西安交通大学

博士学位论文

网络数据平面可编程硬件的研究

学位申请人：乔思祎

指导教师：邹建华 教授

合作导师：邹建华 教授

学科名称：控制科学与工程

2020 年 9 月

Research on Programmable Hardware for Network Data Plane

A dissertation submitted to

Xi'an Jiaotong University

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

By

Siyi Qiao

Supervisor: Prof. Jianhua Zou

Associate Supervisor: Prof. Jianhua Zou

Automation Science and Engineering

September 2020

博士学位论文答辩委员会

网络压缩流量的模式匹配方法研究

答辩人：乔思祎

答辩委员会委员：

西安交通大学教授：嗷嗷 (主席)

西安交通大学教授：宝宝

西安交通大学教授：纯粹

西安交通大学教授：蛋蛋

西安交通大学教授：尔尔

答辩时间：2020 年 12 月 34 日

答辩地点：地点

摘 要

网络通信是支撑当今社会的重要基础设施，当前的发展方向主要集中于建设高性能，高可创新性的网络环境。最近 10 年，软件定义网络 (SDN) 和可编程网络 (SDN2.0) 概念的提出很好的解决了过去网络创新性差、创新难的不足。但随着流量和网络复杂度快速增长，新的网络体系结构也带来了性能和可扩展性两方面的挑战。性能和功能方面: 基于 CPU 的转发平台性能发展逐步减慢，基于 ASIC 的智能网卡硬件可编程性差。可扩展性方面: 数据平面和控制平面分离的 SDN 网络架构带来了稳定性不足和效率低的问题。

将问题从网络的三个维度进行分析：

主机侧网络，在服务器网卡层面，基于 CPU 的智能能网卡的性能难以满足目前虚拟化技术和网络监管细粒度化的发展需求。

交换侧网络，在核心网骨干网层面，基于 ASIC 的转发平面不足以提供网络网络处理的高灵活性，由于其与成本、性能之间平衡困难，网络工程师的创新空间受到了限制。

流表可扩展性研究，硬件流表是一种高效且昂贵的实现网络转发抽象的核心部件，在软件定义网络时代流表稀缺性更加突出。同时，由于流数目和流量的快速增长，控制平面针对流表的操作导致数据平面和控制平面的大量协议开销，导致网络鲁棒性差，易形成安全隐患。近年来，现场可编程门阵列（FPGA）器件快速发展，以可编程硬件技术为首的异构架构已经大量融合到网络领域，带来高用户可定制能力的同时也能保证处理性能。

本文主要探索基于可编程硬件的高性能网络数据平面。本文研究在软件定义网络编程语言内如何将这种可编程硬件抽象层融入整体开发系统，并设计与其配套的控制平面软件和协议，使整体网络系统的软硬件有机结合，增强网络处理性能、灵活性的同时保证安全性。论文从理论抽象分析中提出了体系架构，最后给出了系统实现并进行验证。本文将从以下三方面阐述：

1）研究可编程设备加速主机侧网络方法。本文提出利用基于 FPGA 的智能网卡卸载操作系统层部分网络功能，以达到扩展网络接入层的性能的目的。探讨了不同场景下网络功能的构成，分析并提出一种基于可编程硬件的网络功能定义抽象（Data-Computing，DC 抽象）。本文把服务器网络功能任务中可转化为 DC 抽象的计算密集型功能通过合理转换下放到网卡的 FPGA 可编程器件中。论文基于可编程网卡设计了一套网络流量捕获，统计分析和回放系统。在满足网络功能不受改变的前提下，证明利用基于 FPGA 的智能网卡能有效地提升服务器的网络性能（100x）、抖动 (降低 10^4x) 和效率 (10x)。

2）研究可编程设备加速网络硬件交换层方法。本文提出一种硬件异构型的可编程网络数据平面架构，将 FPGA 与 ASIC 交换芯片有机结合，以增强 ASIC 报文处理报文

的灵活性，同时满足性能需求。论文设计了 ASIC 面向硬件可编程扩展的接口，将数据包头拆分并通过高速数据互联载体发送给 FPGA，利用 FPGA 可重配特性实现完全可编程的报文处理数据平面；同时，本文基于 DC 抽象，将网络随路计算（network-centric computing）模式引入可编程网络体系架构；本文通过分析流量模型在 FPGA 中设计了一种并行化处理单元，在资源消耗可控的前提下大规模提高系统的可扩展性能；另外本文提出了一套基于可编程硬件混合网络架构的软件定义语言编程框架，实现了软件定义需求和可编程硬抽象层分离，以及针对底层数据平面的一种高效自适应的并行单元流分配算法，在可编程性与 FPGA 同等的条件下，比目前 FPGA 交换机性能提升 120x。

3）SDN 硬件流表可扩展性研究。本文针对不同层面网络设备的控制，进行全局优化、分布式优化。在可编程网卡和交换机组成的网络系统中，数据平面内最重要的资源是流表资源（瓶颈资源），本文从全局视野角度，结合可编程硬件的特性，在全网约束的条件下，对流表资源进行优化，以满足未来可扩展性需求。本文分析不同的流量规模和特征，以及系统多模块直接独特的互联协议，提出一种 SDN 网络流表空间全局共享机制。实现了在流量大规模扩展的情形下，保证数据平面稳定性，对受影响的流转发 RTT 时间和安全通道消息风暴数量的优化均达到至少 2 个数量级。

此外，为支持本文提出的相关设计概念，本文实现了一套基于 FPGA 的转发平面设备，包括智能网卡和交换机原型平台。此套平台资源容量大，外设接口丰富，可以满足本文在各类网络架构下实验验证需求。

关 键 词：软件定义网络；网络数据平面；可编程硬件；现场可编程门阵列

论文类型：应用基础

ABSTRACT

英文摘要正文每段开头不缩进，每段之间空一行。

The abstract goes here.

LaTeX is a typesetting system that is very suitable for producing scientific and mathematical documents of high typographical quality.

KEY WORDS: Xi'an Jiaotong University, Doctoral dissertation, LaTeX template

TYPE OF DISSERTATION: Application Fundamentals

目　录

CONTENTS

1 绪论

1.1 研究的背景

1.1.1 研究的意义

21 世纪的新 20 年，网络正以前所未有的速度越来越紧密地参与到民生社会中，对满足国家民生需求、新基建拉动内需和产业升级起到了至关重要的作用。从“百度一下”到网红全民直播带货，从实现“三网通”到发展“新基建”的国家战略，小到优化社会资源效率的办公数字化，大到勾勒出智能交通、智慧城市和万物互联的 5G 海洋，无一不是构建在网络基础设施的快速发展之上。思科公司预计，到 2023 年全球家用互联网总带宽将达到 $5.85Ebps$[①]（是现在的 3.27 倍），移动互联网用户预计达到 57 亿，其总流量可达 $11.3Ebps$ (将达到目前的 5 倍)，其中 5G 流量将占据移动互联网总带宽的 76.5%（0.6%，2019 年）[1-2]。由于深度学习、AI、大数据、云计算、物联网的快速发展，这些新技术将催使新零售、新金融、新医疗、新教育、新制造、云视频和云游戏等行业“云化”，海量的数据会在数据中心内部服务器间网络中及对外网关中交互，这些关键应用将会改变数据中心算力和数据中心内部网络结构特性。

IDC 报告称，2019 上半年中国公有云服务整体市场（Iaas/PaaS/SaaS）达到 54.2 亿美元，并预计在未来 5 年间内以年均复合 46% 的速度快速增长 [3-4]。数据中心内服务器计算力呈现异构化趋势，GPU, AI Chip, FPGA 等使用非通用类型指令集和特殊体系架构计算单元已成为目前分布式计算领域的热点话题。现在超大型数据中心一般可容纳数十万台终端服务器，内部网络链接数量多、拓扑规模大、传送海量数据，这使得现有的网络将变的异常复杂。同时，新的数据包类型层出不穷也使得现有网络变得异常脆弱。

传统网络技术已经无法满足当前的网络环境的需求，最近十年来网络技术和架构经历了快速地演进和变革，针对数据中心网络尤为明显。传统网络的互连包含了经典的二三层网络。为增强交换机的扩展能力，二层网络增加了广播，桥接等复杂功能。这种网络架构在小规模应用时可以展现强大的智能性与可扩展性，但当网络规模进一步增加，网络中容易出现的广播风暴、链路收敛等一系列尖锐问题变得难以解决。现代的大型网络设计思想摒除了略显冗余看似小聪明的功能设计，事先规划好网络拓扑层次，完整地保留网络的第三层，从而将网络扁平化。网络拓扑结构演化出可进行大规模扩展的 CLOS 型架构，为了降低系统复杂性，在各个层次之间的网络设备功能也逐步变得统一透明。网络设备统一化，可降低网络功能开发部署的难度。通常，研究人员需要持续地投入对网络进行测量、监控、容错、提升效能的工作。由于思想的创新和技术的推进，设备厂商不断开发出具备各种高级功能交换芯片。设备、芯片功能强大的同时，复杂的网络功能不断地对于网络的管理层又提出了新的挑战。

[①] $1Ebps = 10^6 Tbps = 10^{18} bps$

为解决设备制造复杂和设备管理复杂的问题，软件定义网络（Software Defined Network，SDN）概念的提出拨开了笼罩在网络体系结构发展道路上的迷雾。SDN 将数据平面和控制平面解耦。在数据平面上，对数据包的处理统一做查找-转发（Match-Action）抽象。控制平面复杂建立网络拓扑，控制并下发流表。这样所有的数据包转发行为都又控制平面的软件逻辑完成，数据平面可以支持任意一种网络协议的处理。由于软件具有强大的灵活性以及开发的敏捷性，SDN 大大加速了网络创新和智能化进程。数据平面和控制平面的安全通道由 OpenFlow 协议进行规范，将数据平面统一化、简单化，使得网络交换设备向白盒化方向发展。

大规模网络无论是在底层设备架构还是运维方式上仍不能停止变革的脚步，这为可编程硬件的发展带来了巨大空间。随着云服务概念和大规模机器学习的落地，近年来以云计算为代表的数据中心网络规模指数增长。网络功能虚拟化在数据中心内部是关键一环。虚拟交换机则是主机内各虚拟机之间数据包转发的核心软件。随着众核 CPU 架构快速发展，服务器内虚拟机布置资源大幅扩张，促使主机出口吞吐量从 40GbE 向 100GbE 甚至 400GbE 演进。不但如此，复杂的网络安全规则、流量监控等模组进一步导致 CPU 过多地消耗在处理网络功能上面。研究人员可能需要花费大量时间去解决目前网络架构的大规模扩展的方案。（创新变的异常艰难）。传统 x86CPU 架构适合于处理灵活多变的计算控制任务，对于做重复、常规流式数据处理，通用指令集架构并不能得到最优的效率，厂商往往不得不依靠大量部署 server 来解决。为缓解主机内 CPU 消耗过大，目前提出的智能网卡是一种新思路。智能网卡采用 FPGA，网络处理器，ARM 等器件，或以他们的组合形式形成在网卡端的新的算力集合，这种算力集合对于处理网络流量会有更高的效率。我们可以把转发动作，网络安全规则等功能下放进来，以削减服务器 CPU 的额外消耗。ASIC 具有最好的性能和最高的能量效率，但每次大批量的部署消耗时间长，投入研发资金大。对于运营商来说，设备、仪器等一次性支出都叫做 CapEx（Capital Expenditure，资本性支出）。对于目前快速发展的网络环境架构，设备的更新换代周期也在变短，在优化 CapEx 时已经不能把固定设备投入当做一次性支出。在探索新一代网络架构时，CapEx 也会成为重要的参考因素。

随着创新性和需求的进一步发展，让底层硬件拥有灵活的可控制能力才能满足目前行业变革的需求。因此，网络领域提出了编程协议无关（Programming Protocol-Independent Packet Processors，P4）概念。P4 协议不但支持 SDN 网络控制和管理的可编程性，还提出了数据平面可编程的概念。数据包在数据平面内的处理模型遵循解析-查找-匹配的抽象模式。P4 规定了一种编程语言 [5]，它可以控制数据平面对数据包的任意解析行为，也可以自由配置查找表的数据位宽和多级流表之间的查找流水线 [6-7]。这种更高阶的数据平面可编程模型使交换机设备更加白盒化，交换机与任意网络协议解绑，带来了具备灵活性的创新实践。除此之外，端到端大带宽、低时延的网络需求引申出了网络功能硬件卸载、网络随路计算等概念，这进一步增强了对高性能的网络数据平面可编程性的需求。

综上所述，现代网络在向软件定义、数据平面可编程的方向发展。网络架构的变迁

的核心是有一套可以映射上层可编程逻辑的硬件数据平面。本文主要探索一种面向网络数据平面的可编程硬件，能够满足快速迭代的网络创新性需求，同时能够提供与目前主流设备相仿的处理性能，以及可扩展性高的全局优化方法。

1.1.2 技术简介

软件定义网络的基本设计概念是将数据平面与控制平面分离 [8-9]。其中，网络数据平面是指完成计算机之间通信数据包的匹配、修改、传送、转发的软硬件设备。数据平面的可编程性要求网络管理员拥有对数据平面的各个特性做快速个性化定制。网络的控制平面维护全网视野数据，配置针对流的转发条目，控制平面中的应用程序几乎都由软件构成。当前数据平面的设计思想如图1-1所示，主要有软件方法实现，专用硬件实现和新设计的可编程硬件。

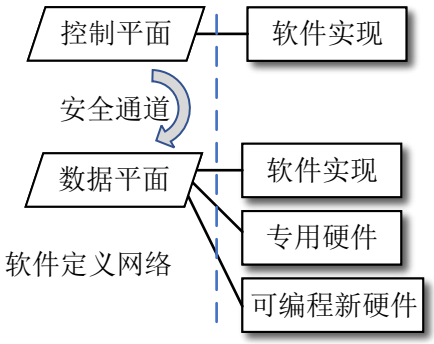

图 1-1 软件定义网络结构及其实现方案

在不同场景下，网络对于数据平面的需求千差万别，研究人员一般根据场景的流量大小，处理过程复杂度来思考并选取数据平面的实现方案，本文将在第2章详细介绍各类数据平面的实现方案的优缺点，并着重于可编程性的分析。目前两种最重要的数据平面是“软件交换机”和“专用硬件交换机”。两者在功能上都是针对数据包做一系列处理，包括匹配、查找、统计、传送、转发和安全校验等等，其中“流表”是实现数据平面核心功能的函数（器件）。数据平面、都包含一个可以与远端控制器沟通的软件代理，这部分功能着重于通信协议的实现以及通道安全性加解密，主要由轻量级通用处理器完成。其二者的主要区别在于处理数据包的性能以及交换容量。数据包处理性能主要看数据吞吐量（字节每秒）和包吞吐量（包每秒），目前软件交换机做高性能的包转发几乎可以达到 60G/60Mpps[10]。当数据包处理复杂度增加时，软件交换机的性能会直线下降，几乎与操作步骤数成反比。专用硬件交换机有接口数目多，交换容量大的特点，一般能满足 64 口乘以每口 25$Gbps$ 的总交换容量。而且硬件交换机的性能与数据包处理步骤几乎无关，它拥有良好的性能稳定性，低转发时延等特性。虽然在核心网络和高性能网关领域主要使用硬件交换机，但是硬件交换机的功能固定，更换成本高昂，如果需要修改网络或者更改网络功能那么选用专用硬件交换机的场景将无从下手。所以目前在数据中心网络或服务器 NFV（网络功能虚拟化）等场景中，软件交换机依然占据很大份额。由于软件交换机的灵活性高，开发人员能够快速迭代部署新功能，且

传统单机 CPU 通信速率需求不高，软件交换机尚能满足在数据处理时延高、吞吐率低的前提下，提供足够的可编程灵活性。但随着人工智能领域、5G 的发展，数据中心网络内通信容量需求快速增长，转发时延需求快速收紧，软件交换机性能瓶颈快速到来，将不得不面对大量无谓堆叠 CPU 的情形。本文将主要侧重于研究主机侧网络和核心交换网络中使用可编程硬件来大大提高交换机的性能瓶颈。针对控制平面，本文将从单点优化开始用分布式优化和全局优化的思想，实现对网络中的瓶颈资源（如流表资源）的可扩展性和安全性提升。

1.1.3 国内外应用与研究现状

为增强数据平面的可编程性，工业界学术界互相促进、广泛研究并已经提出了许多方案。

1）基于软件的数据平面

这类技术着重于开发便捷，价格低廉，无需在网络中部署专用设备场景，是快速实现功能的首选方案。目前在虚拟化的云服务系统中，已经部署了大量基于软件的功能：a）转发层，华为 CE1800V[11] 是专为数据中心云计算虚拟化环境部署的一种分布式虚拟交换机。其支持标准 Open Flow1.3 控制协议，以及 Open vSwitch 数据库管理协议（OVSDB），基于英特尔 DPDK（Data Plane Development Kit）技术提供每核 12Gbps 的转发吞吐，比业界平均水平高出 20%。b）流量监管，Activelogic[12] 是一个提供安全可靠、流量分类、提高 QoE（Quality of Experience）能力的网络管理工具。它基于软件可自动化部署，依靠超大规模性能、人工智能技术以及云计算场景优化的能力，在数据平面解决流量监管的问题。基于软件的数据平面功能可以依靠堆叠 CPU 核数来实现大规模的性能扩展，但由于计算复杂度过高、基于指令的图灵机在高速内存共享和海量数据处理场景中效率低下，即使简单转发的性能达到 100Gbps 线速也需要占用 8 个核心以上 [10, 13]。综上所述，我们发现单纯地依靠软件处理器扩张来增加网络性能边界收益将越来越小。

2）基于白盒交换机和 P4 专用芯片的数据平面

在网络性能方面大幅超越基于通用服务器的 NFV 数据平面 [5, 13]。符合 OpenFlow 规范的白盒交换机可将控制平面移交给远端软件层，从而大幅提升设备的再开发能力，在 DDoS 防护、负载均衡等基础网络转发设备的智能化和可定制化方面给出了比较好的灵活性。阿里巴巴在其云计算网络场景中，通过可编程硬件交换机和通用服务器结合来实现公有云的网关服务。此架构既享受到芯片带来的网络转发性能提高（6.4Tbps，400ns 延迟）和可编程能力带来的网络功能快速部署迭代，又能实现软件所擅长的复杂网络调度功能 [14]。这样同时兼顾了性能、灵活性，在大规模扩展网络体系结构时达到降低成本，满足业务需求和简化网络架构同时提升服务稳定性。数据平面可编程芯片提供了硬件层面上的可编程包头抽取器、可编程流表以及可编程执行器，他们的设计思想是依靠快速查表（TCAM，SRAM）法，或经过后期编程选取特定的冗余逻辑模块（在 ASIC 芯片内部的空间上堆叠的可编程单元）法，来完成专用电路（ASIC）的直接

描述逻辑 [6, 15]。不过这类可编程芯片架构提供的可编程执行器是不完备的，前后堆叠的流表限制了流表的宽度、深度范围，会造成逻辑资源浪费以及流水线处理延迟过长。同时，ASIC 设计定型之后无法增加新的用户特性（状态转发、随路计算、监测计数和包调度特性），导致这类 P4 专用芯片的可编程性是大大受限的。

3）基于 FPGA 的自主设计的数据平面

现场可编程门阵列（FPGA）是一种灵活性可以与软件媲美可编程硬件，性能和效率与专用硬件比较接近。现代高速度云架构依赖于每个专用硬件（ASIC）网络节点的支持，随着网络功能需求多变与复杂化，ASIC 类型的网络处理芯片已经不能提供足够的可编程性，然而 CPU 核心无法提供高的处理性能。业界已经开始将网络堆栈向基于 FPGA 的自研网卡中卸载 [16-17]。为了推广可编程硬件，学术界牵头推出了基于 FPGA 的智能网卡开源项目 NetFPGA[18]，业界龙头企业 Xlinx、Intel 等也纷纷推出了基于 MPSoC/FPGA 的自适应计算加速平台 Alveo[19-20] 系列智能网卡和 N3000[21]。目前，基于 FPGA 的可编程数据平面已经广泛应用在 5G 接入边缘网络 [22]、数据中心计算存储 [23]、核心网络低延迟加速器 [24] 以及高性能高可靠性高安全性的数据中心防火墙 [25] 加密通信 [26] 等领域。FPGA 的高灵活性由全可编程的逻辑门带来，目前一般用硬件描述语言 Verilog、VHDL 等开发。一个合格的硬件工程师的培养周期要远大于软件工程师，这也是目前网络领域硬件卸载最难所在。为了解决这种不足业界也推出了一系列类似 C 语言的高层次综合工具 HLS[27]，但使用这类工具必须学习 1000 多页的开发文档 [28]。并不是所有代码都可以直接被工具转译，而且还需要考虑到硬件细节，降低 FPGA 资源消耗；需要自主决定并行区块；需要在代码中融入这种编译器的特性标记字符，总体来看，目前并没有从本质上改善对硬件编程的困难程度。除此之外，由于在 FPGA 中复杂逻辑对并行总线宽度的时延敏感度高，一个大型工程的主频一般不会超过 200MHz，即使每个时钟节拍都可以处理一个数据包，那么 FPGA 流水线在处理最小包时的最高吞吐量也只有 134Gbps[②]，这对进一步需求性能的核心网包交换场景也形成了瓶颈。

1.2 研究内容

本文主要探索基于可编程硬件的高性能网络数据平面。论文提出基于可编程硬件的网络数据平面，对主机侧网络和交换层网络的数据平面实现加速，并研究在软件定义网络（SDN）概念下控制平面对全网核心流表资源的全局优化方法。如图1-2所示，论文把操作系统软件网络堆栈的大负载的网络存储和计算功能向网卡硬件卸载，利用 FPGA 与交换芯片使能交换网络数据平面的高性能高可编程性，把数据平面的主机侧网络、交换层网络的普通转发设备替换为具有硬件可编程特性的网络设备。流表资源是网络转发数据包的核心指令依据，本文基于软件定义网络控制面数据面分离的特点，对全网的流表资源进行了全局效率、可扩展性和安全性优化。

1）研究可编程设备加速主机侧网络方法

[②] 134Gbps=200Mpps*(64+20)*8bits

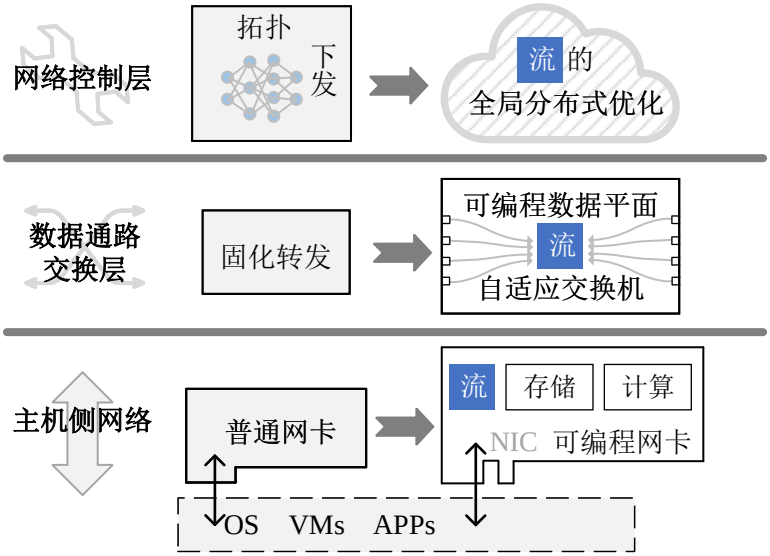

图 1-2 基于可编程硬件的 SDN 数据平面研究框架

本文提出利用基于 FPGA 的智能网卡卸载操作系统层部分网络功能，以达到扩展网络接入层的性能的目的。探讨了不同场景下网络功能的构成，分析并提出一种基于可编程硬件的网络功能定义抽象（Data-Computing，DC 抽象）。本文把服务器网络功能任务中可转化为 DC 抽象的计算密集型功能通过合理转换下放到网卡的 FPGA 可编程器件中。论文针对网络流量捕获，统计分析和回放等功能场景，将其功能利用 DC 抽象方法，合理化地卸载到硬件网卡。在满足网络功能不受改变的前提下，证明利用基于 FPGA 的智能网卡能有效地提升服务器的网络性能、时延和效率。

2）研究可编程设备加速网络硬件交换层方法

本文提出一种硬件异构型的可编程网络数据平面架构，将 FPGA 与 ASIC 交换芯片有机结合，以增强 ASIC 报文处理报文的灵活性，同时满足性能需求。论文设计了 ASIC 面向硬件可编程扩展的接口，将数据包头拆分并通过高速数据互联载体发送给 FPGA，利用 FPGA 可重配特性实现完全可编程的报文处理数据平面；同时，本文基于 DC 抽象，将网络随路计算（network-centric computing）模式引入可编程网络体系架构；本文通过分析流量模型在 FPGA 中设计了一种并行化处理单元，在资源消耗可控的前提下大规模提高系统的可扩展性能；另外本文提出了一套基于可编程硬件混合网络架构的软件定义语言编程框架，实现了软件定义需求和可编程硬抽象层分离，以及针对底层数据平面的一种高效自适应的并行单元流分配算法，可以稳定实时地保障系统交换层的高性能。

3）SDN 硬件流表可扩展性研究

本文针对不同层面网络设备的控制，进行全局优化、分布式优化。在可编程网卡和交换机组成的网络系统中，数据平面内最重要的资源是流表资源（瓶颈资源），本文从全局视野角度，结合可编程硬件的特性，在全网约束的条件下，对流表资源进行优化，以满足未来可扩展性需求。本文分析不同的流量规模和特征，以及系统多模块直接独

特的互联协议，提出一种 SDN 网络流表空间全局共享机制。实现了在流量大规模扩展的情形下，保证数据平面稳定性，降低系统中关键通信通道失效风险。

1.3 关键科学问题

1）精度高、性能可扩展性强的软件网络流量功能卸载方法

面对当前数据量庞大复杂的操作系统网络环境，业界一般会使用专门的软件传输加速工具库（例如，DPDK[29]）, 也会使用到例如 SR-IOV[30] 的专有硬件加速。新一代的网卡还会支持 VXLAN、GENEVE 等封装技术的卸载，同时基于硬件的使远距离直接内存访问（RDMA[31]）大有取代 TCP 协议栈的趋势。然而这些基于固定转发平面的卸载技术只能将虚拟化的转发层或者 TOE（TCP Offloading Engine[32]）卸载下去得到硬件加速，对于一些基于随路流量的有状态计算、并行计算以及灵活的流量工程却依然难以享受硬件加速带来的优势。目前基于 FPGA 硬件可编程网卡同时提供了高性能收发和足够强大的灵活性已经可以满足主机侧网络的性能需求，为更复杂功能的卸载提供了有力支持 [33-34]。如何利用可编程网卡实现高精度、高性能保障的网络功能硬件卸载，并且提出网络功能抽象、合理部署、合理划分任务是本文要解决的第一个问题。

2）高资源利用率、高动态性的高性能硬件可编程数据平面设计方法

在云、服务器–客户端的计算网络体系结构下，由于新兴的内容应用（社交，虚拟/增强，混合现实）以及工业网络应用（移动性，大数据，机器学习）导致网络追求高的实时性、可扩展性和可靠性。网络设备数量和多样性随着数据中心、边缘设备的发展而壮大，因此，现在学界对交换层、核心网场景快速创建灵活解决方案的需求也愈发强烈。可编程数据平面交换机拥有很高的灵活性，可以快速重新定义新的数据包处理协议，为应对新形态网络发展提供了良好前景。其有三类典型设计架构但目前都存在缺陷：1）软件交换机性能普遍低下，2）基于 ASIC 的交换机无法拥有完全可编程性，3）基于 FPGA 的交换机资源有限，交换性能无法满足业界需求。综上所述，本文第二个研究问题：如何设计一款转发性能强，而又拥有硬件可编程性的交换机设备？如果这种设备所需求的资料是目前产业界无法提供的，有没有一种对现有设备进行科学合理的具有最小改动可能性的方法？如何实现高资源利用率、高灵活性的高性能硬件可编程数据平面设计方法？

3）流表关键资源的全局优化方法

网络数据包的转发动作依赖于数据平面内查找表的匹配结果，SDN 架构下亦是如此，当前 SDN 数据平面内将网络数据包的处理流程抽象为 Match-Action（匹配-执行）。在此基础上还交换机内增加了多种匹配域、多级流表结构，绝大多数平台中都视转发表为最核心以及成本占用最大的模块。以 OpenFlow 协议为代表，为更好的服务动态的新流，一般规定控制器与交换机之间流表安装流程为 Reactive 模型：交换机收到一条新流首先会上报控制器，随后控制器计算路径并下发流表到数据平面设备。基于硬件

的高性能 TCAM（三态内容地址查找表）拥有单周期流水、掩码匹配等优秀性能，然而昂贵的价格使得用户无法购置容量足够大的表。因此，交换机内极易引发流表溢出的现象，若此时新流到达此交换机节点并按照 Reactive 模型处理，由于可能需要频繁更替活跃流表内容，这会进一步直接引发控制平面和数据平面之间安全通道的消息风暴，否则会造成丢包或服务任务中断等异常现象。本文第三个研究问题：如何在维持交换机中原有流表容量的前提下，缓解流表溢出所带来的危害？在保持 SDN 网络平面分离优点的条件下，如何利用其全局化优势高效利用网络设备资源？

1.4 主要研究成果

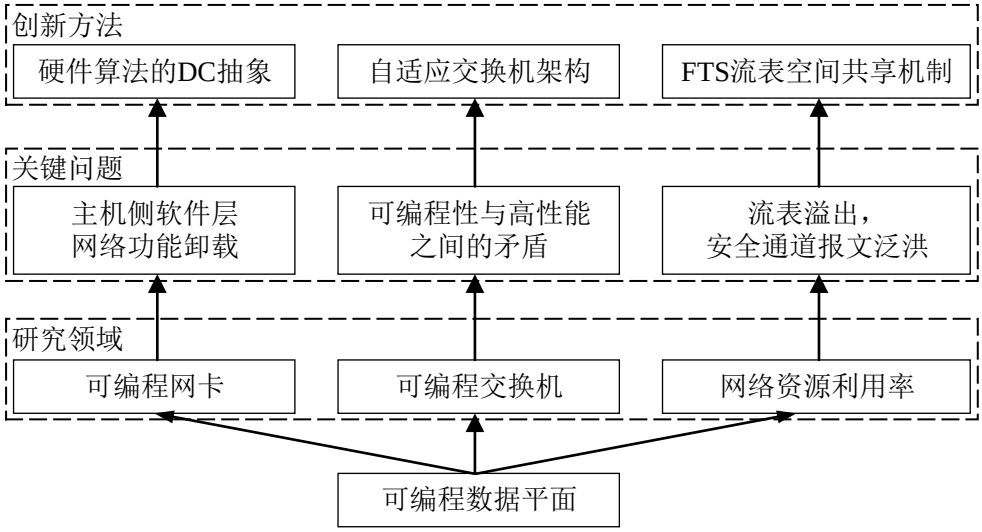

图 1-3 论文主要研究内容以及成果

论文针对可编程网卡卸载有状态计算和流量工程、基于 FPGA 可编程硬件性能不足，流表溢出威胁风险大网络资源利用率低等问题展开分析和创新方法设计。如图1-3具体研究成果概况如下：

1）提出了针对流量随路计算的网络功能卸载抽象模型

本文提出一种适用于网络功能硬件卸载的抽象模型：数据—计算抽象（DATA—COMPUTING, DC 抽象）。根据 DC 抽象，分离软件中适用于硬件加速的繁杂计算，使原本经 X86 计算架构需要频繁访存的任务，转换到硬件中做流水线式流计算，可在不影响功能精度的前提下，释放 CPU 资源，大规模扩展性能，提升系统效率。同时，再配合数据包的分类—查找抽象（Classification—Matching，CM 抽象），论文在可编程硬件的网卡中实现了更高精度、更高性能、资源利用率更好的流量捕获-统计-回放应用。在满足网络功能不受影响的前提下，证明利用可编程硬件能使原有软件性能效提升 100x、抖动降低 4 次方数量级、能源效率提升 10x。

2）提出了 FPGA 与交换芯片（Switching ASIC）结合的自适应交换机架构

本文提出一种高性能的可重配交换层数据平面架构：自适应交换结构（Adaptive

Switch, AS）。通过 FPGA 与交换芯片联合的设计思想，AS 架构将 FPGA 的高灵活性与交换芯片的强大性能同时对外表现。论文在前述 DC 抽象的基础上，继续研究 FPGA 可编程硬件中高度并行的大规模性能扩展方法。为了保证 FPGA 低资源消耗，论文设计了一种基于硬件的灵活负载均衡机制。综上，AS 架构解决了 FPGA 性能差与资源少的限制，与交换芯片的有机连接更进一步增强了 AS 架构的整体性能。综上在可编程性与纯粹 FPGA 等同的条件下，论文将目前基于 FPGA 的可编程数据平面性能提升 120x。

3）提出了一种针对流表资源不足场景下的网络内流表共享机制

网络转发层核心资源不足的问题，本文提出一种全局流表共享方法（Flow Table Sharing, FTS）。本文分析目前 OpenFlow 协议中有关 Table-Miss（流表缺失）的处理过程，并论证即使单纯依靠增加流表容量的资源堆叠方案，并不能使流表溢出的概率降低为零。本文在维持 SDN 网络控制面悬离特性不变的前提下，提出新的 Table-Miss 处理机制。FTS 方法通过控制器层面、交换机数据层面的软硬件联合设计方法，使得新的 Table-Miss 机制能实现对原先受影响的转发流量 RTT 时间和安全通道消息风暴数量的优化均达到至少 2 个数量级，并且能够容易回退、向下兼容现阶段的传统方案。

1.5 论文组织结构

根据主要研究内容的讨论，本文的组织结构安排如下：

第 2 章对相关工作进行调研，主要介绍网络中主要数据平面，以及其可编程化发展趋势，分析应对网络软件定义化的主要挑战。

第 3 章网络计算、流量工程卸载方法。

第 4 章自适应交换机，可编程数据平面，网络交换层。

第 5 章网络资源全局优化方法，table-miss 处理方法。

第 6 章总结。

2 数据平面可编程综述

虽然本模板不讲解 LaTeX 的详细使用方法，但是为了方便大家使用本模板撰写论文，本章对论文写作中经常用到的 **图、表、公式**等内容的排版方法做一个简单介绍。

3 参考文献格式

参考文献格式应符合国家标准 GB/T-7714-2005《文后参考文献著录规则》。中国国家标准化管理委员会于 2015 年 5 月 15 日发布了新的标准 GB/T 7714-2015《信息与文献参考文献著录规则》。因为二者的差别非常小，所以采用了新的标准。标准的 BiBTeX 格式网上资源非常多，本模板使用了李泽平开发的版本，该版本提供了多种参考文献的排序规则。学校博士论文规范指定了两种排序方法：一是按照文献的引用顺序进行排序，二是按照作者姓氏加出版年份进行排序。本模板采用第一种排序规则，第二种排序规则的使用方法请参考文献 [36]。

致　谢

致谢中主要感谢导师和对论文工作有直接贡献和帮助的人士和单位。致谢言语应谦虚诚恳，实事求是，字数不超过 1000 汉字。

用于盲审的论文，此页内容全部隐去。

参考文献

[1] 华为公司年报[EB/OL]. 2019. https://www-file.huawei.com/-/media/corporate/pdf/annual-report/annual_report_2019_cn.pdf.

[2] 思科公司互联网发展跟踪白皮书（2018-2023）[EB/OL]. 2019. https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html.

[3] 国际数据公司（IDC）. 上半年中国公有云市场[EB/OL]. 2019. https://www.idc.com/url.do?url=/getdoc/pdf_download.do?containerId=prCHC45634819&position=15&transactionId=39032154&term=&page=5&perPage=100.

[4] IREASEARCH. 中国公有云服务市场跟踪[EB/OL]. 2020. http://news.iresearch.cn/yx/2020/02/315730.shtml.

[5] BOSSHART P, DALY D, GIBB G, et al. P4: Programming protocol-independent packet processors [J]. ACM SIGCOMM Computer Communication Review, 2014, 44(3): 87-95.

[6] BOSSHART P, GIBB G, KIM H S, et al. Forwarding metamorphosis: Fast programmable match-action processing in hardware for sdn[J]. ACM SIGCOMM Computer Communication Review, 2013, 43(4): 99-110.

[7] HONDA M, HUICI F, LETTIERI G, et al. mswitch: a highly-scalable, modular software switch[C]// Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research. [S.l.: s.n.], 2015: 1-13.

[8] 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.

[9] CASADO M, MCKEOWN N, SHENKER S. From ethane to sdn and beyond[J]. ACM SIGCOMM Computer Communication Review, 2019, 49(5): 92-95.

[10] SHAHBAZ M, CHOI S, PFAFF B, et al. Pisces: A programmable, protocol-independent software switch[C]//Proceedings of the 2016 ACM SIGCOMM Conference. [S.l.: s.n.], 2016: 525-538.

[11] HUAWEI. 1800V 虚拟交换机[EB/OL]. 2018. https://carrier.huawei.com/~/media/CNBG/Downloads/Product/Fixed%20Network/b2b/0920/1800-en.pdf.

[12] SANDVINE. Hyperscale data plane for next generation telco networks[EB/OL]. 2020. https://www.sandvine.com/hubfs/Sandvine_Redesign_2019/Downloads/2020/Datasheets/Network%20Optimization/Sandvine_DS_ActiveLogic.pdf.

[13] CENTEC. Hybrid v580 sdn switch[EB/OL]. 2019. http://www.centecnetworks.com/cn/DownView.asp?ID=2272&SortID=153.

[14] 高山渊, 蔡德忠, 赵晓雪, 等. 企业数字化基石-阿里巴巴云计算基础设施实践[M]. 北京市海淀区: 电子工业出版社, 2020.

[15] BAREFOOT. Second generation of world's fastest p4 programmable ethernet switch asics[EB/OL]. 2020. https://www.barefootnetworks.com/products/brief-tofino-2/.

[16] LU G, GUO C, LI Y, et al. Serverswitch: a programmable and high performance platform for data center networks.[C]//Nsdi: volume 11. [S.l.: s.n.], 2011: 2-2.

[17] FIRESTONE D, PUTNAM A, MUNDKUR S, et al. Azure accelerated networking: Smartnics in the public cloud[C]//15th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 18). [S.l.: s.n.], 2018: 51-66.

[18] ZILBERMAN N, AUDZEVICH Y, COVINGTON G A, et al. Netfpga sume: Toward 100 gbps as research commodity[J]. IEEE micro, 2014, 34(5): 32-41.
[19] XILINX CO. L. [EB/OL]. 2020. https://www.xilinx.com/about/company-overview.html.
[20] XILINX. Smartnics for diverse workloads[EB/OL]. 2020. https://www.xilinx.com/applications/data-center/network-acceleration.html#smartnics.
[21] INTEL. Fpga programmable acceleration card n3000 for networking[EB/OL]. 2020. https://plan.seek.intel.com/psg_WW_psgcom3_LPCS_EN_2019_PACN3000ProductBrief.
[22] INTEL. 5G 前传边缘网络 FPGA（IP）方案[EB/OL]. 2020. https://plan.seek.intel.com/5GFrontHaulGatedFormCN_LP?erpm_id=8235613&erpm_id=8235613&elq_cid=6511651.
[23] XILINX. Stand alone nvme-of acceleration solution[EB/OL]. 2020. https://www.xilinx.com/publications/solution-briefs/partner/nvme-of_solutionbrief.pdf.
[24] INTEL. SDN/NFV 低延迟 GRE 处理加速器[EB/OL]. 2020. https://www.intel.cn/content/dam/altera-www/global/zh_CN/pdfs/literature/wp/low-latency-gre-processing-accelerator-evaluation-cn.pdf.
[25] INTEL. 电信解决方案 FPGA PAC N3000 助力在云环境中实现大容量 DDoS 防护[EB/OL]. 2020. https://www.intel.cn/content/dam/www/programmable/cn/zh/pdfs/literature/solution-sheets/sb-high-capacity-ddos-protection-in-cloud-environments-cn.pdf.
[26] INTEL. 英特尔 FPGA 可编程加速卡 N3000 的 IPSec 加速解决方案[EB/OL]. 2020. https://www.intel.cn/content/dam/altera-www/global/zh_CN/pdfs/literature/solution-sheets/sb-accelerating-ipsec-arrive-technology-intel-fgpa-pac3000-cn.pdf.
[27] XILINX. Vivado high-level synthesis accelerates ip creation by enabling c/c++ and system c specifications[EB/OL]. 2020. https://www.xilinx.com/products/design-tools/vivado/integration/esl-design.html.
[28] XILINX. Vivado hls documentation[EB/OL]. 2020. https://www.xilinx.com/products/design-tools/vivado/integration/esl-design.html#documentation.
[29] INTEL. Data plane development kit[EB/OL]. 2020. https://www.dpdk.org/.
[30] VMWARE. Single root i/o virtualization(sr-iov)[EB/OL]. 2019. https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.networking.doc/GUID-CC021803-30EA-444D-BCBE-618E0D836B9F.html.
[31] MELLANOX. Remote direct memory access(rdma)[EB/OL]. 2019. https://community.mellanox.com/s/global-search/rdma.
[32] MICROSOFT. Information about the tcp chimney offload, receive side scaling, and network direct memory access features[EB/OL]. 2008. https://support.microsoft.com/en-us/help/951037/information-about-the-tcp-chimney-offload-receive-side-scaling-and-net.
[33] XILINX. Alveo u250 data center accelerator card[EB/OL]. 2020. https://www.xilinx.com/products/boards-and-kits/alveo/u250.html.
[34] NETFPGA. A line-rate, flexible, and open platform for research, and classroom experimentation. [EB/OL]. 2020. https://netfpga.org/site/#/about/.
[35] VETTERLI M, KOVACEVIC J, GOYAL V K. Foundations of signal processing[M]. Cambridge: Cambridge University Press, 2014.
[36] LEE Z. GB/T 7714-2015 参考文献 BiBTeX 样式[EB/OL]. 2016. https://github.com/zepinglee/gbt7714-bibtex-style.

附录 A 公式定理证明

附录编号依次编为附录 A，附录 B。附录标题各按一级标题编排。附录中的图、表、公式另行编排序号，编号前加“附录 A-”字样。这部分内容非强制性要求，如果论文中没有附录，可以省略。

排版数学定理等环境时最好给环境添加结束符，以明确定理等内容的起止标志，方便阅读。官方模板未对这些内容进行规范，本模板中定义的结束符采用 ◊，例子的结束符采用 ⧫，定理的结束符采用 □，证明的结束符采用 ■。

定义 A.1 (**向量空间**)： 设 X 是一个非空集合，$\mathbb{F}$ 是一个数域 (实数域 $\mathbb{R}$ 或者复数域 $\mathbb{C}$)。如果在 X 上定义了加法和数乘两种运算，并且满足以下 8 条性质：

1. 加法交换律，$\forall\, x, y \in X$，$x + y = y + x \in X$；
2. 加法结合律，$\forall\, x, y, z \in X$，$(x + y) + z = x + (y + z)$；
3. 加法的零元，$\exists\, 0 \in X$，使得 $\forall\, x \in X$，$0 + x = x$；
4. 加法的负元，$\forall\, x \in X$，$\exists\ -x \in X$，使得 $x + (-x) = x - x = 0$。
5. 数乘结合律，$\forall\, \alpha, \beta \in \mathbb{F}$，$\forall\, x \in X$，$(\alpha\beta)x = \alpha(\beta x) \in X$；
6. 数乘分配律，$\forall\, \alpha \in \mathbb{F}$，$\forall\, x, y \in X$，$\alpha(x + y) = \alpha x + \alpha y$；
7. 数乘分配律，$\forall\, \alpha, \beta \in \mathbb{F}$，$\forall\, x \in X$，$(\alpha + \beta)x = \alpha x + \beta x$；
8. 数乘的幺元，$\exists\, 1 \in \mathbb{F}$，使得 $\forall\, x \in X$，$1x = x$，

那么称 X 是数域 $\mathbb{F}$ 上的一个**向量空间** (linear space)。

例 A.1 (**矩阵空间**)： 所有 $m \times n$ 的矩阵在普通矩阵加法和矩阵数乘运算下构成一个向量空间 $\mathbb{C}^{m\times n}$。如果定义内积如下：

$$\langle A, B\rangle = \mathrm{tr}(B^H Q A) = \sum_{i=1}^{n} b_i^H Q a_i \tag{A-1}$$

其中 a_i 和 b_i 分别是 A 和 B 的第 i 列，而 Q 是 Hermite 正定矩阵，那么 $\mathbb{C}^{m\times n}$ 构成一个 Hilbert 空间。 ⧫

定理 A.1 (Riesz **表示定理**)： 设 H 是 Hilbert 空间，H^* 是 H 的对偶空间，那么对 $\forall\, f \in H^*$，存在唯一的 $x_f \in H$，使得

$$f(x) = \langle x, x_f\rangle, \qquad \forall\, x \in H \tag{A-2}$$

并且满足 $\|f\| = \|x_f\|$。 □

证明： 先证存在性，再证唯一性，最后正 $\|f\| = \|x_f\|$。 ■

附录 B 算法与代码

对于数学、计算机和电子信息专业，算法和代码也是经常用到的排版技巧。

B.1 算法

算法描述使用 algorithm2e 宏包，效果如算法 B-1 所示。

Input: $\mathbf{x}(k), \quad \mu, \quad \mathbf{w}(0)$
Output: $y(k), \quad \varepsilon(k)$
for $k = 0, 1, \cdots$ **do**
  $y(k) = \mathbf{w}^H(k)\mathbf{x}(k)$ // output signal
  $\varepsilon(k) = d(k) - y(k)$ // error signal
  $\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\varepsilon^*(k)\mathbf{x}(k)$ // weight vector update

算法 B-1 LMS 算法详细描述

B.2 代码

源代码使用 listings 宏包，LMS 算法的 Verilog 模块端口声明如代码 B-1 所示。

代码 B-1 空时 LMS 算法 Verilog 模块端口声明

```
module stap_lms
#(
parameter     M          = 4,    // number of antennas
              L          = 5,    // length of FIR filter
              W_IN       = 18,   // wordlength of input data
              W_OUT      = 18,   // wordlength of output data
              W_COEF     = 20    // wordlength of weights
)(
output signed [W_OUT-1:0]   y_i,  // in-phase component of STAP output
output signed [W_OUT-1:0]   y_q,  // quadrature component of STAP output
output                      vout, // data valid flag of output (high)
input         [M*W_IN-1:0]  u_i,  // in-phase component of M antennas
input         [M*W_IN-1:0]  u_q,  // quadrature component of M antennas
input                       vin,  // data valid flag for input (high)
input                       clk,  // clock signal
input                       rst   // reset signal (high)
);
```

攻读学位期间取得的研究成果

研究成果包括以下内容：

1. 已发表或已录用的学术论文、已出版的专著/译著、已获授权的专利按参考文献格式列出。
2. 科研获奖，列出格式为：获奖人 (排名情况). 项目名称. 奖项名称及等级, 发奖机构, 获奖时间.
3. 与学位论文相关的其它成果参照参考文献格式列出。
4. 全部研究成果连续编号编排。

用于盲审的论文，只列出已发表学术论文的题目和刊物名称，可以备注自己为第几作者，及期刊影响因子。

学位论文独创性声明 (1)

本人声明：所呈交的学位论文系在导师指导下本人独立完成的研究成果。文中依法引用他人的成果，均已做出明确标注或得到许可。论文内容未包含法律意义上已属于他人的任何形式的研究成果，也不包含本人已用于其他学位申请的论文或成果。

本人如违反上述声明，愿意承担以下责任和后果：

1. 交回学校授予的学位证书；
2. 学校可在相关媒体上对作者本人的行为进行通报；
3. 本人按照学校规定的方式，对因不当取得学位给学校造成的名誉损害，进行公开道歉；
4. 本人负责因论文成果不实产生的法律纠纷。

论文作者 (签名)：　　日期：　　年　　月　　日

学位论文独创性声明 (2)

本人声明：研究生_____ 所提交的本篇学位论文已经本人审阅，确系在本人指导下由该生独立完成的研究成果。

本人如违反上述声明，愿意承担以下责任和后果：

1. 学校可在相关媒体上对本人的失察行为进行通报；
2. 本人按照学校规定的方式，对因失察给学校造成的名誉损害，进行公开道歉；
3. 本人接受学校按照有关规定做出的任何处理。

指导教师 (签名)：　　日期：　　年　　月　　日

学位论文知识产权权属声明

我们声明，我们提交的学位论文及相关的职务作品，知识产权归属学校。学校享有以任何方式发表、复制、公开阅览、借阅以及申请专利等权利。学位论文作者离校后，或学位论文导师因故离校后，发表或使用学位论文或与该论文直接相关的学术论文或成果时，署名单位仍然为西安交通大学。

论文作者 (签名)：　　日期：　　年　　月　　日

指导教师 (签名)：　　日期：　　年　　月　　日