

利用 SDN 实现可编程无线局域网多径接入管理^①



朱 格, 李明政, 刘 童, 童浩杰, 田 野

(中国科学技术大学 计算机科学与技术学院, 合肥 230027)

通讯作者: 田 野, E-mail: yetian@ustc.edu.cn

摘 要: WiFi 接入点 (Access Point, AP) 被广泛部署在学校、商场和机场等公共场所, 满足用户随时随地的使用需求. 然而, 管理大规模的 WiFi 网络仍是一个亟待解决的难题, 设备间的差异性造成难以在无线局域网中提供集中的接入管理. 同时, 由于各种新服务、新应用的推动, 使得传统无线局域网中的网络资源利用率低及接入点负载不均衡的问题日益严重. 因此, 本文引入 SDN 思想, 将协议无关转发 (POF) 技术应用到无线局域网中, 提出一种深度可编程无线局域网架构, 实现集中接入管理. 同时, 综合考虑信号强度 (RSSI)、实时网络流量等因素, 提出一种基于多路径 TCP (MultiPath TCP, MPTCP) 的无线 AP 接入管理机制, 充分利用各 AP 的带宽资源, 避免在拥塞的路径上继续传输数据, 从而均衡底层 AP 负载.

关键词: SDN; WLAN; 集中接入管理; POF; MPTCP

引用格式: 朱格, 李明政, 刘童, 童浩杰, 田野. 利用 SDN 实现可编程无线局域网多径接入管理. 计算机系统应用, 2019, 28(9): 125-132. <http://www.c-s-a.org.cn/1003-3254/7075.html>

Exploiting SDN for Multi-Path Access Management in Programmable WiFi Networks

ZHU Ge, LI Ming-Zheng, LIU Tong, TONG Hao-Jie, TIAN Ye

(School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China)

Abstract: WiFi Access Points (AP) are widely deployed in public places such as schools, shopping malls, and airports to meet the network access demands of users at any time from any location. However, managing a large-scale WiFi network is still a difficult problem to be solved, and the diversity of devices makes it difficult to provide centralized access management in WLAN. Meanwhile, due to the promotion of various new network services and applications, the problem of low utilization of network resources in traditional WLAN and unbalanced load of access points is becoming more serious. In this study, we adopt the idea of Software-Defined Network (SDN) and apply the Protocol Oblivious Forwarding (POF) technology in the WLAN network. We propose a deep programmable WLAN architecture to realize centralized access management. Moreover, considering the factors such as Received Signal Strength Indicator (RSSI) and real-time network traffic, we design a wireless AP access management mechanism based on MultiPath TCP (MPTCP). It can make full use of the bandwidth resources of each AP, avoid the continuous transmission of data on the congested path, and hence achieve the balance of work loads of substrate APs.

Key words: SDN; WLAN; centralized access management; POF; MPTCP

目前, WiFi 作为重要的无线接入方式, 其以便利、无处不在等特点为终端设备的网络访问提供了极大的

便利, 在人们的生活和工作中占据着举足轻重的地位. 然而, 不同生产厂商研制的 AP 设备在协议标准、操作

① 基金项目: 国家自然科学基金面上项目 (61672486); 国家科技重大专项子课题 (2017ZX03001019-004)

Foundation item: General Program of National Natural Science Foundation of China (61672486); National Science and Technology Major Program (2017ZX03001019-004)

收稿时间: 2019-03-08; 修改时间: 2019-04-02; 采用时间: 2019-04-08; csa 在线出版时间: 2019-09-05

系统、使用方式等方面都存在很大的差异,考虑到终端设备的多样性和用户的移动性,无线局域网中集中接入管理仍是一个亟待解决的难题。

同时,随着移动互联网的发展,无线数据业务的应用呈现指数增长,尤其是社交网络、媒体共享和智能终端等应用的快速发展,使得 WiFi 也面临着性能瓶颈。由于 WiFi 技术的局限性,特别是终端用户需求的多样化,使得很难灵活动态的满足终端客观需求,特别是相邻 AP 信号覆盖的区域可能存在重叠,位于信号重叠区域的用户仅依靠 RSSI 作为选择 AP 的衡量标准,会导致某些 AP 负载过大,而有些 AP 长时间处于空闲状态,使得网络中的负载分布不均衡造成网络资源的浪费^[1-4]。这种问题主要是有物理资源和分配策略两个方面造成,一方面是接入点带宽资源有限,即物理资源不足;另一方面是没有充分利用有限的带宽资源,这也就是所谓的资源分配策略问题。因此,如何利用有限的资源为终端用户提供更好的服务、避免拥塞,这是一个很有价值的研究方向。

软件定义网络^[5-9](Software Defined Network, SDN)技术的出现,给无线局域网带来了新的机遇和挑战,同时也带来了新的挑战和机遇。通过将网络控制平面和数据平面的分离,SDN 对底层网络设备进行抽象,为上层应用提供简单高效的编程接口,将管理操作集中在控制器。SDN 构建一种开放的可编程网络体系结构,通过编程的方式实现报文解析和数据转发,彻底打破硬件设备对数据平面的限制,提供开放的网络接口。POF^[10-12](Protocol-Oblivious Forwarding)作为数据平面可编程的代表性技术,通过部署底层多级流水线实现特定的网络策略。

近年来,将 SDN 技术与无线局域网技术相结合取得了可观的研究成果。如 OpenRadio^[13]通过模块化实现对数据平面的编程,支持 WiFi 和 LTE 等不同的无线网络协议,使得配置和重构协议得到了简化。不同于 OpenRadio, Odin^[14,15]不需要对 IEEE802.11 标准进行修改就可以兼容现有的无线设备,Odin 通过轻量级虚拟接入点(Lightweight Virtual Access Point, LVAP)与物理 AP 解耦合,使得物理 AP 对控制器来说是透明的。文献[16]提出基于 SDN 的无线局域网架构,该架构提出一种针对终端用户无感知的移动切换机制。

因此,将 SDN 技术应用到无线接入点中,使得 AP 设备具有可编程性将极大地促进网络服务革新,对

于用户的接入和迁移过程的研究具有创新意义。本文引入 POF 技术,实现终端设备的集中接入管理,同时结合多路径 TCP (MPTCP)^[17-19]技术,提出多路 AP 集中接入管理机制,提高网络资源利用率,避免网络拥塞。

1 架构介绍

在 WiFi 网络中,不同的终端应用具有多样化的 QoS 需求,每个终端可能同时运行一些具有不同服务需求的应用程序,不同的应用需要的网络资源也不尽相同(如带宽资源),这会造成 AP 之间负载不平衡;同时,无线网络带宽通常具有局限性和不稳定性。鉴于以上问题,本文通过引入基于 MPTCP 的无线 AP 接入管理机制,实现多个 AP 同时为终端设备服务,在满足终端设备对高带宽的需求,提高整个网络的资源利用率的同时,还能平衡拥塞,避免因 AP 负载过重导致的通信中断或效率低下等问题。

本文采用基于 MPTCP 的 AP 接入管理机制,实现细粒度的多路径传输,提高网络资源的利用率。如图 1 所示,本研究将可编程的概念拓展到无线 AP,构建一种基于 POF 的可编程无线局域网架构,实现对终端设备的统一接入管理,同时引入面向用户级的虚拟 AP 的概念(Virtual Access Point, VAP),负责实际的流量传输。

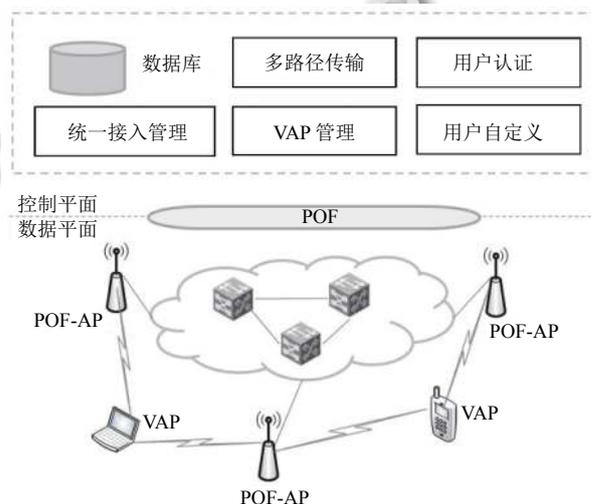


图 1 基于 POF 的可编程无线局域网架构

该架构分为基于 POF 的控制平面和数据平面,控制平面包含统一接入管理、多路径传输、VAP 管理、用户认证和用户自定义等多个模块;数据平面由多个兼容 POF 的无线 AP(POF-AP)组成。POF-AP 根据流

表规划将数据帧和管理帧分别交给不同的流水线处理,所有的管理帧都交由位于控制平面的控制器统一处理。

1) 集中式控制器: 控制器统一处理探测、认证和关联等所有的管理帧, 并负责 POF-AP 中流表的生成独配置, 消除了不同设备间的差异性。

2) VAP: 引入 VAP 概念是为了简化网络功能的实现, VAP 是与终端用户相关联的逻辑 AP。当终端用户和终端设备 VAP 的分配; 此外, 通过将 OpenWrt^[20] 系接入 AP 时, 控制器为其分配 VAP, 终端用户所有的网络流都将由相应的 VAP 进行处理, VAP 以流表的形式存在。对于终端用户而言, 与其关联的 VAP 可以同时关联多个 POF-AP; 当然, 单个 POF-AP 的网络资源(如带宽等)也可以被多个 VAP 共享, 即 VAP 和 POF-AP 之间是多对多的映射关系。统刷入商用路由器, 控制器不需要对每个设备进行单

3) POF-AP: 兼容 POF 技术的 AP, 利用 POF 技术对无线数据帧进行处理, 并以二元组 {offset, length} 的形式描述协议字段, 其中 offset 表示协议字段到数据字段的偏移量, length 表示协议字段的长度。所以, 处理数据帧时, 只需提取相应的协议字段, 不需要预知关于协议格式的任何信息, 然后根据流表匹配并执行相应的操作, 从而实现协议无感知转发。

FIS (Protocol-oblivious Instruction Set) 作为 POF 协议无关报文处理指令集, 包含编辑操作和数据转发两种操作。在指令执行阶段, 编辑操作实现协议头部的添加和删除, 数据转发实现数据包的发送和流表间的跳转。

4) 多路径传输: 在该网络架构中利用 MPTCP 对数据进行分流, 使得不同的子流通过不同的路径(即不同的 AP) 传输, 实现终端用户同时连接多个 AP, 平衡无线设备间的负载, 避免网络拥塞。

2 集中接入过程

基于上文提出的可编程无线局域网架构, 在设备接入 AP 时, 所有的管理帧都由集中式管理器统一处理, 该管理器和所有与之相连的无线 AP 一起构成可编程无线局域网, 在该无线局域网覆盖的范围内, 被认证的用户可以享受随时随地的网路服务。具有接入管理功能的控制器和具有转发功能的无线 POF-AP 协同工作, 完成终端设备的接入操作。

在控制器的管理下, 终端设备接入时具体的交互

过程如图 2 所示。

1) 控制器周期性的向终端设备发送无线信号, 统计与之相连的每个 AP 上承载的虚拟 VAP, 通过查找数据库获取与 VAP 相关联的终端用户, 用其对信标帧中的单播地址进行填充, 经由物理 AP 发向终端。

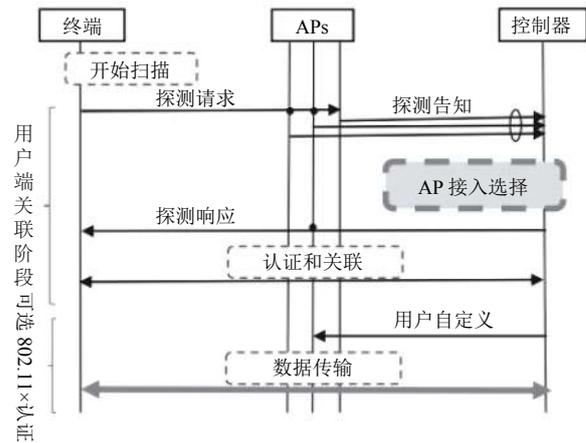


图 2 终端设备接入过程

2) 由于终端用户和 VAP 之间是一对一的映射关系, 并且 VAP 间是相互隐蔽的, 所以采用终端用户发送探测请求帧的方式主动寻找网络。通过上述方式, 所有在终端设备通信范围内的 AP 都会收到该探测请求帧, 并上报给控制器。控制器对首次接入的终端用户进行身份验证, 同时为终端设备构造并初始化 VAP, 并构建包含虚拟网络名称、认证方式、传输速率等信息的探测响应帧。根据 AP 接入选择算法, 控制器为终端用户选取一组 AP 进行后续的交互。

3) 选取 AP 后, 终端设备的认证请求帧和关联请求帧会依次发出, 并由选取的 AP 上报给控制器, 控制器生成响应帧反馈给终端用户。

4) 在加密认证过程中, 基于四次握手原则进行密钥信息交互, 控制器和终端设备进行加密方式的协商。如接入的虚拟网络不需要加密, 则直接跳过该操作; 如采用 WPA2 方法加密, 则双方需安装 GTK 和 PTK。

5) 接着, 控制器将 IP 获取请求发给无线局域网的 DHCP 服务器, 终端接入设备获得 IP 地址。至此, 终端设备完成接入过程。随后控制器将之前为终端设备构造的 VAP 配置到物理 AP 上, 同时, 所有与终端设备有关的流表也会随之下发, 接下来终端设备和网络之间的所有数据传输将通过 AP 完成。

然而,在传统的终端设备接入过程中,只存在终端设备和 AP 之间的信息交互,不采用集中式控制,不存在 AP 与控制器之间的交互过程;与传统无线网络不同,本研究中设备接入必须在控制器的参与下完成,AP 与控制器之间存在多次信息交互过程,这会造成接入耗时增加.针对该问题,做出以下优化:

在优化前,接入过程中探测、认证、关联和转发配置每步都需要上报控制器,相比较传统的接入过程,该方案中 AP 与控制器之间的多次的信息交互造成接入时延较长.为了进一步优化该问题,这里提出一种改进的接入方案,如图 3 所示,通过减少 AP 与控制器之间的交互,进而降低接入时延.具体来说,在探测阶段,与优化之前该阶段的操作一致,控制器初始化虚拟 VAP,构造探测响应帧,根据 AP 接入选择算法选举 AP,将响应帧发出;接下来,控制器预先下发配置好的指令集到选出的物理 AP 上,以便后续的认证、关联和转发配置等接入过程无需再上报控制器,而是由 AP 生成响应帧反馈给终端.优化后的接入方案,能够在一定程度上降低接入时延.

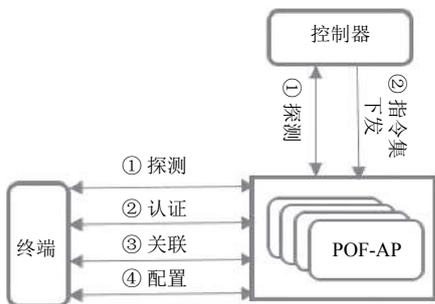


图 3 终端设备接入过程改进

3 集中接入策略

3.1 AP 性能指标

当终端用户接入 AP 时,按照传统的接入方式仅依靠 RSSI 来衡量 AP 的性能,并不能保证 AP 间的负载均衡.如在由多个 AP 覆盖的区域按照 RSSI 值选择 AP 接入,会导致某些 AP 因接入较多的用户而导致拥塞,而其他 AP 却长时间处于空闲状态.另外,有学者提出依靠终端接入量来衡量 AP 负载,但是这也并不能有效的反应 AP 的真实负载,如某个 AP 接入了大量通信量极低的用户,此时 AP 的负载并不重;相反,当 AP 接入的终端数量并不多,但是通信量很大时,AP 的负载

反而会比较重.

如何有效的衡量 AP 实时的负载状况,使得新加入的终端能够择优选择 AP 接入,使得 AP 间负载尽可能均衡,避免拥塞状况,提高整个网络系统的性能.由于终端设备的移动性和无线信道的不稳定性,使得网络流量波动较大,所以在考虑 AP 实时负载时,不仅要考虑 RSSI,还要从网络实时流量的角度综合考虑.

1) RSSI

RSSI 作为影响 AP 性能的重要指标之一,与终端设备接收距离的关系如下:

$$RSSI = -(10n \log 10d + A) \quad (1)$$

其中, n 为环境衰减因子, d 为终端与 AP 间的距离,用户与 AP 的距离越近接收到的 RSSI 值就越大,相应的数据传输质量也就越好.研究表明,在数据传输时,信号强度与抗干扰能力呈正相关,AP 的信号强度越高,表示其能提供的负载能力越强.

2) AP 吞吐量

该研究中 AP 和终端设备之间是多对多的映射关系, $ST = \{S_1, S_2, \dots, S_m\}$ 表示与 AP 相关联的 m 个终端设备组成的集合,而每个终端即为 AP 的一个子流.因此,在衡量 AP 吞吐量时要综合考虑集合 S 中全部终端(即每个子流)的吞吐量.

而对于 AP 中的每个子流 S ,丢包率和传输时延是影响其吞吐量的关键因素.丢包率过高也就意味着发送数据帧的时间会延长,相应的 AP 能够获得的吞吐量也会下降.

在 802.11DCF 架构中,AP 成功发送 MAC 数据帧的时间 T_t 为:

$$T_t = DIFS + \frac{RTS}{R} + SIFS + \frac{CTS}{R} + SIFS + \frac{DATA}{R} + SIFS + \frac{ACK}{R} + T_b(n) \quad (2)$$

其中, $DIFS$ 和 $SIFS$ 分别为分布式帧和短帧时间间隔, RTS 和 CTS 分别为请求和允许发送的控制帧大小, $DATA$ 和 ACK 分别为数据帧和确认帧大小, R 为传输率, T_b 为平均回退时间:

$$T_b(n) = \begin{cases} \frac{2^k(W_{\min} + 1) - 1}{2} T_s, & (0 \leq n \leq 6) \\ \frac{W_{\max}}{2} T_s, & (n > 6) \end{cases} \quad (3)$$

其中, W_{\min} 和 W_{\max} 分别为竞争窗口的最大和最小值,

T_s 为时隙时间.

并且,数据的丢包率 $loss_p$ 主要受 MAC 数据帧的错误率 E_p 和碰撞率 C_p 的影响.

$$loss_p = (E_p + C_p - E_p C_p)^N \quad (4)$$

其中, N 为数据帧最大重传次数.

丢包率 $loss_p$ 直接影响成功发送 MAC 数据帧的平均时延 \bar{T}_t , 从而影响吞吐量的大小.

$$\bar{T}_t = \frac{T_t}{1 - \sqrt[N+1]{loss_p}} \quad (5)$$

因此, AP 子流 S 的吞吐量 S_{tps} 为:

$$S_{tps} = \frac{DATA}{\bar{T}_t} = \frac{DATA \cdot (1 - \sqrt[N+1]{loss_p})}{T_t} \quad (6)$$

假设, AP 中每个子流的吞吐量都能达到饱和状态, 这时 AP 的吞吐量将达到理论上的峰值 TPS_{max} :

$$TPS_{max} = \sum_{S \in ST} \max\{S_{tps}\} \quad (7)$$

3) 负载比

从 AP 的角度考虑, 在 AP 接入选择时, 不仅要根据实际网络状况考虑 AP 理论的吞吐量峰值 TPS_{max} , 还要考虑 AP 当前实时的吞吐量 TPS_{cur} . 为了更好的衡量 AP 的实时负载状况, 本文引入负载比 ϕ 作为 AP 接入选择的一个重要参数:

$$\phi = \frac{TPS_{cur}}{TPS_{max}} \quad (8)$$

3.2 基于 MPTCP 的接入算法

基于 MPTCP 协议, 终端设备可以同时接入多个 AP, 通过在通信中使用多条链路, 将数据流分发到多条路径上进行传输, MPTCP 可以实现网络带宽利用率的提高和网络拥塞的平衡.

MPTCP 在一条 TCP 连接中建立多条子流, 当一条子流按照三次握手的方式建立起来后, 可以按照同样的方式建立其他子流, 这些子流都绑定于 MPTCP 会话连接. 子流可以通过两种方式构建, 一种是当 IP 相同时, 利用不同的端口号建立子流; 另一种是利用多个网络接口, 即不同 IP 建立多个子流. 文中利用第一种方式建立子流, 每个子流与一个特定 AP 建立连接. 根据负载比 ϕ 对 AP 实时负载进行评估, 按照性能择优选取一组 AP. AP 的选取过程主要有以下步骤:

1) 根据 RSSI 值, 选择大于阈值 λ 的 AP, 组成

AP 候选集 $A = \{AP_1, AP_2, \dots, AP_n\}$, 其中, n 表示满足要求的 AP 总数.

通常 RSSI 值在 0~50 dbm 时, 信道的传输质量较佳, 为了获得较好的数据传输效率和用户体验, 本文设定阈值 $\lambda = -40$ dbm.

2) 然后, 通过计数的方式获取 AP 的实时吞吐量 TPS_{cur} , 综合考虑网络的实际状况和 AP 的实际性能, 将 AP 理论吞吐量峰值 TPS_{max} 作为 AP 的负载阈值, 通过式 (8) 计算每个候选 AP 的负载比, $\phi = \{\phi_1, \phi_2, \dots, \phi_n\}$ 为候选集 A 中 AP 相应的负载比.

3) 最后, 根据负载比对候选集 A 中的 AP 进行排序, 选取 ϕ 值较小的 k 个 AP:

即 $A_{select} = \{A_1, A_2, \dots, A_k\}$ 为选取的 k 个 AP, 其中, k 为终端设备子流总数, 即需要接入的 AP 总数. 按照大需求小负载比的原则, 即有大流量需求的子流接入有小负载比的 AP, 终端设备依次与选取的 AP 相关联.

因此, 基于 MPTCP, 提出 AP 接入选择算法, 算法 1 描述了整个接入选择过程.

算法 1. AP 接入选择算法

- 1) 当终端设备接入 APs 时, 通过探测响应帧获取 AP 的 SSID、RSSI 等信息, 同时更新控制器中相应的信息;
- 2) 根据得到的 RSSI 值, 选取 RSSI 值高于阈值 λ 的 AP 作为候选集 A ;
- 3) 根据吞吐量峰值 TPS_{max} 和实时吞吐量 TPS_{cur} , 评估候选 AP 的负载比 ϕ ;
- 4) 选取负载比 ϕ 较小的 k 个 AP, 即 $A_{select} = \{A_1, A_2, \dots, A_k\}$, 与终端设备进行关联;
- 5) 更新控制器中 AP 相应的 RSSI 值和吞吐量等信息;
- 6) 当集合 A_{select} 中 AP_i 的负载超过负载阈值 (即 $\phi \geq 1$) 时, 将 AP_i 承载的相应子流切换到 A_{select} 中负载比 ϕ 较小的 AP_j 上;
- 7) 重复 5)、6), 直到负载均衡.

4 实验结果及分析

4.1 平台搭建

在实验室内, 搭建基于 POF 的可编程无线局域网, 并通过测试对其进行性能评估. 该实验平台所使用的硬件设备和参数如表 1 所示. 该实验平台利用简单的星型网络拓扑结构, 如图 4 所示, 直接将每个配有 POF Switch 1.4.015 的 POF-AP 与多端口 Linux 服务器相连接, 该服务器又与 MPTCP & Iperf 服务器相连. POF-AP 为控制平面和数据平面配置不同的接口, 终端设备支持 MPTCP v0.92.

表1 硬件实验平台参数

设备	型号与配置
AP	Netgear R6100 openwrt15.05
	TP-Link WR1043ND v2 openwrt15.05
服务器	CPU: Intel Xeon E5 RAM: 64 G
	Ubuntu14.04
	CPU: Intel Core I7-8700 RAM: 16 G
终端设备	Ubuntu14.04
	iPhone A1586 ThinkPad E550

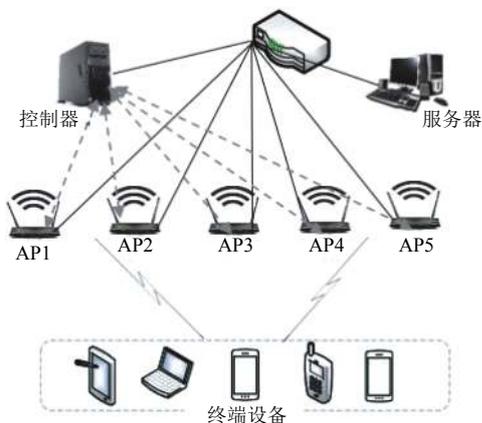


图4 实验网络拓扑

通过设置不同的试验条件,对文中搭建的可编程无线局域网的性能进行评估,下文章节给出了实验结果,并对结果进行了分析。

4.2 结果分析

1) AP 限流对终端设备带宽的影响

为了验证上文提出的集中接入策略的可行性,在实验中采用笔记本接入测试,不同于传统的仅接入一个 AP;按照上述支持 MPTCP 的集中接入策略,实验中选取 3 个 POF-AP 与笔记本相关联。同时,利用 shell 脚本,采用 tc 和 iptables 命令实现 AP 的限流,从而模拟现实网络中 AP 带宽资源的不稳定性。

实验中通过编程方式实现在第 31.5 s-46.5 s 间将接入的 AP1 的带宽限制在 1 Kbps,图 5 是在对 AP1 进行限流时,笔记本测试的带宽结果。分析实验结果可知,按照传统方式接入 AP 时,因为只接入 AP1,所有的数据流只能由 AP1 传输,所以在将 AP1 的带宽限制在 1 Kbps 的时间段内,终端总带宽极速下降到接近于零的状态;然而,按照上文提出的支持 MPTCP 的集中接入策略,由于同时择优接入了 3 个 AP(即 AP1、AP2、AP3),将 AP1 的带宽限制在 1 Kbps 时,AP1 的

负载比变大,其负载的子流将会被切换到另外两个 AP 上,而总的带宽会基本保持稳定。同时,与传统的接入策略相比,上文提出的接入策略在一定程度上提升了整体网络的吞吐量,提高了网络资源的利用率,避免在拥塞的路径上继续传输数据,从而均衡底层 AP 负载。

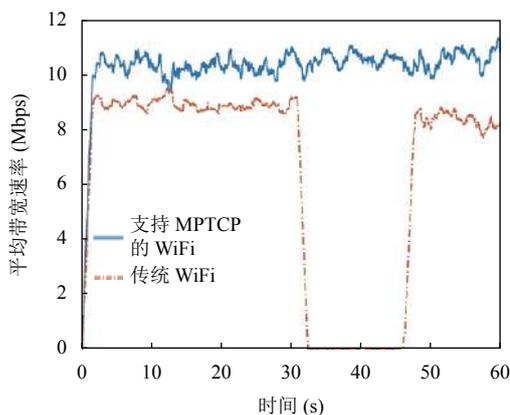


图5 AP 限流后终端带宽速率对比

2) AP 限流对其他 AP 带宽的影响

上小节是从终端用户的角度分析集中接入策略的可行性,接下来从 AP 的角度进行分析。为了体现一般性,实验中部署两个 POF-AP(AP1 和 AP2)和十个终端设备,每个终端具有两个子流分别与上述两个 AP 相关联,即 20 个子流均匀分布在两个 AP 上。

如图 6 所示,在非限流情况下,两个 AP 的带宽速率相当,保持在 19 Mbps 左右,总体上振幅差不多;当把 AP1 的带宽速率限制在 15 Mbps 时,由实验结果可知,AP1 的振幅变小,而 AP2 的带宽速率上升到 23 Mbps 左右,振幅变大。经过分析可知,在 AP1 限流时,AP1 负载比变大,其上的子流大部分被切换到 AP2,达到 AP 间的负载均衡,保证整体网络的吞吐量。

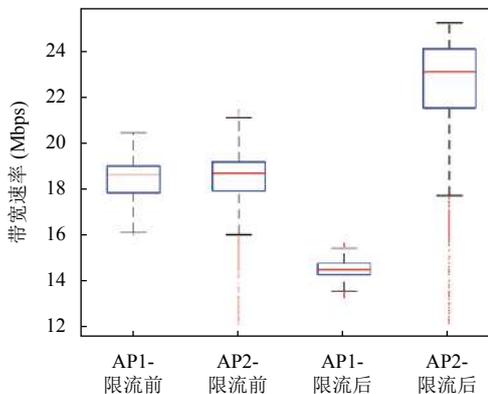


图6 AP 限流后各 AP 间的带宽速率对比

3) 控制平面性能测试

采用笔记本接入 AP1 测试, 编写 shell 脚本, 比较 OpenWrt 自带 WiFi 和支持 MPTCP 的 WiFi 情况下的接入时间. 共测得 40 组数据, 将每 10 组求平均, 得到 4 组数据如表 2 所示. 在终端接入时, 由于 AP 与控制器之间存在数据流交互, 而且集中控制器还要进行 AP 接入选择决策, 这些都会间接造成终端接入延迟. 相比较 OpenWrt 自带的 WiFi, 本研究原来提出的支持 MPTCP 的 WiFi 接入耗时是前者的 2 倍左右, 但是经过优化, 接入耗时降到前者的 1.5 倍左右, 其中, 算法运行耗时平均在 800~900 ms 之间, 在可接受范围之内.

表 2 终端接入测试 (ms)

OpenWrt 自带 WiFi	支持 MPTCP 的 WiFi	优化后的 WiFi
2654	5908	4026
2642	4028	3963
2651	4181	3896
2649	5564	3947

由于本研究引入软件定义的思路对物理 AP 虚拟化, 采用集中式管理方式实现终端设备的接入, 所以仅考虑接入时的延迟确实比传统的接入方式耗时一些. 但是, 相比较该架构带来的优势, 这些延时代价完全在可接受范围; 因为该架构不仅可以很好的做到 AP 设备之间的负载平衡, 避免网络拥塞; 同时, 该架构未来还可以拓展到用户移动的环境下, 如果在终端用户移动的情况下, 终端用户可能从 AP1 覆盖的范围移动到 AP2 的覆盖范围, 这时, 如果是传统的 WiFi 环境, 终端就需要先与 AP1 断开关联, 再接入 AP2, 不仅要重新接入而且终端的服务也会被迫断开; 但是如果利用该架构就可以实现无缝切换, 不需要重新接入, 因为该架构中终端是与 VAP 绑定的, 无论终端如何移动, 与之关联的 VAP 并不变, 底层只需要 VAP 在物理 AP 之间迁移, 而 VAP 是以流表的形式存在, 相比较传统的接入方式, 流表的迁移更便捷、更容易实现, 从而实现终端用户无感知的 AP 切换, 这也是未来会进一步研究的方向.

接下来进行多终端接入的仿真测试, 由 AP 不间断上报不同的 probe_request 报文来模拟设备的接入请求, 测试不同负载情况下控制面的 CPU 占用率, 固定

接入 AP 的总是为 5, 调整每个 AP 上报的 probe_request 数目, 即模拟的主机个数. 由图 7 分析可知, 随着终端数的增加, 控制平面 CPU 占用率呈现上升趋势,

在终端数目一定的情况下, 随着子流数的增加, 控制平面 CPU 占用率呈现小幅度提升趋势, 即随着终端数或子流的增加, CPU 的占用率呈现明显上升趋势.

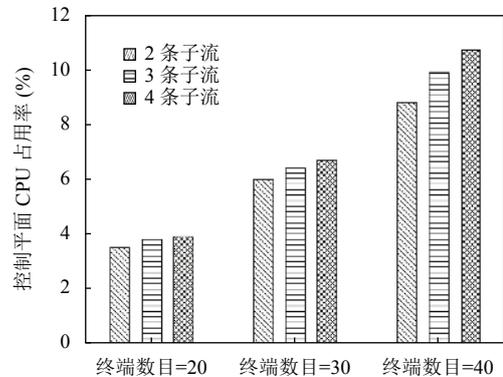


图 7 控制平面 CPU 占用率

综上分析各实验结果可知, 本文提出的支持 MPTCP 的集中接入策略在一定程度上能提升网络的吞吐量, 提高网络资源的利用率; 同时, 能充分利用各 AP 的带宽资源, 避免在拥塞的路径上继续传输数据, 从而达到平衡拥塞和均衡 AP 间负载的目的, 保证终端用户的服务质量. 然而, 除了上述优势之外, 该集中接入策略也存在一些不足, 该策略会造成接入耗时的延长, 随着终端数和子流的增加, 控制平面 CPU 的占用率也会有一定程度的提高.

5 结论与展望

本文引入 POF 技术, 提出一种深度可编程 WLAN 架构, 该架构实现终端设备的集中接入管理. 同时, 提出一种基于 MPTCP 的无线 AP 接入管理机制, 相对于传统的仅仅按照信号强度 RSSI 来选择 APs 接入, 该机制从信道相对空闲的 APs 中选取接入点, 对整个网络资源的利用率进行调优, 解决了因 AP 负载过重造成的连接中断、效率低下等问题, 保证了终端用户的服务质量.

然而, 实验环境也仅仅是在实验室很小的局域网内进行, 同时也未考虑终端用户的移动性. 所以, 未来的工作就是将文中提到的 AP 接入管理机制应用到更大范围的网络, 同时考虑终端用户移动的情况.

参考文献

- 1 Namal S, Ahmad I, Gurtov A, et al. SDN based inter-technology load balancing leveraged by flow admission

- control. Proceedings of 2013 IEEE SDN for Future Networks and Services. Trento, Italy. 2013. 1–5.
- 2 Papanikos I, Logothetis M. A study on dynamic load balance for IEEE 802.11 b wireless LAN. Dennis: IEEE, 2001.
- 3 刘志宏. 802.11 无线局域网接入式负载均衡技术研究[硕士学位论文]. 长沙: 国防科学技术大学, 2011.
- 4 李云, 周亚峰, 曹侯. 基于软件定义网络的 WLAN 架构及负载均衡切换. 重庆邮电大学学报 (自然科学版), 2016, 28(5): 613–619. [doi: [10.3979/j.issn.1673-825X.2016.05.002](https://doi.org/10.3979/j.issn.1673-825X.2016.05.002)]
- 5 Kreutz D, Ramos FMV, Verissimo PE, *et al.* Software-defined networking: A comprehensive survey. Proceedings of the IEEE, 2015, 103(1): 14–76. [doi: [10.1109/JPROC.2014.2371999](https://doi.org/10.1109/JPROC.2014.2371999)]
- 6 Zhang CK, Cui Y, Tang HY, *et al.* State-of-the-art survey on software-defined networking (SDN). Journal of Software, 2015, 26(1): 62–81.
- 7 Wang XD, Tian Y, Zhao M, *et al.* PNPL: Simplifying programming for protocol-oblivious SDN networks. Computer Networks, 2018, 147: 64–80. [doi: [10.1016/j.comnet.2018.09.018](https://doi.org/10.1016/j.comnet.2018.09.018)]
- 8 谈小冬, 邹山, 郭浩然, 等. 面向协议无感知转发技术的 SDN 试验床. 计算机系统应用, 2016, 25(4): 237–241.
- 9 Zhao M, Li MZ, Mei L, *et al.* FlowWatcher: Adaptive flow counting for source routing over protocol independent SDN networks. Proceedings of the 8th International Conference on Electronics Information and Emergency Communication. Beijing, China. 2018. 237–242.
- 10 Song HY. Protocol-oblivious forwarding: Unleash the power of SDN through a future-proof forwarding plane. Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking. Hong Kong, China. 2013. 127–132.
- 11 Song HY, Gong J, Chen HF, *et al.* Unified POF programming for diversified SDN data plane devices. Proceedings of the 11th International Conference on Networking and Services. Rome, Italy. 2015. 92–97.
- 12 Tan XD, Zou S, Guo HR, *et al.* POFOX: Towards controlling the protocol oblivious forwarding network. In: Park JJH, Yi GM, Jeong YS, *et al.*, eds. Advances in Parallel and Distributed Computing and Ubiquitous Services. Singapore. 2016. 21–30.
- 13 Bansal M, Mehlman J, Katti S, *et al.* Openradio: A programmable wireless dataplane. Proceedings of the 1st Workshop on Hot Topics in Software Defined Networks. Helsinki, Finland. 2012. 109–114.
- 14 Schulz-Zander J, Suresh L, Sarrar N, *et al.* Programmatic orchestration of wifi networks. Proceedings of 2014 USENIX conference on USENIX Annual Technical Conference. Philadelphia, PA, USA. 2014. 347–358.
- 15 Sequeira L, de la Cruz JL, Ruiz-Mas J, *et al.* Building an SDN enterprise WLAN based on virtual APs. IEEE Communications Letters, 2017, 21(2): 374–377. [doi: [10.1109/LCOMM.2016.2623602](https://doi.org/10.1109/LCOMM.2016.2623602)]
- 16 Sen A, Sivalingam KM. An SDN framework for seamless mobility in enterprise WLANs. Proceedings of the IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications. Hong Kong, China. 2015. 1985–1990.
- 17 Ford A, Raiciu C, Handley M, *et al.* TCP extensions for multipath operation with multiple addresses. RFC 6824, IETF, 2013. [doi: [10.17487/RFC6824](https://doi.org/10.17487/RFC6824)]
- 18 Croitoru A, Niculescu D, Raiciu C. Towards wifi mobility without fast handover. Proceedings of the 12th USENIX Conference on Networked Systems Design and Implementation. Oakland, CA, USA. 2015. 219–234.
- 19 Liu L, Li MZ, Mei L, *et al.* Design and Implementation of a novel SDN-based architecture for Wi-Fi networks. Proceedings of the 19th International Conference on Parallel and Distributed Computing: Applications and Technologies. Jeju Island, South Korea. 2018. 41–49.
- 20 Dutt S, Habibi D, Ahmad I. A low cost Atheros system-on-chip and OpenWrt based testbed for 802.11 WLAN research. Proceedings of the TENCON 2012 IEEE Region 10 Conference. Cebu, Philippines. 2012. 1–4.