doi:10.3772/j.issn.1002-0470.2015.05.013

基于移动平台的文件传输策略的设计和实现①

黄家辉② 冯冬芹③ 还约辉

(浙江大学工业控制技术国家重点实验室 杭州 310027)

摘 要 提出了一种基于移动平台的文件可靠传输策略:发送节点采用与接收节点约定好的协议将文件内容按特定的长度进行分包,按特定的字段进行组包,按特定的数量进行分组;发送节点每传输一组数据包后都进行丢包的及时检查,只有等到接收节点接收了该组的全部数据包后才允许发送下一组数据包,直到所有数据包发送完毕;在传输过程中采用循环冗余校验(CRC)码对每个数据包进行校验,传输结束后采用哈希算法对整个文件的完整性进行校验,以增加文件传输的可靠性。实验结果表明,该文件传输策略能够同时减小链路开销、丢包率和传输时间。

关键词 移动平台,可靠传输,哈希算法,链路开销,丢包率

0 引 言

基于移动平台的通信系统,是一种起源于欧洲的技术标准,其开发目的是让全球各地可以共同使用一个移动电话网络标准[1]。移动平台的短信业务具有随时随地获取信息的便利,但由于短信链路的不确定性,使得通过短信来传输数据在可靠性方面存在一定的隐患。因而,如何提高基于移动平台的文件传输的可靠性的问题日益受到重视。本文在深入研究的基础上提出了一种高可靠性传输策略,并从理论上分析了该策略对传输指标的影响。

1 相关研究

传输可靠性可以通过丢包率、重传包数、传输时间等传输指标来反映。丢包率越低、重传包数越少、传输时间越短,则可靠性越高。为此,一些研究学者提出了一系列的传输指标用来反映传输的可靠性:Cerpa 等人[2]从重传包数的角度,通过使用要求发送数据包数(required number of packets,RNP)计算传输和重传数据包的数目与成功接收数据包的数目的比值,RNP 值越高,则重传的数据包数越多,可靠性越低;Srinivasan 等人[3]从丢包率的角度,使用数据包接收率(packet reception rate,PRR)计算成功接收的数据包数目与总共发送数目的比值,PRR 值越高,则发送节点总共发送的数据包越少,丢包率越低,可靠性就越高;Tran 等人[4]也从丢包率的角度,考虑了链路的上行质量和下行质量不对称,采用期望发送次数(expected transmission count,ETX)计算发送节点处的 PRR 值与接收节点处的 PRR 值(接收节点理论上应该响应的数据包数目与实际发出的数据包数目的比值)的乘积的倒数,ETX 值越高,表明 PRR 值越小,可靠性越低;Zahmati 等人[5]从传输延迟的角度,使用平均传输延迟(average transmission delay,ATD),它定义了成功接收所有数据包的整体延时与数据包数目的比值,ATD 越小,则传输时间越少,可靠性就越高。单个指标仅能反映可靠性的某一个方面,不能完全反映可靠性。只有使上述指标综合达到最优,才能真正说明传输可靠性的高低。

① 国家自然科学基金(61433006)资助项目。
② 男,1990 年生,硕士;研究方向:工业网络通信和脆弱性研究;E-mail:elninohjh@163.com
③ 通讯作者,E-mail:dqfeng@iipc.zju.edu.cn
(收稿日期:2014-04-14)

基于短信传输的系统在可靠性方面存在一系列隐患:短信稳定性差;实时性较差,短信延迟;在移动网络中运行的 SIM 卡会收到垃圾短信,可能导致系统故障。考虑到上述隐患,杨高洁等人[6]提出在传输数据包时采用点对点信息应答机制,对每一个传输的数据包都进行确认,这虽然增加了传输的可靠性,却大大增加了链路开销,增加了传输时间;Paek 等人[7]通过控制传输速率避免信道拥塞来提高数据包传输的可靠性,考虑通道噪声等外在因素,这增加了系统的复杂性,并且也会增加传输时间;Kang 等人提出的冗余数据传输[8]将数据包进行备份,全部发送完之后再根据接收方的响应决定是否重传丢失的数据包,这虽然减小了链路开销,但大量数据存放在缓冲区,容易造成缓冲区溢出,且丢包率难以保证。此外,上述三种传输策略都将文件以数据包的形式进行传输,仅仅对传输的数据包进行校验,而没有考虑到短信延迟导致数据包失序而使文件内容遭到破坏。

针对上述问题,本文提出了一种文件可靠传输策略,主要分为以下几步:将文件内容进行分包处理,对数据包分组发送,考虑可能存在的丢包状况以及对每个数据包进行校验,对整个文件进行完整性校验。本文所采用的数据包校验方法为循环冗余校验(cyclic redundancy check, CRC),文件完整性的校验方法为哈希校验。通过对文件进行分包、组包,能够使收发双方实时了解文件传输的状态,避免出现接收缓冲区溢出等安全性问题;采用分组发送,能够减少点对点应答机制所带来的链路开销大的问题;采用一种实时丢包重传机制,能够在一定程度上减小丢包率;采用完整性校验,能够解决文件内容混乱的问题。

2 传输策略的设计和分析

2.1 设计思想

前面提到的采用冗余数据传输[8]的策略将文件进行分包备份,存放在内存中,依次从内存中读取数据包进行发送,全部发送完后再进行“是否全部接收”的响应,若收到“已全部接收”的响应则清空内存中的数据包,若收到数据包号则重传对应的数据包,直到收到“已全部接收”的响应;杨高洁等人[6]提出的方案将文件进行分包备份后,每发送一个数据包就等待接收节点的响应,若收到“已接收”的响应,则删除已发送成功的数据包,继续发送下一个,直到全部发送完成。本文在上述研究的基础上,提出了一种新的策略,该策略的传输步骤包括完整性校验、分包处理、分组发送及响应和实时丢包重传等过程。发送节点首先对文件内容进行完整性校验,之后以 m 个字节为单位对文件进行分包,以 n 个数据包为单位进行分组。在与接收节点建立连接关系后,连续传输 n 个数据包后等待接收节点的响应,只有收到“接收成功”的响应才发送下一组数据包,直到文件全部发送完毕。每个数据包都经过循环冗余校验(CRC)来确保内容的正确性。接收节点在收到全部文件内容后,同样进行完整性校验,将校验结果返回给发送节点,与之前计算的校验结果进行比较,若两者相等则代表文件传输成功。该策略整体流程如图 1 所示。

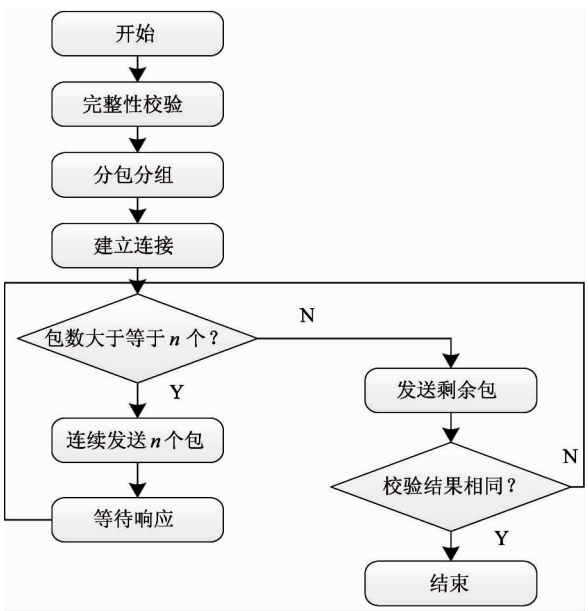

图 1 传输策略流程

2.2 策略分析

2.2.1 完整性校验

完整性校验用于检验数据包在传输过程中是否被篡改或丢包导致文件内容损坏。常用的完整性校验方法有 MAC 算法[9]、MD5 算法[10]和哈希算

法[11]。由于 MAC 算法和 MD5 算法的校验方式比较复杂,本文采用面向字符的哈希算法。哈希算法包括 MD4、MD5、SHA-0、SHA-1 和 SHA-2。其中针对 SHA-0 和 SHA-1 算法的模差分攻击方法及明文修改技术,可以在较小的运算复杂度情况下找到 SHA-0 和 SHA-1 算法的碰撞;MD4 和 MD5 算法在运算上比较复杂,需要一定的计算开销。SHA-2 被认为是最安全的,SHA-256 由于加密速度快、有很好的非线性度和很高的安全性而备受欢迎,因此本文采取 SHA-256 算法。

冗余数据传输策略和杨高洁策略对传输的每个数据包进行校验,能够保证收到的包的内容是正确的,却无法保证接收节点最后将所有数据包的内容拼接以还原初始文件时不会出现重组错误,因此有必要在还原文件内容后再进行校验。

2.2.2 分包分组

假设文件的大小为 M 字节,每个数据包存放 m 字节的数据。由于移动规定每条短信的长度最大为 140 个字节,故满足 $1 \leqslant m \leqslant 140$。当 $M \leqslant m$ 时,只需发送一个数据包即可完成传输;当 $M \geqslant m$ 时,必须对文件进行分包。

考虑所需传输的数据包的个数 $TotalNum$ 定义为

$$TotalNum = \begin{cases} \dfrac{M}{m}, M \text{ 能整除 } m \\ \dfrac{M}{m} + 1, M \text{ 不能整除 } m \end{cases} \quad (1)$$

之所以加 1,是因为可能出现最后剩余的文件内容不足 m 字节。由于 M 为定值,m 越大,则链路开销越小,传输时间也会减少。但是如果 m 取 140,接收节点就无法判断这 m 字节的数据是否出错,此外由于短信存在延迟现象,假设第 $k+1$ 个数据包先于第 k 个数据包到达接收节点,接收节点在还原文件时会默认第 $k+1$ 个数据包是第 k 个,导致重组错误。因此需要在数据包中添加必要的字段。考虑最后一个数据包存放的数据不足 m 字节,此时由于该数据包的大小是随机的,接收节点不能采用处理之前数据包的方法,而要通过计算来解析出实际的数据部分。

为了防止数据包失序,添加分包号字段来为每个数据包添加编号,便于最后重组文件,假设该字段大小为 $p1$ 字节;为了判断每个数据包的正确性,添加校验字段,假设大小为 $p2$ 字节;为了解析出不定长度的数据,添加数据长度字段,该字段的值减去分包号字段和校验字段的大小,即为数据的大小,假设大小为 $p3$。由于短信在传输过程中出现碰撞导致丢失或损坏只与发送的频率有关,而与短信长度无关,因此为了减少链路开销,缩短传输时间,满足 $p1 + p2 + m + p3 = 140$,且 m 取值尽量大。本文采取的数据包校验方式为 CRC,发送节点在发送数据包前先对数据包的内容进行校验得到 CRC 值,置于短信末端。接收节点根据特定协议解析出数据包内容,采取相同的算法进行校验后与发送节点计算的 CRC 值对比,若相等则接收该数据包,否则丢弃。CRC 值的大小与生成多项式有关,常用的生成多项式为 CRC-12、CRC-16 和 CRC-32,由于 CRC-32 出错的概率最低,本文采用 CRC-32 校验码,得到 4 字节的 CRC 值,即 $p2 = 4$;由于一条短信最长为 140 字节,故数据长度只需 1 个字节,即 $p3 = 1$;移动运营商为了防止垃圾短信的骚扰,规定 24 小时内不能发送超过 2000 条的短信,因此分包号采取 2 字节,即 $p1 = 2$。综上所述,本文策略以每 133 字节对文件进行分包。

数据包传输采取分组传输,即以 n 个数据包为一组,每次发送一组数据包后再等待接收节点的响应,只有收到“接收成功”的响应,才发送下一组数据包,直到文件发送完毕。如果数据包数不足 n 个或者剩余的数据包数不足 n 个,则一次性发送所有的数据包。

文件在传输前,收发节点之间进行连接,否则如果接收节点由于接收缓冲区已满,若发送节点继续传输,就会造成数据包浪费;文件在传输过程中,发送节点只有在接收节点处于“正常接收”状态,才能进行发送。因此本文采用请求报文询问接收节点的状态,采用响应报文来通知发送节点可传输数据包。其中请求报文包含总包数字段,接收节点根据收到的不同分包号数量之和是否等于总包数来判断文件是否全部接收。为了让发送节点知道哪些数据包需要重传,或者接收节点计算的完整性结果是多少,本文采取结果报文。

2.3 实时丢包重传

以响应报文为例,发送节点在发完一组数据包后未收到正确的“接收成功”的响应,原因可能是:(1)数据包丢失导致接收节点未收满 n 个数据包;(2)响应报文丢失;(3)响应报文出错被丢弃。

基于此,通过在发送节点设置定时器。当定时器溢出时仍未收到正确的响应报文,则发送请求报文。接收节点在收到请求报文后,如果是原因(1),则将未收到的分包号返回给发送节点;如果是原因(2)和(3),则重新发送响应报文。发送节点请求 3 次仍未收到任何信息,则终止后续文件的传输。

3 传输指标分析

假设发送节点每隔 $t1$s 发送一条短信,等待响应的时间为 $t2$s,由于 $t2$s 包含了收、发两条短信的时间,而 $t1$s 仅包含发送一条短信的时间,因此满足 $t2 > 2t1$。在不出现丢包的情况下,假设发送节点发送短信到接收节点所需的时间与接收节点发送短信到发送节点相等。

3.1 要求发送数据包数(RNP)

冗余数据传输策略由于最后再对数据包进行确认,发送节点需发送 *TotalNum* 条短信,接收节点要回复 1 条“已全部接收”的短信,故传输文件共需要的短信数量 *MsgNum*1 用下式表示:

$$MsgNum1 = TotalNum + 1 \tag{2}$$

杨高洁策略对每条发送的短信进行确认,发送和接收节点均要发送 *TotalNum* 条,总共需要的短信数量 *MsgNum*2 用下式表示:

$$MsgNum2 = 2 \times TotalNum \tag{3}$$

本文策略采用分组传输,以 n 个数据包为一组进行传输,再加上之前用于建立连接关系的报文(2 条)、传输过程中的响应报文($\frac{TotalNum}{n}$ 条)以及传输结束的结果报文(1 条),短信数量 *MsgNum*3

$$MsgNum3 = TotalNum + 3 + \frac{TotalNum}{n} \tag{4}$$

根据 RNP 的定义,接收数据包数目一定时,因 *TotalNum* 大于等于 1,所以 $MsgNum2 \geqslant MsgNum1$。显然 $MsgNum3 \geqslant MsgNum1$。而对比 *MsgNum*2 和 *MsgNum*3 的大小,当满足 $TotalNum > \frac{3n}{n-1}$ 时,$MsgNum2 \geqslant MsgNum3$。计算 $\frac{3n}{n-1}$ 的导数可知其小于 0,而 $n < TotalNum$,当 *TotalNum* 大于 6 时,$MsgNum2 \geqslant MsgNum3$,而且 *TotalNum* 越大,两者的短信数量差越大。因此当 *TotalNum* 大于 6 时,本文策略的要求发送数据包数(RNP)大于冗余数据传输策略而小于杨高洁策略。

3.2 平均传输延迟(ATD)

冗余数据传输策略在传输过程没有等待响应,全部发送完以后才进行一次响应,故所需的传输时间 *TransTime*1 的表达式为

$$TransTime1 = TotalNum \times t1 + t2 \tag{5}$$

杨高洁策略对每一个数据包都进行响应,所需的传输时间 *TransTime*2 的表达式为

$$TransTime2 = TotalNum \times t2 \tag{6}$$

本文策略建立连接关系需要 $t2$s,连续发送 n 个数据包并等待响应需要 $(n-1)t1 + t2$ s,发送最后不足 n 个数据包并等待响应需要 $(TotalNum \bmod n - 1)t1 + t2$ s,其中 mod 表示求余运算。故传输时间 *TransTime*3 的表达式为

$$\begin{aligned} TransTime3 = {} & (TotalNum - \frac{TotalNum}{n} \\ & + TotalNum \bmod n - 1)t1 \\ & + (\frac{TotalNum}{n} + 2)t2 \end{aligned} \tag{7}$$

首先假设 *TransTime*1 > *TransTime*2,化简得 $TotalNum < \frac{t2}{t2 - t1} < 2$。显然 *TotalNum* 为大于等于 2 的正整数,因此 *TransTime*1 < *TransTime*2;同理当 $t2 > 2t1$ 且 $TotalNum \bmod n > 0$ 的时候,*TransTime*1 < *TransTime*3;当 *TotalNum* 大于 6 时,*TransTime*2 > *TransTime*3,而且 *TotalNum* 越大,两者的传输延迟差越大。因此当 *TotalNum* 大于 6 时,本文策略的平均传输延迟(ATD)大于冗余数据传输策略而小于杨高洁策略。

3.3 数据包接收率(PRR)

上述传输指标的分析都是在无丢包的前提下进行的,当存在丢包率 p 时,由于丢包率与移动信道有关,数据包的丢失是不确定的,但是可以求出丢包的

期望。假设一共要发送 $MsgNum$ 个包,每个包的丢失服从二项分布,即要么丢包,要么不丢包,设丢包率为 p,满足 $TotalNum \sim B(MsgNum,p)$。二项分布的期望 $E = MsgNum \times p$。

冗余数据传输策略一共要传输 $MsgNum1$ 个包,且每个包之间独立,上一个包的丢失不会影响下一个包丢失的概率,故丢包的期望 $E(1)$ 的表达式为

$$E(1) = (TotalNum + 1) \times p \tag{8}$$

杨高洁策略在传输过程中能够实时了解数据包传输的状态,在遇到丢包情况时,会采用增大传输间隔来退避丢包率高的时段,从而减小丢包率。假设出现一次丢包后,之后丢包的概率变为原来的 p 倍,则当存在 m 次丢包时,丢包的期望 $E(2)$ 的表达式为

$$E(2) = \sum_{i=1}^{m-1} k_i \times p^i \tag{9}$$

其中 $m-1$ 表示前 $m-1$ 次丢包(最后一次丢包没有影响),k_i 为连续两次丢包期间传输的数据包数,满足 $\sum_{i=1}^{m-1} k_i = 2 \times TotalNum$。$p^i$ 为第 i 次丢包后的丢包概率。由于短信中心的调度使得短信丢失的概率很低,因此式(9)可以简化为

$$E(2) = k_1 \times p + k_2 \times p^2 \tag{10}$$

本文策略出现的丢包发生在传输 n 个数据包之前或者传输 n 个数据包期间,但不会影响这一组的丢包率,而是降低了下一组传输的丢包率。同样一次丢包后下一组丢包的概率变为原来的 p 倍。当存在 m 次丢包时,丢包的期望 $E(3)$ 的表达式为

$$E(3) = \sum_{i=1}^{m-2} [k_i' + (n+1)] \times p^i + [k_{m-1}' + q] \times p^{m-1} \tag{11}$$

其中 k_i' 为连续两次丢包期间丢包那一组之前传输的数据包数,$n+1$ 表示传输 n 个数据包加上一个 1 个响应报文,k_{m-1}' 为最后一次丢包和倒数第二次丢包之间传输的数据包数,q 为最后剩余的可能存在的不足 $(n+1)$ 个的数据包数。同样式(11)可以简化为

$$E(3) = [k'_1 + (n+1)] \times p + [k'_2 + (n+1)] \times p^2 \tag{12}$$

对比 $E(1)$ 和 $E(2)$,当 $k_1 > TotalNum + 1$ 时,即若冗余数据传输策略和杨高洁策略同时传输数据包,杨高洁策略第一次出现丢包发生在冗余数据传输策略已经完成传输的情况下,此时 $E(2)$ 才大于 $E(1)$,相当于冗余数据传输策略未发生丢包情况下的丢包数才小于杨高洁策略;对比 $E(1)$ 和 $E(3)$,当 $k_1' > TotalNum - n$ 时,即本文策略第一次出现丢包时,冗余数据传输策略最多还剩下 $n+1$ 个数据包未传,此时本文策略的丢包数才大于冗余数据传输策略;对比 $E(2)$、$E(3)$,只有当 $k_1 > k_1' + n + 1$ 和 $k_2 > k_2' + n + 1$ 同时满足时,$E(2)$ 才恒大于 $E(3)$,否则杨高洁策略和本文策略的丢包数根据具体环境而定。因此,冗余数据传输策略的丢包数最多的可能性最大,从而其数据包接收率(PRR)最小。

3.4 期望发送次数(ETX)

ETX 的分析与 PRR 相似,冗余数据传输策略的 PRR 最小,其 ETX 最大。

4 实验及结果分析

本研究采用如图 2 的测试平台进行上述传输策略的测试。

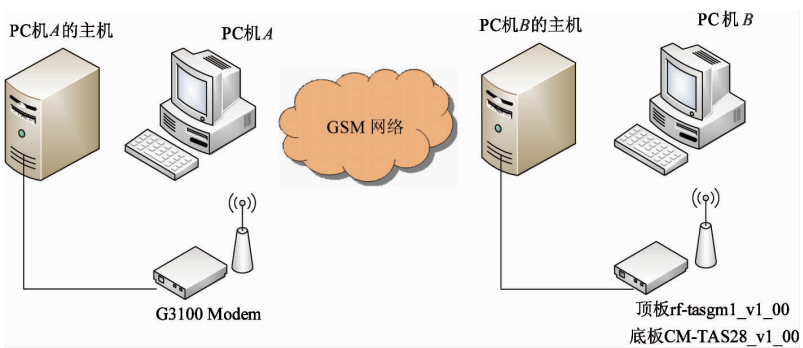

图 2 测试平台

其中发送节点 *A* 采用一台 PC 和一个佐格 G3100 Modem。接收节点 *B* 采用一台 PC 机、顶板 rf-tasgm1 _ v1 _ 00、底板 CM-TAS28 _ v1 _ 00。*A* 的软件平台为 Microsoft Visual Studio 2010，*B* 的软件平台为 IAR6.40 版本。测试的文件来自一个大小为 100KB 的二进制文件，每个数据包装载 133 字节的数据。理想情况下一共要发送 $100 \times 1024/133 + 1 = 770$ 个数据包。取 n 为 10。

4.1 RNP

在 *B* 成功接收到全部文件内容的情况下，双方节点所需传输的短信数量与对应的 RNP 的对比结果如表 1 所示。

表 1 三种策略的要求发送数据包数对比

	传输的短信量（条）	接收的短信量（条）	RNP
冗余数据传输策略	836	770	1.086
杨高洁策略	1544	770	2.005
本文策略	858	770	1.114

由表 1 可知，冗余数据传输策略的 RNP 值最小，本文策略次之，杨高洁策略几乎是冗余数据传输策略的 2 倍。根据 RNP 值越小越优的原则，冗余数据传输策略最优，这是因为冗余数据传输策略中 *B* 只有在收到最后一条短信时才会进行确认。杨高洁策略中 *A* 要对每一条发送的短信进行确认，相当于 *A* 和 *B* 要发送相同数量的短信。本文策略，*A* 每发送 10 个数据包进行一次确认，在无丢包的情况下，其所需发送的短信量介于其他两种策略之间。

4.2 PRR

在 *B* 成功接收到全部文件内容的情况下，*A* 所传输的短信数量与 PRR 的对比结果如表 2 所示。

表 2 三种策略的数据包接收率对比

	发送的短信量（条）	接收的短信量（条）	PRR
冗余数据传输策略	826	770	0.932
杨高洁二	772	770	0.997
本文策略	778	770	0.990

由表 2 可知，杨高洁策略的 PRR 值最大，本文策略次之，冗余数据传输策略最小。根据 PRR 值越大越优的原则，杨高洁策略最优。因为杨高洁策略对每个报文都进行确认，了解数据包传输的状态及链路质量，在质量差时进行退避，否则加快数据的传输，因此重传的短信数量少。冗余数据传输策略的 PRR 值具有偶然性，一旦链路质量较差，可能造成大量的短信丢失，增加重传短信的数量。本文策略采取的定时器超时重传，也能很好地根据链路质量来自适应调整发送速率，减小链路质量对文件传输的干扰。

4.3 ETX

B 实际发送的短信数量如表 3 所示。

表 3 接收节点发送的短信数量

	传输的短信量（条）	发送短信量（条）	*B* 发送短信量（条）
冗余数据传输策略	836	826	10
杨高洁策略	1544	772	772
本文策略	858	778	80

理论上，冗余数据传输策略 *B* 只需最后发送 1 条确认报文即可，因此接收的 PRR 为 1/10；杨高洁策略 *B* 需发送 772 条确认报文，因此接收的 PRR 为 770/772；本文策略 *B* 需发送 79 条报文，因此接收的 PRR 为 79/80。ETX 的对比如表 4 所示。

表 4 三种策略的期望发送次数对比

	发送 PRR	接收 PRR	ETX
冗余数据传输策略	0.932	0.1	10.730
杨高洁策略	0.997	0.997	1.006
本文策略	0.990	0.988	1.022

由表 4 可知，杨高洁策略的 ETX 值最小，本文策略次之，冗余数据传输策略远远大于其他两种策略。根据 ETX 越小越优的原则，杨高洁策略最优。原因还是与链路质量有关，杨高洁策略和本文策略都会根据链路质量的状况进行自适应调整传输策

略,避开了链路质量差、丢包概率高的时期,而本文策略的传输策略是固定的,在 *A* 到 *B* 这条链路上计算的 PRR 值与其他两种策略差别不大,但在 *B* 到 *A* 的链路上,报文传输的成功率只有 10%,原因可能是 *B* 在最后进行确认的那个时期移动网络十分繁忙。

4.4 ATD

在 *B* 成功接收到全部文件内容的情况下,传输所有数据所需要的时间与对应的 ATD 的对比结果如表 5 所示。

表 5 三种策略的平均传输延迟对比

	传输时间(h)	接收短信量(条)	ATD(s)
冗余数据传输策略	1.11	770	5.190
杨高洁策略	6.22	770	29.081
本文策略	1.53	770	7.153

由表 5 可知,冗余数据传输策略的 ATD 值最小,本文策略次之,杨高洁策略最大。根据 ATD 值越小越优的原则,冗余数据传输策略最优。这是因为冗余数据传输策略对数据包的确认少,且发包速率一定,但是如果所需传输的短信量很大并且链路质量差,重传报文的次数会大大增加,相应增加 ATD。杨高洁策略对每条短信进行确认,传输延迟最大。本文策略根据链路质量自适应的调整发包的速率,链路质量好的时候就会减少 ATD。图 3 为 3 种策略基于 4 个指标的对比柱形图。

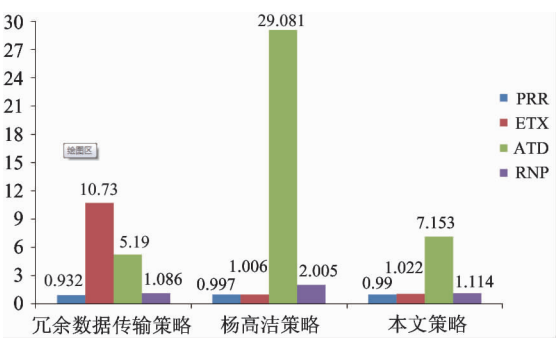

图 3 对比图

综上所述,本次测试与之前的理论性能分析结果基本吻合,从而本文策略能够从整体上使链路开销、丢包率和传输时间达到最优。

5 结 论

基于移动平台的文件传输,由于移动网络本身的特性,使得传输充满了不确定性,如何提高数据传输的可靠性正越来越受到人们的关注。本文提出了一种高可靠性的文件传输策略,并从完整性校验、分包操作、分组发送、丢包重传等方面对该策略进行了介绍,并从理论的角度分析了该策略对于传输指标的影响。测试结果表明,该传输策略能够减少丢包率,同时减小短信开销和传输时间,增加了文件传输的可靠性。

参考文献

[1] 曾鹏, 于海斌. 工业无线网络 WIA 标准体系与关键技术. 自动化博览, 2009, (1): 24-27

[2] Cerpa A E, Wong J L, Potkonjak M, et al. Temporal properties of low power wireless links: modeling and implications on multi-hop routing. In: Proceedings of the 6th ACM IntSymp on Mobile ad Hoc Networking and Computing, New York, USA, 2005. 414-425

[3] Srinivasan K, Dutta P, Tavakoli A, et al. Understanding the causes of packet delivery success and failure in dense wireless sensor networks. In: Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, New York, USA, 2006. 419-420

[4] Tran A T, Kim M K. Characteristics of ETX link quality estimator under high traffic load in wireless networks. In: Proceedings of the 2013 IEEE International Conference on High Performance Computing and Communications & Embedded and Ubiquitous Computing (HPCC _ EUC), Zhangjiajie, China, 2013. 611-618

[5] Zahmati A S, Fernando X, Kojori H. Transmission delay in wireless sensing, command and control applications for aircraft. In: Proceedings of the 4th Annual Caneus on Fly by Wireless Workshop (FBW), Montreal, 2011. 1-4

[6] 杨高洁,商建东、范志辉. 一种基于 GSM 的高可靠性安全监控系统的设计,计算机安全. 2009,7:30-32

[7] Paek J, Govindan R. RCRT: Rate-controlled reliable transport for wireless sensor networks. In: Proceeding of the 5th ACM Conference on Embedded Networked Sensor

Systems, Sydney, Australia, 2007. 305-319

[8] Kang J A, Kim H K. An adaptive packet loss recovery method based on real-time speech quality assessment and redundant speech transmission. *International Journal of Innovative Computing, Information and Control*, 2011, 7(12): 6773-6783

[9] Alhazbi S, Khan K M, Erradi A. Preference-based semantic matching of web service security policies. In: Proceedings of the 2013 World Congress on Computer and Information Technology (WCCIT), Sousse, 2013. 1-5

[10] Gupta P, Kumar S. A Comparative Analysis of SHA and MD5 Algorithm. *architecture*, 2014. 1-5

[11] Garcia R, Algredo-Badillo I, Morales-Sandoval M, et al. A compact FPGA-based processor for the secure Hash algorithm SHA-256. *Computers & Electrical Engineering*, 2014, 40(1): 194-202

Design and implementation of a file transmission strategy based on mobile platform

Huang Jiahui, Feng Dongqin, Huan Yuehui

(State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027)

Abstract

A strategy for reliable transmission of files based on mobile platform is proposed below: The sending node uses the protocol neogiated with the receiving node to subcontract the content of a file by a specific length, pack packets for the file by a specific field and group packets by a specific number; After sending a group of packets, the sending node timely checks whether any packets of the group are lost, and only after the receiving node receives all the packets of the group can the sending node send the next group of packets, with the sending continued until all the packets of the file are sent; Using cyclic redundancy check (CRC) codes to check every packet during the transmission, and using the Hash algorithm to check the integrity of the file after the transmission to increase the reliability of transmission. The experimental results show that this strategy can simultaneously reduce the link cost, the packet loss rate and the transmission time.

Key words: mobile platform, reliable transmission, Hash algorithm, link cost, packet loss rate