

# 面向 M2M 通信的传输控制机制与方法

王 群<sup>1,2</sup> 钱焕延<sup>1</sup>

(南京理工大学计算机科学与技术学院 南京 210094)<sup>1</sup> (江苏警官学院公安科技系 南京 210031)<sup>2</sup>

**摘 要** M2M 通信是物联网的重要组成部分,其中基于 IP 的 M2M 通信方式实现了网络节点的统一编址、统一协调和统一管理,成为当前的研究重点和热点。传统 TCP 差错控制机制把数据丢失原因简单地归结为网络拥塞,并采用拥塞控制机制来解决,而忽略了无线连接存在的链路高比特误码率、拓扑频繁变化、信道非对称性、MAC 协议的不公平性等缺陷,使得传统 TCP 不适合在高可靠性 M2M 通信中使用。分析了影响 M2M 通信的 TCP 因素和改进方法,以端到端连接、拆连接和混合连接 3 种方式分别讨论了相应的解决方案,提出了 TCP 在 M2M 通信中的改进思路和研究方向。

**关键词** M2M 通信, TCP 协议, 拥塞控制机制, 优化方法

**中图分类号** TP393 **文献标识码** A

## Methods and Mechanisms on Transmission Control for M2M Communications

WANG Qun<sup>1,2</sup> QIAN Huan-yan<sup>1</sup>

(School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China)<sup>1</sup>

(Department of Science & Technology, Jiangsu Police Institute, Nanjing 210031, China)<sup>2</sup>

**Abstract** M2M communications are important parts of the Internet of Things(IoT). Among them, the IP-based M2M communications achieve a unified addressing, coordinated and unified management for nodes, is also a hot research topic for many years. In traditional TCP error control mechanism, the loss of data is simply as network congestion, and congestion control mechanisms are used to solve these problems. The traditional TCP ignores wireless link imperfections including high bit error rate, frequent topology changes, channel asymmetry, unfairness of MAC protocol and so on. Therefore, the traditional TCP is not suitable for secure M2M communications. This paper analyzed the TCP deficiencies and corresponding improving methods that impact the M2M communications, and discussed the appropriate solution respectively from end to end connections, split connections and mixed connections. Finally, the paper proposed TCP improving thinking and research directions for M2M communications.

**Keywords** M2M communication, Transmission control protocol, Congestion control mechanism, Optimization based technique

## 1 引言

被人们期望成为第三次信息产业浪潮和后互联网时代的物联网(Internet of Things, IoT),从广义上讲就是一个分别代表机器之间(Machine to Machine)、人机交互(Man to Machine, Machine to Man)和人与人之间(Man to Man)通信的 M2M 系统。其中,人机交互和人与人之间的通信在互联网时代已经成为现实,而实现机器与机器之间通信的 MTC(Machine Type Communication)方式则成为 M2M 系统体现物联网泛在互联特点的主要通信方式,也是 M2M 通信研究和应用的重点与热点<sup>[1]</sup>,同时也是本文主要讨论的通信模式。

在众多的 M2M 通信方式中,作为 IP 子网接入的纯 IP 技术不仅解决了物联网中 M2M 节点的寻址问题,还可以通过 NAT(Network Address Translation,网络地址转换)、IPv6 等

技术解决物联网地址不足以及缺乏应有的安全机制等问题。同时, M2M 网络中的信息感知节点具有了与核心网络中的节点在结构上完全一致的 IP 地址后,感知节点之间、感知节点与互联网节点之间将具有端到端的通信能力,实现了对物联网的全局统一编址、统一协调和统一管理功能,进而实现了以地址为中心的互联网与以数据为中心的物联网的融合。

在基于全 IP 的 M2M 通信方式中,节点间的通信不管是采用端到端的直通模式,还是利用 NAT 的间接通信模式, TCP 在无线网络中存在的先天不足<sup>[2]</sup>,都会影响到 M2M 通信性能。本文针对全 IP 的 M2M 通信模式,分析了 TCP 在无线通信中存在的主要缺陷及其产生原因,在此基础上针对 M2M 通信要求,将整个 M2M 通信过程划分成不同的连接类型(端到端连接、拆分连接和混合连接),分别介绍了典型的 TCP 改进技术,对比分析了其在 M2M 通信中的应用价值和

到稿日期:2013-01-23 返修日期:2013-04-15 本文受公安技术学,江苏省高等学校重点学科建设专项资金资助。

王 群(1971—),男,博士生,副教授,CCF 会员,主要研究方向为网络安全管理、物联网, E-mail: wqun@jspi.edu.cn; 钱焕延(1950—),男,教授,博士生导师,主要研究方向为下一代网络协议与通信模型、计算机网络应用等。

借鉴意义,并提出了进一步的研究方向。

## 2 基本概念

### 2.1 M2M 通信

如图 1 所示,根据功能域(Domain)的不同,将物联网中的 M2M 系统分为设备域、网络域和应用域共 3 个组成部分。其中,设备域与网络域之间通过 M2M 网关连接,而应用域与网络域之间由称为服务能力(Service Capabilities)的中间件互联。M2M 通信是指以下 3 类节点之间的通信:M2M 网络内部 M2M 节点(M2M 设备)之间、不同 M2M 网络的 M2M 节点之间以及 M2M 应用系统(M2M 应用服务器节点)与 M2M 网络中 M2M 节点之间。

M2M 网络是物联网中具有感知功能的系统,主要包括 WSN(Wireless Sensor Network)、MANET(Mobile Ad hoc Networks)、EPC(Electronic Product Code)系统、ZigBee、IEEE 802.15 等无线通信系统,也包括少量的有线网络(本文不作讨论),它通过各种有线(如 xDSL、有线电视网络等)和无线(如 WLAN、WinMAX、卫星通信等)方式接入核心网络。核心网是基于 IP 的统一数据格式、高宽带、可扩展的分组交换网络,具有异构互联和终端的移动接入与管理等特性,主要包括目前广泛使用的互联网、电信网和广电网以及演进中的 NGN(Next Generation Network,下一代网络)。M2M 应用通过核心网络,提供对感知数据的动态分析、处理、管理和控制功能,同时根据不同行业的需求提供相应的智能化应用和服务。

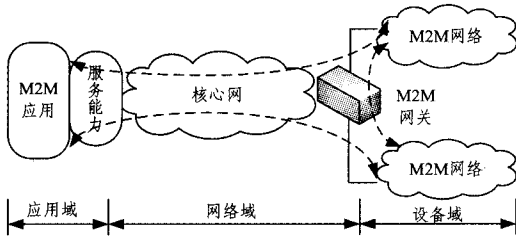


图 1 物联网中的 M2M 通信结构

### 2.2 TCP 的拥塞控制机制

TCP(Transmission Control Protocol,传输控制协议)是早期专门为有线网络设计的传输层协议,它在不可靠的网络层(IP 层)基础上,利用窗口、序列号、重传定时器和确认技术,实现流量控制、差错控制和拥塞控制功能,提供可靠的、面向连接的、端到端的数据传输服务。其中,拥塞控制对 M2M 通信的影响最大,本文进行重点分析。

目前,Internet 广泛使用的 TCP Reno 协议通过慢启动(Slow Start)、拥塞避免(Congestion Avoidance)、快速重传(Fast Retransmit)和快速恢复(Fast Recovery)共 4 种算法及其相互之间的动态切换(如图 2 所示)来实现拥塞控制功能。具体过程为:当开始建立 TCP 连接时,拥塞窗口(cwnd)初始化为一个最大报文段(Maximum Segment Size, MSS)值,即  $cwnd=1$ (1 表示一个 MSS 值,但实际以字节为单位表示),随后进行如下判断(其中,  $ssthresh$  表示慢启动门限值,1 个  $ssthresh$  一般设置为  $2^{16}$  字节):

if ( $cwnd > ssthresh$ )

当值为否时,采用慢启动算法来探测网络的可用带宽,当接收到一个 ACK(Acknowledgement)报文后,按式(1)让  $cwnd$

值按指数方式增长;

$$cwnd_{new} = cwnd_{old} + 1 \quad (1)$$

当值为真时,采用拥塞避免算法,避免可能发生的拥塞,并尽可能地探测可用带宽。当接收到一个 ACK 报文后,按式(2)让  $cwnd$  值按线性增长;

$$cwnd_{new} = cwnd_{old} + \frac{1}{cwnd_{old}} \quad (2)$$

发送端连续收到 3 个重复应答包(Duplicate Acknowledgement, DupACK)时,采用快速重传算法重发 DupACK 对应的包,并采用式(3)重新为  $cwnd$  与  $ssthresh$  赋值,以避免进入慢启动阶段;

$$\begin{cases} cwnd_{new} = cwnd_{old} / 2 \\ ssthresh = \max\{2, cwnd_{old} / 2\} \end{cases} \quad (3)$$

重传定时器(Retransmission Timeout, RTO)超时,进入慢启动阶段。

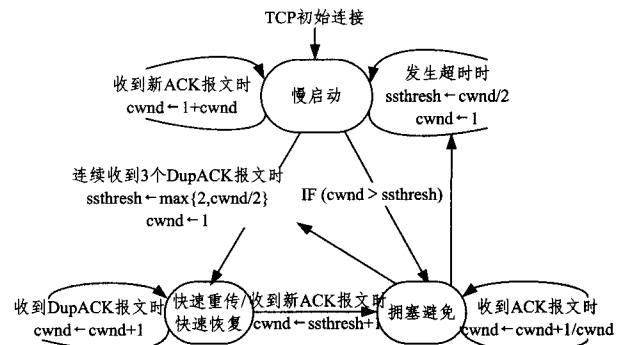


图 2 TCP 拥塞控制机制的有限状态机

由以上分析可以看出,保持 TCP 状态的相对稳定,并尽量减少状态切换,是保持 TCP 传输稳定性的前提,而要实现这一点,必须避免在网络未出现拥塞时发生超时和发送端收到 DupACK 报文,这也是 M2M 通信中需要重点研究和解决的问题。

## 3 M2M 通信中的 TCP 机制评价与分析

### 3.1 影响 M2M 通信的 TCP 因素

#### 3.1.1 TCP 协议对无线链路的不适应性

TCP 的拥塞控制机制在有线网络中表现出了其应用优势,但却严重影响了以无线方式为主的 M2M 通信的性能。在早期专门针对有线网络环境设计的 TCP 协议中,由于链路误码率非常低,因此 TCP 将丢包认为是由于网络发生拥塞或产生意外时延(如网络中断)引起的,从而启动拥塞控制机制。而在无线网络中, TCP 报文段和 UDP 数据报的丢失主要是由无线信道与生俱来的高比特差错率(Bits Error Rate, BER)和移动切换(Mobile Handoff)引起的<sup>[3]</sup>。但是,在无线网络中, TCP 将这种因链路层出错而产生的丢包误认为是网络发生拥塞而引起的,进而启动拥塞控制机制,导致网络性能下降。BER 对 TCP 性能的影响可以用式(4)来分析<sup>[4]</sup>:

$$T \approx \frac{S}{RTT \times \sqrt{\frac{2p}{3}} + RTO \times \min[1, 3\sqrt{\frac{3p}{8}}] \times p \times (1 + 32p^2)} \quad (4)$$

式中,  $T$  为发送速率,单位为 B/s;  $S$  为数据包大小,单位为 B;  $RTT$  为网络往返时间(Round Trip Time),单位为 s;  $RTO$  的单位为 s;  $p$  为稳定状态下 TCP 报文段的丢包率。

然而, M2M 网络主要以无线通信为主, 无线链路本身就是一种无导向的有损链路, 其信号传输机制与 TCP 协议对链路层的定位存在着重大偏差。为此, 在 M2M 网络中, 需要通过 TCP 协议的研究, 提出必要的改进措施, 以适应无线链路的信号传输要求。

### 3.1.2 TCP 协议对无线网络拓扑频繁变化的不适应性

MANET 是 M2M 网络的主体, 而 MANET 是一种没有固定通信设施的移动自组织网络, 移动节点既是感知设备也是分组转发设备。移动节点位置的快速变化导致网络拓扑频繁变化, 而传统有线网络中的分组转发机制无法适应拓扑动态变化的要求。当拓扑发生变化后, 只要源和目的 IP 地址保持不变, 当路由由恢复或重新建立后, TCP 后续的传输就能保持, 但这时 TCP 将进入慢启动阶段, 并重传丢失的包。

对于因路由中断而产生的丢包, TCP 协议仍然将其当作拥塞来处理, 这将大大降低 TCP 的性能。同时, 因节点移动而出现的路由中断, 还会导致接收端数据包的乱序, 乱序会让接收端发送大量的 DupACK 报文, 这会使发送端重发不必要的报文段, 并进入拥塞避免阶段。

### 3.1.3 无线信道的非对称性与 TCP 机制的不适应性

M2M 网络是一种典型的无线多跳连接网络, 存在着因信道占用、速率动态分配、多路径路由等因素引起的 TCP 传输的不对称性, 从而有可能产生 TCP 数据流与 ACK 流的不对称现象。尤其当 ACK 流相对延迟时, 发送端会因 RTO 延时而重发, 并进入慢启动阶段, 接收端也会收到大量的重复包, 对重复包的处理会消耗较多的节点资源。

另外, 由于 TCP 数据流和 ACK 流的不对称性, TCP 很难准确确定 RTT 值, 上一次确定的 RTT 值在本次传输中有可能不再适合。而 RTT 值是确定  $cwnd$  值变化的依据, RTT 值的振荡性决定了对网络拥塞判定的不确定性和网络的不稳定性。

同时, 当 M2M 网络为卫星通信网络时, 将会存在上行链路带宽和下行链路带宽的严重不对称, 再加上卫星链路的高延时特点, 使得标准的 TCP 因出现 ACK 报文传输超时或发送端 RTO 超时而进入拥塞控制, 降低了 TCP 性能。

### 3.1.4 802.11 MAC 协议的不公平性与 TCP 流间的不适应性

Ad hoc 网络是 M2M 网络的主体, IEEE 802.11 DCF (Distributed Coordination Function) 提供了一种分布式信道竞争接入协议, 它已成为 Ad hoc 网络事实上的 MAC 规范。然而, IEEE 802.11 DCF 协议的不公平性与 TCP 流所需要的稳定性之间存在着不适应性。

在 802.11 DCF 协议中, 当节点成功发送一个数据帧后, 便会在  $[0, CW_{min}-1]$  之间按均匀分布随机选择一个值作为下一个数据帧的竞争窗口, 其中  $CW_{min}$  为最小竞争窗口; 如果节点发送失败, 则采用二进制指数退避算法, 从  $[0, 2^m \times CW_{min}-1]$  之间随机选择一个值作为重发数据帧的竞争窗口, 其中  $m$  为尝试失败的次数。显然, 当一个节点成功发送后, 其拥有的竞争窗口则最小, 进而竞争优势也越明显, 这将导致节点在争用信道时的不公平性。同时, 802.11 DCF 的隐藏站点和暴露站点问题, 加剧了这种不公平性。

802.11 DCF 站点在争用信道时的不公平性, 导致在共享无线链路的 M2M 网络中, 节点间因同时发送数据帧而产生碰撞后必须回退再检测信道, 正在发送中的数据帧也会因为

无线链路的有损性而产生传输错误需要重传。无线网络 MAC 层的这些现象对于高层的 TCP 来说都是不可知的, TCP 机制都将其看作是网络拥塞, 从而启动拥塞控制机制, 降低了网络的性能。

另外, 802.11 MAC 协议采用分段(fragment)机制将上层服务数据单元(MSDU)分成多个较小的 MAC 层协议数据单元(MPDU)后分别发送, 这种机制称为 TCP 分割(Split TCP)。对于 TCP 接收端来说, 它只有在接收到一个完整的 TCP 报文段后才能进行转发, 这将使 TCP 报文段的转发延时增大, 导致 RTO 延时进入慢启动阶段, 影响传输性能。

还有, 当 M2M 网络为蜂窝网络或由多 AP (Access Point) 组成的 WLAN 时, 移动节点在蜂窝或 AP 之间切换时会存在一个信号消失阶段。在此期间, 将会造成发送节点的 RTO 超时, 进而引起 TCP 重发数据, 并启用拥塞控制机制, 导致 TCP 性能下降。

## 3.2 M2M 通信中 TCP 协议改进方法评价

M2M 通信同时包括移动节点之间、固定节点之间以及移动节点与固定节点之间的通信, 是一个复杂的通信过程。为此, 对于 M2M 通信中 TCP 协议的改进, 需要针对具体网络环境来提出相应的方案, 以最大可能地解决 M2M 通信与传统 TCP 协议的不适应性。

### 3.2.1 显式拥塞通知(ECN)

显式拥塞通知(Explicit Congestion Notification, ECN)<sup>[5]</sup>通过修改 IP 分组头部的 TOS (Type of Service) 字段值, 使分组收发双方在网络层显式交换当前网络状态参数, 并提前调整网络运行机制, 避免上层(传输层) TCP 拥塞发生。ECN 采用 RED (Random Early Detection, 随机早期检测) 算法来探测网络可能发生的拥塞, 并通过修改 ECN 字段来告知网络当前的拥塞情况。ECN 不需要依赖 TCP 粗粒度的超时重传机制, 当 M2M 网络的延时不确定时, ECN 将会取得很好的应用效果。

ECN 属于捎带机制, 具体由 IPv4 头部 TOS 字段的第 6bit 和 7bit 显式实现。其中, 第 6bit 称为 ECT (ECN-Capable Transport) 位, 用于协商网络是否支持 ECN 机制, 当值为 1 时表示支持, 当值为 0 时表示不支持; 第 7bit 称为 CE (Congestion Experienced) 位, 主要由路由器设置, 用以通知终端节点、路由节点当前缓存的使用情况, 当 CE 设置为 1 时表示缓存已满, 即将出现拥塞, 当 CE 设置为 0 时表示缓存使用正常。终端节点在接收到捎带有 ECN 的分组后, 通过返回 ECN-echo ACK 报文通知发送端网络的拥塞情况, TCP 发送端据此能够提前发现网络的拥塞, 并避免了路由器的缓存溢出, 减小了分组的丢失, 缓解了网络的拥塞, 提高了 TCP 的效率。

ECN 机制在许多 TCP 增强协议中得到了应用。例如, RBCC (Rate-Based Congestion Control)<sup>[6]</sup> 在分组头部增加了一个 ECN 反馈域, 分组经过的每个中间节点首先计算本节点允许的最大传输速率, 然后与反馈域中的值进行比较, 若小于反馈域中的值则替换之。通过这样的机制, 发送端可以精确控制发送速率, 避免进入拥塞控制过程。再如, 文献[7]针对无线网络通信的不稳定性问题, 提出了节点用 MRAI (Minimum data Rate Adjustment Index) 值来表示本节点能够为 TCP 流提供的最大带宽, 发送端在 TCP 头部设置一个 ECN

反馈域,每个节点在接收到该 TCP 报文段时,将本节点的 MRAI 值与反馈域中的对应值进行比较,若小于反馈域中的值时则替换之。发送端在收到接收端的 ACK 报文后,便根据最后的 MRAI 值对发送流量进行控制。

在 M2M 网络中,ECN 机制通过对 IP 头部 TOS 字段部分位的设置来实现对分组传输状态的显式通知,并未改变头部字段的长度,在不增加路由和终端节点负担的情况下,提升了 M2M 设备的 TCP 通信性能。

### 3.2.2 显式丢失通知(ELN)

使用显式丢失通知(Explicit Loss Notification, ELN)<sup>[8]</sup>可以及时通知发送端当前的 TCP 传输延时或丢失是由于 MAC 层的数据帧丢失引起的,而与 TCP 拥塞无关。与 ECN 一样,ELN 也通过捎带方式来实现。但在 ELN 中 TCP 头部通过 1bit 来显式地通知终端节点 TCP 时延或丢失是由 MAC 层引起或是由传输层引起的。MAC 层通过 CRC 等校验技术来验证数据帧在端到端链路传输中是否出错,如果出错,发送端将重传该帧,直到接收端成功接收为止。而此过程,终端节点却一无所知。通过采用 ELN 机制,可以将 MAC 层出现的错误显式地通知给 TCP 协议,避免进入拥塞控制过程。同样,ELN-echo ACK 报文也可以将 MAC 层信息告诉发送端,使发送端暂缓发送数据。另外,当链路层的重传超过一定次数时,MAC 协议将认为链路出现故障。此时,ELN 也会将此事件捎带给传输层。

ELN 机制的工作原理类似于 ECN,只是 ELN 通过定义 TCP 头部字段的某一位来实现对低层差错的显式通知,ELN 的启用只需要在相互通信的两个终端节点间协商即可,不涉及对中间路由节点操作模式的修改,非常适合 M2M 中端到端的 TCP 通信。

### 3.2.3 ARQ 和 FEC

链路层协议的主要功能是在不可靠的链路上实现可靠的数据传输。在整个网络体系中,只有链路层两端最清楚产生数据帧差错的原因,并做到局部问题局部解决,其中 ARQ(Automatic Repeat reQuest,自动请求重传)和 FEC(Forward Error Correction,向前纠错)是典型的链路层差错处理技术。

在链路层的发送端,ARQ 将上层下传给它的分组(如果中间路由由节点没有进行分片操作,1 个分组中将承载着 1 个 TCP 报文段)分成多个数据帧后分别传输,每个数据帧对应一个计时器以防数据帧丢失。接收端在收到数据帧后需要立即回传 ACK 确认帧,若发现数据帧出错或丢失则回传 NACK(Negative ACKnowledgement)确认帧。发送端在收到 NACK 或计时器超时后,将重传丢失的数据帧。接收端将数据帧排序并还原成原始的分组,最后提交给传输层协议处理。ARQ 的按序提交机制确保了分组提交给高层时不会因重传而打乱,解决了 M2M 无线通信中存在的乱序问题,而且 ARQ 机制的实现非常高效,对 M2M 终端配置要求较低。同时,作为 MAC 层协议,ARQ 对 IP 和 TCP 头部信息是透明的,上层的协议修改或加密数据的传输对 ARQ 是未知的,这使 ARQ 具有普适性,非常适合采用不同技术的 M2M 通信。

FEC 协议对需要发送的原始数据帧进行编码以加入冗余数据,当接收端发现错误时可利用冗余数据恢复原始数据。FEC 是一种典型的本地纠错机制,在延时较大的网络中由于避免了重传,因此非常适合传输信道容量远大于单个用户通

信流量的网络(如 CDMA2000、WCDMA 等)。不过,FEC 码率(即冗余数据占编码后数据的比例)的设置会影响数据传输的可靠性,冗余数据传输也会增加系统开销,而且无线网络信道质量不稳定,往往需要根据当前具体的通信质量状况动态调整 FEC 码率,以达到应用的最优化,但这在规模较大的 M2M 通信中很难做到。因此,FEC 机制在小规模、单一模式的 M2M 无线通信中很适用,但在大规模、跨网络互联的复杂 M2M 通信中以及 M2M 设备性能受限的网络中不适合采用。

## 4 TCP 改进协议在 M2M 通信中的应用分析

根据目前的研究成果及 M2M 网络的通信特点,可以将 TCP 协议在 M2M 通信中的应用改进<sup>[9-11]</sup>分为以下 3 种类型:端到端连接、分段连接和混合连接。

### 4.1 端到端连接

端到端连接技术由收发双方直接建立 TCP 通信机制,保持了连接的端到端 TCP 语义,不需要对中间设备进行改进,易于在异构网络中实现。目前,典型的端到端 TCP 连接技术主要有以下几种:

(1)F-TCP。F-TCP(Freeze-TCP)<sup>[12]</sup>协议主要解决移动节点信道切换所引起的数据丢失问题,其主要实现原理为:位于 M2M 网络中的 MN(Mobile Node)探测无线信号的能量,检测出可能会发生的信道切换事件;当切换操作即将发生时,MN 向发送端发送一个窗口值为零的通告包,从而迫使发送端进入 ZWP(Zero Window Probe)状态;在 ZWP 状态下,发送端将保持当前拥塞窗口并增加 RTO 超时时间;一旦 MN 信道切换成功,MN 就向原先的数据发送端连续发送 3 个针对链路中断前最后接收到的包的 DupACK 报文,告知发送端可以解除 ZWP 状态并立即发送数据。

使用 F-TCP 协议,发送端可以在接收端成功切换信道后按原有窗口大小立即发送数据,提高了 TCP 协议的性能。但是,当 MN 从一个子网切换到另一个子网时,发送端仍然按照原来窗口大小发送数据,这是否能够适应新子网的通信要求,在协议中还未考虑。

(2)Fixed-RTO。文献[13]通过对比分析 MANET 中多种路由协议环境下的 TCP 改进时发现,路由中断引起的 RTO 呈幂指数增长,并基于此现象提出了 Fixed-RTO 协议。该协议在端节点通过分析 RTO 超时来判断路由断开或恢复事件,其中当连续发生 RTO 超时时,发送端认为这是因路由中断引起并避免进入拥塞控制过程,同时仅重传没有得到确认的包而不改变 RTO 的值。

Fixed-RTO 协议在发现因路由断开而进行重传时保持固定的 RTO 值,抑制了 RTO 的增长,使 M2M 无线通信保持在一个相对稳定的状态。但是,Fixed-RTO 容易产生因链路层重传(路由中断)的延时而导致的 TCP 超时重传,即伪重传(Spurious Retransmission),导致接收端重复包的出现,这对处理能力和存储空间相对有限的 M2M 移动节点来说增加了部署难度。另外,该协议在纯 M2M 无线网络中应用较好,但在有线和无线混合网络中性能较差。

(3)TCP-DOOR。在传统的 TCP 协议中,因链路层重传和路由改变引起的接收端乱序现象,将导致 TCP 快速重传。TCP-DOOR(TCP with Detection of Out-of-Order and Response)<sup>[14]</sup>通过在收发节点的同时分析 TCP 报文段的序号和

ACK 报文来检测乱序的产生是因路由中断还是路由改变引起的,之后发送端将禁止 TCP 的拥塞控制,并暂时冻结 TCP 的各个参数;当路由恢复后,TCP 立即启用 TCP 拥塞控制,并解冻 TCP 的参数。

TCP-DOOR 在固定或低速移动节点中的应用较为理想,但当 M2M 节点快速移动、网络拓扑频繁发生变化时,该协议在响应时间、检测精度等方面的表现较差。

(4) ABRA。ABRA (Adaptive Back-off Response Approach)<sup>[15]</sup>采用了类似于 TCP-DOOR 的 TCP 参数冻结机制,当发生 RTO 超时时将冻结当前 TCP 的各个参数,当收到重传 TCP 报文段的 ACK 报文时将解冻,避免了 RTO 呈幂指数增长。

与 TCP-DOOR 不同的是,ABRA 不需要网络层和链路层的反馈信息,而直接在传输层操作,避免了对网络中间节点的修改,提高了协议的效率。

(5) EXACT。EXACT (EXplicit rAte-based flow Control)<sup>[16]</sup>是一种基于速率的拥塞控制协议,发送端在包头部添加 ER(Explicit Rate)和 CR(Current Rate)两个域,其中 ER 表示期望速率,其初始值为发送端期望的最大速度;CR 表示当前速率,其初始值为发送端实际发送该数据流的速率,一般情况下 ER 要远大于 CR(例如,某一发送端通过一条低速网络进行 FTP 大文件传输时便是这样)。当某一数据流经中间路由由节点时,路由节点首先读取并记录该数据流中的 CR 值,然后根据自身带宽计算本节点可以分配给该数据流的速率,如果小于该数据流包头部的 ER 值则替换之。接收端将收到的 ER 值通过反馈告知发送端,这样发送端便可以根据该 ER 值控制发送速率。

EXACT 协议的一个特点是提出了安全窗口的概念,发送端要求未经应答的包数量要小于该窗口值,这样可以避免在路由发生中断时发送端过度发送数据;另一个特点是采用 SACK(Selective ACKnowledgement)方式代替传统 TCP 中的 RTO,只有当未应答序号与最大应答序号相差较大时才重传该序号对应的包。EXACT 主要针对 MANET 提出,非常适合于 M2M 无线网络节点之间的通信。

## 4.2 分段连接

分段连接是以 M2M 网关(见图 1)为中间分割节点,将 M2M 节点之间的 TCP 通信拆分为两个阶段。分段连接可以分为两种模式:MN↔GW↔FN 和 MN↔GW↔MN,其中,MN(Mobile Node)为 M2M 网络中的移动感知节点,GW(gateway)为 M2M 网关,FN(Fixed Node)为有线固定接入方式的 M2M 应用主机。目前,典型的分段连接技术主要有以下几种:

(1) I-TCP。I-TCP(Indirect TCP)<sup>[17]</sup>适用于 MN↔GW↔FN 环境,MN↔GW 之间使用无线连接,而 GW↔FN 之间使用有线连接,每段分别采用自身线路特点和要求的 TCP 协议,TCP 在有线和无线网络中的差异性由 GW 区分开来,优化了 MN 与 FN 间的通信性能。

但是,GW 在收到 FN 的包后,立即回复 ACK 报文。当 FN 收到该 ACK 报文时,便判断对应的包已被对端(MN)接收到。但实际上,由于 GW↔MN 之间无线链路的不稳定性,GW 收到的包并不一定正确地传输给 MN。以上现象,破坏了端到端的 TCP 语义规定。

(2) M-TCP。M-TCP (TCP for Mobile Cellular Net-

works)<sup>[18]</sup>的原理是:位于逻辑连接中间节点的 GW 通过相应的算法检测发送到 MN 的包和回传的 ACK 报文,判断与该 MN 之间的连接状态;如果发现与该 MN 之间的链接发生中断,GW 就向另一侧的 MN 或 FN 发送一个窗口值为零的通告包,迫使其 TCP 协议进入保持状态(Persist Mode);当 MN↔GW 之间的连接中断恢复后,GW 便通知原来处于保持状态的 MN 或 FN 退出该状态,恢复到先前状态。通过这种机制,接收端的 TCP 中断信息会及时反馈给发送端,发送端通过进入保持状态而避免进入 TCP 拥塞控制,当连接恢复后发送端仍然以原先的窗口大小发送数据,提高了带宽的利用率。

M-TCP 与 F-TCP 一样使用了零窗口机制,但 M-TCP 协议的实现由 GW 负责相关信息的转发,而 F-TCP 则是由 MN 通过探测信号强度来决定是否切换信道,当切换到新的信道后再利用 3 个 DupACKs 报文通知发送端重发数据,在相同环境下 F-TCP 的效率要高于 M-TCP。

在分段连接机制中,所有分段的控制信息和缓存数据都保存在 GW 中,GW 的可靠性和稳定性在分段连接中起着决定作用。但在 M2M 通信过程中,承担 GW 功能的 M2M 网关一般具有较强的处理能力和较大的存储空间,为分段连接的实现提供了基础保障;另一方面,当 M2M 节点位于不同的 M2M 网络中时,M2M 网关之间的通信由高速核心网络连接,能够确保数据传输的可靠性。

(3) TCP-F。TCP-F(TCP-Feedback)<sup>[19]</sup>是 Ad hoc 网络中较早出现的通过拆分连接技术来解决路由中断的 TCP 改进方法。网络在正常情况下运行时使用传统的 TCP 协议,当 GW 监测到接收端所在的下行网段中出现路由中断时,便向上行网段中的发送端发送一个 RFN(Route Failure Notification,路由失败通知)包。当发送端接收到 RFN 包时,将进入休眠(snooze)状态,且立即停止发送数据,并冻结重传超时计时器和拥塞窗口的参数;当 GW 监测到路由恢复后,便向发送端反馈一个 RRN(Route Re-establishment Notification,路由重建通知)包。当发送端接收到该 RRN 包时,便解除 snooze 状态,恢复正常的 TCP 传输。

TCP-F 协议通过 GW 监测节点两侧网络运行状况,利用 RFN 和 RRN 包来告知发送端路由中断和恢复情况,使发送端通过暂时冻结 TCP 相关参数而避免进入拥塞控制,实现原理相对简单,且易于实施,非常适应在 M2M 网络中的应用。

(4) TCP-Bus。TCP-Bus<sup>[20]</sup>是对 TCP-F 协议的改进,也是一种在网络层由 GW 监测路由中断并将这一事件反馈给发送端,以避免发送端采取不适当操作的机制。类似于 TCP-F,TCP-Bus 协议也采用两个控制包:ERDN(Explicit Route Disconnection Notification,显式路由中断通知)和 ERSN(Explicit Route Successful Notification,显式路由重建通知)。GW 在监测到路由中断时,便通过向发送端发送 ERDN 包告知此事件,发送端在接收到该包时立即停止数据发送;当 GW 监测到路由恢复后,便通过发送 ERSN 包将此事件反馈给发送端,发送端在接收到该包后恢复数据发送。

与 TCP-F 相比,TCP-Bus 协议的主要改进为:将路由中断事件通过网络层路由协议 ABR(Associativity-Based Routing)通告给传输层;考虑到了 ERDN 和 ERSN 包在传输过程中可能出现的丢失,将发送端因超时重传的可能性降到最低;使用 SACK 机制;在 GW 中缓存一些尚未得到接收端已成功接收的数据,当路由重建后由 GW 快速重传这些数据,避免

发送端出现重传超时。

### 4.3 混合连接

混合连接技术的实现不单纯依赖某一层的功能,而是同时结合多层功能,通过多层功能的联动来实现。目前,典型的混合连接技术主要有以下几种:

(1) Snoop。Snoop<sup>[21]</sup>技术的实现同时涉及链路层和网络层,其中通过链路层的 ARQ 机制来解决无线信道存在的高 BER 问题,而在网络层采用 IP 组播技术解决不同信道间切换而引起的数据丢失问题。

Snoop 技术将数据流向分成两种情况,分别采取不同策略来处理:当数据从 FN 发往 MN 时, GW 采取传统网络中常用的存储转发方式,将发往 MN 的数据在缓存中保留一份。对于无线信道上传输的数据,可采取两种方式防止数据的丢失。一种是设置适合链路层的 RTO,实现链路层数据帧的超时重传。另一种是检测从 MN 到 FN 的 DupACK,当收到第一个 DupACK 时, GW 重发丢失的数据帧,同时丢弃后到的 DupACK,防止 DupACK 到达 FN 进而启动不必要的拥塞控制机制;当数据从 MN 发往 FN 时, GW 检测是否存在能量空洞(靠近 GW 的节点容易过早耗尽自身的能量,从而形成能量空洞现象),一旦发现存在能量空洞,则将其通过带有 ELN 的 NACK 报文反馈给 MN,通知重新发送数据包。

在 Snoop 协议的基础上,提出了一些改进协议,如 EBSN (explicit bad state notification, 显式坏状态通知)<sup>[22]</sup>是在 GW 上监测 GW↔MN 间无线链路的性能,并将该信息反馈给 FN, FN 根据该反馈信息动态调整 TCP 的 RTO 的触发时间,提高了数据从 FN 向 MN 的发送性能。

Snoop 技术非常适合基于 MN↔GW↔FN 模式的 M2M 通信,不但确保了端到端的 TCP 语义,而且对 GW 的修改较少(只需要增加一个用于监测 TCP 连接的 Snoop 代理模块),非常适合分布式 M2M 节点的部署。

(2) TCP-ELFN。TCP-ELFN (TCP with Explicit Link Failure Notification)<sup>[23]</sup>是将 ELFN (Explicit Link Failure Notification, 显式链接失败通知)方式应用到 TCP 的一项技术。TCP-ELFN 的工作原理类似于 TCP-F, 当发送端接收到含有 ELFN 信息的包时,进入类似于 TCP-F 中“snooze”状态的“stand-by”模式,冻结 TCP 的各项参数,避免进入拥塞控制。但与 TCP-F 不同的是:TCP-ELFN 通过低层反馈感知链路层中断或网络层路由中断;TCP-ELFN 没有引入另外的控制包,而是通过 DSR (Dynamic Source Routing, 动态源路由协议)的反馈信息或 ICMP (Internet Control Message Protocol, Internet 控制包协议)的主机不可到达 (Host Unreachable) 包来得知链路或路由中断。当收到发送队列中第一个包的确认包时,得知路由恢复。ECP-ELFN 因在实现方式上的简洁性而更适合在 M2M 通信中应用。

为了进一步优化 ELFN 反馈机制,在 TCP/RCWE (TCP with Restricted Congestion Window Enlargement)<sup>[24]</sup>协议中提出了 RCWE 算法,该算法实时监测 RTO 值,当其增大时拥塞窗口保持不变,当其值减小或不变时将按照 TCP 自身的机制增加拥塞窗口。仿真结果显示, TCP/RCWE 采用有限拥塞窗口扩大机制,可以较好地解决 MANET 中随机丢包问题,其性能优于 TCP-ELFN 协议。

(3) ATCP。ATCP (Ad hoc TCP)<sup>[25]</sup>通过在网络层与传输层 (IP 与 TCP) 之间增加一个中间协议层,并利用显式反馈

方式来向发送端及时反映当前网络状态参数,进而采取相应的控制机制,避免网络拥塞的发生。ATCP 通过 3 种方式来监测并判断网络状态参数:1) 判断任意两个相邻数据包的达到时间间隔,当某一时间间隔超过正常值时意味着可能发生了拥塞;2) 在某一时间段(一般设置为 10ms~30ms)内网络的吞吐量突然减少时意味着可能发生了拥塞;3) 数据包到达的顺序如果发生乱序意味着可能发生了路由中断。接收端将以上监测值通过 ECN 和 ICMP 的目标不可到达“Destination Unreachable”反馈给发送端。发送端根据接收到的反馈信息作出相应的操作,可选择 persist, congestion 和 retransmit 3 种状态。

ATCP 通过增加一个中间层来屏蔽传输层以下的差异性,实现了低层对传输层的透明传输,使得 ATCP 协议不但兼容传统的 TCP 协议,而且很好地适应了有线和无线混合网络的通信要求;ATCP 协议的设计初衷是通过分析干扰对 MANET 网络状态参数的影响来提出判断网络发生拥塞方法的可靠性和准确性,所以 ATCP 协议除能够判断和处理路由中断事件外,还可以处理无线链路高 BER 问题,显示了该协议较为全面的功用。ATCP 所具有的对低层协议的屏蔽性、对有线和无线混合网络的适用性以及对 TCP 协议的兼容性等特点和实现细节,值得 M2M 通信借鉴。

### 4.4 不同连接技术比较

端到端技术通过估算信号能量、监测链路状态及路由中断与恢复、选择性重传等方法实现了 TCP 协议从有线网络向无线网络的迁移,进而实现在有线与无线混合网络环境中的应用,对 M2M 通信过程中局部问题的解决提供了参考。但是,这种就问题而解决问题的方法缺乏全局性考虑,不适应对整体要求较高的 M2M 通信的直接应用。

物联网是一个融合现有和演进中的各类网络技术的泛在互联网,拆分连接技术将一个网络连接拆分成多个由有线和无线网络组成的网段,不同网段分别使用适合自身网络通信要求的控制协议来增加传输的可靠性,并提高了协议的综合效率。拆分连接技术的主要缺陷是部分协议违反了端到端的 TCP 语义,这在部分对节点身份信息要求较高的 M2M 通信中是不允许的。另外, GW 的设计也因需要完成协议转换、数据缓存与重传、链路和路由状态监测等操作而变得较为复杂,在一定程度上增加了大规模部署时的难度。

混合连接技术试图采取低层问题由低层解决,而不是将问题全部上交给传输层,该技术将传输控制功能交给链路层和网络层来完成,具有如下优势:1) 充分发挥了低层协议(如 MAC 协议、IP 协议、ICMP 协议等)的功能;2) 低层协议更容易知道网络状态参数,也能够更早地知道发生的链路和路由中断等事件;3) 分担了传输层的任务,简化了传输协议设计上的复杂度。其缺点是必须协调不同层协议的实现,有可能发生不同层控制功能之间的冲突。

**结束语** 随着移动 IP 网络应用的快速发展,基于无线和有线混合的 M2M 通信成为物联网中一种主要的通信方式。但是,传统 IP 网络中的 TCP 协议毕竟是针对有线网络而设计的,当其直接应用到移动环境中时遇到了大量问题,并成为阻碍 M2M 通信 IP 化的主要因素之一。为了让 TCP 协议能够适应物联网复杂环境下的 M2M 通信,对原有协议革命性地推倒重来肯定是不合适的,为此只能在原有协议上进行打

补丁式的修改。改良意味着吸收借鉴与摒弃,是在可用性、可靠性、可管理性等诸多性能之间的取舍与博弈。本文的主要目的不是一一列举各种算法的实现原理,而是有针对性地提出问题、分析问题,并通过典型解决方案的讨论来开拓思路,为尚处于起步阶段的 M2M 通信研究提供更广阔的思维空间和更多可供借鉴的方法。基于 IP 网络的 M2M 通信在拥有许多应用优势的同时,也存在大量现存和潜在的问题需要解决,这也是笔者需要进一步研究的内容和方向。

### 参 考 文 献

[1] 王群,钱焕延.一种面向物联网的 M2M 通信解决方案[J]. 微电子学与计算机,2012,29(11):13-17

[2] Mukaidani H, Cai L, Shen X M. Stable Queue Management for Supporting TCP Flows over Wireless Networks[C]// Proceedings of 2011 IEEE International Conference on Communications(ICC). New York: IEEE Press, June 2011: 1-6

[3] Chaskar H, Lakshman T V, Madhow U. TCP Over Wireless with Link Level Error Control: Analysis and Design Methodology[J]. IEEE/ACM Transactions on Networking, 1996, 7(5): 605-615

[4] Padhye J, Firoiu V, Towsley D F, et al. Modeling TCP Reno performance: a simple model and its empirical validation [J]. IEEE/ACM Transactions on Networking, 2000, 8(2): 133-145

[5] Floyd S. TCP and Explicit Congestion Notification[J]. ACM Computer Communication Review, 1994, 24(5): 10-23

[6] Zhai H Q, Chen X, Fang Y G. Rate-Based Transport Control for Mobile Ad Hoc Networks[C]// Proceedings of IEEE WCNC'05. New Orleans: IEEE Computer Society, 2005: 2264-2269

[7] Lien Y N, Hsiao H C. A New TCP Congestion Control Mechanism over Wireless Ad Hoc Networks by Router-Assisted Approach[C]// Proceedings of the 27th International Conference on Distributed Computing Systems Workshops(ICDCSW'07). June 2007: 84-84

[8] Gergö B, Adam G, Thomas Z, et al. Explicit Loss Notification to Improve TCP Performance over Wireless Networks[C]// Proceedings of High Speed Networks and Multimedia Communications(HSNMC). 2003: 481-492

[9] 冯颜君,孙利民,钱华林,等. MANET 中 TCP 改进研究综述[J]. 软件学报, 2005, 18(3): 434-444

[10] Al-Jubari A M, Othman M, Ali B M, et al. An Adaptive Delayed Acknowledgment Strategy to Improve TCP Performance in Multi-hop Wireless Networks[J]. Wireless Personal Communications, 2013, 69(1): 307-333

[11] 陈立家,江昊,吴静,等. 车用自组织网络传输控制研究[J]. 软件学报, 2007, 18(6): 1477-1470

[12] Jung W S, Son S W, Rhee B H. A study of enhanced TCP for vertical Handover using explicit congestion notification (ECN) [C]// Proceedings of the 2011 Third International Conference

on Ubiquitous and Future Networks(ICUFN). Dalian, China, June 2011: 305-308

[13] Dyer T D, Boppana R V. A comparison of TCP performance over three routing protocols for mobile ad hoc networks[C]// Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc'01). ACM New York, 2001: 56-66

[14] Wang F, Zhang Y G. Improving TCP performance over mobile Ad-hoc networks with out-of-order detection and response[C]// Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc'02). ACM New York, 2002: 217-225

[15] Ghanem T F, Elkilani W S, Hadhoud M M. Improving TCP performance over Mobile Ad Hoc Networks using an adaptive backoff response approach[C]// Proceedings of the International Conference on Networking and Media Convergence (ICNM 2009). Cairo, March 2009: 16-21

[16] Chen K, Nahrstedt K, Vaidya N. The Utility of Explicit Rate-Based Flow Control in Mobile Ad Hoc Networks[C]// Proceedings of the 2004 IEEE Wireless Communications and Networking Conference(WCNC 2004). March 2004: 1921-1926

[17] Liu W R, Wu M, Peng J, et al. A Stable Congestion Control Mechanism in Ad Hoc Network[C]// Proceedings of the Second International Conference on Information and Computing Science (ICIC'09). May 2009: 225-228

[18] Brown K, Singh S. M-TCP: TCP for Mobile Cellular Networks [J]. Computer Communication Review, 1997, 27(5): 19-43

[19] Chandran K, Raghunathan S, Venkatesan S, et al. A feedback-based scheme for improving TCP performance in ad hoc wireless networks[J]. IEEE Personal Communications, 2001, 8(1): 34-39

[20] Kim D, Toh C K, Choi Y. TCP-BuS: Improving TCP Performance in Wireless Ad Hoc Networks[J]. Journal of Communications and Networks, 2001, 3(2): 1-12

[21] Balakrishnan H, Seshan S, Katz R H. Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks [J]. ACM Wireless Networks, 1995, 1(4): 469-481

[22] Bakshi B S, Krishna P, Vaidya N H, et al. Improving performance of TCP over wireless networks[C]// Proceedings of the 17th International Conference on Distributed Computing Systems. Baltimore, MD, May 1997: 365-373

[23] Holland G, Vaidya N. Analysis of TCP performance over mobile ad hoc networks[J]. Wireless Networks, 2002, 8(2): 275-288

[24] Gunes M, Vlahovic D. The performance of the TCP/RCWE enhancement for ad-hoc networks[C]// Proceedings of the 7th International Symposium on Computers and Communications (ISCC 2002). July 2002: 43-48

[25] Liu J, Singh S. ATCP: TCP for mobile ad hoc networks[J]. IEEE Journal on Selected Areas in Communications, 2001, 19(7): 1300-1315

(上接第 57 页)

[8] 魏旻,王平,金基天.一种适用于 WIA-PA 网络的分层入侵检测方法[J]. 仪器仪表学报, 2012, 33(7): 1453-1459

[9] Burri N, von Rickenbach P, Wattenhofer R. Dozer: ultra-low power data gathering in sensor networks[C]// IPSN'07. ACM Press, New York, USA: Apr. 2007

[10] Meier A, Weise M, Beutel J, et al. NoSE: efficient initialization of wireless sensor networks[C]// SenSys 2008. 2008: 397-398

[11] Ahn G-S, Miluzzo E, Campbell A T, et al. Funneling-MAC: A Localized, Sink-Oriented MAC For Boosting Fidelity in Sensor Networks[C]// Proc. of Fourth ACM Conference on Embedded Networked Sensor Systems (SenSys 2006). Boulder, Colorado, USA, 2006: 1-3