# 面向下一代网络的端到端多路径传输层架构

薛淼，高德云，张思东，张宏科
(北京交通大学 下一代互联网互联设备国家工程实验室，北京 100044)

**摘 要**：为了解决传统网络无法有效同时使用多家乡终端的多个接口传输数据的问题，提出了一种面向下一代网络的端到端多路径传输层架构—E2EMP。E2EMP 通过自适应的根据路径特性分发数据，采用双层序列空间，实施灵活的端到端路径管理，提高了多家乡终端的传输性能。实验仿真表明，E2EMP 能够有效地聚合终端多家乡的出口带宽，同时提高了数据传输的安全性和可靠性。

**关键词**：端到端多路径传输；乱序；架构；下一代网络
**中图分类号**：TP393 **文献标识码**：A **文章编号**：1000-436X(2010)10-0026-10

# End-to-end multipath transport layer architecture oriented the next generation network

XUE Miao, GAO De-yun, ZHANG Si-dong, ZHANG Hong-ke
(National Engineering Lab for Next Generation Internet Interconnection Devices, Beijing Jiaotong University, Beijing 100044, China)

**Abstract:** To solve the problem of inefficient transmission using multiple interfaces of multihome terminal in the tradition network, an end-to-end multipath transport layer architecture—E2EMP oriented the next generation network was presented. Through distributing data adaptively following characters of the end-to-end paths, adopting dual sequence space, implementing smart path management policies, the performance of the multihome terminal using E2EMP has significant improvement. The simulation results show that E2EMP aggregates bandwidth of the multihome terminal interfaces efficiently, and meanwhile promotes the security and reliability of end-to-end multipath transport.

**Key words:** end-to-end multipath transport; reordering; architecture; next generation network

## 1 引言

在过去的 30 年里，Internet 秉承其开放性的设计理念，通过承载分布式应用和互联异构网络，取得了公认的巨大成功，并成为推动经济发展和社会进步的主要引擎之一。Internet 是一个基于分组交换的网络，采用端到端的原则在网络终端间传输数据。由于设计之初价格因素和设计理念的影响，终端通常只配置一块网络接口卡（NIC），并且相应只配置一个网络层地址，端到端的数据传输在此基础上也只能建立单一的传输层连接传输数据。

随着网络技术的发展，一方面网络接入技术不断多样化，包括广域网、局域网和个域网接入技术，有线和无线接入技术都不断成熟并且商用，例如

**收稿日期**：2010-05-31；**修回日期**：2010-09-26
**基金项目**：国家重点基础研究发展计划（“973”计划）基金资助项目（2007CB307100）；国家科技支撑计划基金资助项目（2008BAH37B03）；国家自然科学基金重点基金资助项目（60833002）；国家自然科学基金资助项目（60903150）；中央高校基本科研业务费专项基金资助项目（2009YJS017）
**Foundation Items:** The National Basic Research Program of China (973 Program) (2007CB307100); The National Key Technology R&D Program (2008BAH37B03); State Key Program of National Natural Science of China (60833002); The National Natural Science of China (60903150); Fundamental Research Funds for the Central Universities (2009YJS017)

xDSL、xPON、3G、WLAN、WiMAX、Bluetooth 等；另一方面，接入设备的成本不断降低，越来越多的终端开始配置多个网络适配器，例如移动终端 laptop 通常配有 LAN 接口和 WLAN 接口，也可以支持 3G 无线接入；而一般 PDA 都支持 WLAN 和 3G 接入；一些固定终端也同时支持 LAN 和 ADSL 接入。各种集成了 USB 接口的网络适配器也使得终端在选择接入方式上变得更加灵活。

下一代网络结构的演化，使得终端能够配置多个网络层接入地址。IPv6 的初步部署使得网络层地址更为丰富,终端可以配置多个 IPv6 地址或者同时配置 IPv6 地址和 IPv4 地址，以接入不同子网；而网络虚拟化[1,2]可以使由不同种类的网络层地址、不同的路由机制构成的虚拟网络部署在相同的基础设施上，终端可以同时选择 IP 或者其他虚拟网络层地址，例如平面标识，接入网络。

尽管基于多种接入技术和多个网络地址的多家乡终端正在成为下一代互联网的主要特征，但是传统的端到端传输协议，例如 TCP、UDP 仍然只能基于一个 NIC 的网络地址建立连接，在端到端单路径上传输数据，无法充分发挥多个 NIC 的特性。

由于现有协议的不足，端到端多路径的研究逐渐成为关注的焦点。欧洲的 Trilogy[3]计划将端到端的多路径传输作为其资源池(resource pooling)概念的一部分；文献[4~6]中描述的一体化网络架构提出的多连接多路径[7](MCMP, multi-connection multi-path)也将端到端多路径作为新网络体系研究的一个方面。

为了充分使用终端的多家乡特性进行数据传输，适应下一代网络终端多家乡化的趋势，提出了一种基于传输层的端到端多路径架构(E2EMP, end-to-end multipath)，在传输协议 SCTP(stream control transmission protocol)[8]上实现了 E2EMP，并对其进行性能评估。本文第 2 节介绍了与 E2EMP 相关的工作；第 3 节描述端到端多路径的优点和 E2EMP 的设计目标；第 4 节描述了 E2EMP 的架构设计；第 5 节对其性能进行评估；第 6 节是结束语。

## 2　相关工作

近年来，很多研究尝试在各个层面使用多家乡终端的多个接口并行传输数据。本文根据其实现的层次将这些研究分为 3 类。

### 2.1　应用层实现

MuniSocket[9]是一个在用户空间实现的 socket 中间件，它通过在多个网络接口上建立多个 TCP 连接分发数据，从而达到聚合带宽和提高可靠性的目的。但其实现需要手动配置将要建立的 TCP 连接地址对和端口号，可扩展性不强。SmartSockets[10]能够为多家乡主机智能的选择建立连接的地址和端口，但其建立的连接是基于端到端单路径进行数据传输的,需要在内存中开辟新的存储空间用于缓存不同路径到来的乱序数据分组，因此内存的利用率不高。

### 2.2　传输层实现

通过传输层实现端到端多路径传输主要包括基于 TCP 的实现和基于 SCTP 的实现。

pTCP[11]通过在传输层为多个接口上的 TCP “管道”分发数据，实现带宽聚合。M/TCP[12]在内核中实现在多个接口上建立 TCP 连接分发数据，并利用新的 TCP 选项管理不同接口 TCP 的连通性，健壮的 ACK 机制保证其可靠性。mTCP[13]的实现与 M/TCP 类似，但其加入了共享拥塞检测机制。上述这些实现均对路径差异性考虑不足，仅仅依赖于 TCP 的滑动窗口机制，忽视了数据发送调度算法的重要性，无法适应路径差异大的网络环境。R-MTP[14]通过带宽估计调度数据分组在多个接口的分发比例，达到带宽聚合的目的，但其发送速率易受带宽估计的影响。MPTCP[15]是最近一个活跃的关于端到端多路径分支，但是 MPTCP 对 TCP 的分组格式做了大量修改，尤其是多个选项的添加使其难以穿越现有网络的防火墙设备。

由于 SCTP 自身对多家乡的支持，很多端到端多路径设计方案也采用 SCTP。LS-SCTP[16]通过修改 SCTP 发送机制，使得 SCTP 能够同时在多条路径并行发送数据，达到负载均衡的目的。cmpSCTP[17]引入了路径序列号和修改的 SACK，完善了 LS-SCTP 的设计。但二者都对已有分组格式做了大量修改，后向兼容性不强。W-SCTP-PR[18]通过动态估计路径带宽，基于带宽在各路径上发送数据，实现了 PR-SCTP 并行多路径传输，但其在与 TCP 共存时带宽容易被挤占。CMT-SCTP[19]在尽量不改变标准 SCTP 分组格式的基础上修改了 SCTP 基于路径的拥塞控制算法和多种重传算法，实现 SCTP 在多个接口的并行传输，但其只是简单在路径间进行 Round-Robin 调度发送，没有考

虑路径间的特性差异。

### 2.3 链路层实现

为了聚合多个网络接口的带宽，Linux 的 bonding[20]技术可以将多个网络接口绑定成一个虚拟接口，用户数据在各接口间按照一定的算法调度，从而实现负载均衡和带宽聚合。在 Solaris8 和 Solaris 9 中引入的 IPMP[21]实现了在 SUN 操作系统的多接口的带宽聚合和并行数据传输。但 bonding 和 IPMP 不能精确地得到基于端到端路径的时延和网络可用带宽等参数，只能根据接口默认的性能参数设置数据调度算法，容易造成传输层 RTT 估计不准，产生不必要的快速重传等问题，使得传输性能下降。

## 3 E2EMP 设计目标

### 3.1 端到端多路径优势描述

使用端到端多路径进行数据传输能够带来如下几个优点。

带宽聚合：端到端多路径可以有效的聚合多条路径的带宽，从而为用户提供更好的 QoS 保障。

可靠性：由于同时存在多条端到端的路径，单条路径失败不会影响服务的连续性，从而为端节点提供网络层冗余，保证了传输的可靠性。

负载均衡：多条端到端路径同时使用，可以根据网络中的拥塞状况动态地调整在不同路径的发送速率，从而实现在网络边缘处的负载均衡。

安全性：所有的应用数据都从多条路径传输，对任一单条端到端路径的监听嗅探无法有效地恢复初始数据内容，具有更好的安全性。

### 3.2 E2EMP 设计目标

针对上述端到端多路径传输的优点，对在 E2EMP 的设计中需要解决的问题进行描述，同时提出 E2EMP 的设计目标。

1) 同时使用终端所有活动接口带宽。

这个目标是 E2EMP 设计的初衷，即最大限度利用终端接口汇聚带宽。

2) 减少乱序。

端到端多路径传输的数据经历不同的路径环境，到达接收端的顺序可能与其进入网络的顺序不一致，即发生数据分组乱序。而乱序会引起传输协议接收缓冲区阻塞，数据递交延时增大，引起不必要的重传等，降低了端到端多路径传输的性能。因此根据路径特性进行数据分组分发调度，降低乱序是 E2EMP 要解决的主要问题之一。

3) 减少不必要的快速重传。

现有拥塞控制 RFC 5681 将 3 个重复 ACK 认为路径分组丢失，进而将拥塞窗口减半进入快速重传阶段。而在端到端多路径传输过程中，数据分组的乱序到达现象会大量出现，不必要的快速重传会严重影响端到端多路径的传输性能。E2EMP 需要降低识别不必要的快速重传以保证多路径传输的性能。

4) 路径管理与路由设置。

多家乡主机能够依据其网络层地址建立多个端到端路径，而主机能够准确地对每条端到端路径识别、操作和调度。因此 E2EMP 需要能够灵活准确地对端到端路径进行管理。

5) 分组丢失响应。

由于采用多条路径并行传输数据，不同的路径具有不同的分组丢失率。RFC5681 对分组丢失的响应就是降低路径的发送速率。E2EMP 需要确保一条路径的分组丢失行为不能影响未分组丢失路径的使用带宽，同时也要保证尽快重传丢失数据防止接收端缓冲区阻塞[22]。

## 4 E2EMP 架构设计

### 4.1 E2EMP 层次结构

E2EMP 通过对传统网络的传输层进行重新架构，改变现有端到端单路径的传输方式，充分利用多个网络接口带来的优势。E2EMP 将传输层功能分成 3 个子层，如图 1 所示。其中第 1 个子层是基于

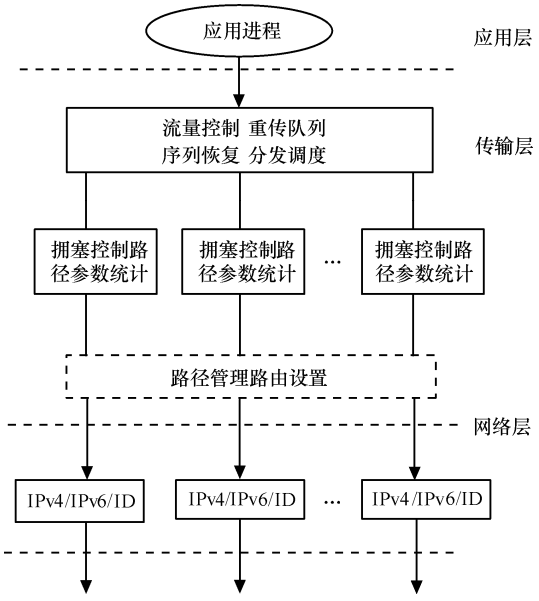



图 1 E2EMP 架构设计

整个连接的，与应用程序中创建的一个 socket 接口对应。在该子层中，除了实现面向应用的数据重新排序、错误恢复等功能，还要维持面向连接的重传队列维护、路径权值计算、数据分发调度等功能。

在 E2EMP 架构中，既然每条端到端路径都经历独自的网络环境，因此每条端到端路径都有各自独立的拥塞控制功能；而传输的所有数据都是来自同一个应用进程，因此采用共同的接收缓冲区进行流量控制。E2EMP 第 2 个子层是面向端到端单路径的，提供端到端的拥塞控制和基于端到端路径的参数统计，如单路时延、带宽估计、分组丢失率计算等。E2EMP 的最底层实现端到端的路径选择和动态主机路由配置。

### 4.2　路径管理

E2EMP 架构在建立端到端多路径传输初始阶段，发送端会得到一个目的端的网络层地址，这个地址可以从 DNS 查询得到或者人为输入。发送端向这个目的地址发起连接，并通告本地可以使用的网络层地址，而接收端回应并携带接收端的网络层地址。当双方彼此了解对方的地址信息后开始按照相应规则建立端到端路径。本文提供了一种路径建立策略应用于 E2EMP 中，路径管理算法见附录。

在本文的算法中，端到端的路径建立并不要求穷尽所有地址对的组合，而是尽可能地利用多个接口所提供的带宽。同时算法中并不要求前向路径与后向路径保证对称性。在路径管理中引入了一个标识端到端路径的变量——连接标识（CID, connection identifier）。连接标识是一个本地变量，不会在网络中出现，用于取代四元组<源地址，源端口号，目的地址，目的端口号>表示端到端路径的方法，从而对端到端路径进行灵活的管理。每个数据分组的连接序列号都与唯一一个连接标识对应，进而与连接标识所标势路径的路径序列号对应，关于连接序列号和路径序列号的概念在下一节将会介绍。图 2 给出了基于连接标识的路径管理。

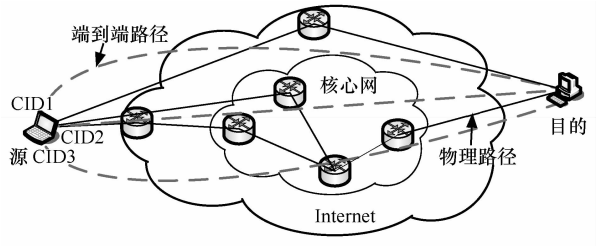



图 2　基于连接标识的路径管理

E2EMP 中的路径在端到端层面是完全不重叠的，但在网络路由层面可能会有一跳或多跳路由重叠。假定核心网络带宽是足够的，在核心网络的路由重叠不会带来性能下降的问题。

### 4.3　缓解乱序影响

端到端多路径并行传输经历的网络时延、分组丢失等环境参数都不同，到达接收端时会引起大量的数据分组乱序现象。RFC 5681 中描述的基于分组丢失的拥塞控制机制将乱序作为拥塞分组丢失的一种指示，因此如果有 3 个乱序的数据分组到达接收端，接收端会通过 3 个重复 ACK 触发发送端的快速重传，进而造成拥塞窗口下降，可用带宽降低。如图 3 描述了端到端多路径的不必要的快速重传。这种不必要的快速重传严重影响端到端多路径的性能。E2EMP 通过 2 种方法缓解上述影响：①设计发送端的数据分发调度算法，使在不同路径传输的数据虽然经历不同路径环境，但仍然尽量按序到达接收端；②采用双层序列空间，优化现有拥塞控制算法，消除对乱序的过度反应。

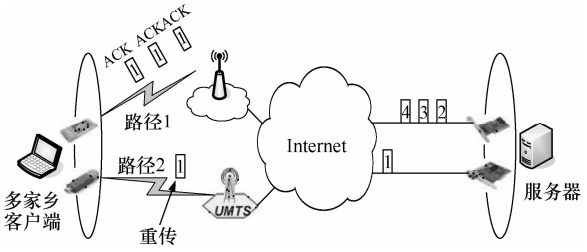



图 3　乱序引发多路径快速重传

#### 4.3.1　数据分发调度算法

为了减少乱序，提出了一种数据分发调度算法。算法根据路径的特性动态生成一个路径权值(PW, path weight)，每一轮发送调度算法根据权值比例在每条路径上调度数据。由于 E2EMP 是一个端到端多路径架构，每一条端到端路径的都可以看作一跳单跳虚拟路由，数据的分发调度依赖于虚拟路由的度量值 PW。定义 PW 为

$$PW_i(t)=\frac{\alpha}{Bandwith_i(t)}+\beta RTT_i(t)+\gamma Loss_i(t) \quad (1)$$

其中，$Bandwith_i(t)$ 为 $t$ 时刻路径 $i$ 的估计可用带宽，$RTT_i(t)$ 为 $t$ 时刻路径 $i$ 测得的平均往返时延，$Loss_i(t)$ 为到 $t$ 时刻路径 $i$ 统计的重传比率，这里重传包括快速重传和超时重传。乘子 $\alpha$，$\beta$，$\gamma$ 分别

表示带宽，时延，分组丢失在路径权值计算中所占的比例。这 3 个参数可以根据用户需求进行配置和修改。可以看到 PW 值越小则路径状态越好。每条路径的实时带宽可以通过动态估计获得。记录 $t$ 时刻带宽估计样本为 $s_i(t)$。

$$s_i(t)=\frac{M_i(t)}{\max(MRTT_i(t),40\text{ms})} \tag{2}$$

其中，$M_i(t)$ 为 $t-1$ 时刻到 $t$ 时刻发送的数据量，$MRTT_i(t)$ 为路径 $i$ 在 $t$ 时刻测量的即时往返时延样本。将样本值经过低通滤波进行平滑即得到路径的带宽估计值 $Bandwith_i(t)$。

$$Bandwith_i(t)=\lambda Bandwith_i(t-1)+(1-\lambda)s_i(t) \tag{3}$$

其中，$\lambda$ 为滤波因子。

每条端到端路径单独维持变量 $P^i_{\text{send}}(t)$、$P^i_{\text{Tresend}}(t)$ 和 $P^i_{\text{Fresend}}(t)$，用于记录路径 $i$ 到 $t$ 时刻共发送数据分组数目、超时重传数据分组数目和快速重传数据分组数目。可得 $Loss_i(t)$：

$$Loss_i(t)=\frac{1\,000\,000(P^i_{\text{Fresend}}+100P^i_{\text{Tresend}})}{P^i_{\text{send}}} \tag{4}$$

连接起始阶段，可以设置 $Bandwith_i(0)$ 为 1 000 000，$RTT_i(0)$ 为 1，$Loss_i(0)$ 为 0。

当端到端传输进入发送状态后，计算各路径的权值比例，然后根据该比例进行数据调度。E2EMP 架构保留每条路径的基于滑动窗口的拥塞控制算法，以维持与现有传输协议的公平性。权值越小的路径将优先发送数据直到拥塞窗口全部占满，然后是权值次小的路径，依次直到所有路径遍历一次，或者所有待发送数据全部发送出去。依据路径权重的数据分组调度算法见附录。

#### 4.3.2 双层序列空间

由于现有传输协议设计假定一个连接的数据只经过一个端到端路径传输，其采用单一序列空间标识所传输的数据，例如 TCP 的字节号和 SCTP 的 TSN（transmit sequence number）。在端到端单路径传输时，可以用该单一序列空间判断数据是否乱序，以进行顺序调整进而递交给应用层，又可以通过该序列空间中某些序号的缺失或混乱对路径拥塞情况作出判断进而调整发送端拥塞窗口。

在端到端多路径并行传输情况下，通过单一序列空间标识数据变得不再适用，如图 3 所示。路径 2 的数据分组 1 由于延迟大到达接收端较晚，而由于路径 1 的时延小数据分组 2，3，4 已经先于数据分组 1 到达，产生 3 个重复的 ACK，使得发送端错误的认为路径 2 上数据分组 1 丢失，从而重传数据分组 1，并将路径 1 的拥塞窗口减半。对于一条端到端路径拥塞状况的判断应该基于该条端到端路径的反馈，而不能受到其他端到端路径的影响，由于缺乏路径信息，现有单一序列空间标识数据的方法无法做到这一点。

在 E2EMP 架构中，采用双层序列空间标识数据流，路径序列空间和连接序列空间。路径序列空间按照升序标识端到端路径上发送数据分组的数目，连接序列空间按照升序标识整个连接发送的应用数据分组或者字节。为了与现有端到端单路径传输协议兼容，并不将路径序列封装到数据分组中，以防止其经过网络中间设备，如被防火墙，被过滤掉。E2EMP 架构中在发送每一个数据分组时会记录其发送的端到端路径，并将其连接序列号与该端到端路径的路径序列号进行一对一映射。而每个端到端路径根据该映射关系记录判断返回的 SACK 中是否确认了该路径的数据和需要重传数据。如图 4 所示，序号 3,4,5 为路径 1 的路径序列号，序号 9,10,11 为路径 2 的路径序列号，而序号 11~15 则是连接序列号。

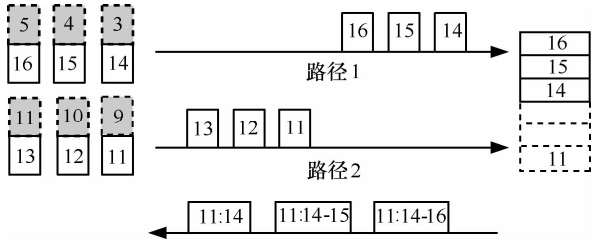



图 4　E2EMP 双层序列空间

### 4.4 拥塞控制与流量控制

由于每条端到端路径经过的网络环境存在差异，例如延时带宽参数不尽相同，E2EMP 架构基于每条端到端路径实施拥塞控制。拥塞算法采用 RFC5681 建议的算法，也是当前 TCP，SCTP 等传输协议采用的拥塞算法。

在 E2EMP 架构中，每条端到端路径维持一个路径序列号，用于判断本路径的分组丢失情况，进而采取拥塞调整。同时端到端路径基于 SACK 分组，

统计路径参数如 4.3.1 节中 $Bandwith_i(t)$，$RTT_i(t)$，$Loss_i(t)$。发送端接收到 SACK 的处理算法见附录。

在 E2EMP 架构中，与 cmpSCTP，MPTCP 的设计不同，路径序列号不在数据分组中携带，因此接收端的 SACK 生成机制无需修改。但由于乱序的大量出现，SACK 机制无需发现连接序列出现 gap 就立即发送 SACK 包。E2EMP 架构中设置连续收到 2 个连接序列号发送 SACK 包，或者延迟 200ms 发送 SACK 包，而不管数据分组是否乱序。

SACK 信息是基于分组丢失的拥塞控制算法检测拥塞分组丢失的唯一信息。因此 E2EMP 架构将所有 SACK 包从 PW 最小的路径发送，以使得路径拥塞控制算法能迅速对路径拥塞作出响应。

## 5　仿真验证

### 5.1　仿真方法

为了验证 E2EMP 的性能，将其设计部署在传输层协议 SCTP 上。采用 SCTP 主要因为：①SCTP 自身支持多家乡的操作，易于实现 E2EMP 的路径管理；②无需为每条端到端路径单独建立连接，降低初始传输时延；③无需为管理多条端到端连接添加额外的分组格式或者选项；④扩展性好，并且能够后向兼容。需要强调的是，E2EMP 是一个传输层架构，其结构层次和设计算法的实现并不仅限于 SCTP，同样可以部署在其他传输层协议上。

本文采用 NS2 作为仿真平台，对 E2EMP 的性能进行仿真分析。实验拓扑如图 5 所示，这里简单采用 2 条端到端路径的情况，边缘链路表示网络中最后一跳，核心链路表示网络中端到端路径状况，没有引入路由影响产生的乱序情况。拓扑中接入带宽为 100Mbit/s，链路时延为 5ms，瓶颈链路 R1-R2、R3-R4 的参数设置将根据实验设置调整，瓶颈链路队列为 50 个分组。其中 S3 和 D3 分别为部署了 E2EMP 的 SCTP 发送端和接收端，S1 和 S2 为 TCP Reno 的发送端，D1 和 D2 分别为对应的接收端，其中接收端接收缓冲区为 65 535byte。为了简化本文将经过 R1-R2 链路的端到端路径称为路径 1，经过 R3-R4 链路的端到端路径称为路径 2，之后涉及到路径 1 和路径 2 的时延、带宽、分组丢失等参数均指在 R1-R2、R3-R4 的单向参数。

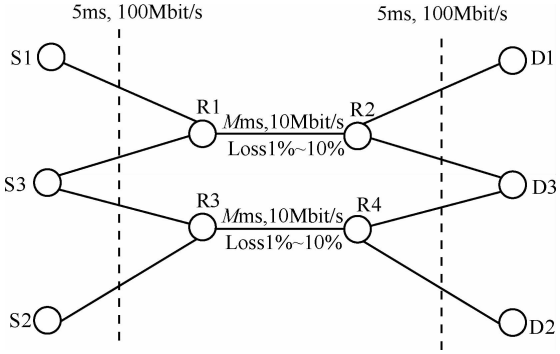


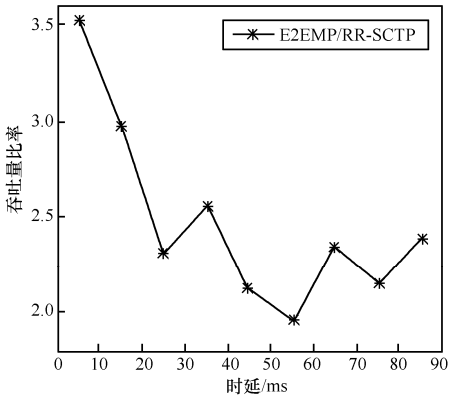

图 5　仿真拓扑

### 5.2　数据分发调度算法仿真与分析

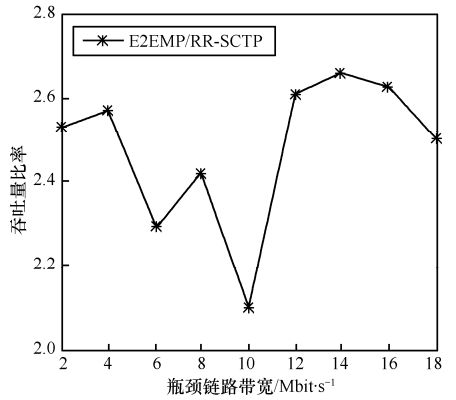
当路径 1 和路径 2 的特性参数差异较大时，端到端多路径传输性能会受到很大影响。分别对路径的带宽、时延、分组丢失存在差异时，对 E2EMP 和仅采用 Round-Robin 算法在路径间分发数据的 SCTP(RR-SCTP)的吞吐量进行比较。图 6(a)、图 6(b)和图 6(c)分别对应 2 条路径带宽、时延、分组丢失有差异的情况下 E2EMP 与 RR-SCTP 的吞吐量比值。其中，设定图 6(a)的仿真参数为路径 2 带宽 10Mbit/s，时延 55ms，分组丢失率 1%，路径 1 带宽 10Mbit/s，时延可变，分组丢失率 1%；设定图 6(b)的仿真参数为路径 2 带宽 10Mbit/s，时延 55ms，分组丢失率 1%，路径 1 带宽可变，时延 55ms，分组丢失率 1%；设定图 6(c)的仿真参数为路径 2 带宽 10Mbit/s，时延 55ms，分组丢失率 1%，路径 1 带宽 10Mbit/s，时延 55ms，分组丢失率可变。

如图 6(a)所示，路径 1 的时延偏离 55ms 越大，则 E2EMP 与 RR-SCTP 的吞吐量比值越大。当路径 1 的时延值小于 55ms 时，由于路径时延较小，相比于路径 1 时延大于 55ms 的情况，E2EMP 与 RR-SCTP 的吞吐量比的峰值更大。当路径 1 的时延在 55ms 附近时，E2EMP 的 2 条路径权值近似，因此发送调度类似于 Round-Robin 调度。此时 E2EMP 和 RR-SCTP 吞吐量比值最小，也说明 RR-SCTP 在 2 条路径差异不大的情况下可以达到最大吞吐量。

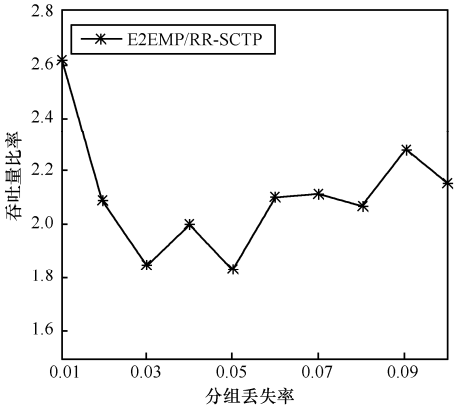
类似于图 6(a)，从图 6(b)和图 6(c)中可以看出，当 2 条路径特性差异较小时，基于 RR-SCTP 的端到端多路径可以达到较大的吞吐量，而此时与 E2EMP 的吞吐量比值最小。而一旦路径差异变大，则 E2EMP 的吞吐量远远超过 RR-SCTP。这是因为 Round-Robin 调度不考虑路径特性，向 2 条路径发送相同数量的数据分组，产生了大量的乱序，进而



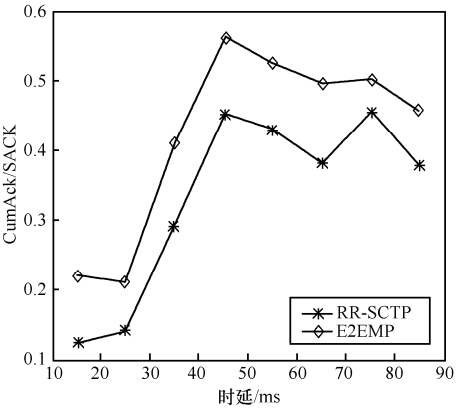

(a) 路径时延差异下吞吐量比率




(b) 路径带宽差异下吞吐量比率




(c) 路径分组丢失差异下吞吐量比率




(d) 路径时延差异下的乱序比较

图 6　路径参数差异对吞吐量影响

使 2 条路径的拥塞窗口频繁地触发快速重传而始终维持在一个较低的值，限制了基于 RR-SCTP 端到端多路径传输的吞吐量。而 E2EMP 可以根据动态生成的路径权重优先选择好的路径发送，甚至在路径差异大的情况下只在优选路径发送数据，降低了乱序的发生，提高了端到端多路径的吞吐量。

由于乱序的数据分组到达接收端会触发一个 SACK 数据分组，但其累积确认值(cumulative acknowledgement)却没有得到增长，因此接收端收到 Cum-SACK 的数量与总的 sack 的数量比值在一定程度上可以判断发生乱序的情况。图 6(d)为在图 6(a)对应仿真参数下得到的 CumAck/SACK 比例情况，可以看到 E2EMP 的基于路径权重的数据分发算法能够有效降低乱序的发生，进而验证了上面的分析。

### 5.3　双层序列空间算法仿真与分析

采用双层序列空间，可以将端到端数据传输的可靠性和拥塞控制解耦。连接序列空间仅用来纠正数据分组的乱序从而达到按序递交，而路径序列空间仅用来判断路径拥塞状况，进行相应的拥塞状态调整。设定仿真参数为路径 2 带宽 10Mbit/s，时延 55ms，分组丢失率 1%；路径 1 带宽 10Mbit/s，时延可变，分组丢失率 1%，仿真时间 70s。

如图 7 所示，当路径 1 的时延在 55ms 附近时，RR-SCTP 发生的快速重传处于较低值，此时，路径状况较接近，乱序情况相对较少。随着路径 1 的时延逐渐偏离 55ms，乱序发生情况增多，采用单一序列空间的 RR-SCTP 的快速重传次数不断上升，而 E2EMP 的快速重传次数始终维持在稳定的波动范围区间，说明采用双层序列空间能够有效减少不必要的快速重传。

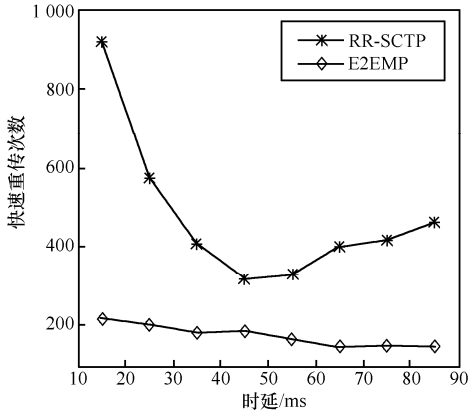
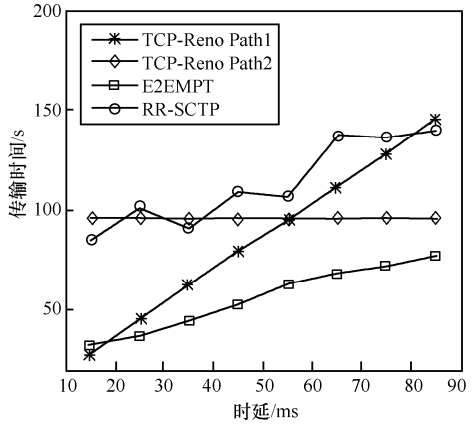


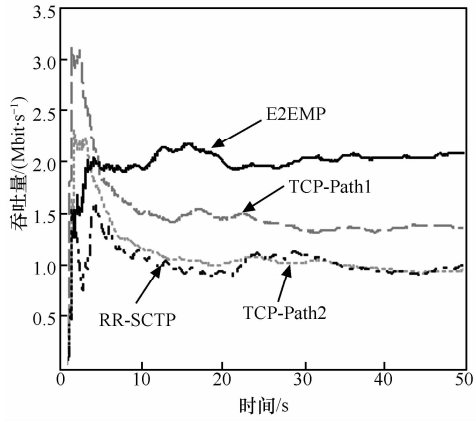

图 7　基于双层序列空间的拥塞控制

### 5.4 E2EMP 性能测量

本节对 E2EMP 与 RR-SCTP、TCP Reno 进行性能比较。如图 8(a)所示，显示传输 15MB 大小的文件时三者所用的时间比较。设定仿真参数为路径 2 带宽 10Mbit/s，时延 55ms，分组丢失率 2%；路径 1 带宽 10Mbit/s，时延可变，分组丢失率 1%。其中 TCP Reno 分别在路径 1 上和延时为 55ms 的路径 2 上传输。从图 8(a)中可以看到 RR-SCTP 传输时间总是接近于最差路径的传输时间，即在路径 1 时延小于 55ms 时，传输时间接近较差路径路径 2 采用 TCP 时的传输时间，而在路径 1 时延大于 55ms 时，由于路径 1 状况变差，因此 RR-SCTP 的传输时间又近似于路径 1 采用 TCP 时的传输时间。而 E2EMP 的传输时间远远小于 RR-SCTP，甚至采用最佳路径的 TCP 的传输时间。图 8(b)截取图 8(a)在路径 1 时延为 35ms，路经 2 时延为 55ms 时的剖面稳态吞吐量，可以看到 E2EMP 的性能要明显好于基于 RR-SCTP 的多路径传输，也好于其他端到端单路径传输。




(a) 传输时间比较




(b) 35ms 处稳态吞吐量比较

图 8　传输相同大小文件所需时间比较

## 6　结束语

本文分析了现有网络接入不断多样化背景下的数据传输特征，提出了一种面向下一代网络的端到端多路径传输架构 E2EMP，该传输架构可以根据路径权重，动态地在路径间调度分发数据分组，从而有效地降低了多路径传输情况下的数据分组乱序现象；双层序列空间的采用可以有效降低乱序引发的不必要的快速重传，提高了多条路径并行使用的吞吐量，达到了带宽聚合的目的；基于连接标识的路径管理可以灵活的调整切换路径，减少路径失败带来的性能下降。通过将 E2EMP 部署到 SCTP 上进行仿真验证，实验结果表明，即使在路径参数差异较大的情况下，E2EMP 仍然能够达到较好的性能。

E2EMP 架构是一个通用的传输层架构，可以在现有 TCP、SCTP 上修改实现其设计，也可以作为下一代网络传输层协议设计的参考。此外由于 E2EMP 架构不对任何数据分组格式进行修改，易于后向兼容，可以很好地满足传统网络和下一代网络对端到端多路径数据传输的需求。

## 附录：算法 1、算法 2 和算法 3

**算法 1**：路径管理算法

```
Connection initialization
[Sender side behavior]
if ( one of the destination addresses has been known) then
    send request for establish connection
        with local network layer addresses list;
[receiver side behavior]
On receipt of packet for request connection establishment
    send back the local network layer addresses
        list along the backward path;
    process the connection establishment request;
Path management [both sides beheivor]
On receipt the addresses list of the peer
  for each address add_local_i in the local address list do
    for each address add_peer_j in the peer address list do
      if (add_local_i and add_peer_j are not the
          same type address ) then
        continue;
      if (add_local_i is a private address ) then
        if (add_peer_j is a private address) then
          continue;
        else if (add_peer_j is the longest match to add_local_i
            and not in the used list )
          set the add_peer_j the destination of add_local_i;
          generate a connection identifier;
          add the add_peer_j to the used list;
          addl to the add_peer_j used times;
```

```
                    break;
                else
                    set the add_peer_j with min used times in used
                        list the destination of add_local_i;
                    generate a connection identifier;
                    addl to the add_peer_j used times;
                    break;
            end;
            create the end-to-end path management struct;
        end;
```

**算法 2**：基于路径权重的数据分组调度算法

```
Distributionalogrithm
 for each CID_i do
    update path[CID_i].PW;
 end;
 if (sent_path_list is empty) then
    path[CID_i].issend ==0;
    add path[CID_i] to the not_sent_path_list;
    clear sent_path_list;
 for each CID_i do
    if (path[CID_i].issend ==0) then
       if (path[CID_i].PW is the min in the not_sent_path_list ) then
          if (path[CID_i].cwnd-path[CID_i].flightsize>0) then
             add path[CID_i] to the sent_path_list;
             path[CID_i].issend=1;
             if (retransimssion queue is not empty) then
                for each packet in the retransimssion queue do
                   if (path[CID_i].cwnd-path[CID_i].flightsize>0) then
                      path[CID_i].havesent+=1;
                      send the packet;
                   else
                      break;
                end;
                continue;
             else if (transimssion queue is not empty ) then
                for each packet in the transimssion queue do
                   if (path[CID_i].cwnd-path[CID_i].flightsize>0) then
                      path[CID_i].havesent+=1;
                      send the packet;
                   else
                      break;
                end;
                continue;
             else
                break;
          else
             continue;
       else
          continue;
    else
       continue;
 end;
```

**算法 3**：基于路径序列的 SACK 处理

```
On receipt a DATA numbered TSN [receiver side behavior]
 if ( there is TSN-1 in the buffer and TSN arrive) then
    find the path[CID] with the min path[CID].PW;
    send back a SACK for TSN+1 with gap;
 else if (the 200ms timer is expired ) then
    find the path[CID] with the min path[CID].PW;
    send back a SACK with a gap of TSN;
On receipt a SACK[Sender side behavior]
 if (SACK carries a new cum ack or SACK carries a new gap ack) then
    for each connection TSN in new cum ack and gap ack do
       for each TSN in the retransmission queue do
          find the match CID for TSN;
          get corresponding path[CID].PSN to the TSN;
       end;
       if (path[CID].PSN == path[CID].cum+1) then
          path[CID].cum += 1;
          if (path[CID].RTTlabel ==1) then
             path[CID].RTT=RTTnow;
       else if (path[CID].PSN > path[CID].cum+1) then
          path[CID].fastretran +=1;
          if (path[CID].fastretran == 3) then
             resent packet TSN;
             path[CID].resent +=1;
       else
          continue;
 end;
```

## 参考文献：


[1] CARAPINHA J, JIMENEZ J. Network virtualization: a view from the bottom[A]. Proceedings of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures[C]. Barcelona, Spain, 2009, 73-80.

[2] PETERSON L, SHENKER S, TURNER J. Overcoming the Internet impasse through virtualization[A]. HOTNETS III[C]. San Diego, CA USA, 2004.31-41.

[3] Trilogy[EB/OL].http://www.trilogy-project.org。

[4] 董平, 秦雅娟, 张宏科. 支持普适服务的一体化网络研究[J]. 电子学报, 2007, 35(4): 599-606.
DONG P, QIN Y J, ZHANG H K. Research on universal network supporting pervasive services[J]. Acta Electronica Sinica, 2007, 35(4): 599-606.

[5] 杨冬, 周华春, 张宏科. 基于一体化网络的普适服务研究[J], 电子学报, 2007, 35(4): 607-613.
YANG D, ZHOU H C, ZHANG H K. Research on pervasive services based on universal network[J]. Acta Electronica Sinica, 2007, 35(4): 607-613.

[6] 张宏科, 苏伟. 新网络体系基础研究——一体化网络与普适服务[J]. 电子学报, 2007, 35(4): 593-598.
ZHANG H K, SU W. Fundamental research on the architecture of new network universal network and pervasive services[J]. Acta Electronica Sinica, 2007, 35(4): 593-598.

[7] 杨冬, 李世勇, 王博等. 支持普适服务的新一代网络传输层架构[J]. 计算机学报,2009,32(3): 359-70.
YANG D, LI S Y, WANG B, *et al*. New transport layer architecture for pervasive service[J]. Chinese Journal of Computer, 2009, 32(3): 359-370.

[8] STEWARD R, Ed. RFC 4960，Stream Control Transmission Protocol[S]. 2007.

[9] MOHANED N, Al-JAROODI J. Self-configured multiple-network interface socket[J]. Journal of Network and Computer Applications, 2010, 33(1): 35-42.

[10] MAASSEN J, BAL H B. Smartsockets: solving the connectivity problems in grid computing[A]. Proceedings of the 16th International

Symposium on High Performance Distributed Computing[C]. California, USA, 2007.1-10.

[11] HSIEH H Y, SIVAKUMAR R. pTCP: an end-to-end transport layer protocol for striped connections[A]. ICNP[C]. Washington, DC, USA, 2002. 24-33.

[12] KULTIDA R, HITOSHI A. An evaluation of multi-path transmission control protocol (M/TCP) with robust acknowledgement schemes[J]. IEICE Transactions on Communications, 2005, 87(9): 2699-2707.

[13] ZHANG M, LAI J, *et al*. A transport layer approach for improving end-to-end performance and robustness using redundant paths[A]. Proceedings of the Annual Conference on USENIX Annual Technical Conference[C]. Boston, MA, 2004.

[14] MAGALHAES L, KRAVETS R. Transport level mechanisms for bandwidth aggregation on mobile hosts[A]. ICNP[C]. Riverside, California, USA, 2001. 165-171.

[15] RAICIU C, HANDLY M, FORD A. Multipath TCP Design Decisions[R]. July 2009.

[16] El Al A A, SAADAWI T, LEE M. LS-SCTP: a bandwidth aggregation technique for stream control transmission protocol[J]. Computer Communications, 27(10): 1012-1024

[17] LIAO J, WANG J, ZHU X. cmpSCTP: an extension of sctp to support concurrent multi-path transfer[A]. ICC[C]. Beijing, China, 2008. 5762-5766.

[18] FIORE M, CASETTI C. An adaptive transport protocol for balanced multihoming of real-time traffic[A]. GLOBECOM[C]. St Louis, Missouri, 2005.1091-1096.

[19] IYENGAR J, AMER P, RTEWART R. Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths[J]. IEEE/ACM Transactions on Networking, 2006, 14(5): 951-964.

[20] STERLING T L, BECKER D J, *et al*. Achieving a balanced low-cost architecture for mass storage management through multiple fast ethernet channels on the beowulf parallel workstation[A]. Proceedings of the 10th International Parallel Processing Symposium[C]. 1996. 104-108.

[21] SOLARIS. IP Network Multipathing(IPMP) Administration Guide[R].

[22] IYENGAR J, AMER P, RTEWART R. Receive buffer blocking in concurrent multipath transfer[A]. GLOBECOM[C]. St Louis, Missouri, USA, 2005. 367-371.

**作者简介：**

**薛淼**（1983-），男，河北保定人，北京交通大学博士生，主要研究方向为下一代网络服务理论、端到端传输协议和拥塞控制。

**高德云**（1973-），男，江苏淮阴人，北京交通大学副教授、博士生导师，主要研究方向为无线传感器网络、无线局域网、移动互联网等。

**张思东**（1945-），男，山东寿光人，北京交通大学教授、博士生导师，主要研究方向为下一代互联网与无线传感器网络路由、资源分配与管理、网络安全等。

**张宏科**（1957-），男，山西大同人，北京交通大学教授、博士生导师，主要研究方向为下一代信息网络关键理论与技术、下一代网络服务理论、新一代移动互联网络路由、协议理论与技术等。