doi:10.11959/j.issn.1000-436x.2015061

# OpenFlow 网络测量分析系统的设计实现

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

**摘 要**：OpenFlow 网络目前缺少支持定量测量分析各种创新应用或机制的有效手段。以升级 OpenFlow 网络设备为具有本地日志功能的 OpenFlow 测量实体为基础，设计了一种基于集中式服务器控制测量实体进行分布式测量的机制，制定了其间的通信规程 OpenFlow 测量控制协议(OMCP)，同时基于正则表达式、散列技术和可扩展的统计函数库等方式设计了一种分析测量日志的功能。原型系统的实验表明，OpenTrace 服务器能够灵活部署和控制分布式测量任务，OpenTrace 系统不仅能够定量地重现数据平面的数据流传输过程而且能够重现控制平面的控制事件交互过程，从而可为量化分析 OpenFlow 网络应用和新型机制提供广泛的性能数据。
**关键词**：未来互联网；OpenFlow；分布式测量；集中式控制
**中图分类号**：TP393 **文献标识码**：A

# Design and implementation of a network measurement and analysis system in OpenFlow networks

WENG Xi, CHEN Ming, ZHANG Guo-min, XU Bo, XING Chang-you
(College of Command Information Systems, PLA Univ. of Sci. & Tech., Nanjing 210007, China)

**Abstract:** Nowadays OpenFlow networks lack an effective measurement means supporting for quantificationally analyzing and measuring various innovative applications or mechanisms yet. On the basis of upgrading equipments in OpenFlow networks to be measurement entities which function local log, a mechanism that the measurement entities can carry out the distributed measurement controlled by a centralized server OpenTrace server is designed, and a communications specification called OpenFlow measurement control protocol (OMCP) is established. Meanwhile, a function to analyze the measurement logs, which is based on the regular expression, hash technique and extended statistical function base, is designed. The experimental results of the prototype show that OpenTrace can flexibly deploy and control the distributed measurements; not only the transmission process of dataflow in the data plane but also the interactive process of controlling events in the control plane can recur quantificationally by OpenTrace system, and the comprehensive performance data can be provided for the quantificational analysis of applications and new mechanisms in OpenFlow networks.
**Key words:** future internet; OpenFlow; distributed measurement; centralized control

## 1 引言

在因特网取得极大成功的同时，人们也在关注它现存的流量剧增、安全性和移动性等难题的解决[1]。由于“端到端原则”使因特网核心成为数字传输管道，封闭了各种网络设备的功能，端系统中的应用程序只能通过套接字接口来利用网络的分组传输功能，定制、更新和演进网络的其他功能则变得步履维艰。

计算机软件工作模式的成功为网络变革提供了启示，一种称为软件定义网络(SDN, software defined networking)的网络体系结构应运而生[2]。SDN 的核心概念可以归纳为：硬件平台标准化、控制功能集中化

**收稿日期**：2013-10-30；**修回日期**：2014-08-12
**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(2012CB315806)；国家自然科学基金资助项目(61379149, 61070173, 61103225)
**Foundation Items:** The National Basic Research Program of China (973 Program) (2012CB315806); The National Natural Science Foundation of China (61379149, 61070173, 61103225)

和高层应用开放化。SDN 能够以软件编程的方式控制网络行为，为强化网络功能、缩短网络创新周期和解决因特网难题提供了一种新思路。与此同时，OpenFlow 的问世为研究人员提供了一种将 SDN 理念具体化的网络创新平台，促进了 SDN 技术的发展。

OpenFlow 网络一般由多台 OpenFlow 交换机和一个控制器 2 部分组成。OpenFlow 交换机承担了网络中数据流的转发功能，在充当中央控制节点的控制器上，通过软件编制的逻辑能够定制该网络的行为和功能。例如，由开放网络基金会(ONF, open networking foundation)发布的 OpenFlow 1.4.0 规范[3]指出，OpenFlow 交换机依据内部的多级流表(flow table)和组表(group table)对分组分别进行查询与转发。它通过 OpenFlow 安全通道与控制器连接，并遵循 OpenFlow 协议进行交互。典型的控制器如 NOX[4]，对用户屏蔽了底层网络细节，提供灵活的上层编程接口，能够以软件的形式控制网络行为和监测网络状态，主要表现为对 OpenFlow 交换机进行流表配置与信息查询。

能够在网络环境中支持进行某种网络机制的实验仅是问题的起点。只有通过量化分析才能指导 SDN 技术的科学发展，而有效测量 SDN 是关键的第一步。目前，与 OpenFlow 网络性能测量分析相关的工作可分为对网络设备本身性能的测量和对网络承载流量的测量 2 类。其中对设备本身性能的测量工作有：文献[5]在 Linux 上模拟了 OpenFlow 交换机的数据报转发功能，并与传统路由转发和交换转发的软件模拟方式进行了对比，得到了有关 OpenFlow 转发性能的一些结论；文献[6]采用排队模型对 OpenFlow 交换机和控制器的转发时延、转发速率和分组丢失率进行了测量分析；文献[7]将硬件实现与可扩展的开放式软件架构相结合，从多角度评估具有 OpenFlow 支持的交换机。

本文重点关注对网络承载流量的测量分析。文献[8]利用 OpenFlow 协议已有的统计信息查询接口，结合控制器中的路由信息，分析了从不同交换机实时获取流统计数据以构建全网流量矩阵的网络负载问题。文献[9]利用 OpenFlow 架构的工作特点，通过在交换机安装测量相关的流表项，由控制器读取相应计数器完成如大聚集流量识别等网络异常检测任务。然而，尽管 OpenFlow 协议已提供流统计功能并支持从控制器实时获取流信息[10]，但这种集中式测量方式有如下缺陷:其一，可能导致在控制平面和数据平面分别产生巨大的测量流量，尤其在控制器处形成严重的流量和计算瓶颈；其二，无法测量控制平面的性能；其三，无法掌握控制平面和数据平面间的交互轨迹。为此，需要在 OpenFlow 架构下研究分布式网络测量机制，使其支持对 OpenFlow 应用或新型机制进行测量和分析。

在 OpenFlow 网络的分布式测量机制研究方面，文献[11]提出了 OpenSketch 架构用于网络管控，通过扩展交换机的处理逻辑并向控制器提供灵活的配置接口，由控制器定义交换机的操作方式，由交换机对分组执行过滤、分类与计数处理，从而发现特定的流量模式。该方法以交换机为基本测量单元，能够就预先设定的数据平面网络流量特性进行高效地在线识别。但在 OpenFlow 网络发展尚未成熟，研究尚在初期的现阶段，尤其是在实验环境中对新应用和新机制进行验证评估时，需要更加充足的网络行为信息作为评判依据。综上所述，目前，OpenFlow 网络测量工作大都是基于 OpenFlow 现有集中式测量机制，面向交换机性能评价或明确的建模目标进行，而有关如何设计和构建科学合理、通用的 OpenFlow 网络测量架构的研究未见报道。

本文提出一种基于集中式测量服务器协调测量实体进行分布式测量的机制，以简化全局测量过程；提出一种对聚合测量日志预分析处理的方法，以提高测量数据分析效率。将具有上述机制和方法的 OpenFlow 网络测量分析系统称为 OpenTrace。

## 2 OpenTrace 分布式测量机制

为实现 OpenTrace，首先通过升级 OpenFlow 网络设备（即交换机和控制器）使其增加时钟同步和在本地记录流（控制流和数据流）轨迹的功能，这些升级的网络设备被称为 OpenTrace 测量实体。其次，以实时或待实验结束等方式聚合来自多个测量实体的本地日志，形成一个按时间戳排序的全局日志。这就将 OpenFlow 原先的集中式测量方式转变为分布式测量方式，从而大大降低了像集中式测量方式那样引发性能瓶颈的可能。

OpenTrace 不仅能够获取数据平面流信息，而且能够获取控制平面流信息以及两平面之间的交互信息。具体来说，OpenFlow 交换机中的本地日志包括 5 类流表项统计结果记录：1) 流表项插入；2) 流表项删除；3) 流表项修改；4) 流表项超时，5) 根据预定义周期从流表中读取的流表项。控制器

中的本地日志包括：1) 交换机连接；2) 交换机断开；3) 收到待处理分组；4) 安装流表项；5) 收到流表项超时；6) 删除流表项；7) 收到其他自定义类型分组等记录。为提高处理效率，所有记录最先以特定数据结构存入 OpenTrace 测量实体的内存中；然后根据测量分析模式，这些测量记录将保存在本地文件中或者直接发送给网络指定接收方，即 OpenTrace 服务器；最后以时间戳为关键字段，利用插入排序算法完成聚合，并对聚合的本地日志进行预分析处理。

## 3　OpenTrace 集中式控制机制

由于 OpenTrace 测量实体以分布式方式工作，因此协同位于不同地理位置的众多测量实体完成测量工作是一件十分繁琐的事。尤其是测量实体可能位于不同建筑物中，故必须寻求自动化方式加以协调控制。

### 3.1　OpenTrace 系统体系结构

设置一台 OpenTrace 服务器是一种自然的选择: 服务器通过定制的控制策略对测量实体进行集中式管理，而服务器和测量实体之间交互通信需要一种协议。OpenTrace 系统的构成如图 1 所示。该系统包含 $n$+1 个测量实体、1 台 OpenTrace 服务器和其间的通信协议 OMCP 三部分。其中测量实体除了履行 OpenFlow 交换机或控制器职能外，还具有本地测量生成测量日志的功能，并且能够通过 OMCP 协议与 OpenTrace 服务器通信。应注意到，OpenTrace 遵从了 OpenFlow 的框架，只是在控制平面与数据平面的关键节点上构建新的测量观察点，用于监测评价网络行为。

OpenTrace 服务器具有人机接口，管理员通过定义策略控制下列行为：1) 定义启动/停止本地测量的时机；2) 定义读取流表信息的粒度；3) 确定测量记录存放位置，如暂存本地日志或直接发送到 OpenTrace 服务器上组成聚合日志；4) 确定日志处理方式，如在线处理或离线处理。在适当时候，该服务器通过 OMCP 协议用这些策略来控制测量实体的行为。OpenTrace 服务器的另一项重要任务是预分析处理日志的启动，相关内容将在第 4 节讨论。

OpenTrace 系统支持在线和离线 2 种处理测量结果的方式。当测量频率低且测量流量小，即对网络应用的侵扰可忽略时，OpenTrace 服务器可以要求测量实体直接将本地测量记录发送给它，从而以在线方式直接生成聚合日志，可供其他应用（如网络安全或网络管理）实时利用。而当测量流量可能会对网络应用产生影响或对测量数据没有实时性要求时，可采用离线方式。离线方式则要求测量实体将本地测量记录先存放于本地，待测量任务结束再自动传至服务器，由服务器对多个本地日志进行聚合处理。离线方式可以提供更为精确的网络测量信息，可用于评价新型体系结构或机制、调试新协议和流量特征建模等。不失一般性，下面的讨论以离线方式处理测量结果为背景进行。

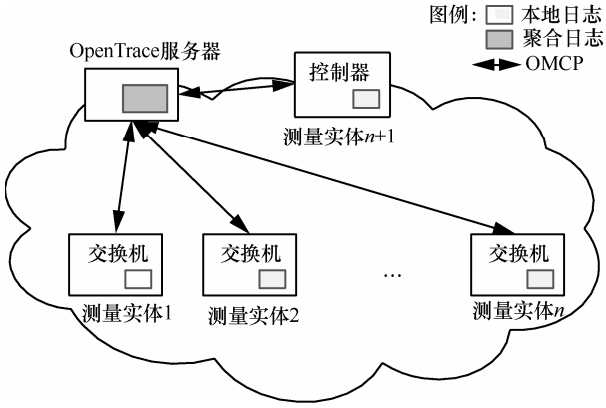


图 1　OpenTrace 系统的构成

### 3.2　OpenTrace 交互模型

为了使 OpenTrace 服务器能够有效地控制测量实体，达到全面、有效获取测量数据的目的，必须精心设计两者之间的交互机制。考虑到该测量过程是 OpenTrace 服务器协同 $n$+1 个测量实体进行本地测量的分布式计算过程，两者的交互可分为 2 个阶段。在阶段 I，$n$+1 个测量实体需要向 OpenTrace 服务器注册并与其在时间上取得同步。仅当所有 $n$+1 个测量实体都已完成上述过程，交互才能进入阶段 II。图 2 和图 3 分别以有限状态机形式给出了 OpenTrace 服务器与测量实体的交互模型。

在图 2 所示的 OpenTrace 服务器有限状态机中，有 4 个状态：等待实体注册、测量就绪、等待测量结束以及等待日志传输。OpenTrace 服务器中的测量事务一经启动便开启注册监听端口，进入等待实体注册状态。对于每个测量实体的注册请求，服务器都将给出包含本地时钟的应答。一旦本测量事务所需的实体均完成注册则进入测量就绪状态(即开始阶段 II)。当完成测量配置并启动测量后，进入等待测量结束状态。待测量结束后，服务器通过发送日志获取报文就能得到各测量实体的本地日志，经

过聚合分析后便完成了此次测量事务，重新进入就绪状态。注意到在测量中，由于存在测量实体的本地日志传输失败、取消测量或测量实体异常退出等事件，在状态机中也存在相应的状态回退或测量事务中止异常。

在图 3 所示的测量实体有限状态机中，有 4 个基本状态和 1 个终止状态。测量实体一经启动便向 OpenTrace 服务器发送实体注册请求，在接收到应答后设置本地时钟并进入等待测量任务状态。若接收到测量任务定义报文，则在解析后进入测量状态。测量完成后，转入等待日志获取状态；当接收到日志获取报文后，传输本地日志文件，此后回到等待测量任务状态。图 3 中也给出了出现异常时应执行的处理，这里不再解释。实现中，为确保测量的可靠性，可设置额外的监视线程在实体或实体代理退出后及时重启该实体或其测量代理，随后自动进行重新注册与测量等过程。

### 3.3 OpenFlow 测量控制协议

为支持 OpenTrace 服务器与测量实体之间的交互，定义的 OpenFlow 测量控制协议包含 4 种报文，主要信息如表 1 所示。

表 1 中的实体注册报文用于测量实体到 OpenTrace 服务器的注册请求以及服务器到测量实体的注册应答过程，承载在 UDP 报文段中。在注册请求报文中，“时钟信息”字段可设为 0。在服务器返回的注册应答报文中该字段放置服务器的时

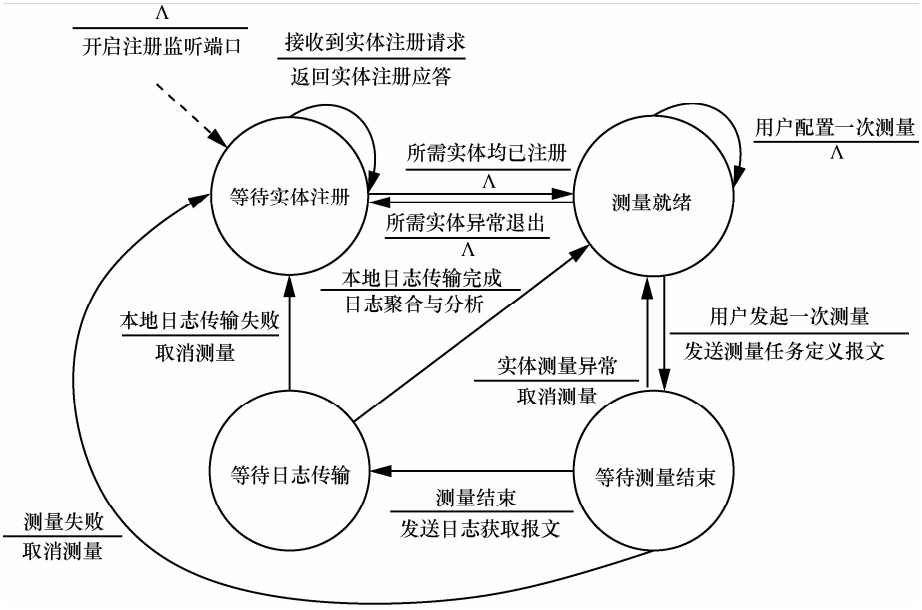


图 2 OpenTrace 服务器有限状态机

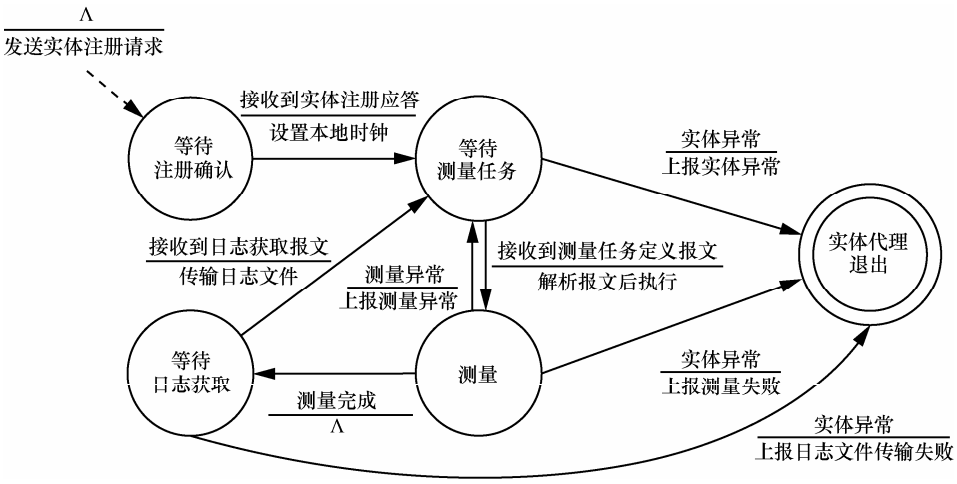


图 3 测量实体有限状态机

**表 1　　OMCP 报文的主要信息**

| 报文类型 | 发送方到接收方 | 主要字段 |
|---|---|---|
| 实体注册报文 | 服务器到测量实体或相反 | 实体名称、实体类型、实体 IP 地址、服务类型和时钟信息 |
| 任务定义报文 | 服务器到测量实体 | 测量任务组名、任务操作类型、测量模式、开始时刻、持续时间、记录间隔和记录类型 |
| 日志获取报文 | 服务器到测量实体 | 测量任务组名 |
| 日志描述报文 | 测量实体到服务器 | 日志文件标志位、当前日志名称、当前日志大小 |

钟信息，测量实体由此调整本地时钟。“实体名称”用于标识测量实体，“实体类型”用于声明测量实体为控制器或 OpenFlow 交换机，“实体 IP 地址”用于 OpenTrace 服务器与其进行的后续通信。“服务类型”表明当前实体支持的服务类型，由于测量任务大都针对特定服务进行，故设置该字段将利于后续选择测量任务组成员。

任务定义报文承载在 UDP 报文段中，用于 OpenTrace 服务器向测量实体分发测量任务及参数。其中，“测量任务组名”用于标识一次测量。“任务操作类型”可为 0、1 和−1(0 表示任务暂停，1 表示任务开始，−1 表示任务取消或删除)。“测量模式”即为在线(0)或离线(1)处理测量结果的工作模式。将“开始时刻”设为 0 表示任务立刻开始，若设为未来某个时刻，则是等待该时刻到时再启动测量。“持续时间”表示测量所要持续的时间，“记录间隔”是指周期性获取流表项统计信息时的时间间隔，而“记录类型”则是定义日志文件中所需记录的信息类型，对 OpenFlow 交换机类型的测量实体来说，该值由 5 个 0/1 数值表示，分别对应于 5 类流表项记录类型，控制器类型的测量实体与之类似。测量实体对服务器回复的任务应答报文未列出，它可为 ACK 或其他错误信息，其中 ACK 字段表示测量实体的当前状态支持此次测量任务的执行，而返回错误信息则与之相反。在整个测量过程中，若测量实体出现异常均将返回错误信息，所有错误信息均由错误标识和错误说明构成，表中略去。

本地日志传输到服务器基于 TCP 连接进行，在离线模式中共包含 2 类报文：服务器向测量实体发送的日志获取报文和反方向的日志描述报文。日志获取报文仅包含“测量任务组名”字段，用于声明服务器想要获取哪次测量的本地日志文件集合。而每一个日志描述报文对应于其中的一个日志文件，“日志文件标识位”取“0”表明当前将要传输的日志是该测量任务中生成的最后一个本地日志，该位取“1”表明还有其他本地日志要传。在日志描述报文中同时包含“当前日志名称”与“当前日志大小”字段，便于服务器做相应处理。对于在线模式则无需经历上述 2 类报文的交互。

## 4　测量日志的预分析处理功能

聚合日志包括了数据平面、控制平面及这 2 个平面间交互的时空测量数据，但若节点多、测量粒度细则测量踪迹记录的数量可能十分庞大，为分析揭示各种 OpenFlow 应用或新型机制规律带来困难。通常可以使用某些专用数学工具完成多种统计分析工作，但考虑到聚合日志中存在大量与分析主题无关的冗余数据，首先需要根据测量踪迹日志的特点提供某些预分析处理功能(如筛选与分类等)，去除原始聚合日志中的冗余信息，从而提升分析效率。

图 4 所示为 OpenTrace 服务器中聚合日志预分析处理功能的模块划分。其中分析模板管理模块用于新建与管理分析模块，而分析模板是指对具有某种规范格式的文本进行特定分析的过程描述，存储在模板库中。该模板定义了分析时所进行操作的序列及操作的约束条件，如筛选操作和筛选条件、分类操作和分类所依据的标准字段以及统计函数类型和统计关键字段等。分析任务生成模块用于创建分析任务实例，主要为选择或配置自定义的模板，该模块输出的配置信息将指导分析任务的执行。分析任务执行模块用于分析聚合日志文件，分析功能包括筛选、分类与统计，具体操作由配置信息定义。其中，对每条记录进行字段内容提取是基于正则表达式。筛选器依据筛选条件对特定流记录进行筛选并按指定字段排序。分类器则基于散列技术，以所选定的分类标准字段的值作为索引完成记录分类。统计器能够根据求均值、极值或相邻差值等统计需求的描述，调用相应的函数接口完成基本的数据统计功能，其统计函数库由自行编写的统计函数构成，支持后期扩展。

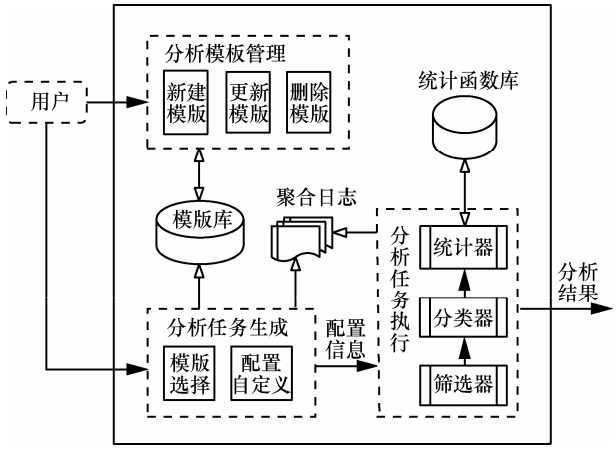


图 4 聚合日志预分析处理功能的模块划分

离线模式下聚合日志预分析处理的过程是:首先，由用户创建分析任务，确定分析的对象与模板；其次，根据用户的任务定义，配置筛选器、分类器和统计器；第三，筛选、分类和统计分析聚合日志中的记录；第四，呈现预分析处理结果。注意到在线模式的主要区别是，需在最初定义测量任务时一并选择或自定义分析模板。

## 5 原型系统实验

### 5.1 实验环境

为验证 OpenFlow 网络测量分析系统的有效性，在如图 5 所示的多楼层实验环境中运行测试了原型系统。其中 3 台 OpenFlow 交换机由运行开源的 Linux 操作系统和 OpenFlow 软件(版本号 1.0.0)的多吉比特以太网卡 PC 机充当，控制器由运行 NOX 程序(版本号为 0.9.1)的 PC 充当。控制器通过带内方式与其他 OpenFlow 交换机互联。

实验环境中的所有 OpenFlow 交换机与控制器均已升级为测量实体。OpenTrace 服务器程序运行在 1 台 PC 机上，通过交换机 B 接入网络。3 台 OpenFlow 交换机分布在不同楼层，其连接方式如图 5 所示。本原型系统采用 Python2.7 编程实现，其中 OpenTrace 服务器提供 GUI 操作接口以及交互过程的控制台显示，其代码约为 300 行。由于测量实体一经启动就无需用户参与，仅提供了报文交互过程的控制台显示，其代码约为 150 行。系统涉及的所有配置信息均采用 Json 字符串存储。

### 5.2 实验过程及分析

为了测试 OpenTrace 系统是否达到了设计要求，设计了一组简单的 TCP 流量竞争实验。实验的基本过程如下：1) 当 OpenTrace 系统准备就绪后，主机 A 经由 OpenFlow 交换机 A、B 和 C 与主机 C 进行持续 100 s 的 TCP 数据传输，该数据流称为流 1；2) 其间，主机 B 经由 OpenFlow 交换机 B 和 C 与主机 C 分别在第 18~28 s、第 43~53 s 和第 79~89 s 3 个时段进行 3 次 TCP 数据传输，每次持续 10 s，该数据流称为流 2。其中 TCP 流量都是由开源 D-ITG 工具[12]产生。下面分阶段讨论 OpenTrace 系统的工作过程。

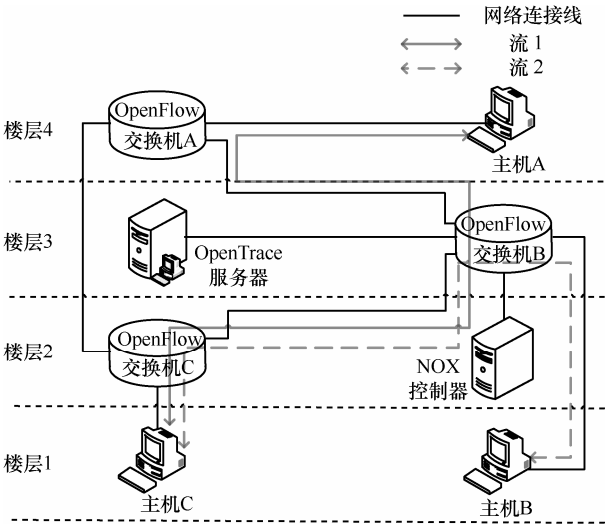


图 5 OpenTrace 原型系统实验环境

**阶段 1** 定义测量任务并实施测量。首先，当 OpenFlow 交换机 A、B 和 C、控制器以及相应的测量代理启动后，这些测量实体分别用 OMCP 报文向 OpenTrace 服务器进行注册并实现时钟同步。此时，可以通过 OpenTrace 服务器的 GUI 界面来定义此次测量任务的参数。例如，对于 TestGroup1 测量任务组的成员而言，测量任务的部分参数设置如表 2 所示。第二，若所有测量实体返回确认，则此次测量任务开始执行；若测量过程中出现任何异常，则此次测量任务将自动中止。当测量开始时，控制器测量实体以事件触发记录，其中参数“记录间隔”无效，参数“记录类型”设为安装流表项与流表项超时；交换机测量实体则以 1 s 为周期，读取流表项信息形成本地日志记录。至此，OpenTrace 系统准备就绪。第三，在测量持续 120 s 后，该测量过程终止。

分析。1) 基于 OpenTrace 服务器的集中式控制，能够方便地协同网络中分布在不同物理位置的测量实体对数据平面和控制平面进行全方位地测量。如果分别对这些测量实体进行控制的话，需要大量的人工交互，并可能产生大量无用的测量记录，也不能适时终止。2) 根据 TCP 的友好特性，流 2 的 3 次

注入都会使流 1 通信流量迅速减少为可用带宽的一半，而当流 2 停止后，流 1 又会恢复到占用全部带宽。希望能够通过对日志进行处理分析重现这一现象。

**表 2　OpenFlow 交换机测量实体的测量任务部分参数**

| 参数 | 值 | 说明 |
|---|---|---|
| 测量任务组名 | TestGroup1 | 标识测量实体集合 |
| 任务操作类型 | 1 | 任务开始 |
| 测量模式 | 1 | 结果离线分析 |
| 开始时刻 | 0 | 立刻开始 |
| 持续时间 | 120 | 持续 120 s |
| 记录间隔 | 1 000 | 间隔为 1 s |
| 记录类型 | 00001 | 仅周期性记录流表项 |

**阶段 2**　本地日志传输与聚合。OpenTrace 服务器向测量任务组中的各测量实体发送日志获取报文，而测量实体提交日志描述报文后便将本地日志传输至服务器； OpenTrace 服务器将 4 个本地日志中的记录按时间戳进行聚合，在服务器上形成一个聚合日志文件。

分析。1) 由于传输和聚合过程是在实验结束后进行的，不会对实验产生额外的测量误差。2) 由于 3 台交换机和 1 台控制器上的所有关键信息都包括在其本地日志上，因此聚合日志应当包含该网络 2 个平面上的所有关键信息。这使重现 OpenFlow 网络中的交互过程成为可能。

**阶段 3**　聚合日志分析与利用。使用 OpenTrace 服务器提供的测量日志预分析处理功能，在选择被分析的文件及其分析模式后，系统将自动处理。通过给出对流 1 和流 2 进行过滤的条件，即按 OpenFlow 1.0.0 规范的 12 元组描述流 1 和流 2 的特征，筛选出聚合日志中与流 1 和流 2 相关的数据平面记录，最终从 3 000 多条记录中得到了关键的 100 多条记录，图 6 显示了由此分析得到的结果。

图 6 中的横轴为时间，纵轴为流量大小(每秒字节数)。可见，在时间为 18~28 s、43~53 s 和 79~89 s 的区间内，流 1 因受到流 2 的干扰而出现了速率下降并在最大值一半的地方两者趋于平衡，而当流 2 消失则流 1 恢复到带宽最大值。分析可知，在第 1 次和第 3 次流 2 加入时，流 1 出现分组丢失使其拥塞窗口变小(检测到多个冗余的 ACK)，继而流 2 调整拥塞窗口为拥塞避免状态。此时，流 1 由慢启动状态逐渐增加拥塞窗口大小，直至流 1 和流 2 相互影响达到平衡。在流 2 第 2 次加入的过程中，图 6 分析结果显示与另外 2 次有明显不同，即流 1 和流 2 均出现短时间流速率为 0 的情况，对应着流 1 和流 2 都在严重拥塞后分组丢失，使两者都由拥塞窗口为 1 开始进入慢启动，直至遵循 TCP 拥塞控制机制达到两者在资源竞争与共享中的均衡。

还可以对控制平面中与流 1 和流 2 相关的行为进行分析，图 7 给出了相关事件的时序分析结果。例如，来自主机 1 的流 1 首个分组到达交换机 A 时被转发给控制器，由控制器在交换机 A、B 和 C 上为流 1 建立相应流表项；而当来自主机 2 的流 2 首个分组到达交换机 B 时也被转发给控制器，由控制器在交换机 B 和 C 上为流 2 建立流表项；最后，由于流 2 的间断性，在流表项空闲计时器超时后，交换机 B 和 C 删除流 2 的表项，而当下次流 2 传输时都将再次请求控制器建立表项等。

图 7 中横轴为时间，纵轴的不同位置表示对聚合日志中控制平面流表项安装与超时删除事件分析的结果。图 7 给出了流 1 和各段流 2 的起始时间。流 1 约开始于第 7 s 并终止于第 107 s，而流 2 分别起始于第 18 s、43 s 和 79 s，终止于 28 s、53 s 和 89 s，这一分析结果与真实发生的过程完全一致。

分析。借助于聚合日志中的分布式测量记录和预分析处理功能，OpenTrace 系统不仅能够记录控制流信息，并能够确定控制流之间的交互过程。

上述实验表明了 OpenTrace 系统的有效性。尽管上述实验环境较为简单，但对于具有更为复杂的环境和应用的同等规模的 OpenFlow 网络而言，本

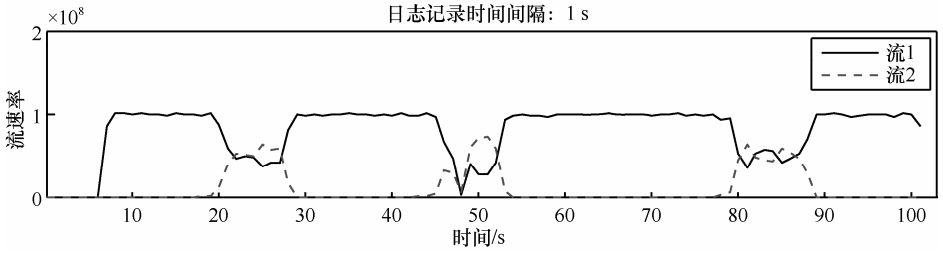


图 6　聚合日志的数据平面分析结果

文采用的分布式 OpenTrace 系统也总会比原先的集中式方案更具优势，也更具扩展性。

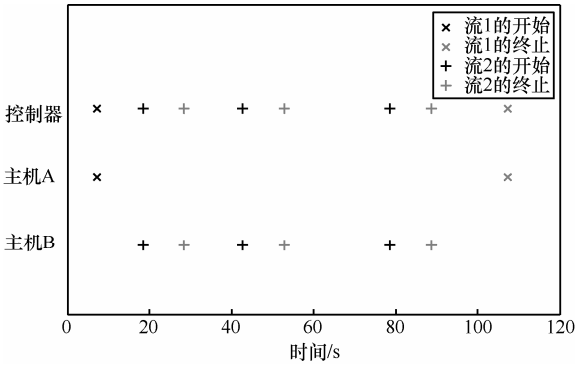


图 7 聚合日志中的控制流分析结果

## 6 结束语

任何网络技术的成熟和科学发展都离不开网络测量手段。OpenFlow 网络与 IP 网络相比有很多差异，这导致 IP 网络中的测量技术无法在 OpenFlow 网络中使用，因此目前 OpenFlow 网络测量是一个正在起步的研究领域。 OpenTrace 能够为 OpenFlow 网络全面测量和量化分析各种新应用或机制提供一种有效手段。它通过 OpenFlow 测量控制协议，以集中式的方式灵活控制测量实体进行分布式协作测量，并提供了自动筛选、排序和分类统计等测量日志预分析处理功能。原型系统的实验结果表明，OpenTrace 服务器能够灵活部署和控制分布式测量任务，OpenTrace 系统不仅能够定量地重现数据平面的数据流传输过程而且能够重现控制平面的控制事件交互过程，从而可为量化分析 OpenFlow 网络应用和新型机制提供必要的性能数据。然而考虑到 OpenFlow 网络及其相关技术尚处于快速发展阶段，当前 OpenTrace 系统的设计实现重点关注更为全面和详细的数据获取，虽能支持实时的日志信息传输，却难免占用较多的网络资源。当 OpenFlow 技术趋于成熟，各种网络应用规范有序运行，对 OpenFlow 网络进行测量的实时性要求也会愈加强烈，因此必须就测量日志信息的高效压缩问题做深入研究。此外，OpenTrace 被动测量技术仍存在如何更好地应用于网络管理、网络优化和网络安全等方面的问题。研究 OpenTrace 随着 OpenFlow 网络规模增大而产生的扩展性问题，以及 OpenFlow 网络中的主动测量技术等，也是下一阶段的研究目标。

## 参考文献：

[1] ROSCOE T. The end of Internet architecture[A]. Proceedings of the 5th Workshop on Hot Topics in Networks[C]. Irvine, CA, USA, 2006.
[2] 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.
[3] ONF. OpenFlow Switch Specification version 1.4.0 (Wire Protocol 0x05)[S]. 2013.
[4] GUDE N, KOPONEN T, PETTIT J, *et al*. Nox: towards an operating system for networks[J]. ACM Computer Communication Review, 2008, 38(3): 105-110.
[5] BIANCO A, BIRKE R, GIRAUDO L, *et al*. OpenFlow switching: data plane performance[A]. IEEE International Conference on Communications(ICC)[C]. Cape Town, South Africa, 2010.1-5.
[6] JARSCHEL M, OECHSNER S, SCHLOSSER D, *et al*. Modeling and performance evaluation of an OpenFlow architecture[A]. Proceeding of The 23rd International Teletraffic Congress[C]. San Francisco, USA, 2011. 1-7.
[7] ROTSOS C, SARRAR N, UHLIG S, *et al*. OFLOPS: an open framework for OpenFlow switch evaluation[A]. Passive and Active Measurements Conference[C]. Vienna, Austria, 2012.85-95.
[8] TOOTOONCHIAN A, GHOBADI M, GANJALI Y. OpenTM: traffic matrix estimator for OpenFlow networks[A]. Passive and Active Measurements Conference[C]. Zurich, Switzerland, 2010.201-210.
[9] JOSE L, YU M, REXFORD J. Online measurement of large traffic aggregates on commodity switches[A]. Proceeding of the USENIX Hot ICE[C]. Boston, MA, 2011.13.
[10] ONF. Software-Defined Networking: the New Norm for Networks[S]. 2012.
[11] YU M, JOSE L, MIAO R. Software defined traffic measurement with OpenSketch[A]. NSDI[C]. Lombard, IL, 2013.29-42.
[12] D-ITG. http://traffic.comics.unina.it/software/ITG[EB/OL]. 2012.

## 作者简介：

**翁溪**（1990-），女，江苏兴化人，解放军理工大学硕士生，主要研究方向为网络管理与测量、软件定义网络、未来互联网。

**陈鸣**（1956-），男，江苏无锡人，博士，解放军理工大学教授、博士生导师，主要研究方向为网络测量、网络性能分析与建模、分布式系统、未来网络。

**张国敏**（1979-），男，山东济南人，博士，解放军理工大学讲师，主要研究方向为网络管理、分布式计算。

**许博**（1980-），男，甘肃兰州人，博士，解放军理工大学讲师，主要研究方向为网络性能测量、网络流量识别、软件定义网络。

**邢长友**（1982-），男，河南杞县人，博士，解放军理工大学讲师，主要研究方向为网络与分布式计算、未来网络、网络流媒体。