doi:10.11959/j.issn.1000-436x.2015199

主动测量 SDN 性能的机制

陈鸣，代飞，许博，邢长友，李兵，张国敏
（解放军理工大学 指挥信息系统学院，江苏 南京 210007）

摘 要：提出了一种能够主动测量 SDN 中任何两点间的端到端路径性能的机制，设计了 OpenFlow 测量协议 OFMP，实现了无需改变交换机转发规则就能测量两点间特定流的多种性能参数的原型系统 OFMd。实验结果表明，OFMd 只需发送一个测试报文就能快速高效地获取多种端到端路径性能参数。
关键词：软件定义网络；主动测量机制；测量协议；端到端路径性能
中图分类号：TP393 **文献标识码**：A

Active measurement mechanism measuring SDN performances

CHEN Ming, DAI Fei, XU Bo, XING Chang-you, LI Bing, ZHANG Guo-min
(College of Command Information Systems, PLA University of Science and Technology, Nanjing 210007, China)

Abstract: An active SDN measurement mechanism was proposed, which could measure the end-to-end path performance between any two nodes, then the OpenFlow measurement protocol OFMP was designed, and a prototype OFMd was implemented, which could measure multiple performance metrics of the flow between two nodes without changing the switch forwarding rule. The experiment results show that OFMd can acquire multiple end to end path performance metrics only by sending a single probe packet.
Key words: software defined networking; active measurement mechanism; measurement protocol; end-to-end path performance

1 引言

网络测量是理解网络行为的基本手段和定量评估网络性能的重要方法。伴随着因特网技术的发展，包括主动测量和被动测量 2 种模式的网络测量技术逐渐走向成熟。目前，SDN 中的 OpenFlow 规范提供了从控制器实时获取流信息的接口[1]，这在一定程度上提供了一种集中式的被动测量方式。被动测量有其局限性，难以测量诸如两点之间连通性以及连通性能等多种参数并且实现技术复杂。IP 网络的经验表明，网络端到端路径性能是网络中最重要的指标之一，而获取其性能参数的主要手段是主动测量。主动测量通过向路径源端发送测试报文（序列），然后观察分析测试报文(probe)在网络传输过程中产生的变化，从而推测出网络状态和相关性能参数。这种主动注入测试报文并跟踪分组真实路由的测量方式更能反映问题真实情况。然而，SDN 目前缺乏实用的主动测量机制和方法。

对比 IP 网络中的主动测量机制，不难发现在 SDN 中发展高效易用的主动测量机制存在着很大困难。首先是可行性，IP 网络中任何两点之间都默认存在着端到端路径，这使测量端到端性能成为可能，而在 SDN(以下以 OpenFlow 网络为例)中两点之间的路径可能并不存在，即便存在，测试报文也要遵从控制器下发给交换机的转发规则。其次是易用性，IP 网络具有支持网络测量的协议如网际控制

收稿日期：2014-12-17；**修回日期**：2015-03-17
基金项目：国家自然科学基金资助项目（61379149,61103225）；国家重点基础研究发展计划（“973”计划）基金资助项目（2012CB315806）；江苏省科技计划基金资助项目（BY2013095-1-06）；江苏省自然科学基金资助项目（BK20140070）
Foundation Items: The National Natural Science Foundation of China (61379149, 61103225); The National Basic Research Program of China (973 Program) (2012CB315806); S&T Supporting Project of Jiangsu Province (BY2013095-1-06); The Natural Science Foundation of Jiangsu Province (BK20140070)

报文协议(ICMP, Internet control messages protocol)，这使主动测量易于进行，而在 SDN 中并不存在这样一种协议。第三是高效性，IP 节点协议栈内置支持主动测量功能，这使测量任务能够高效完成，而当前 SDN 节点却不能提供这种支持。实现高效易用的 SDN 主动测量机制必须要面对和解决这些问题。毕竟 SDN 正处于成长期，为其增加良好的测量机制并形成规范，取得比在 IP 网络中更好的测量效果是可能的。本文致力于研究一种高效、易用的主动测量 SDN 端到端路径性能的机制，为解决上述难题研制原型系统。

2 相关工作

IP 主动测量可以在不同协议层次实现，即可基于 TCP/IP 协议（如 ICMP、UDP 和 TCP 等协议）进行，也可在应用层定制特定的测量协议进行主动测量。应用层性能测量如 IETF 的 IPPM (IP performance metrics)工作组提出的单向主动测量协议(one-way active measurement)[3]和双向主动测量协议(two-way active measurement)[4]等，存在着应用层性能与网络层性能难以映射的问题。此外，由 NLANR(national laboratory for applied network research)提出的基于 IP 的主动测量协议 (IPMP, IP measurement protocol)[2]试图通过修改 IP 协议栈，以提高主动测量的功能和效率，然而这在因特网高度发展的现状下并不具备增量部署性，但其可以为在 SDN 网络中提高测量效率提供借鉴。

软件定义网络正处于发展初期，尽管在 SDN 进行主动测量的需求非常迫切，但目前的研究成果并不多。即使在当前比较成熟的 SDN 南向接口 OpenFlow 各种版本的规范[5]中，均没有涉及测量 OpenFlow 网络端到端路径性能的规范，目前仅能从控制器提供的 API 中获取有限、零散的性能信息。ATPG(automatic test packet generation)[6]通过生成探测报文来验证数据平面的转发行为，主要关注全网络范围的行为而不是特定路径的端到端性能。OFRewind[7]通过记录和回放 SDN 控制平面的流量来诊断网络故障。NetSight[8]通过收集存储底层网络中所有分组的摘要信息，再剖析这些历史信息来诊断网络故障，该文主要关注网络故障的精确诊断而非端到端的实时性能。OpenSketch[9]设计了一套软件定义的测量架构并在 NetFPGA 中进行了实现，关注用 SDN 来解决 IP 网络的有关测量问题。DREAM[10]提出了一种动态自适应的测量框架以动态调整每个测量任务的资源分配和使用，使测量资源和测量精度达到平衡。SDN traceroute[11] 与本文相关性较大，该文方法为了解决在 SDN 网络中的流路径跟踪问题，利用给交换机着色的方法在不改变网络转发行为的前提下实现了对流路径的跟踪，但其测量效率不够高、时延不够准确且无法获得其他性能参数。

3 设计主动测量 SDN 的机制

在设计主动测量机制时，应当遵循 SDN 的控制与转发分离理念，同时要使测量过程对网络转发性能的影响应尽可能得小。具体而言，首先，控制器执行主动测量的逻辑，而端到端路径性能测量则由 OpenFlow 交换机执行，即由控制器承担测量控制任务，它通过 OFMP 与交换机实体通信来协调测量行为，而 OFMP 交换机承担网络性能测量任务。OFMP 交换机是指能够识别并支持 OFMP 协议的交换机。第二，测量不应当改变节点中原有的数据转发规则，非 OFMP 交换机可以透明地传输测试报文。第三，控制器能够同时测量任意 2 台 OFMP 交换机之间多条路径的性能。最后，测量行为对网络节点的转发性能的影响应尽可能小。

注意到控制器是 SDN 的集中控制点，也是主动测量端到端路径性能的控制点，控制器既能够构造测试报文使之经过特定流的路由发送与接收，如果被测流路由不存在甚至能够建立起这样一条流及其转发规则再加以测量，也能够分析和显示测量结果，这可以使主动测量易于进行。

3.1 基本流程

基于控制器控制调度端到端路径的性能测量通常包括以下几个基本步骤。

1) 控制器构造测试报文，指定测量源、目的地和测量流路径的标识，以及根据测量测度定义发送、接收和处理测试报文的方式。

2) 测试报文经过被测路径时，OFMP 节点识别并在报文写入相关测量信息。

3) 测试报文到达目的地时，由 OFMP 节点控制测试报文流向，直至该测试报文返回控制器。

4) 控制器处理显示带有测量信息的测试报文。

例如，在网络节点均为 OFMP 节点并且它们时钟同步的情况下，控制器 C_0 需要确定测量源点为 n_0、测量目的地为 n_m 的流的具有特定标识和测量模

式，在 t_0 时刻，从 C_0 构造并发送具有时间戳 t_0 的测试报文 p_1，在 t_1 时刻 n_1 在流表中匹配到该流标识并在 p_1 上设置节点标识和时间戳 t_1，直至在 t_m 时刻到达 n_m 设置节点标识和时间戳 t_m。由于节点 n_m 是测量路径终点，它可以沿路径的反方向发送该测试报文，并依次追加到达的节点标识和时间戳，直至测试报文返回控制器 C_0，即可计算包括路由、往返时延(RTT)及逐跳时延等性能参数。可见，在主动测量中，测试报文应当具有被测路径的流标识，能被路径两端节点及部分中间节点识别并写入节点标识和时间戳，从而获得路由信息 C_0 $n_1 \cdots n_m \cdots n_1 C_0$ 和相应时间戳等性能参数。

3.2　测试报文在交换机中的传输

根据网络节点是否支持 OFMP，可将其分为 2 类：OFMP 节点是指能够识别并支持 OFMP 协议的网络节点，如支持 OFMP 的控制器和 OpenFlow 交换机；而非 OFMP 节点则不知晓 OFMP 协议。SDN 支持主动测量的最小化条件是控制器和被测路径 2 个端点必须支持 OFMP。如果测量性能指标需要涉及被测路径的某个节点，则要求该节点必须为 OFMP 节点。考虑到目前 OpenFlow 交换机通常是非 OFMP 交换机，要保证它们对测量呈透明性，只要这些交换机像对待该流的普通分组那样来对待测试报文，就能够测量到端到端路径 RTT 等性能参数。OFMP 控制器可由控制器运行 OFMP 组件分组而成；OFMP 交换机可由普通交换机扩展 OFM 守护进程而成，或作为特殊的测量仪器而存在。

举例来说，图 1 给出了测量 SDN 端到端单向时延的一个典型场景，其中控制器和被测路径 2 个端点为 OFMP 节点，而中间交换机有部分是非 OFMP 节点。图 1 下半部分示意性地描述了在路径端点间以单向测量方式处理测试报文的过程。首先，控制器根据测量应用逻辑确定端到端路径的源和目的地以及路径的流标识，构造并在发送 OFMP 测试报文前添加控制器标识和时间戳，然后用 PACKET-OUT 报文发给测量源点（参见图 1 左下部分分组 1）。

当该测试报文到达源点时，OFMP 交换机将根据测试报文中的流信息匹配查找下一跳的转发接口，此后在报文中添加源点标识和时间戳(如图 1 中的分组 2)。如果测试报文到达某非 OFMP 交换机，该交换机将其识别为普通分组，转发至下

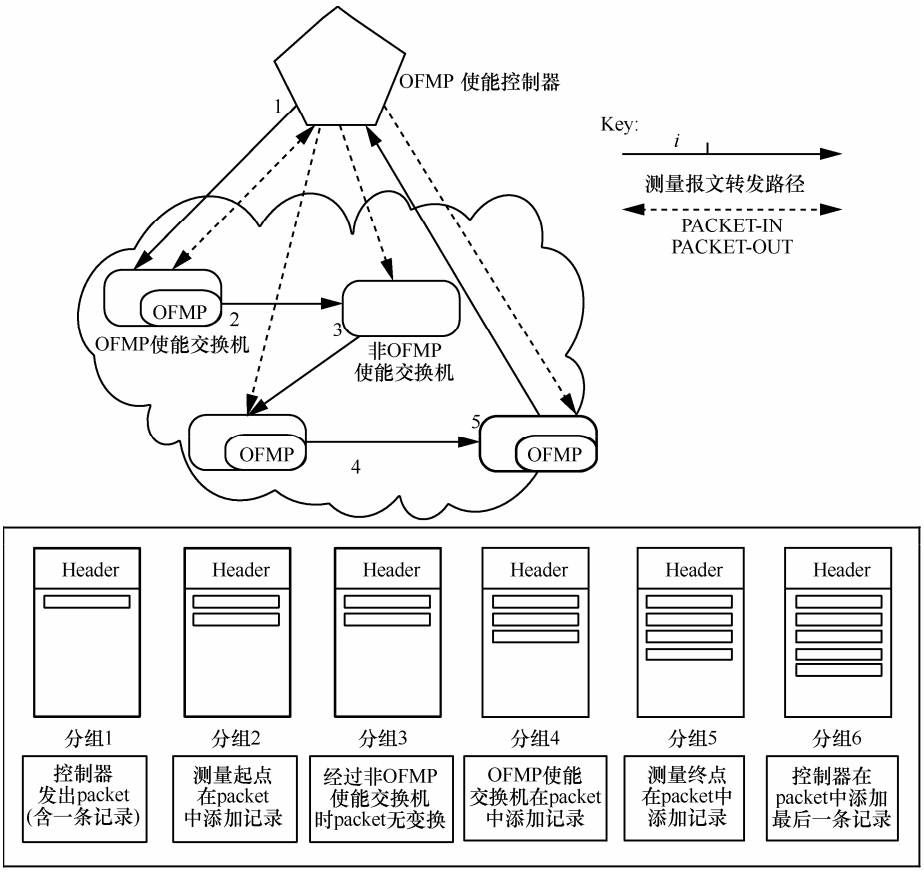


图 1　单向端到端测量中 OFMP 测试报文的传输

一跳交换机(如图 1 中的分组 3)，这时不会在报文中添加任何信息。后续的交换机按上述方式处理测试报文，故得到分组 4，直至测试报文到达终点，该交换机在添加本跳标识和时间戳后，将该报文通过 PACECK-IN 转交给控制器(如图 1 中的分组 5)。最后，控制器收到测试报文后再为其添加自己的标识和时间戳(如图 1 中的分组 6)，至此单向测量结束。显然，在满足时钟同步时，该过程能够测得路径交换机序列、单向时延和逐跳单向时延等性能参数。如果以双向测量方式工作，终点则要经过与前面相似但相反的过程返回该测试报文，直至到达测量源点，再由源点将测试报文转交给控制器。

显然，这种由控制器发起主动测量的设计，可以灵活控制和组织测量 SDN 网络中任何一条端到端路径的性能，从而能够为需要性能测度的网络应用提供实时性能参数。

4 设计 OpenFlow 测量协议

考虑到测量环境的复杂性，测试报文既能测量 OpenFlow 交换机网络，也能穿越 IP 路由器，选择将 OFMP 作为应用层协议，即将 OFMP 报文封装在 TCP 或 UDP 报文中。控制器与端点交换机交互再将该构造的测试报文封装在 OpenFlow 协议的 PACKET-OUT 或 PACKET-IN 中，测试报文通过 OFMP 交换机时能够被识别和处理并进行转发，而通过非 OFMP 交换机时也能随工作流一样被转发。为使测试报文被 OFMP 交换机识别，测试报文的标识利用了 IP 首部的 ToS 字段特征，即该字段的最低两位均置 1，便于 OFMP 交换机识别并且进行处理。

4.1 OFMP 报文主要字段

设计的 OFMP 协议必须具有以下能力：描述主动测量功能的能力、描述被测数据流标识的能力以及承载多跳测试信息的能力。从功能上讲，OpenFlow 测量协议首先应当为控制器与交换机中的 OFMP 实体之间的通信提供语法、语义和定时等要素；其次，应支持测量 SDN 端到端路径的 RTT、单向时延、逐跳时延以及分组丢失率等性能指标；第三，测量可覆盖数据平面和控制平面以及两者的交互。

在 OpenFlow 规范中，OpenFlow 交换机以流为基本操作对象，而流可由链路层、网络层和运输层首部等信息定义。由于 OFMP 报文被封装在运输层报文中，即 SDN 测试报文已经具备了链路层、网络层和运输层等关键字段的信息，因此根据流进行端到端测量时，仅需在 OFMP 首部中声明测量的源点和终点。

OFMP 报文定义的主要字段包括：12 bit 的 Flag 字段，其中，2 bit T 是 type 标识位，0 表示测量请求，1 表示测量响应；2 bit M 是 mode 标识位，0 表示单向测量模式，1 表示双向测量模式；2 bit S 是 single/continuous 标识位，0 表示单次测量，1 表示连续测量；2 bit P 是 plans 标识位，0 表示测量路径性能，1 表示测量控制平面和数据平面间的性能。

各为 2 byte 的 Identification 和 Sequence Number 字段，用于标识不同的主动测量过程，即不同的测量过程所发送报文中的 Identification 不相同，而在同一个测量过程发送报文中的 Identification 应当相同，但后一个报文中的 Sequence Number 应当比前个报文的加 1。

1 byte 的 Path Pointer 字段，每当插入一个路径测量新记录时，须将该流路径指针加 1。它累计了当前路径记录的总数，即指出下一个可用路径记录位置的起始点。

各为 6 byte 的 Measurement Start 和 Measurement End 字段是路径两端交换机的 Dpid 值，用于标识测量的起点和终点。

Path Record 为 OFMP 的数据部分，内容为节点标识和时间戳小节。

4.2 路径记录格式

OFMP 报文中 Path Record 部分的单跳路径记录占 18 byte，其格式如图 2 所示。

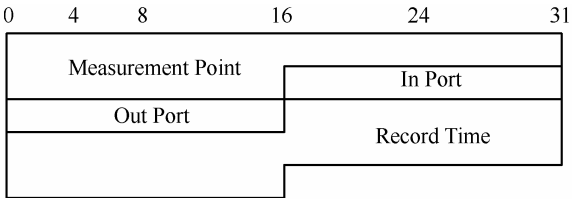


图 2 OpenFlow 测量协议中的路径记录格式

6 byte 的 Measurement Point 字段用于放置本节点标识，填充 OpenFlow 交换机的 Dpid 或主机的 MAC 信息。

各为 2 byte 的 In Port 和 Out Port 字段为入接口和出接口号，用于标识流数据在设备接口间的交换关系。

6 byte 的 Record Time 为本地到达时间戳，其值为自 1970 年 1 月起计数的毫秒数。

一个 OFMP 报文仅支持记录单次主动测量过程，测量的关键在于 OFMP 节点在接收到测试报文时，能够识别它并将其交给 OFM 守护进程来追加本节点的标识和时间戳。即使是控制器，也都要在请求和响应报文中添加相应的信息。对于一个长为 1 289 byte 的典型 UDP 分组，由于可容纳约 70 跳记录，这对大规模网络也是够用的。完整的节点标识和时间戳信息是计算端到端路径各种性能参数的基础。为了避免差错，在插入新路径信息前，应当先检查分组是否有足够的剩余空间。

4.3　OFMP 有限状态机

涉及 OFMP 的通信实体有 OFMP 交换机和 OFMP 控制器。前者的有限状态机（FSM）较为简单，它只有“等待 probe 报文”和“等待转发 probe 报文”2 个状态。如图 3 所示，当交换机在状态“等待 probe 报文”中时，则一旦从流中解析出 probe 报文，就需要在测试报文中写入测量信息并进入“等待转发 probe 报文”状态，此后或者向控制器或者向下一跳交换机发送 probe 报文，并再次转入“等待 probe 报文”状态。

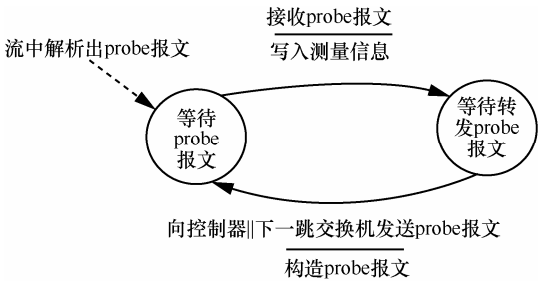


图 3　OFMP 交换机的 FSM

如图 4 所示，OFMP 控制器有 4 个状态。当用户发送测量指令后，控制器指定测量任务并在测试报文中写入测量信息，并从“等待解析指令”状态进入“等待发送 probe 报文”状态。控制器发送 probe 报文后，进入“等待接收 probe 报文”状态，若接收 probe 报文且测量结束，就进入“分析测量结果”状态，此后输出测量结果并再次进入“等待解析指令”状态。若“等待接收 probe 报文”状态出现“接收 probe&&序号正确”事件时，标识该次测量仍在进行中，则进入“等待解析指令”状态，等待后继测量指令。

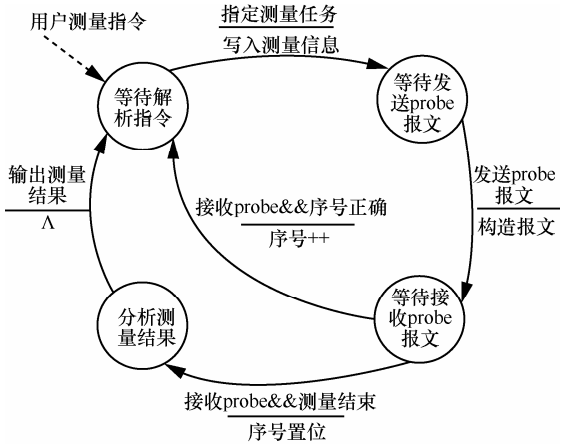


图 4　OFMP 控制器的 FSM

5　OFMd 系统实现

5.1　OFMd 的主要模块

本节主要研究为支持主动测量机制，在控制器和交换机需要设置的主动测量模块及其关键技术，由此实现主动测量系统 OFMd。

假定控制器和部分 OpenFlow 交换机(例如被测路径的两台边缘交换机)为 OFMP 节点。为了验证 SDN 主动测量机制和 OFMP 协议，基于 Stanford 软件 OpenFlow 交换机代码，设计了 OFM 守护进程；基于 POX 控制器代码，设计了 OFMP 组件分组，该分组具有类 POX 的事件触发机制。图 5 描述了 OFMd 的主要模块及其交互关系。

控制器部分包含 3 个接口：操作接口、测量结果接口以及通信接口。在控制器中，管理员通过操作接口输入测量指令，指令解析模块处理指令并转换成 OpenFlow 交换机可执行的测量任务，由组织测量模块启动主动测量过程。当 OpenFlow 通信接口接收到交换机反馈的测量结果后，经结果分析模块的分析处理，由结果输出接口呈现测量结果。其中，操作接口和测量结果接口可共用相同的应用接口，而 OpenFlow 通信接口为 OpenFlow 规范中的控制器与交换机通信的安全通道。

在 OFMP 交换机中，流处理的工作过程与普通 OpenFlow 交换机类似。控制器下发的 OFMP 测试报文从通信接口来，先经 OFMP 任务模块解析交由 OFMP 报文处理模块进行路径记录等必要操作，最终通过 OFMP 通信接口发送测量结果报文。为简明起见，图 5 仅显示了 2 个端点的 OFMP 交换机，而略去中间的其他交换机。注意到当测量单向时延和分组丢失率时，目的端交换机可直接向控制器提交

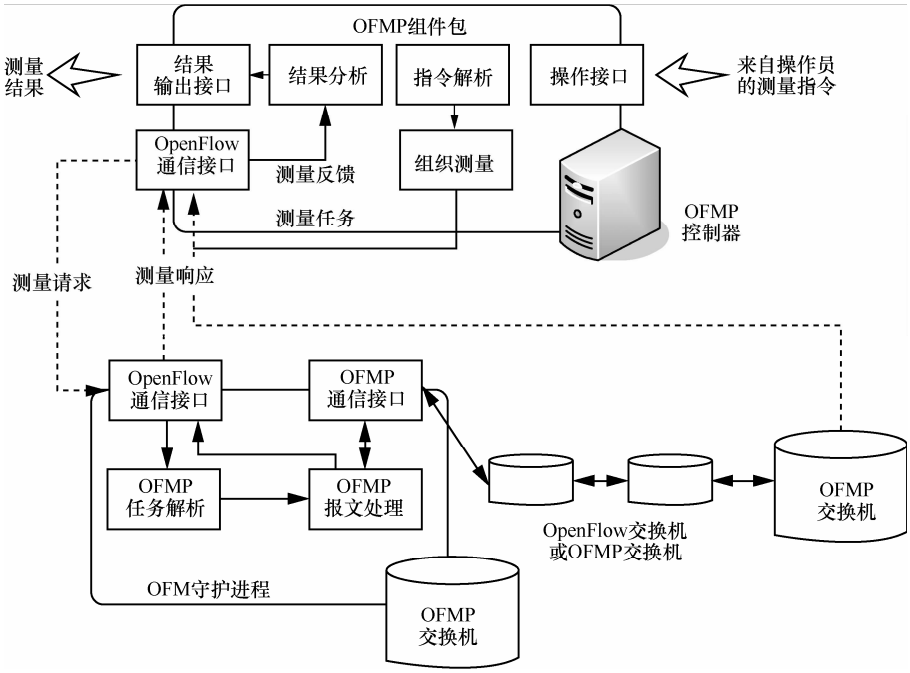


图 5 OFMd 测量系统的功能模块

测量结果，而当测量 RTT 和双向分组丢失率时，仍由源端交换机向控制器提交测量结果。

5.2 性能测度及其计算

SDN 端到端路径的性能测度主要包括：路由(路径节点序列)、RTT、单向时延、分组丢失率，以及逐跳时延和控制平面与数据平面间的时延等。

OFMd 能够具体测量何种性能测度，与网络中的设备(控制器和交换机)时钟是否同步以及是否对 OFMP 提供支持有关。例如，在所有网络节点均为 OFMP 节点并且时钟同步的情况下，该主动测量机制不仅能够测量路径的端到端 RTT，也能测量其单向时延；不仅能够测量端到端总时延，也能测量其逐跳时延；不仅能够测量数据平面的时延，也能测量控制器与数据平面间的时延。而在仅有控制器和两端交换机为 OFMP 节点并且时钟不同步的情况下，该主动测量机制也许只能测量较少的性能参数，如路径的 RTT。此外，通过连续发送多个测试报文，在上述 2 种情况下统计计算发送分组数与接收分组数之比，得到端到端路径的分组丢失率。

表 1 归纳了 OFMd 提供的性能测度及其所需测量条件。

5.3 某些实现问题

在实现主动测量机制时，还有一系列技术问题需要处理，下面讨论其中的几个。

测试报文操作问题。OpenFlow 交换机执行 OFMP 测量的过程，实际就是匹配相应字段、添加路径记录和封装报文等操作的过程。当添加和封装报文时，报文长度的变化将导致内存操作。此时需要遵从 OpenFlow 专用的内存使用规范，进行结构 ofpbuf 的操作。由于交换机接收到的报文都用 ofpbuf 结构体来存储，报文的起始位置、长度以及报文每个字段的位置都能得以确定。因此，在测量

表 1 OFMd 提供的性能测度及其所需测量条件

测度	交换机类型	时钟同步
路由	所有节点均为 OFMP 交换机	不需要
RTT	仅端节点为 OFMP 交换机	不需要
单向时延	OFMP 交换机之间	需要
逐跳时延	所有节点均为 OFMP 交换机	需要
丢包率	仅端节点为 OFMP 交换机	不需要
控制平面与数据平面间的时延	OFMP 控制器与 OFMP 交换机间	不需要

过程的内存操作中，首先要用 xmalloc 重新申请一块内存，然后将路径记录复制到新的位置，最后给各个指针赋以相应的值来完成操作。

主动测量的精度问题。当采用网络时间协议（NTP）同步一个 OpenFlow 网络中网络实体的时钟时，它们之间通常约有几十微秒的误差。测量表明由于每台交换机在网络流量轻载时的时延约为 0.12 ms，在网络流量较大时的时延约为 0.2 ms，因此，将时延测量精度定在 100 μs 量级，在 10 μs 量级则采用四舍五入规则处理实验误差。

6　原型系统实验与数据分析

为验证 SDN 主动测量机制的可行性和可用性，本文搭建了如图 6 所示的 OFMd 实验环境。其中包括 OpenFlow 软件交换机 5 台、PICA8 公司的 P3290 交换机 1 台、POX 控制器 1 台和 Linux 端主机 2 台以及 Spirent 流量发生器 1 台。5 台 OpenFlow 软件交换机 S1、S2、S3、S5 和 S6 均为运行 openflow-1.0.0 软件的 Linux PC。这些 PC 采用 i5-3470CPU，主频为 3.2 GHz，内存为 4 GB，具有 4 端口吉比特以太网，它们的 Dpid 值分别设为 1、2、3、5、6，交换机 S4 的 Dpid 设为 4。其中交换机 S1、S2、S3、S5 和 S6 和控制器都运行了 OFM 守护进程，是 OFMP 节点，交换机 S4 则是非 OFMP 节点的。控制器采用了 pox-carp 版本，与交换机之间使用带内方式互连。控制子网为 10.0.0.0/24，而数据平面的 OpenFlow 子网 IP 地址为 192.168.1.0/24。此外，所有交换机均可运行 NTP 协议，它们与控制器能够进行几十微秒级的时钟同步。

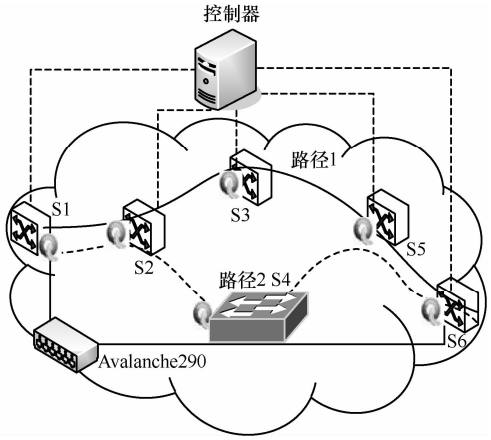
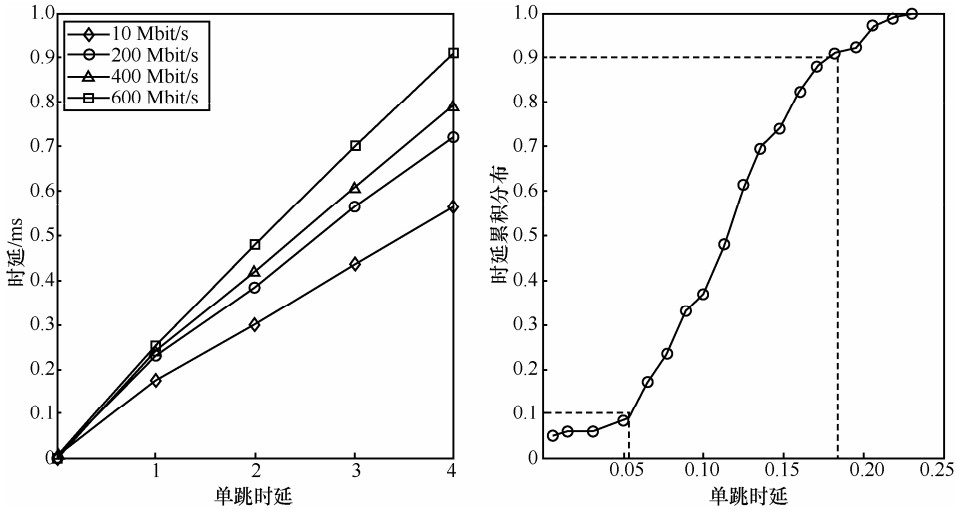


图 6　OFMd 的实验环境

6.1　测量全 OFMP 交换机路径的实验

实验 1　当路径全部节点均为 OFMP 节点时，考查 OFMd 是否能够正确、方便、高效地测量端到端路由、单向时延和 RTT 以及逐段单向时延等性能参数。实验中，首先用 Spirent Avalanche 流量发生器在 S1 到 S6 之间产生 TCP 背景流，并由控制器为背景流建立流表，使其路由为 S1-S2-S3-S5-S6(即图 6 中的路径 1)。交换机与控制器之间使用 NTP 协议进行时钟同步，然后，控制器调用测量应用程序测量该路径的端到端性能，用 PACKET-OUT 发起测量，测量的起点为 S1，终点为 S6。当采用双向测量模式时，测试报文到达 S6 后按原路径返回到 S1，由 S1 用 PACKET-IN 向控制器返回具有控制器和各交换机时间戳的测试报文，控制器测量应用程序分析处理这些节点标识和时间戳，得到测量结果。图 7 给出了沿路由逐跳单向时延的平均值



(a) 单向累积时延的平均值　　(b) S2 至 S3 间单跳时延的累积分布

图 7　全 OFMP 交换机路径时延测量

(100 次测量的平均值，单位为 ms)以及在背景流为 10 Mbit/s 时交换机 S2 至 S3 之间单跳时延的累积分布。

实验结果表明，利用 OFMd 发送的主动测量报文上所获取的 Dpid 和时间戳序列，得到了逐跳时延和端到端单向时延等性能参数。图 7(a)显示随着背景流量的增大，交换机处理的流分组数据增多，逐跳时延和端到端单向时延也随之增大，这符合网络实际的情况。从图 7(b)可以发现，单跳时延在一定范围内变化，但对于 80%的测量结果，变化范围在 0.13 ms 以内，这可能是网络背景流量的波动和测量误差等所致。这种波动范围很小，它对最终测量结果影响不大。为了进一步对测量结果的稳定性进行分析，对不同背景流情况下逐跳时延的均值和方差进行了计算，结果如表 2 所示。

从图 7 和表 2 的测量数据可知，当背景流量从 10 Mbit/s 逐步增大到 600 Mbit/s 时，网络每跳时延都在逐步加大，这是由于网络分组排队时延的增加而导致的。在 100 次测量下，逐跳时延的方差较小，这表明 OFMd 的测量结果比较稳定。对于网络交换机来说，由于测量报文和背景流的流标识一致，故转发行为也是一致的，因此测量报文携带的 Dpid 和时间戳序列真实地反映出网络当前被测端到端路径的性能参数。

该实验的其他相关工作还包括：1）在网络无法进行时钟同步的情况下，仍然能够快速测出 OpenFlow 网络任何流的路径信息；2) 若测量前路径 1 不存在，OFMd 仍可以先建立路径 1，然后再测量该路径的性能；3）在上述 100 次测量中，没有发现分组丢失现象，因此可估算此时网络路径 1 的分组丢失率为 0。

实验小结：控制器发起一次测量就能得到所测流的路由、单向时延和 RTT 以及逐段单向时延等端到端性能参数。

6.2 测量部分 OFMP 交换机路径的实验

实验 2 两端为 OFMP 交换机，而路由中间有部分非 OFMP 交换机，其他条件同于前面的实验。具体而言，控制器建立的路由为：S1-S2-S4-S5-S6 (图 6 中的路径 2)。其中商用 OpenFlow 交换机 S4 运行 OpenvSwitch1.9.2，OpenFlow 协议版本为 1.0，它不支持 OFMP，而其他节点均为 OFMP 节点。这时控制器发起双向测量，起点为 S1，终点为 S6，S6 收到测试报文后沿该路径反方向传输，最后由 S1 向控制器返回测试报文。分析表明，该测试报文具有控制器、S1、S2、S5 和 S6 写入的节点标识和时间戳序列，但没有 S4 的任何信息。

表 3 对比了实验 1 和实验 2 中测试报文携带的节点序列以及路径 RTT。可以看出，尽管路径 1 和路径 2 有所不同，但两者的 RTT 平均值基本相同。从两条路径的相似性或许可以解释所得结果的合理性。

实验小结：OFMd 能够跨越非 OFMP 节点进行测量，并能够方便、快速、高效地得到所测流的部分端到端性能参数。

根据上述实验，OFMd 主动测量功能的特点可总结如下：1) 从控制器集中实施 SDN 主动测量，要求控制器和路径两端交换机必须为 OFMP 节点；2) 当路径全为 OFMP 节点并且时钟同步，OFMd 能够用一次测量得到许多种端到端路径性能信息；3) 当路径中有部分为非 OFMP 节点，OFMd 仍能得到一些重要的端到端路径性能信息；4) OFMd 能

表 2 不同背景流情况下的逐跳平均时延和方差

背景流量大小 /(Mbit·s^{-1})	第 1 跳 S1-S2		第 2 跳 S2-S3		第 3 跳 S3-S5		第 4 跳 S5-S6	
	时延/ms	方差	时延/ms	方差	时延/ms	方差	时延/ms	方差
10	0.19	0.080 2	0.11	0.002 7	0.14	0.003 6	0.12	0.003 2
200	0.23	0.007 4	0.15	0.004 3	0.17	0.005 4	0.16	0.005 2
400	0.23	0.006 8	0.17	0.004 8	0.18	0.004 9	0.19	0.004 2
600	0.25	0.004 2	0.22	0.004 5	0.22	0.003 8	0.20	0.002 8

表 3 实验 1 和实验 2 中测量结果对比

单向流路由	测试报文中的路由(不含控制器)	*RTT* 平均值/ms
S1→S2→S3→S5→S6	S1→S2→S3→S5→S6→S5→S3→S2→S1	1.247
S1→S2→S4→S5→S6	S1→S2→S5→S6→S5→S2→S1	1.112

够测量已存在或不存在路径的端到端性能参数，无需改变节点的转发规则。

实验 3　此实验比较交换机类型对端到端性能测量的影响。分 2 种情况对路径端到端 RTT 进行测试。情况 1：图 6 中路径 1 的所有交换机均为 OFMP 交换机；情况 2：图 6 中路径 1 的 S2、S3 和 S5 不启动 OFM 守护进程，即仅 2 个端点支持 OFMP。测量这种差异是否会对 RTT 的大小有显著的影响。

图 8 显示了在背景流量为 200 Mbit/s 时对路径端到端 RTT 进行 100 次测量的累积分布曲线。实验表明，为交换机增加主动测量功能对交换机传输性能的影响微不足道，全 OFMP 节点的 RTT 均值为 1.2 ms，方差为 0.038 1；仅 2 个端点为 OFMP 节点的 RTT 均值为 1.1 ms，方差为 0.008 8。而当背景流量为 600 Mbit/s 时，全 OFMP 节点的 RTT 均值为 1.3 ms，方差为 0.052 0；仅 2 个端点为 OFMP 节点的 RTT 均值为 1.2 ms，方差为 0.003 6。

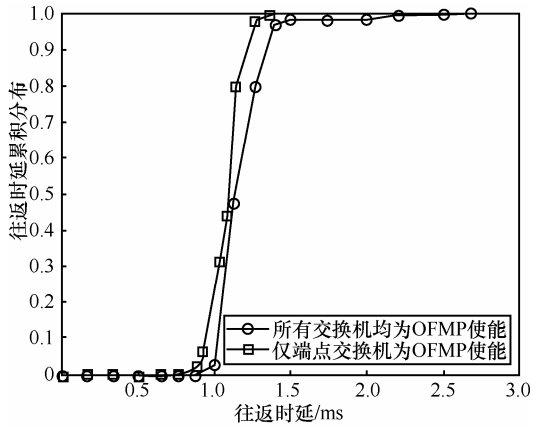


图 8　路径交换机全为 OFMP 节点和仅端点为 OFMP 节点的端到端 RTT 累积分布对比

实验小结：交换机类型对测量路径性能并无显著影响。这是因为测试报文数量不多并且处理 OFMP 协议的计算量很小，这些处理不会成为交换机负担的缘故。因此，建议将 OFMP 纳入 OpenFlow 规范中，这将给 SDN 的发展带来利好。

6.3　控制平面与数据平面间的性能测量

SDN 分离了控制平面与数据平面，但目前控制平面对路径端到端性能有什么影响并没有结论。

实验 4　此实验测量这 2 个平面之间的单向时延/RTT 参数。在 OFMd 设计中，所有测试报文均由 OFMP 控制器发起。该测量报文首先由控制器设置时间戳 t_0，再传递给路径首跳交换机，由它设置时间戳 t_1。在时间同步的情况下，2 个平面之间的单向时延 t_1-t_0；在时间不同步的情况下，设置测量的源点和终点均为首跳交换机，则 2 个平面之间的 $RTT_{\text{planes}}=t_0'-t_0$，其中 t_0' 为当首跳交换机返回测试报文时控制器再次设置的时间戳。测试时保持控制器 CPU 在大约 30%利用率的水平，由控制器对交换机 S3 发起 100 次 RTT 测量，测得控制平面与数据平面之间的 RTT 均值为 0.44 ms，方差为 0.005 3。

实验小结：OFMd 能够测量控制平面与数据平面之间的 RTT 均值，为进一步分析控制平面对数据平面的影响提供了便利。

7　结束语

合理解决主动测量报文在被测路径的传输、主动测量协议的设计以及交换节点对主动测量功能等难题，才能发展实用的 SDN 端到端路径性能测量方法和系统。本文提出了一种主动测量 SDN 中任何两点间的端到端路径性能的机制，设计了 OpenFlow 测量协议 OFMP，实现了原型系统 OFMd。实验结果表明，在所有节点均支持 OFMP 的环境下，OFMd 能够测量端到端路径的序列、单向/双向/逐跳时延和分组丢失率等性能参数；在控制器和 2 个路径端点为 OFMP 节点且其他部分节点为非 OFMP 节点的环境下，OFMd 能够测量端到端路径的 RTT 和分组丢失率等性能参数。下一步，将根据 SDN 程序开发、网络管理和定量评估新型网络机制等方面的需求，继续改进 OFMP 并研制相应的实用系统或工具。

参考文献：

[1] GUDE N, KOPONEN T, PETTIT J, *et al*. NOX: towards an operating system for networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(3): 105-110.

[2] LUCKIE M, MCGREGOR A. IPMP: IP measurement protocol[A]. Passive and Active Measurement Workshop[C]. 2002.

[3] SHANLUNOV S, TEITELBAUM B, KARP A, *et al*. A one-way delay measurement protocol (OWAMP)[R]. Internet Draft, 2003.

[4] HEDAYAT K, KRZANOWSKI R, Morton A, *et al*. A two-way active measurement protocol (twamp)[R]. RFC 5357, October, 2008.

[5] MCKEOWN N, ANDERSON T, BALAKRISHNAN H, *et al*. OpenFlow: enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74.

[6] ZENG H, KAZEMIAN P, VARGHESE G, *et al*. Automatic test packet generation[A]. Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies[C]. ACM, 2012. 241-252.

[7] WUNDSAM A, LEVIN D, SEETHARAMAN S, *et al*. OFRewind: enabling record and replay troubleshooting for networks[A]. USENIX Annual Technical Conference[C]. 2011.
[8] HANDIGOL N, HELLER B, JEYAKUMAR V, *et al*. I know what your packet did last hop: Using packet histories to troubleshoot networks[A]. Proc USENIX NSDI[C]. 2014.
[9] YU M, JOSE L, MIAO R. Software defined traffic measurement with OpenSketch[A]. NSDI[C]. 2013. 29-42.
[10] MOSHREF M, YU M, GOVINDAN R, *et al*. DREAM: dynamic resource allocation for software-defined measurement[A]. Proceedings of the 2014 ACM Conference on SIGCOMM[C]. ACM, 2014. 419-430.
[11] AGARWAL K, ROZNER E, DIXON C, *et al*. SDN traceroute: tracing SDN forwarding without changing network behavior[A].Proceedings of the Third Workshop on Hot Topics in Software Defined Networking[C]. ACM, 2014.145-150.

作者简介：

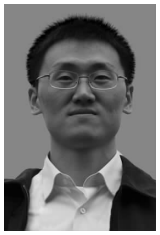
陈鸣（1956-），男，江苏无锡人，解放军理工大学教授、博士生导师，主要研究方向为分布式计算、网络测量、网络管理等。

代飞[通信作者]（1989-），男，四川达州人，解放军理工大学硕士生，主要研究方向为未来网络、网络测量。E-mail：daifei08@163.com。

许博（1980-），男，甘肃兰州人，博士，解放军理工大学讲师，主要研究方向为分布式计算、网络测量等。

邢长友（1982-），男，河南杞县人，解放军理工大学副教授，主要研究方向为未来网络、P2P 多媒体通信等。

李兵（1967-），男，四川金棠人，解放军理工大学副教授，主要研究方向为计算机网络、网络测量等。



张国敏（1979-），男，山东济南人，解放军理工大学讲师，主要研究方向为分布式计算、网络测量等。