TCP-Shape 一种改进的网络拥塞控制算法研究

程 京[1,2], 沈永坚[1], 张大方[2,1], 黎文伟[1]
(1. 湖南大学计算机与通信学院, 湖南长沙 410082; 2. 湖南大学软件学院, 湖南长沙 410082)

摘 要: 网络拥塞是由于网络业务流不可预测的流量突发现象造成的. 文章从考虑网络业务流突发现象产生的特点出发, 采用可用带宽测量技术和流量整形技术, 提出了一种针对传统网络拥塞控制算法的改进算法(TCP-Shape). 改进后的拥塞控制算法能够快速地探测到网络链路中可用剩余带宽并能够有效地消除网络业务流中的突发现象. 使得在网络业务流的吞吐量和数据报文段的丢失率等性能上, 更加优越于传统拥塞控制算法所获得的性能.

关键词: 拥塞控制; 可用带宽测量; 流量整形技术; 网络测量
中图分类号: TP393 文献标识码: A 文章编号: 0372-2112 (2006) 09-1621-05

TCP-Shape: Study on a Renovation Method of Network Congestion Control Algorithms

CHENG Jing[1,2], SHEN Yong-jian[1], ZHANG Da-fang[2,1], LI Wen-wei[1]
(1. *College of Computer and Communication, Changsha, Hunan 410082, China*; 2. *Software School, Hunan University, Changsha, Hunan 410082, China*)

Abstract Network congestion is caused by the unexpected burst. According to characteristic analysis of network traffic burst, the paper proposes a renovation method of congestion control algorithm (TCP-Shape) against the traditional congestion control algorithms, which uses dummy data segment probe to probe the available bandwidth in a TCP path and use flow shape techniques to make the flow look like more smoother. The experiments also show that the new congestion control algorithms can do better on data segment drop and throughout performance against the traditional congestion control algorithms.

Key words congestion control; available bandwidth measurement; flow shape technique; network measurement

1 引言

网络业务流的突发性一直是造成网络拥塞现象发生的重要原因. 如何有效地降低网络业务流由于流量突发原因造成的 IP数据报文段的丢失从而达到提高网络服务性能, 已经成为许多研究工作者所关注的对象.

为了避免和减少网络拥塞现象的发生和提高网络服务的性能, 1988年 Jacobson[1]提出了基于 TCP(Transmission Control Protocol)流的端到端网络拥塞控制算法。该算法通过采用慢启动算法和在拥塞避免阶段采用 AIMD(Additive Increase and Multiplicative Decrease)[2]算法来探测网络链路中的可用资源, 并相应地调整 TCP发送端发送窗口, 以防止网络拥塞现象的发生。然而, 随着网络经过几十年的发展, 当年 Jacobson提出的拥塞控制算法已经不能很好的满足今天提高网络服务性能的需要。因此, 为了继续提高网络服务性能, 在 Jacobson提出的网络拥塞控制算发的基础上, 许多文献提出了大量的新的和改进的算法, 这些算法主要是从下面两个角度来进行分析和讨论的: 一是通过探测网络拥塞现象是否发生来进行的[3,4]; 二是通过预测网络拥塞是如何发生来进行的[5,6]。他们通过探测和预测网络拥塞情况是否发生和如何发生, 来探讨和确定网络拥塞控制策略。然而, 通过这些方式探讨的网络拥塞控制算法并没有从根本上消除促使网络拥塞现象产生的根本原因——突发现象, 这也是造成网络数据报文段频繁丢失的根本原因。

因此, 本文试图从尽量消除网络业务流的突发现象和提高网络链路带宽的利用率的角度出发, 分析传统端到端 TCP拥塞控制算法发送端发送数据报文段的方式, 提出一

收稿日期: 2005-01-12; 修回日期: 2006-05-18
基金项目: 国家自然科学基金 (No. 60473031); 国防基础科研"十一五"项目 (No. A1420060162)
* 通信作者: dfzhang@hnu.edu.cn

种改进的网络拥塞算法 (TCP- Shape)。改进后的拥塞控制算法采用了虚拟数据报文段探针快速地探测网络链路中可用剩余带宽, 并且结合在文献[7]中提到的流量整形技术, 使得改进后的拥塞控制算法所产生的网络业务流显得更加的平滑。

2 突发业务流的产生和改进方法的提出

下面就传统端到端 TCP拥塞控制算法发送端发送数据报文段业务流的特点和存在的一些问题进行分析并提出流量整形方法.

为了讨论方便起见, 在对传统端到端 TCP拥塞控制算法进行分析之前我们作两点前提假设: (1)在一个端到端 TCP拥塞控制会话连接中, TCP发送端需要有足够的数据报文段发送; (2)在网络链路拥塞状况发生时, 网络拥塞将持续一段相对较长的时间. 因此, 一个端到端 TCP会话连接能够访问到的网络链路资源可以表示为在一个 RTT往返时间内, TCP会话连接发送端所拥有的拥塞控制窗口 (cwnd congestion window)的大小. 也就是说: 在一个 RTT时间区域内, 端到端 TCP发送端能够以背靠背 (back to back)的形式发送最大数据报文段数量. 这种以背靠背发送数据报文段的方式使得一个端到端 TCP会话连接产生的网络业务流具有更大的突发性特性, 也正是由于这些突发性特征的存在, 使得网络链路拥塞现象出现的概率迅速地增加, 同时也增加了业务流数据报文段丢失的概率. 由于网络拥塞的发生, 大大提高了网络的延迟, 严重影响了网络所提供服务的性能.

为了避免网络链路拥塞现象的发生和尽量消除网络业务流所具有的突发性特征, HyoungWoo Park和 JinWook Chung[7]应用流量整形技术对传统端到端 TCP拥塞控制算法发送的网络业务流进行整形. 在该过程中, 当 TCP拥塞控制算法发送端从接收端接收到反馈回来的确认数据报文段后, 在一个 RTT往返时间内, TCP拥塞控制算法发送端不是以背靠背的方式连续发送拥塞控制窗口允许发送的所有数据报文段, 而是在一个 RTT往返时间内, 相等时间间隔均匀地发送拥塞控制窗口允许发送所有数据报文段. 实验证明: TCP拥塞控制发送端这种发送数据报文段方式虽然减少了网络业务流的突发现象, 减少了由于网络链路拥塞而引起的数据报文段的丢失现象, 但是却降低了网络链路的吞吐量. 这是因为: 在网络传输的实际业务流中, 小文件传输 (例如 HTTP流)产生的业务流在当前网络传输业务流中仍然占据着重要部分, 传统端到端 TCP拥塞控制算法在探测网络链路可用剩余带宽时由于采用了慢启动算法, 使得它在探测到网络链路中的最大可用有效带宽之前, 传统端到端 TCP拥塞控制算法发送小文件过程早就已经完成了[5], 因此, 无论是从网络链路的利用率上来说, 还是从网络链路的吞吐量上来说, 其与传统端到端 TCP拥塞控制算法相比, 都有所下降. 正是因为如此, 为了快速地探测到网络链路中可用剩余带宽, 本文通过采用虚拟数据报文段主动探测网络链路中可用剩余带宽的方法, 并采用上面提到的流量整形方法, 提出了一种端到端 TCP拥塞控制改进算法, 以达到提高网络链路中可用剩余带宽的利用率和提高网络链路吞吐量的目的.

3 传统端到端 TCP拥塞控制算法改进

3.1 对慢启动算法的修改

3.1.1 虚拟数据报文段探针思想

为了快速有效地探测到网络链路中可用剩余带宽以达到提高网络链路利用率的目的, 本文提出一种主动探测网络链路中可用剩余带宽方法. 该方法的基本思想是: 在一个端到端 TCP会话连接 RTT往返时间内, TCP拥塞控制算法发送端等时间间隔均匀地向网络中发送低优先级的数据报文段, 该数据报文段称之为虚拟数据报文段探针, 当 TCP拥塞控制算法发送端收到来自接收端反馈回来的所有虚拟数据报文段探针的确认数据报文段时, 则这时我们认为网络链路中拥有发送端所探测的可用带宽; 否则我们认为网络链路中不拥有网络链路中所需要的可用带宽. 并且大量文献[8, 9]表明这种主动测量网络链路可用剩余带宽技术方法是一种非常有效的网络测量测试技术方法.

3.1.2 网络链路中有效剩余带宽的探测

当一个端到端 TCP拥塞控制算法会话连接建立后, 端到端 TCP拥塞控制发送端收到从接收端反馈回来的接收端可接受拥塞控制窗口 (RWND)大小, 然后与发送端可接受拥塞控制窗口 (CWND)大小进行比较, 取其最小值作为端到端 TCP拥塞控制会话连接的拥塞控制窗口. 之后, 在一个端到端 TCP拥塞控制算法 RTT往返时间内, 发送端等时间间隔均匀地向网络中发送所要探测网络链路带宽数量的虚拟数据报文段探针, 如果发送端从接收端接收到所有虚拟数据报文段反馈回来的所有确认数据报文段, 则认为网络链路拥有端到端 TCP拥塞控制算法发送端所需要探测的网络链路中网络链路的可用带宽; 否则认为网络链路不拥有端到端 TCP拥塞算法发送端所需要探测的可用带宽, 在这种情况下, 端到端 TCP拥塞控制算法发送端便进入下一轮探测网络链路可用剩余带宽进程. 其算法如下:

```
Dummy_Packet_Start()
{
cwnd= 1;
r= RTT/rwnd; //rwnd为从接受端发送给发送端能
send(Dummy_Segment); //够接受的最大发送窗口的大小
for(i= 1; i< rwnd-1; i++)
{
    wait(r);
        send(Dummy_Segment);
}
}
```

图 1 虚拟数据报文探针探测网络链路中有效可用带宽算法

3.1.3　流量整形技术对拥塞避免阶段算法改进

端到端 TCP拥塞控制会话连接发送端通过向网络中发送虚拟数据报文段探针探测到网络链路中最大可用带宽之后, 端到端 TCP拥塞控制算法便开始进入拥塞避免阶段. 在该阶段中, 端到端 TCP拥塞控制会话连接的拥塞窗口进入线性增加的过程. 同时为了使得端到端 TCP拥塞控制算法发送端产生的数据报文段业务流显得更加地平滑, 本文采用了流量整形技术方法来对传统端到端 TCP拥塞控制会话连接在拥塞避免阶段的算法进行改进. 其发送数据报文段业务流的过程算法如下:

```
Congestion_ avoidance( )
{
    win= min( cwnd, rwnd);
    while( seqno< highest_ ack+ win) //if there is data to
send
  {
  if( cwnd> ssthresh) //if congestion avoidance mode
    {
      wbst= sRTT/win;
      wait( wbst);
       send(Data_ Segment);
       seqno++;
      }
    }
  }
```

图 2　拥塞避免阶段发送端发送数据报文段改进算法

3.2　算法分析

3.2.1　探测网络链路中最大有效可用带宽时间分析

为了比较传统端到端 TCP拥塞控制算法所采用的慢启动算法探测网络链路中最大可用带宽所需要的时间和通过采用虚拟数据报文段探针探测网络链路中最大可用带宽所需要的时间, 假设在网络链路不发生拥塞的情况下, 网络链路可用剩余带宽采用端到端 TCP拥塞控制算法在一个 TCP会话连接 RTT往返时间内所拥有的拥塞控制窗口来表示. 为分析简单起见, 这里我们假设网络链路中最大可用剩余带宽在一个 RTT往返时间内, 端到端 TCP拥塞控制会话连接所拥有的拥塞控制窗口为 64个数据报文段大小.

当一个端到端 TCP拥塞控制会话连接建立后, 发送端从接受端收到反馈回来的允许接收的最大拥塞控制窗口, 通过与发送端允许接收的最大拥塞控制窗口进行比较, 取其最小值作为 TCP拥塞控制会话连接的拥塞控制窗口, 假设这时所获得的端到端 TCP拥塞控制会话连接的拥塞控制窗口为 64个数据报文段. 对于本文所提出的虚拟数据报文段探针快速探测网络链路可用剩余带宽算法来说, 端到端 TCP拥塞控制算法发送端在一个 RTT往返时间内, 向网络链路中等时间间隔均匀地发送 64个虚拟数据报文段探针, 在一个 RTT时间内, 当 TCP拥塞控制算法发送端收到从接收端反馈回来的所有虚拟数据报文段确认数据报文段时, 则认为网络链路中拥有端到端 TCP拥塞控制发送端所探测的网络链路有效可用带宽; 否则, 认为网络链路中不拥有端到端 TCP拥塞控制发送端所探测的网络链路中有效可用带宽. 从上面的叙述中我们能够看出, 在网络链路不发生拥塞的情况下, 虚拟数据报文段探针快速探测网络链路中可用带宽算法在一个 RTT的往返时间内, 就能够探测到网络链路中最大的可用带宽. 然而, 对于传统端到端 TCP拥塞控制算法采用的慢启动算法来说, 由于其采用指数增加的方式来探测网络链路中是否拥有所需要探测的可用带宽, 每经过一个 RTT往返时间, TCP发送端所探测到的网络链路可用带宽都是 2倍于前一个 RTT往返时间内所探测到的网络链路可用带宽. 因此, 慢启动算法探测到网络链路中最大可用带宽所需要的时间为以 2为底的最大可用带宽的对数值个 RTT往返时间.

3.2.2　流量行为分析

为了能够方便地对上述所讨论的两种端到端 TCP拥塞控制算法发送端所产生的网络流量行为进行比较, 定义下面一些需要用到的物理量: MSS为 TCP发送端发送数据报文段大小, 单位为比特; T 为 TCP发送端发送一个数据报文段所需要的时间; ΔT 为 TCP发送端连续发送两个背靠背数据报文段的时间间隔; cwnd为端到端 TCP会话连接的发送窗口; RTT为端到端 TCP会话连接往返时间大小. 由 CPU处理器速度和端到端 TCP会话连接的 RTT往返时间范围我们知道: $\Delta T \ll RTT$, $T \ll RTT$, 所以有 $\Delta T \ll \frac{RTT}{cwnd} T \ll \frac{RTT}{cwnd}$ 因此, 我们有如下一些等式成立:

传统端到端 TCP拥塞控制算法发送端由于采用的是以背靠背的方式发送数据报文段, 因此, 发送端发送一个数据报文段平均所需要的时间为发送一个数据报文段所需要的时间 T 和发送两个背靠背数据报文段的时间间隔 ΔT 之和, 因此, 传统端到端 TCP拥塞控制算法发送端发送数据报文段的最大速率为:

$$v_{old} = \frac{MSS}{T + \Delta T} \tag{1}$$

然而, 通过对传统端到端 TCP拥塞控制算法发送端发送数据报文段过程的改进, 本文所提出的端到端 TCP拥塞控制算法发送端由于采用了如图 2所示等时间间隔均匀地发送数据报文段的方式, 在一个 RTT往返时间内, 发送一个数据报文段平均所需要的时间间隔为 $\frac{RTT}{cwnd}$, 因此, 本文所提出的端到端 TCP拥塞控制算法发送端发送数据报文段的最大速率为:

$$v_{new} = \frac{cwnd * MSS}{RTT} = \frac{MSS}{RTT / cwnd} \tag{2}$$

由于 $\Delta T \ll \frac{RTT}{cwnd}$, $T \ll \frac{RTT}{cwnd}$, 则有 $\Delta T + T \ll \frac{RTT}{cwnd}$, 于是有

下面的等式:

$$\frac{MSS}{T+\Delta T} \gg \frac{MSS}{\frac{RTT}{cwnd}} \Rightarrow v_{old} \gg v_{new} \quad (3)$$

从等式(3)中能够看出: 在网络链路吞吐量相等情况下,传统端到端 TCP拥塞控制算法发送端发送数据报文段的速率远远大于本文所提出的端到端 TCP拥塞控制算法发送端发送数据报文段的速率. 然而,传统端到端 TCP拥塞控制算法发送端由于采用的是背靠背的发送数据报文段的方式,也就是说,在一个 RTT往返时间内,当发送端收到从接收端反馈回来的已经发送数据报文段的确认数据报文段时,发送端便以背靠背的方式发送在拥塞控制窗口中允许发送的所有数据报文段,然后便进入一个空闲等待状态. 这种以量发送数据报文段的方式大大加重了网络流量突发的特性[10,11],网络流量的突发是造成网络数据报文段传输丢失的根本原因. 但是,由于本文所提出的端到端 TCP拥塞控制算法发送端发送数据报文段方式采用了流量整形技术,当有数据报文段发送时,TCP发送端等时间间隔均匀地发送需要发送的数据报文段,这种以速率发送数据报文段的方式使得其产生数据报文段流显得更加的平滑,有利于降低网络数据报文段的丢失率. 因此,从以上的分析能够得出如下结论: 在网络链路吞吐量相等的情况下,在微观时间范围内,本文所提出的端到端 TCP拥塞控制算法发送端所产生数据报文段的速率远远小于传统端到端 TCP拥塞控制算法发送端所产生数据报文段的速率,并且其产生的数据报文段流更加的平滑,这有利于降低数据报文段在传输的过程中由于拥塞发送丢失的概率.

4 实验结果

为了有效地评估本文所提出的端到端 TCP拥塞控制算法(TCP-Shape)的性能,本文将在 NS2系统环境下分别对文中所提出的端到端 TCP拥塞控制算法和传统端到端 TCP拥塞控制算法进行模拟试验并对其产生的性能结果进行比较. 在该试验模拟过程中,传统 TCP拥塞控制算法选择 TCP-Reno拥塞控制算法. 为了探测网络链路中可用剩余带宽, TCP-Reno端到端拥塞控制算法采用了慢启动算法和在拥塞避免阶段采用了 AIMD算法来探测网络链路中可用剩余带宽. 为了快速探测到网络链路中可用剩余带宽和提高网络链路的利用率,本文经过改进后的端到端 TCP拥塞控制算法(TCP-Shape)采用了虚拟数据报文段探针主动探测网络链路中可用剩余带宽方法和采用了流量整形技术来平滑发送端所发送的业务流. 为了实现改进后的拥塞控制算法,本文采用修改 TCP-Reno拥塞控制算法对应的慢启动算法部分的源代码和修改其在拥塞避免阶段发送业务流过程对应部分的源代码的方法来实现.

图 3为进行模拟试验时所采用的网络拓扑结构示意图. 在该示意图中,节点 A 和节点 B 作为端到端 TCP拥塞控制算法会话连接发送端来发送 TCP会话业务流,然后经过路由器节点 G 转发,到达端到端 TCP会话连接的接收端 C. 在进行模拟实验时,将把从节点 A 出发的端到端 TCP拥塞控制会话连接产生的 TCP会话业务流作为实验时网络链路的背景网络流量,并且其产生的业务流仍然采用传统的 TCP-Reno拥塞控制算法发送. 从节点 B 中出发的端到端 TCP会话连接产生的业务流将作为本文所观察的业务流. 其网络拓扑结构初始配置如图 3中所示,其中对转发路由器的配置采取数据报文端尾部丢弃(tail-drop)的队列管理存储模式,该队列最大长度设为 10数据报文段大小.

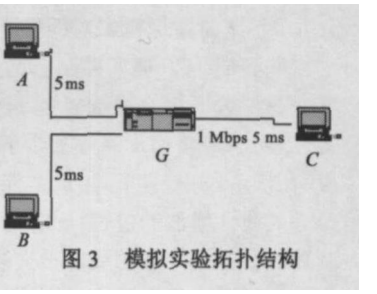
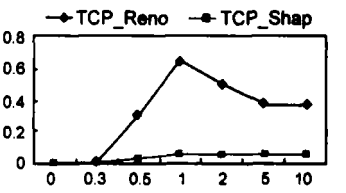


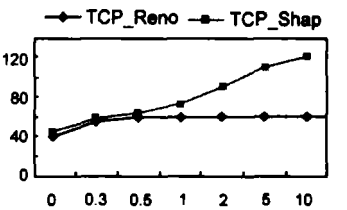
图 3 模拟实验拓扑结构

在进行模拟试验时,设置端到端 TCP拥塞控制算法会话连接的往返时间为 100ms,从发送端节点 A 和发送端节点 B 出发的端到端 TCP业务流都采用服从泊松分布过程方式的 FTP网络业务流来进行试验模拟. 对于发送端节点 A,在进行模拟试验时,将始终采用传统 TCP-Reno端到端拥塞控制算法来发送其所产生的 FTP业务流. 发送端节点 B 则分别采用传统的 TCP-Reno端到端拥塞控制算法和在本文中提出的改进后的拥塞控制算法来发送其所产生的 FTP业务流. 在端到端 TCP拥塞控制算法发送端以不同的速率发送 FTP数据报文段业务流时,观察在两种试验情况下数据报文段的丢失率情况和网络链路数据报文段的吞吐量情况.

图 4显示了本文的实验结果,其中在图 4(a)中显示了数据报文段在传输过程中丢失率的比较结果,从中能够看出,随着端到端 TCP拥塞控制会话连接发送端发送数据报文段速率的增加, TCP-Reno 和 TCP-Shape 会话连接的数据报文段丢失率都会随之增加. 然而在它们增加到了一定程度时,它们的丢失率将会保持不变. 然而,从另外一方面,从图 4(b)中所显示的结果能够看出: 本文所提出的 TCP-Shape网络拥塞控制改进算法却取得了良好的网络数据报文段的吞吐量. 为此我们有如下结论: 无论是在数据报文段丢失率性能上,还是在网络数据报文段吞吐量性能上,本文所提出的基于端到端测量的 TCP拥塞控制改进算法相比传统的端到端 TCP拥塞控制算法都取得了较



(a) 数据报文段丢失率比较示意图



(b) 网络吞吐量比较示意图

图 4 实验结果

好的实验结果.

5 结论

本文利用流量整形的思想,提出了一种针对传统以背靠背形式发送数据报文段的端到端 TCP拥塞控制算法的改进算法.在改进的算法中,采用了虚拟数据报文段探针探测网络链路剩余带宽的方法代替传统慢启动算法来快速地探测 TCP会话连接链路中的可用带宽,并在拥塞避免阶段,由于采用了流量整形技术,使得本文所提出的基于测量的端到端 TCP拥塞控制算法(TCP-Shape)所产生的数据报文段流比传统端到端 TCP拥塞控制算法所产生的数据报文段流显得更加的平滑.实验结果证明:本文所提出的端到端 TCP-Shape拥塞控制改进算法无论是在数据报文段的丢失率性能上还是在数据报文段的吞吐量性能上,与传统的端到端 TCP拥塞控制算法相比,都获得了更加优越的实验结果.

参考文献:

[1] V Jacobson. Congestion Avoidance and Control[A]. Proceedings of ACM SIGCOMM'88[C]. USA: ACM Press, 1988. 314-329.

[2] D Chiu, R Jain. Analysis of the increase and decrease algorithm for congestion avoidance in computer network[J]. Journal of Computer Networks and ISDN, 1989, 17(1): 1-14.

[3] Z Wang, J Crowcroft. Eliminating periodic packet losses in the 4.3-Taboe BSD TCP congestion control algorithm[J]. ACM Computer Communication Review, 1992, 22(4): 9-16.

[4] L S Brakme, S W Omalley. TCP Vegas: New techniques for congestion detection and avoidance[A]. Proceedings of the SIGCOMM'94[C]. Arizona, USA: ACM Press, 1994. 24-35.

[5] Ian F Akyildiz, Giacomo Morabito, Sergio Palazzo. TCP-Peach: A new congestion control scheme for satellite IP networks[J]. IEEE/ACM Transactions on Networking, 2001, 9(3): 307-320.

[6] Shudong Jun, Liang Guo, Ibrabin Matta, et al. A spectrum of TCP-Friendly window-based congestion control algorithms[J]. IEEE/ACM Transaction on Networking, 2003, 11(3): 341-355.

[7] Hyoung-Woo Park, Jin-Wook Chung. A study on reduction of traffic burstness using window based segment spacing[A]. IEEE the 15th International Conference on Information Networking[C]. New York: IEEE Inc, 2001. 41-15.

[8] Hao Jiang, Constantinos Dovrolis. The effect of flow capacities on the burstiness of aggregated traffic[A]. Proceedings of PAM2004[C]. New York: Springer, 2004. 247-256.

[9] Mark Carson, Darrin Santay. Micro-Time-Scale network measurements and harmonic effects[A]. Proceedings of PAM2004[C]. New York: Springer, 2004. 103-112.

[10] Walter Willinger, Murad S Taqqu, Rober Sherman, Daniel V. Willson. Self-Similarity through high-variability: statistical analysis of ethernet LAN traffic at the source level[J]. IEEE/ACM Transactions on Networking, 1997, 5(1): 71-86.

[11] Robert J Adler, Raisa E Feldman, Murad S Taqqu. A practical Guide to Heavy Tails: Statistical Techniques and Applications[M]. Cambridge, MA, USA: Birkhauser Boston Inc, 1998. 27-53.

作者简介:

程　京　女,1963年 7月生于武汉,博士研究生,副教授.曾参加国家自然科学基金、国家 863计划、国防"十一五"基础科研等多个项目.主要研究领域:可信系统与网络、网络测量等.

沈永坚　男,1978年 7月出生于江西都昌,湖南大学计算机与通信学院硕士研究生,研究方向:网络测量.

张大方　男,1959年 4月生于上海,博士,教授,博士生导师.曾主持国家自然科学基金、国家 863计划、国防"十一五"基础科研等多个项目,已发表学术论文 100余篇,其中 SCI, EI, ISTP检索 70余篇.现兼任中国计算机学会容错计算专委会副主任,教育部软件工程教学指导委员会委员等.主要研究领域:可信系统与网络、网络测量、软件容错等.

通信作者 E-mail: dfzhang@hnu.cn

黎文伟　男,1975年 9月出生于湖南沅江,博士研究生,研究方向为网络测量与 QoS.