Bluetooth in the presence of microwave ovens[C]//IEEE VTC 2002 Fall. Vancouver, BC: IEEE VTS, 2002: 168-173.
- [4] 谢希仁. 计算机网络[M]. 第4版. 北京: 电子工业出版社, 2004: 69-77.
- [5] ERIC E J. ARQ protocol features for HF radio links[C]//Proceedings of MILCOM 2003. Boston, Massachusetts: IEEE MCC, 2003: 234-240.
- [6] SAMIR K. Analysis of memory and incremental redundancy ARQ schemes over a nonstationary channel[J]. IEEE Transactions on Communications, 1992, 40(9): 220-226.
- [7] 王承恕. 现代通信网[M]. 北京: 电子工业出版社, 2005.
- [8] 李增智, 陈 研. 计算机网络原理[M]. 第2版. 西安: 西

安交通大学出版社, 2000: 116-119.

[9] ANDREW S T. Computer network[M]. 4th ed. 北京: 清华大学出版社, 2005.

[10] DOUGLAS E C. Internetworking with TCP/IP VOL I:

principles, protocols and architectures[M]. 4th ed. 北京: 电子工业出版社, 2005.

编辑 张俊

(上接第826页)

4 总 结

本文提出了一种基于OGSA网格的工作负载监测体系结构。在网格的运行中,通过对工作单元的分类和端到端的跟踪监控,实现对工作单元状态的报告和运行时间的计算。为了增强可移植性和减少应用程序代码的改变,采用了基于OGSA的网格中间件结构。

在该体系结构中,还建立了Petri网模型,用于理解各服务之间的关系,便于准确计算运行时间。该模型能自动收集工作单元的数据,为网格用户提供监控数据信息浏览,有利于用户对系统性能进行有效管理。

在建立上述体系结构的基础上,提出了未来扩展系统功能方面的思路,以适应网格环境的复杂性,减少通信延迟和通信代价,为用户更好地监测网格系统提供方便。

参 考 文 献

[1] FOSTER I, KESSEHMAN C, NICK J M, et al. Grid services for distributed system integration[J]. Computer,

2002, 35(6): 37-46.

[2] FOSTER I, KESSEHMAN C. The grid: Blueprint for a new computing infrastructure[M]. San Francisco: Morgan Kaufmann Publishers, 1999.

[3] SANDHOLM T, GAWOR J. Globus toolkit 3 core-a grid service container framework[EB/OL]. http://www-unix.globus.org/toolkit/3.0/ogsa/docs/gt3_core. PDF, 2005-05-19.

[4] DAVID R, ALLA H. Petri net and grafcet: Tools for modelling discrete event systems[M]. New Jersey: Prentice Hall, 1992.

[5] DINGLE N J, HARRISON P G, KNOTTENBELT W J. Response time densities in generalized stochastic Petri net models[C]//In WOSP '02: Proceedings of the Third International Workshop on Software and Performance. Rome: ACM Press, 2002.

[6] BERTSEKAS D. Dynamic programming and optimal control[M]. Massachusetts: Athena Scientific, 1995.

[7] GOODWILL J. Apache jakarta tomcat[EB/OL]. <http://jakarta.apache.org/tomcat/>, 2005-06-10.

[8] HAPNER M, BURRIDGE R, SHARMA R, et al. Java messaging service specification[R]. Sun Microsystems, 2002.

编辑 熊思亮