

# 基于乱序反馈的差异化多路径并发传输模型数据分配算法

杜文峰 吴 真

(深圳大学计算机与软件学院 深圳 518060)

**摘 要** 多路径并发传输模型在差异化路径中会出现严重的数据包乱序现象,导致接收端缓冲区阻塞,极大地降低了偶联的整体传输性能。在分析影响路径传输性能因素的基础上,给出了差异化路径性能评估模型,并提出了一种基于乱序反馈的差异化路径并发传输数据分配算法。该算法以乱序反馈来动态调节路径的发送比例,减少了传输性能差的路径对整体吞吐量的影响。分析和实现结果表明,本算法的性能比默认的轮询数据分配算法更好。

**关键词** 多路径并发传输,差异化路径,数据分配,缓冲区反馈

中图分类号 TP393 文献标识码 A DOI 10.11896/j.issn.1002-137X.2015.3.013

## Data Distribution Algorithm with Out-of-order Feedback for CMT over Diversity Network

DU Wen-feng WU Zhen

(College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China)

**Abstract** The performance of CMT degrades remarkably when a large quantity of out-of-order packets are blocked in the receive buffer due to the discrepancy of paths in terms of performance. Based on the analysis of different network configurations, a performance evaluation model was proposed. Meanwhile, a data distribution algorithm was proposed to improve the performance of CMT over diversity paths. This algorithm dynamically adjusts the transmission ratio with the out of order feedback of receiver buffer and reduces the impact of path with bad performance. The result of analysis and simulation reveals that the performance of our scheme outperforms the original round-robin scheme.

**Keywords** Concurrent multipath transfer, Diversity paths, Data distribution, Buffer feedback

### 1 引言

随着网络技术的快速发展,网络终端可使用的接入技术越来越多。同时,由于多宿主具有多个网络接口<sup>[1]</sup>,也让网络终端同时接入多种网络成为可能<sup>[2]</sup>,使其能够同时接入并使用多条可以利用的路径。然而,根据当前 TCP 和 UDP 的研究进展,多宿主仅可以利用 TCP 或 UDP 协议通过某一种网络接入技术来访问网络提供的各种资源。

为了充分利用网络终端的多种网络接入方式,Delaware 大学的协议工程实验室提出了多路径并发传输模型(Concurrent Multipath Transfer, CMT)<sup>[3]</sup>,其利用流传输控制协议(Stream Control Transmission Protocol, SCTP)偶联<sup>[4]</sup>提供的多个数据流来传输待发送数据的不同部分,实现数据各个部分的并发传输。利用多路径并发传输模型,多宿主可以进行多路径并发传输,可以更好地利用可用网络资源进行数据传输,多路径并发传输模型如图 1 所示。

目前,CMT 接收端为多条通信路径设置了统一的接收缓存,对提前到达的数据包进行缓存。然而,由于不同通信路径在带宽、延迟、丢失率等方面存在差异,发送端在多条通信路径中顺序发送的数据包在传输过程中可能引入的延迟各不相

同。当路径的性能差异较大时,乱序的数据包会阻塞在接收端的缓冲区中<sup>[5]</sup>,增加了接收端缓冲区的负担<sup>[6,7]</sup>,影响数据包的及时提交,降低了 CMT 并发传输模型的整体性能。因此,如何保障 CMT 通信模型在差异化路径中的传输性能,以及降低性能较差通信路径带来的影响,成为改进 CMT 通信模型运行性能的关键。

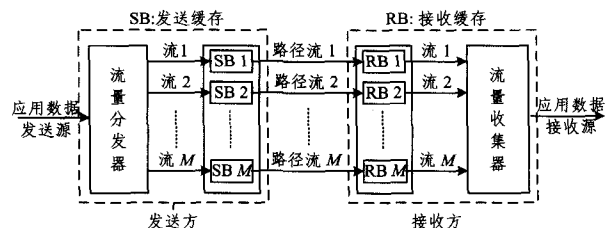


图 1 多路径并发传输

为此,本文提出了一种基于乱序反馈的差异化路径并发传输数据分配算法(Data Distribution Algorithm with Out-of-Order Feedback, DDAOOF)。该算法通过对路径性能差异性的评估和基于乱序情况对缓冲区造成的负载来反馈调节各条路径的数据发送比例,并对多路径并发传输模型的缓冲区负载和路径吞吐量进行改进。

到稿日期:2014-04-09 返修日期:2014-06-23 本文受国家自然科学基金项目(61003271),深圳市战略新兴产业发展专项资金项目(CXZZ2012082015532951)资助。

杜文峰(1977-),男,博士,副教授,主要研究方向为无线网络、多点通信、冲突解析等;吴真(1987-),男,硕士生,主要研究方向为 SCTP 协议、多路径并发通信等。

## 2 相关工作

在 CMT 通信传输模型中,由于各条路径的带宽、延迟、缓冲区不一定相同,会导致各条路径的传输性能存在差异性。如果路径之间的传输性能差异较大,数据包会出现乱序现象。类似于队头阻塞(Head of line, HOL),乱序的数据包阻塞在接收端缓冲区,不能及时提交,影响吞吐量的提高。如图 2 所示,如果两条路径的性能差异较大,主机 B 收到的 TSN 序号会出现不连续的现象,假设 TSN 为 3 的数据包晚到,那么 TSN=4、5、6 的数据包都会阻塞在缓冲区,等到 TSN=3 的数据包到达主机 B 才能实现数据提交。

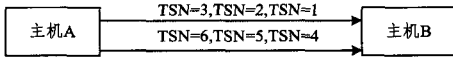


图 2 多路径并发传输阻塞

目前,针对 CMT 传输模型的研究主要集中在路径性能、路径选择、流量分割的研究和讨论上。文献[8]采用包分割的方式,按照路径的权重来分配数据,尽可能实现路径间的负载均衡,但是很难保证数据包的有序性;文献[9]通过流分割的方式,按照哈希函数的方式选择路径,减少了接收端的乱序,但是路径间的负载均衡性降低,哈希函数选取较难;文献[11]基于真实网络环境中的数据包头统计分析,以时间为界限划分流片,更好地保证了路径负载均衡,提高了系统性能,但是网络环境可能不断更新,根据现有的统计分析结果去划分流片,存在一定的局限性;文献[12]提出以最少负载的路径优先调度原则,避免了路径负载过重问题,但是最少负载优先原则不能充分利用各条路径的数据传输能力;文献[13]估算每条路径的发送队列中最后一个分组到达接收端的时延,把将要发送的分组分配给时延最小的路径,力求在多条路径上并发传输的分组尽可能地按序到达接收端,但是这种路径选择方法在性能差异较大的情况下会始终选择性能最好的路径发送数据,未能利用其他路径的数据传输能力;文献[14]提出以最后离开数据发送缓冲区的时间和早进入数据发送时间之差与数据发送缓冲区大小的比值作为评价路径性能的标准,并且数据调度采用的是路径性能和拥塞窗口之积来计算路径的数据发送能力,但是未能考虑到缓冲区大小对路径传输性能的影响;文献[5]提出建立路径和子流的概念(TCP),算法在限制流速情况下对缓冲区负载有较好的改善,但是该方法不适用于流传输控制协议(SCTP);文献[15]建立路径的端到端延时的函数,通过最小最大值的方法,求出最适合的分割流的向量,并对延时的变化和乱序进行讨论,但是端到端的延时函数很难准确表示,最大最小值方法在实际应用中较难实施;文献[10]是一种流片分割的算法,提出了按最大最小路径传输延时之差进行流量分割的方法,将单个流切成多个流片,每个分片按流量均衡算法分配到相应路径,较好地保证了路径间的负载均衡与包的有序性,但是当最大最小传输延时差异较大时,会导致流片大小偏大,负载均衡性得不到保证。

可以发现,目前国内外针对差异化路径下的多路并发传输数据分配问题,基本上都是从数据发送端的角度,以路径性能评估为基础,选择合适的数据分配算法来实现数据并发传输,很少从接收端缓冲区的角度研究多路并发传输。而多路径并发传输模型(CMT)在差异化路径中通过轮询(Round

Robin)算法将会导致数据包的严重乱序,造成接收端缓冲区的阻塞,严重影响传输性能。因此,从接收端缓冲区的角度来研究差异化环境下多路径并发传输模型的数据分配策略是非常有必要的。

本文在了解了 CMT 传输模型的工作原理基础上,对各种传输性能参数对吞吐量的影响进行分析,提出了差异化路径的性能评估方法;同时,本文针对导致传输性能影响较大的乱序问题进行分析,通过乱序反馈来调节路径间的数据分配比例,从而减少数据包的乱序现象,减轻缓冲区负载,提高路径吞吐量。

## 3 路径性能评估模型

本文以吞吐量作为衡量路径性能的主要依据。由于影响偶联性能的因素主要包括带宽、时延、接收端缓冲区大小等因素,本文将以此几个参数为指标来建立路径性能评估模型。

### 3.1 路径性能评价

在 CMT 并发传输过程中,路径的拥塞窗口限制了数据发送速率,拥塞窗口越大,可进入路径发送的数据越多,这里用发送窗口表示路径中允许进入的数据量,令发送窗口大小为  $W$ ,在路径带宽为  $B$  的情况下,发送窗口大小为  $W$  的数据完全进入路径的发送时延  $t_{send}$  为:

$$t_{send} = \frac{W}{B} \quad (1)$$

同时,令  $d$  为路径传播时延,则路径的往返传播时延  $t_{prop}$  为:

$$t_{prop} = 2 * d \quad (2)$$

令  $t_w$  为发送窗口大小为  $W$  的数据从开始发送到收到确认所需要的时间,包括数据进入路径的发送时延、从数据完全进入路径到收到确认需要的往返传播时延、接收端的处理时延,则  $t_w$  为:

$$t_w = t_{send} + t_{prop} + t_{deal} \quad (3)$$

其中,  $t_{deal}$  表示处理时延,指接收端将数据提交给应用层处理所花费的时间。由于单条路径传输数据时,数据乱序问题会大大减轻,避免了数据包阻塞在接收端缓冲区,从而能及时提交到应用层。因此,式(3)可以简化为:

$$t_w = t_{send} + t_{prop} \quad (4)$$

为了体现路径的发送能力,本文引入函数  $Q(W)$  来表示发送窗口  $W$  与窗口大小为  $W$  的数据从进入发送缓冲区到接收到全部确认数据包的所需要时间的比值。

$$Q(W) = \frac{W}{t_w} \quad (5)$$

根据式(5)可知,  $Q(W)$  是关于发送窗口  $W$  的递增函数,当发送窗口为最大值  $W_{max}$  时,  $Q(W)$  可以取到其最大值  $Q(W_{max})$ 。

令  $BufferSize$  为接收端缓冲区大小,根据 CMT 传输原理,其发送窗口最大值  $W_{max}$  为接收端缓冲区的大小,即

$$W_{max} = BufferSize \quad (6)$$

可以得到,  $Q(W_{max}) = Q(BufferSize)$ 。

由于  $Q(W_{max})$  同时考虑了带宽、时延、接收缓冲区大小,本文使用  $Q(W_{max})$  来衡量路径的传输性能。为了简化表达,本文使用  $Q$  来表示特定路径的路径性能参数。  $Q$  值越大,路径的传输性能越好。

$$Q=Q(W_{\max}) \quad (7)$$

### 3.2 路径性能差异系数

令路径的发送窗口为接收端缓冲区大小, 获得其最大发送时延  $\max(t_{\text{send}})$ 。根据路径的传输延迟和发送延迟之间的关系, 可以将路径分为时延敏感型路径和带宽敏感型路径两种。

#### (1) 时延敏感型路径

该类路径的最大发送时延小于往返传播时延, 即  $\max(t_{\text{send}}) < t_{\text{prop}}$ 。时延敏感型路径的传播时延对路径吞吐量影响较大, 带宽对路径吞吐量的影响相对较小。

#### (2) 带宽敏感型路径

该类路径的最大发送时延大于往返传播时延, 即  $\max(t_{\text{send}}) > t_{\text{prop}}$ 。带宽敏感型路径的带宽对路径吞吐量影响较大, 传播时延对路径吞吐量影响相对较小。

假设 CMT 偶联的第  $i$  条路径表示为  $P_i$ 。令路径  $P_i$  的可用带宽为  $B_i$ , 路径传输延迟为  $d_i$ , 路径性能指数为  $Q_i$ 。为了衡量路径之间的性能差异程度, 本文继续引入路径性能差异系数  $D$ ,  $D$  越大表示路径之间的传输性能差异性越大。

在比较两条路径的传输性能时, 可能存在以下 3 种情况。

(1) 两条路径都为带宽敏感型, 可以通过带宽来定义路径性能差异系数  $D$ :

$$D = \left\lceil \frac{\text{Max}(B_1, B_2)}{\text{Min}(B_1, B_2)} \right\rceil \quad (8)$$

(2) 两条路径都为时延敏感型, 可以通过时延来定义路径性能差异系数  $D$ :

$$D = \left\lceil \frac{\text{Max}(d_1, d_2)}{\text{Min}(d_1, d_2)} \right\rceil \quad (9)$$

(3) 两条路径为带宽敏感型和时延敏感型的混合, 可以通过  $Q$  值来定义路径性能差异系数  $D$ :

$$D = \left\lceil \frac{\text{Max}(Q_1, Q_2)}{\text{Min}(Q_1, Q_2)} \right\rceil \quad (10)$$

## 4 基于乱序反馈的差异化多路径并发传输数据分配算法

令  $\text{RecBufferSize}_i$  为第  $i$  次统计接收端缓冲区的负载大小,  $\text{RecCount}$  为接收端缓冲区负载统计次数, 且  $\text{Sum}(\text{RecBufferSize})$  为接收端负载统计总和,  $\text{AvgBufferSize}$  为接收端缓冲区的平均负载。

$$\begin{aligned} \text{Sum}(\text{RecBufferSize}) &= \sum_{i=1}^{\text{RecCount}} \text{RecBufferSize}_i \\ \text{AvgBufferSize} &= \frac{\text{Sum}(\text{RecBufferSize})}{\text{RecCount}} \end{aligned} \quad (11)$$

定义  $\text{SentCount}_i$  为路径  $P_i$  相对最优通信路径的发送次数, 表示在该路径发送数据之前, 最优路径已经发送数据的次数。  $\text{PathRatio}_i$  为路径  $P_i$  相对最优通信路径的发送系数。

$$\text{PathRatio}_i += (D_i - 1) \quad (12)$$

其中,  $D_i$  表示路径  $P_i$  与路径性能最好的路径的差异系数。  $\text{PathRatio}_i$  的初始值为 0。

在分配数据之前, 发送端计算各条通信路径的  $Q$  值, 并将路径按照  $Q$  值降序排序, 得到路径性能集合  $R = \{Q_1, Q_2, \dots, Q_n\}$ 。其中,  $Q_1$  和  $Q_n$  分别对应性能最好和性能最差的路径。

假设当前发送数据的路径为  $P_i$ , 发送端将在路径  $P_i$  上

一直发送数据, 直到路径  $P_i$  的空闲发送窗口低于 1 个 MTU 为止。在本算法中, 发送方在发送数据之前将比较接收缓冲区负载  $\text{AvgBufferSize}$  与缓冲区负载反馈阈值  $\delta$  的关系。接收端将根据实际应用对传输延迟的服务质量需求来确定  $\delta$  值。

(1) 当接收缓冲区负载低于缓冲区负载反馈阈值即  $\text{AvgBufferSize} \leq \delta$  时, 接收端缓冲区的负载较低, 发送方将按照集合  $R$  中的路径排列依次轮询发送数据;

(2) 当接收缓冲区负载高于缓冲区负载反馈阈值即  $\text{AvgBufferSize} > \delta$  时, 接收端缓冲区的负载超过设定的反馈阈值, 发送端将启动乱序反馈来协助数据分配。

由于路径  $P_i$  的数据分配会加重接收端乱序, 本算法将停止在路径  $P_i$  上的数据发送, 选择在性能最好的路径发送数据。此时, 发送端将路径  $P_i$  相对最优路径的发送次数递增一次, 即  $\text{SendCount}_i += 1$ 。

如果当前路径  $P_i$  的相对发送次数  $\text{SendCount}_i$  大于路径发送比例  $\text{PathRatio}_i$  ( $\text{SendCount}_i > \text{PathRatio}_i$ ), 发送端将在路径  $P_i$  上发送一次数据, 并将路径  $P_i$  的相对发送次数  $\text{SendCount}_i$  清零; 同时, 发送端更新路径  $P_i$  的发送系数  $\text{PathRatio}_i$ 。

在数据发送过程中, 发送端及时更新各条路径的  $Q$  值和  $D$  值, 调整集合  $R$  中的路径排列, 直到数据发送过程结束。

## 5 性能分析

在分析算法性能之前, 先做以下假设: 整个 CMT 偶联由  $N$  条独立的通信路径组成, 且第  $i$  条路径用  $P_i$  表示。

根据 SCTP 协议的拥塞控制原理, 发送方的数据到达接收方后, 接收方将缓存所有已收到的数据包, 将低于当前 SACK 对应 TSN 的所有信息包提交给上层应用, 并从接收缓冲区中释放数据空间。接收方向上层应用提交数据后, 将通过 SACK 信息通知接收方当前接收缓存的大小, 允许发送方根据情况发送数据。因此, 使用不同数据分配算法的 CMT 偶联的传输性能将由接收缓冲区的释放速度和执行条件决定。

在本算法中, 发送端将充分利用各条路径的传输性能, 允许传输性能较好的传输路径进行多次传输。由于各条路径在传输过程中是相互独立的, 整个 CMT 偶联的吞吐量并不会因为某一条的传输性能变坏而急剧降低。

根据本算法的运行原理, 发送端将根据接收缓冲区与负载反馈的关系来调整数据分配策略。定义路径  $P_i$  发送数据出现的次数占总分配次数的比例为路径  $P_i$  在传输过程中获得数据包的概率  $p_i$ 。

(1) 当接收缓冲区的负载低于缓冲区负载反馈阈值即  $\text{AvgBufferSize} \leq \delta$  时, 接收端缓冲区的负载较低, 发送方将按照集合  $R$  中的路径排列依次轮询发送数据。

此时, 所有路径被分配数据的概率相等, 且路径的  $P_i$  发送概率  $p_i$  表示为:

$$p_i = \frac{1}{N} \quad (13)$$

CMT 偶联的传输性能与默认的轮询数据分配算法一致。

(2) 当接收缓冲区的负载高于缓冲区负载反馈阈值即

$AvgBufferSize > \delta$  时,接收端缓冲区的负载超过设定的反馈阈值,此时发送端将启动乱序反馈来协助数据分配。

定义  $K$  为 CMT 偶联在遍历完所有路径时的发送次数总和,则  $K$  为

$$K = D_N + \frac{D_N}{D_2} \times (1 + PathRatio_2) + \frac{D_N}{D_3} \times (1 + PathRatio_3) + \dots + \frac{D_N}{D_i} \times (1 + PathRatio_i) + \dots + \frac{D_N}{D_N} \times (1 + PathRatio_N) + \sum_{i=2}^N SentCount_i$$

$$= D_N + \sum_{i=2}^N \frac{D_N}{D_i} \times PathRatio_i + \sum_{i=2}^N SentCount_i + \sum_{i=2}^N \frac{D_N}{D_i}$$
(14)

其中,路径  $P_1$  发送数据的概率  $p_1$  为

$$p_1 = \frac{SentCount_1}{K}$$

$$= \frac{D_N + \sum_{i=2}^N \frac{D_N}{D_i} \times PathRatio_i + \sum_{i=2}^N SentCount_i}{K}$$
(15)

其他路径  $P_i$  被分配数据的概率  $p_i$  为

$$p_i = \frac{SentCount_i}{K} = \frac{\frac{D_N}{D_i}}{K}$$
(16)

由于在集合  $R$  中,序号较小的路径性能相对较优,其路径性能差异系数  $D_i$  值相对较小。因此可以看到,在本算法中路径性能较优的路径被分配数据的概率相对较大。并且,当可能出现乱序时,本算法将选择在性能最好的路径发送数据,路径阻塞率为 0。

令路径  $P_i$  中数据包的平均逗留时间和平均等待时间分别为  $W_i$  和  $W_q$ 。此时,数据包在整个 CMT 偶联中的平均逗留时间  $W$  为

$$W = \sum_{i=1}^N p_i * W_i$$
(17)

且整个数据包在传输过程中的平均等待时间  $W_q$  为

$$W_q = \sum_{i=1}^N p_i * W_{qi}$$
(18)

## 6 仿真分析

实验分别将 Delaware 大学协议实验室提出的 CMT 模型的 RR 算法和本文的 DDAOOF 算法进行测试比较,给出了两种算法在差异化并发传输过程中的实时吞吐量、缓冲区平均负载、发送数据量、平均延时的仿真比较。

为了测试差异化路径对 CMT 偶联带来的影响,本节将从不同的差异系数与不同的缓冲区反馈参数两个角度去进行仿真实验。

### a) 两条路径下的不同差异系数的仿真

分别对  $D=2, 10$  的情况,缓冲区为  $BufferSize=64\text{kB}$  的网络场景,缓冲区负载反馈参数  $\delta$  为 1 进行仿真。

从图 3—图 6 可以看出,当通信路径存在性能差异时,例如  $D=2, 10$ ,使用 DDAOOF 数据分配算法的 CMT 偶联的吞吐量和缓冲区负载相比于标准的 RR 算法均有不同程度的提高;同时,随着性能差异系数的增大,路径性能提高幅度变大。

#### (1) 5Mb/s 20ms, 10Mb/s 20ms, 差异系数 $D=2$

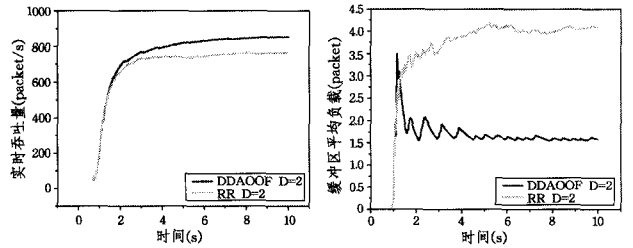


图 3 差异系数为 2 的实时吞吐量 图 4 差异系数为 2 的平均缓冲区负载大小

#### (2) 1Mb/s 30ms, 10Mb/s 20ms, 差异系数 $D=10$

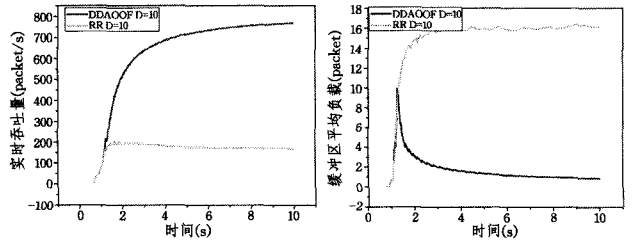


图 5 差异系数为 10 的实时吞吐量 图 6 差异系数为 10 的平均缓冲区负载大小

### b) 多条路径下的不同反馈参数的仿真

设定缓冲区负载反馈参数  $\delta$  为 0.1 和 2, DDAOOF-0.1 表示反馈参数  $\delta$  为 0.1 时的 DDAOOF 算法, DDAOOF-2 表示反馈参数  $\delta$  为 2 时的 DDAOOF 算法,缓冲区大小  $BufferSize=64\text{kB}$ 。仿真参数设置如图 7 所示。

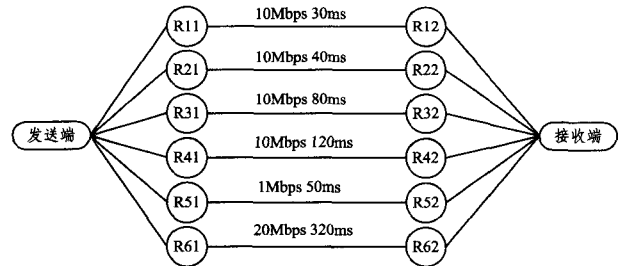


图 7 仿真环境

图 8 记录了 DDAOOF 算法和 RR 算法的实时吞吐量。在差异化的网络环境中,由于 RR 算法没有考虑路径性能的差异性,逐一为所有路径分配数据,而性能差的路径会极大地影响性能好的路径,导致吞吐量下降。DDAOOF 算法考虑到路径性能差异性,通过乱序反馈机制,充分利用性能好的路径多发数据,以减少性能差的路径对吞吐量的影响,提高了路径整体吞吐量。可以看到,在 DDAOOF 算法中,乱序反馈参数设置得越小,性能越好的路径发送数据概率越大,能够获得更好的吞吐量。

图 9 统计了 DDAOOF 算法和 RR 算法的缓冲区平均负载。在差异化的网络环境中,RR 算法大概率地在各条路径上发送数据,由于路径性能的差异性,数据包乱序现象非常严重,乱序的数据包未能及时提交到应用层,阻塞在接收端缓冲区,造成缓冲区负载加大。DDAOOF 算法通过乱序反馈机制,有效地减少了数据包乱序,减轻了缓冲区负载。可以发现,DDAOOF 算法中缓冲区负载反馈参数越小,缓冲区平均负载越小。

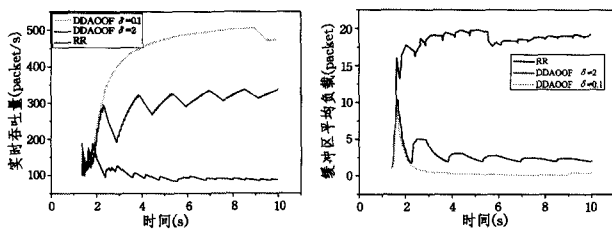


图8 实时吞吐量

图9 缓冲区平均负载

图10统计了DDAOOF算法和RR算法在10s内的发送数据量。可以发现,DDAOOF算法发送的数据量比RR轮询算法的有明显提高。缓冲区负载反馈参数设置越小,发送的数据量越多。由于DDAOOF算法充分利用性能好的路径发送数据,减轻了性能差的路径对整体吞吐量影响。相反,RR算法在传输过程中受到性能差路径影响,发送数据量相对较少。

图11统计了路径在仿真过程中的平均延时,可以发现,DDAOOF算法的路径平均延时比较小;而RR轮询算法受性能差的路径影响,对应的路径平均延时比较大。

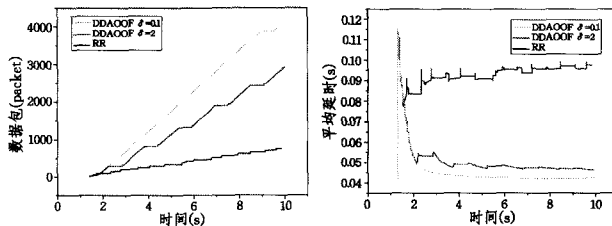


图10 发送数据量

图11 平均延时

**结束语** 由于CMT默认的轮询数据分配算法没有考虑路径传输性能的差异性,等概率地在各条路径分配传输数据将导致接收端缓冲区出现严重的乱序现象,造成缓冲区负载增大,偶联吞吐量降低较大。本文在分析了各种网络参数对吞吐量的影响后,给出了影响路径传输性能的关键原因。通过将传输路径进行分类,本文给出了路径性能评估方法以及路径性能差异化程度的定义。同时,本文提出了基于乱序反馈的差异化路径数据分配算法,其通过乱序反馈调节路径发送的比例,尽可能地避免传输性能差的路径影响传输性能好的路径,在减轻缓冲区负载的同时,提高路径吞吐量。

分析和实验结果表明,传输过程中利用乱序反馈来调整数据分配过程可以较好地保证接收包的到达次序,优化CMT传输模型的运行性能,能够避免性能较差的通信路径所导致的HOL问题。

### 参考文献

[1] Alpcan T, Singh J P, Basar T. Robust rate control for heteroge-

neous network access in multihomed environments[J]. IEEE Transactions on Mobile Computing, 2009, 8(1): 41-51

[2] Meyer D, Zhang L, Fall K. Report from the IAB Workshop on Routing and Addressing[R]. RFC 4984, 2007

[3] Iyengar J R, Aamer P D, Stewart R. Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths[J]. IEEE/ACM Transactions on Networking, 2006, 14(5): 951-964

[4] Stewart R. Stream control transmission protocol[S]. RFC 4960, 2007

[5] 曹宇,徐明伟.一种按需分配的多路径传输分组调度算法[J].软件学报,2012,23(7):1924-1934

[6] Cui L, Cui X, Jin J, et al. Countermeasures to Impacts of Bandwidth and Receiving Buffer on CMT Schemes[J]. Procedia Engineering, 2011, 15: 3723-3727

[7] Yang W, Li H W, Wu J P. PAM: Precise Receive Buffer Assignment Method in Transport Protocol for Concurrent Multipath Transfer[C]//2010 International Conference on Communications and Mobile Computing. Shenzhen, China, 2010: 413-417

[8] Leung K C, Li V O K. Generalized load sharing for packet-switching networks. I. Theory and packet-based algorithm[J]. IEEE Transactions on Parallel and Distributed Systems, 2006, 17(7): 694-702

[9] Shi W, Macgregor M H, Gburzynski P. Load balancing for parallel forwarding[J]. IEEE/ACM Transactions on Networking (TON), 2005, 13(4): 790-801

[10] Kandula S, Katabi D, Sinha S, et al. Dynamic load balancing without packet reordering[J]. ACM SIGCOMM Computer Communication Review, 2007, 37(2): 51-62

[11] Liu B, Sun C H, Yin Z Y, et al. Load-Balancing Multipath Switching System with Flow Slice[J]. IEEE Trans. on Computers, 2012, 61(3): 350-365

[12] Tari Z, Broberg J, Zomaya A Y, et al. A least flow-time first load sharing approach for distributed server farm[J]. Journal of Parallel and Distributed Computing, 2005, 65(7): 832-842

[13] Hasegawa Y, Yamaguchi I, Hama T. Improved data distribution for multipath TCP communication[C]//2005 IEEE Global Telecommunications Conference. St. Louis, USA, 2005: 271-275

[14] Xu C Q, Liu T J, Guan J F, et al. CMT-QA: Quality-aware Adaptive Concurrent Multipath Data Transfer in Heterogeneous Wireless Networks[J]. IEEE Transactions on Mobile Computing, 2013, 12(11): 2193-2205

[15] Prabhavat S, Nishiyama H, Ansari N, et al. Effective Delay-Controlled Load Distribution over Multipath Networks[J]. IEEE Trans. on Parallel and Distributed Systems, 2011, 22(10): 1730-1741

(上接第41页)

[4] Goldsmith A. Wireless Communications[M]. Cambridge University Press, 2005

[5] Gardner W A. Statistical Spectral Analysis, A Non-probabilistic Theory[M]. Prentice-Hall, Englewood Cliffs, 1987

[6] Zivanovic G D, Gardner W A. Degrees of cyclostationarity and their application to signal detection and estimation[J]. Signal Processing, 1991, 22(3): 287-289

[7] Lan Kun-chan, Shih Wen-yuah. Estimating Step Distance Using Simple Harmonic Motion[C]// Vehicular Technology Conference(VTC Spring), 2012 IEEE 75th, 2012: 1-5

[8] Chen Liang-yu, Lan Kun-chan. Estimating walking distance with a smart phone[C]//2012 Fifth International Symposium on Parallel Architectures, Algorithms and Programming(PAAP), 2012: 166-171

[9] Gardner W A. Cyclostationarity in Communications and Signal Processing[M]. IEEE Communication Society, 1994

[10] M S, Hosseini A T, Amindavar H, et al. Cyclostationary Detector in Ultra Wideband Impulse Radar[C]//2011 IEEE CIE International Conference on Radar. Volume 1, Oct. 2011: 24-27