

基于软件定义无线网络的协作式路由实现和性能分析

费宁 徐力杰 成小惠

(南京邮电大学计算机学院 南京 210003)

(fein@njupt.edu.cn)

Implementation and Evaluation of Cooperative Routing in Software Defined Wireless Networking

Fei Ning, Xu Lijie, and Cheng Xiaohui

(School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210003)

Abstract In a mesh network, not all WiFi APs are always taken part in the actual packet transmission which makes network resources redundant. A same packet is usually received by more than one node even though they are not on this packet's original transmission path. These nodes can be chosen to forward the packets to the destination node, which improves the bandwidth of the designated link in a cooperative routing path. However, due to the transmission and computation overhead on WiFi APs, the cooperative routing is difficult to be implemented in traditional wireless networking. The centralized architecture of software defined wireless networking (SDWN) makes it convenient to select helper nodes effectively and globally for a cooperative routing. This paper proposes a cooperative routing algorithm in SDWN. In this algorithm, routing discovery is performed on WiFi APs while the global and computation intensive tasks such as the route validation, the helper node selection and the interferences update are completed by the controller. By extending OpenFlow protocol, the proposed cooperative routing has been implemented and evaluated on a test bed. The evaluation indicates that our QoS oriented SDWN-based cooperative routing achieves greater bandwidth and less packet transmission delay than traditional wireless protocols.

Key words software defined wireless networking (SDWN); cooperative routing; helper node; OpenFlow; centralized architecture

摘要 无线 mesh 网络的冗余式设计往往导致 WiFi AP 的带宽资源不能得到充分利用。由于无线广播的特性,报文会同时被多个节点所接收。在协作式网络中,这些冗余的 WiFi AP 可以作为辅助节点帮助其他链路进行数据传输。这种协作式路由在传统网络中给资源受限的 WiFi AP 带来了额外的通信和计算开销,从而实现较为困难。软件定义无线网络 (software defined wireless networking, SDWN) 集中式架构提供了辅助节点之间及辅助节点与控制器之间的通信机制,使得最佳辅助节点的全局化选择成为可能。在分析现有协作式路由的基础上,提出了基于软件定义无线网络协作式路由算法,其中路由发现

收稿日期:2018-12-26;修回日期:2019-03-15

基金项目:国家自然科学基金项目(61003040,61502251);中国博士后科学基金资助项目(2016M601859);江苏省博士后科研资助计划项目(1701047A),南京邮电大学科研项目(NY215168,NY218024)

This work was supported by the National Natural Science Foundation of China (61003040, 61502251), the China Postdoctoral Science Foundation (2016M601859), the Jiangsu Planned Projects for Postdoctoral Research Funds (1701047A), and the Technology Project of Nanjing University of Posts & Telecommunications (NY215168, NY218024).

由 WiFi AP 完成,而路由有效性验证、辅助节点的选取和链路干扰模型的更新等全局且计算负载较大的工作由软件定义无线网络中的控制器进行.通过对现有 OpenFlow 协议的扩展,在测试平台上验证并实现了基于软件定义无线网络的协作式路由.测试结果表明:该面向服务质量的协作式路由相比传统路由能够显著提高网络传输带宽和降低报文传输延迟.

关键词 软件定义无线网络;协作式路由;辅助节点;OpenFlow;集中式架构

中图法分类号 TP393.02

按照 OSI 七层模型,无线网络和有线以太网网络在网络层以上享有相同的协议栈,两者的区别主要集中在决定传输质量的物理层和数据链路层.无线资源在物理层主要集中在天线、信道和时隙;数据链路层则包括对多个设备协同发送和接收数据报文的控制管理,以及对物理层资源的分配和调度管理.软件定义网络集中式架构在无线领域的应用使得全局性信息收集和无线资源最大化利用成为可能.基于此,本文提出软件定义无线网络 (software defined wireless networking, SDWN) 的协作式路由算法,从而可以更好地优化和使用物理层的无线资源以提供更优质的 QoS 服务.

1 研究现状

在传统无线网络中,每节链路包含 2 个节点,它们在数据传输中担任发送端或者接收端的角色.每个节点可以侦测信道是否已经被占用,若已被占用,则采用 CSMA/CA 等算法来避免频繁的探测.但由于无线信道广播的特性,在报文发送端覆盖范围内所有节点都可以收到数据报文,在一定条件下它们可以再次发送该报文以提高接收端的效率.然而,传统的分布式网络不能做到对节点资源的统一发现和协调,也就无法发挥无线信道广播的优势,很难提高全网带宽利用率.

软件定义网络的核心思想是将控制层和数据转发层相分离,这种集中式的架构同样适用于无线环境.由于软件定义网络原先为有线以太网设计,并没有相应的标准来规定如何控制和管理无线网络的信道、时隙、编码等.目前已经有研究尝试将无线网络中部分控制逻辑转移至控制器,这样的网络架构亦称为软件定义无线网络^[1-3].在文献[1]中,Pantou 项目只是简单地将天线空口直接映射为端口;文献[2-3]尝试将无线资源的参数用隧道模式传送给控制器.这些研究都避开对无线资源的抽象、映射、监控和管理必须进行标准化的问题.本文通过对

OpenFlow 协议栈的扩展有效地解决了无线网络控制信息的交互问题,这是本文研究的重点之一.

协作式路由是一种通过使用空闲节点辅助其他链路进行数据传输,以在无线网络中提高带宽的技术^[4].协作式路由与传统路由的不同之处在于:在寻找满足 QoS 路由的过程中,每一跳都需要决定辅助节点的参与与否.同时在协作式路由中,既需要考虑传统路由中相邻链路之间的干扰,也需要考虑链路和辅助节点之间的干扰.现有研究单跳下的协作式路由主要集中在物理层、数据链路层和网络层.其中,解码转发 (decode-and-forward, DF) 传输机制集中于物理层的研究^[5];数据链路层亦有文献提出了基于 IEEE 802.11 开放源码的协作式 MAC 协议^[6];网络层则主要讨论了如何最大化多条链路的最小带宽^[7-9].在跨层协作方面,已有学者研究如何在物理层和数据链路层合理分配资源,以提高网络带宽,并减少节点之间发送和接收数据报文的冲突^[10];或者尝试在网络层改进路由协议来满足 QoS 以提高网络整体吞吐量^[11];亦或在应用层根据用户请求类型来改进用户的使用体验^[12].但是,这些研究受限于 OSI 分层框架,对现有算法和解决方案的改进仍然局限于某个 OSI 层次或者相邻两个层次的简单结合.虽然这些工作已经能够做到利用辅助节点帮助单条链路进行带宽优化,但仍存在 2 个弊端:1) 辅助节点往往只根据有限的局部信息来选择;2) 辅助节点的计算及链路干扰范围的更新都由路由器完成,这既增加了路由器的计算负载,也增加了路由发现时间.这些关键问题可以通过软件定义网络跨层协作机制解决,这是本文研究的另外一个重点.

传统的分布式路由协议根据使用的出发点可以分为最短路径和最大带宽这 2 种协议.在无线路由中,最短路径往往由最小跳数 (shortest path routing, SPR) 来表示,典型实现包括无线自组网按需距离矢量路由协议 (ad hoc on demand distance vector, AODV)^[13] 和优化链路状态路由协议 (optimized link state routing protocol, OLSR)^[14].这种路由协

议一般不考虑各条链路的传输速率和可用时隙的差别,使用的评价标准是路径的跳数,或者说路径上中间节点的个数.最大带宽协议(widest path routing, WPR)^[15]往往基于迪杰斯特拉算法来实现,旨在发现 2 个节点之间具备最大带宽的路径.与最短路径相反,该路由在寻找下一跳时,总是优先选择最大带宽的链路,并不考虑最终路径所经历的跳数.这 2 种协议是本文在实现 SDWN 协作式路由参照和对比的重点.

本文提出了一种基于软件定义无线网络架构、利用辅助节点增加带宽的协作式路由算法,网络中每个节点将各自的链路信息发送给控制器,控制器负责干扰模型的建立和路径 QoS 有效性验证,统一选择辅助节点,找到最佳转发路径.该协作式路由在 SDWN 集中式架构下,综合了 MAC 层时隙、链路层状态及网络层路由信息,针对传统的 AODV 协议进行 2 次开发.与传统 AODV 及最大带宽协议进行比较的结果表明,该算法可有效提高系统带宽,并易于在 SDWN 环境下部署和实施.

2 算法的数学模型

传统无线网络的协作式路由已经得到了较为充分的研究^[16-17].该网络模型可以用无向图 $G(V, E)$ 来表示,其中 V 为同一个网络中 WiFi AP 的集合, E 为网络中所有链路的集合. $d(u, v)$ 代表节点 u 和 v 之间的物理距离, R_t 是传输距离,若 $d(u, v) \leq R_t$,表示存在一条连接节点 u 和 v 的链路 $e(u, v) \in E$.在该研究模型中,假设所有链路工作在同一个信道,采用 $\Omega(1)$ 模型来约束 2 个无线节点之间的最小距离 d_0 .

协作方案采用 DF 机制来提高带宽,即当直接传输的某节链路带宽不满足要求时,尝试借助辅助节点协作传输,如图 1 所示.

此时无线信号利用 2 个时隙(time slot, TS)进行传输,第 1 个时隙称为广播时隙,即信源节点 s 向所有相邻节点(h 和 t)广播消息;第 2 个时隙称为协作传输时隙,用于辅助节点 h 将从 s 收到的帧传送到信宿节点 t .

考虑到网络拓扑的复杂性,在设计算法时,为了模型简单起见,首先假设每个节点在一条路由中只能充当转发节点或者辅助节点^[18],同时设定每节链路最多只通过一个辅助节点帮助传输,多节链路构成的一整条路由可包含多个辅助节点.而当每节链路都满足带宽要求时,辅助节点为 0,即直接传输,

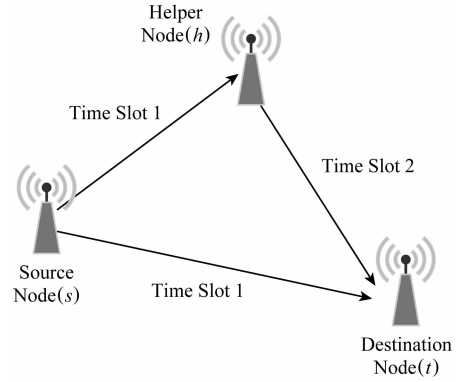


Fig. 1 DF transmission mechanism

图 1 DF 传输机制

此时整条路由在没有任何辅助节点的帮助下完成传输.

直接传输的信道容量可由香农定理得到,已证明协作路由下的 DF 机制在一定情况下可以取得比直接传输速率更快的链路带宽^[19],DF 模式的链路容量 C_{DF} 见式(1)^[20]:

$$C_{DF}(s, h, t) = W \times I_{DF}(s, h, t) = \frac{1}{2} W \times$$

$$\min\{\text{lb}(1 + SINR_{sh}), \text{lb}(1 + SINR_{st} + SINR_{ht})\}, \quad (1)$$

其中, W 表示带宽, $I_{DF}(s, h, t)$ 表示最大平均解码-转发模式相互作用信息, $\text{lb}(1 + SINR_{sh})$ 表示辅助节点 h 有效解码信息源的速率, $\text{lb}(1 + SINR_{st} + SINR_{ht})$ 表示信宿节点 t 在辅助节点 h 帮助下有效解码信息源的速率.协作路由中备选辅助节点的选取可通过式(1)获得,其中信噪比 $SINR$ 是选取辅助节点的关键因素.

协作式通信中,从备选辅助节点中选择合适的节点受限于其所在链路空闲状态.如果一个备选节点同时处于多个节点的传输干扰半径之内且这些节点长期处于数据传输状态,那该备选节点就无法成为有效的辅助节点.链路干扰是选取辅助节点的另一个重要因素,取决于节点的传输半径,也受限制于各个节点所占用的传输时间片.设链路 l 的可用时间片为 $ATF(l)$,其计算方法:

$$ATF = 1 - \sum_{e \in IS(l)} \sum_{f \in F} |TF(t, f)|, \quad (2)$$

其中, $IS(l)$ 表示和路径 l 中链路相干扰的链路集合,网络 $G(V, E)$ 中已存在的流集合定义为 $F = \{f_1, f_2, \dots\}$,每条流占用相应链路 l 的时间片为 $TF(l, F)$.

在实现中, ATF 无法通过式(2)直接获取,但可近似获得:

$$ATF=1-\frac{num_transPackets}{max_transPackets}, \quad (3)$$

其中, $num_transPackets$ 表示单位时间内实际传输数据包的个数, $max_transPackets$ 表示同一信道传输数据包的最高历史记录. 由此可以看出, 相关链路的占空比也是选取辅助节点参数之一.

根据约束条件的分析, 协作路由问题可以定义为: 对于给定网络 $G(V, E)$ 以及每节链路的 $SINR$ 和 ATF , 已有流的集合 $F=\{f_1, f_2, \dots\}$, 以及一个新的连接请求 $\rho(s, t)$, 在保证既有流量不受干扰的情况下, 寻找网络 G 的一个子图, 发现一条从信源节点 s 到信宿节点 t 拥有最大带宽的协作路由由路径 P , 以及对应辅助节点的集合 $H=\{h_0, h_1, \dots, h_n\}$, 此处 h_i 为路径 P 中链路 $l_{i,i+1}$ 的辅助节点. 若 $H=\emptyset$, 则表明该链路采用直接传输模式. 该问题用数学方法建模为

$$\max(BW(P))=C(l_{i,i+1}, h_i) \times TF(l_{i,i+1}, h_i, \rho) \quad (4)$$

s. t.

$$\forall l_{i,i+1}, l_{j,j+1} \in P, C(l_{i,i+1}, h_i) \times TF(l_{i,i+1}, h_i, \rho) = C(l_{j,j+1}, h_j) \times TF(l_{j,j+1}, h_j, \rho) \quad (5)$$

$$\forall l \in IS(P), \sum_{e \in IS(l)} TF(e, \rho) \leq ATF(l) \quad (6)$$

$$\forall i, j, m \in V, \sum_{m \in V} x(m, P) + \sum_{l_{i,j} \in P} y(m, l_{i,j}, P) \leq 1. \quad (7)$$

根据式(5)~(7)这些约束条件, 协作式路由发现问题可以转化为式(4)及其相关约束条件的求解. 其中 $BW(P)$ 表示路径 P 的带宽, $C(l_{i,i+1}, h_i)$ 表明在辅助节点 h_i 帮助下进行协同传输链路 $l_{i,i+1}$ 的信道容量, $TF(l_{i,i+1}, h_i, \rho)$ 是链路 $l_{i,i+1}$ 占用的时隙, ρ 表示连接请求, i, j 表示链路上的节点.

式(4)的约束条件包括式(5)~(7). 其中式(5)表示流预留约束, 规定了2条链路之间流的关系; 式(6)表示对可用时隙的调度约束, 规定了在路径中的节点不能互相干扰, 其中 $ATF(l)$ 表示链路 l 的可用时间片, $IS(P)$ 表示和路径 P 中链路相干扰的链路集合; 式(7)为中继约束, 表明任何节点只能成为转发节点或者辅助节点, $x(m, P)$ 表示是否为转发节点, $y(m, l_{i,j}, P)$ 表示是否为辅助节点, 两者均为布尔型变量.

3 面向软件定义无线网络的协作式路由 (SDWN Coop-routing) 算法设计

传统协作式路由的算法设计有分布式和集中式

2种方案^[21]. 本文在此基础上作了扩展, 提出了融合这2种方案的混合模式, 即面向软件定义无线网络的协作路由算法.

在传统网络环境中, 由于分布式架构的限制, 每个节点只能和其有限个邻居进行信息交互. 这样带来的后果是: 无论是在计算2个节点之间的带宽, 还是进行最优辅助节点选择时, 都无法获取准确的全局信息. 完全的集中式方案可以保证选取全局最优协作式路由, 但该算法是个 NP-hard 问题^[16-17], 无法在多项式时间内发现; 然而, 软件定义无线网络集中式的架构特别适合收集全局信息和进行高强度计算, 这是本文将路由有效性验证、辅助节点选择和链路干扰模型更新这3项功能移植至控制器, 而 WiFi AP 仍然负责路径发现的原因.

面向软件定义无线网络协作式路由算法的核心思想是: 在软件定义无线网络集中式架构中, 各个路由器只负责传统的路径发现, 同时路由器将相关链路信息发送给控制器. 控制器负责完成协作式路由的相关功能, 包括根据拓扑和链路信息建立干扰模型、对各个节点所选路径进行有效性判断以及辅助节点选择, 最后将辅助节点的控制信息发送至路由器执行转发模式的变更, 其具体过程如算法1所示.

算法1. 面向软件定义无线网络协作式路由算法.

输入: 网络 $G(V, E)$ 、每节链路的 $SINR$ 和 ATF 、路由请求 $\rho(s, d)$ 以及相应的带宽要求 RBW ;

输出: 协作路由路径 P 、时隙调度算法 s 、发现路由带宽 $BW(P)$.

- ① 每个节点将链路信息发送至控制器;
- ② 控制器根据拓扑和链路信息建立干扰模型;
- ③ 源节点广播 $RREQ$ 消息, 并按照标准 AODV 协议进行路由发现;
- ④ 源节点将完整的路由信息发送至控制器, 进行 QoS 有效性验证;
- ⑤ if 该 AODV 路由满足最低带宽要求 then
- ⑥ 路由寻找结束, 返回所发现路径;
- ⑦ else
- ⑧ for 所有不满足带宽要求的链路 do
- ⑨ 控制器根据链路信息进行辅助节点查找, 重新计算带宽;
- ⑩ if 如果协作式路由不满足要求 then
- ⑪ 路由寻找失败, 退出;
- ⑫ else
- ⑬ 路由寻找结束, 返回所发现路径;

- ⑭ end if
- ⑮ end for
- ⑯ end if

算法 1 主要涉及 2 个方面:路由器端的路径发现;控制器侧的链路干扰模型的建立、路由 QoS 有效性验证以及辅助节点的选取. 路由器侧运行传统 AODV 协议,并将路由信息发送给控制器进行 QoS 验证;控制器收集网络的拓扑和链路信息,建立干扰模型. 当发现某条链路不满足 QoS 时,迅速有效地为相应链路选取辅助节点. 这使得路由器和控制器充分发挥各自的性能和特点,相较于分布式协作路由算法^[16],算法 1 可以提高辅助节点选取的准确性和最终路由发现的效率.

在一个包含 N 个节点、覆盖直径为 d 的网络中,传统 AODV 的时间复杂度为 $O(2d)$,通信复杂度为 $O(2N)$ ^[22]. 虽然完全的集中式路由选择方案是个 NP-hard 问题,但在本文所提出的面向软件定义网络的协作式通信路由中,仍由路由器运行 AODV 协议进行路径发现,所以并没有增加 AODV 本身路由查找的复杂度,而只是在带宽不满足要求的情况下对相应的链路进行辅助节点的查找和对应带宽的更新,其时间复杂度仍然为 $O(2d)$,但通信复杂度增加至 $O(2d+2N)$. 这是因为在最不理想的情况下,每节链路都需要进行辅助节点的查找,即 AODV 路径上的所有节点都参与了与控制器的双向交互,这是通信复杂度由 $O(2N)$ 增加至 $O(2d+2N)$ 的原因. 当然额外的通信开销包括每个节点与控制器常规的链路状态更新,这里并没有计入协作式路由算法的开销.

4 协作式路由算法的设计与实现

4.1 跨层协作路由框架

在系统实现中,每个 WiFi AP 需要根据链路层的信息来计算路由;此外,这些链路层的信息也会通过用户空间的网络层相关接口汇报给控制器. 由于数据链路层运行在操作系统的内核空间,而网络层运行在用户空间,所以从操作系统角度而言,协作式路由往往涉及到跨层通信. 本文所设计的跨层协作式通信系统的架构如图 2 所示,其主要包含 3 个模块:协作 MAC 层、协作路由层以及控制器侧的辅助节点选择模块.

为了实现 DF 传输机制,协作 MAC 层包含 3 个子模块:NET802.11 模块主要实现了 802.11 数据

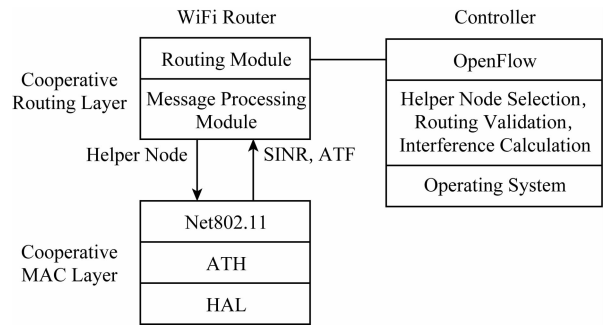


Fig. 2 Framework of Coop-routing protocol
图 2 协作式路由协议架构

帧的解析和封装;ATH 模块(Atheros 无线网卡驱动模块)负责 Atheros 芯片与 NET802.11 模块交互以及通过硬件抽象层(hardware abstraction layer, HAL)来访问硬件的功能;HAL 模块负责将 Atheros 硬件资源进行抽象封装并提供给上层 ATH 模块使用. 相较于传统的 MAC 机制,辅助节点在协作 MAC 层需要判断是否接收目的 MAC 地址与自身 MAC 地址不匹配的报文,如果报文是来自需要进行辅助传输的链路,辅助节点也应该接收该报文;同时辅助节点不需要修改所转发报文的 MAC 地址,收到对应的数据报文后立刻进行转发;此外,辅助节点只须帮助信源节点重传数据报文一次,而无需等待 ACK 确认消息,信源节点需要自行确认所发送报文是否已经被正确接收.

4.2 控制器端在协作式路由中的实现细节

基于软件定义无线网络的协作式通信模块之间的交互信息如图 3 所示. 可以看出控制器主要完成 3 个任务:路由有效性验证、辅助节点的选取和建立干扰模型.

如算法 1 所述,路由器仍然通过传统路由协议 AODV 进行路由发现,所发现的路由送至控制器进行 QoS 验证. 本算法中以带宽为验证因子,也就是说控制器会验证所发现的路由是否满足预设的带宽要求;如果满足,则路由发现结束;如果不满足,再对不满足的链路进行辅助节点的查找和选取,如果所选择的辅助节点满足带宽的要求,则路径发现结束,即为协作式路由;如果有任何一节链路在考虑辅助节点的情况下仍然不能满足带宽要求,算法终止,则路径发现失败.

协作 MAC 层的一个主要功能就是收集链路信息,这些交互信息包括相邻链路的 SINR、传输速率和 ATF. 其中 SINR 和传输速率可以分别从 Madwifi 中的 ATH 模块和接收到数据包的 MAC 头部直接

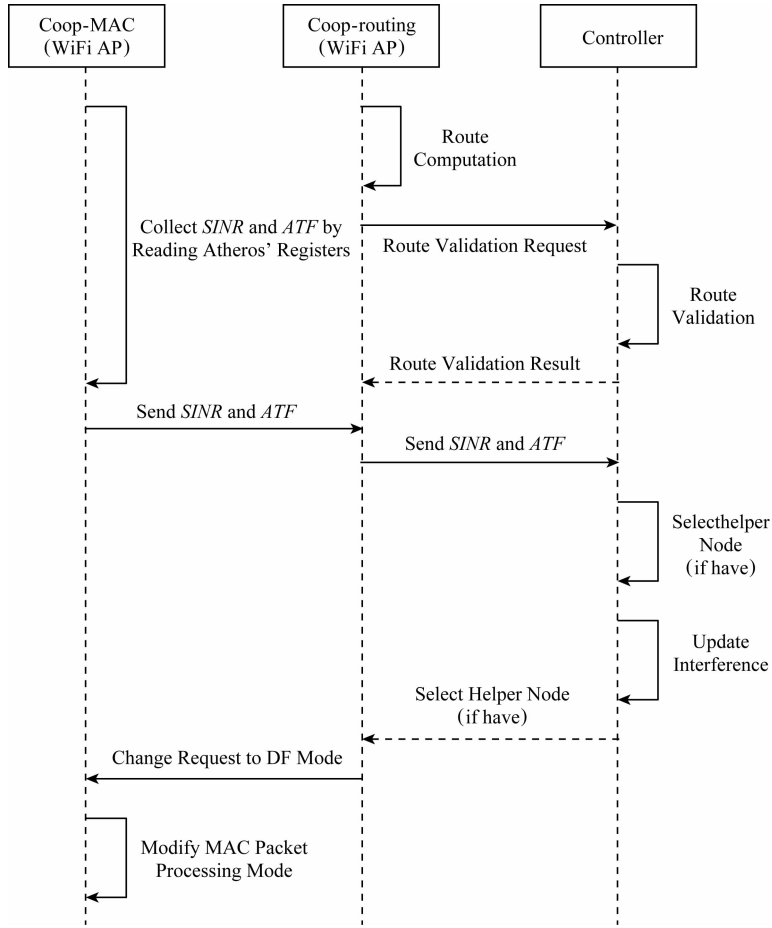


Fig. 3 Message exchange sequence diagram

图3 消息交互时序图

获取;MAC层通过修改 Madwifi 驱动来收集协作式通信需要的链路信息,即计算 ATF 需要的历史报文传输数据 ($num_transPackets$ 和 $max_transPackets$). 最终由协作路由层中的路由模块负责将从 MAC 层收集到的链路状态信息汇报给软件定义无线网络中的控制器,控制器进行辅助节点选取计算之后,发送指令让相关 WiFi AP 切换至辅助转发模式,改变 MAC 层报文接收方式.由此可见,路由计算和辅助节点的选取计算分别交给了路由器端和控制器侧,这种负载分担的方式有助于提高协作式路由计算效率,减少了路由发现的延迟.

协作式路由中的链路干扰条件不同于传统无线网络.在传统无线网络中,链路间形成干扰的前提条件是其中一条链路在另外一条链路终端节点的传输范围之内,然而在协作式路由中,控制器根据拓扑和链路信息建立干扰模型器.如果所有和 l_s 及 l_d 相干扰的节点保持沉默,也就是干扰范围内没有正在进行传输的数据流,表明此时数据包可以在广播时隙成功传输给辅助节点 h 和信宿节点 t .

4.3 基于软件定义无线网络的协作式路由算法 (SDWN-Coop) 的实现

SDWN 协作式路由要求 WiFi AP 将链路信息发送给控制器,由控制器选择辅助节点.所有节点和控制器的消息交互都是和协作式路由这个应用相关联,但是目前 OpenFlow^[3] 协议并没有规定这些消息交互的格式.在实验中,增加了 4 条消息,如表 1 所示:

Table 1 OpenFlow Message Type in SDWN-Coop

表 1 OpenFlow 协作式路由消息交互

Functionality	Message Code
Send link status from router to controller	OFPT_SEND_LINK_STATUS
Send route validation message from router to controller	OFPT_REQ_AODV_ROUTE_VALIDATION
Return the route validation result from controller to router	OFPT_RSP_AODV_ROUTE_VALIDATION
Send the helper node selection status to router	OFPT_SEND_LINK_HELPER_STATUS

如 SDWN 协作式路由算法所述,各个 WiFi 节点通过增加的交互消息 *OFPT_SEND_LINK_STATUS* 向控制器汇报和更新链路状态信息;路由发现由 WiFi AP 完成,所发现的路由通过 *OFPT_REQ_AODV_ROUTE_VALIDATION* 发送给控制器,控制器根据 QoS 要求进行有效性验证;控制器由 *OFPT_RSP_AODV_ROUTE_VALIDATION* 消息返回验证结果,如果路由满足要求,则路由发现结束;如果不满足要求,控制器寻找合适的辅助节点,并由 *OFPT_SEND_LINK_HELPER_STATUS* 消息通告路由器进行状态切换.整个路由发现校验和辅助节点选择可以由表 1 所示的 4 个消息完成.

5 实验平台和性能分析

5.1 实验平台和环境

协作式通信的测试平台采用了基于 OpenFlow 1.3.1 的软件定义无线网络架构.为了定制的需要,实验没有采用传统的 WiFi AP 设备,所有无线节点都由通用笔记本(ThinkPad T61, Linux 系统,内核版本 2.6.31.14)来实现.每个笔记本采用了基于 Atheros 芯片的无线网卡(Atheros AR5005GS),该网卡可以获得驱动相关的源代码(Madwifi-0.9.3.2).WiFi AP 端的路由协议为经过改编的 AODV(AODV-0.95).实验中使用的控制器是基于 Python 语言的轻量级控制器 Ryu,其北向接口清晰,扩展容易.

在实验中总共部署了 9 个这样的测试节点,2 个节点之间的距离在 20 m 至 50 m 之间不等,其测试场景如图 4 所示.在实验过程中,AP 1 和 AP 9 分

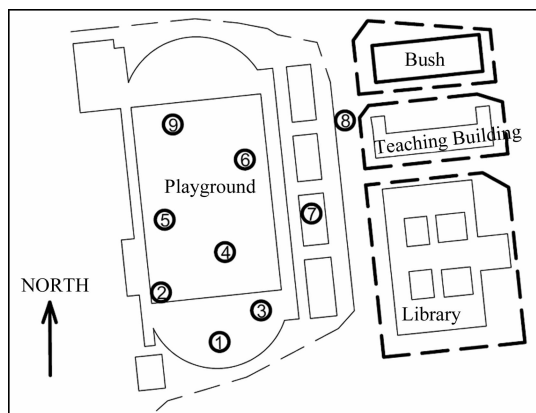


Fig. 4 Experiment scenario

图 4 实验测试场景

别为路由的起点和终点,其他节点在测试过程中根据需要担当转发节点或者辅助节点的角色.在实验中,由于节点较少,在控制器路由校验中,以尝试发现可能的最大带宽为目标,而不是选取一个固定的阈值来做简单比较.另外一个有趣的发现是节点 8 在原本的部署中虽远离其他节点,但实际测试中因为与其他链路冲突较少,反而有更多的机会承担辅助节点的角色.

5.2 实验指标

实验共比较了 4 种路由协议,其中包括 2 个协作式路由算法:本研究所设计的基于 SDWN 集中式算法的协作式路由(SDWN-based cooperative routing, SCR)协议和基于分布式算法的协作式路由(distributed cooperative routing, DCR)协议,另外 2 种传统路由协议分别是基于 AODV^[19]的最小跳数路由(shortest path routing, SPR)协议和基于迪杰斯特拉算法的最大带宽路由(widest path routing, WPR)协议^[15].在本实验中,上述协议都是基于 AODV 开放源代码进行的 2 次开发.

5.3 性能分析

对于上述 4 种路由协议,系统性能通过实际可获得的路由带宽、传输延迟以及路由发现时间这 3 个指标来评估.实验中通过 Iperf 创建了 10 条 TCP 流去验证,并记录平均值和 95% CI 误差值作为结果进行比较.结果汇总如表 2 所示:

Table 2 Summary of Performance Comparison

表 2 常用路由协议性能比较汇总

Routing Protocol	Bandwidth /Kbps	Transmission Delay Time/ms	Route Discovery Time/ms
SCR	393.25±4.3	18.9±0.7	476.8±48.4
DCR	386.5±3.4	19.3±0.5	457.1±48.2
WPR	273.0±3.0	29.9±1.2	408.8±40.6
SPR	185.5±3.5	58.5±1.4	331.2±37.7

图 5~7 是基于软件定义无线网络的协作路由协议 SCR、分布式协作路由协议 DCR 和无线 WiFi Mesh 网络中常用的最大带宽协议 WPR 和最小跳数协议 SPR 的比较结果.图 5 比较了各种路由协议所发现路由的带宽,结果表明 2 种协作路由算法发现的路由带宽比较接近,SCR 协议比 DCR 协议提高约 4.0%,但比 WPR 协议提高约 30%,比 SPR 协议提高约 90%.

从图 6 可以看出网络延迟的测试结果:SCR 协议和 DCR 协议结果比较接近,SCR 协议略好于

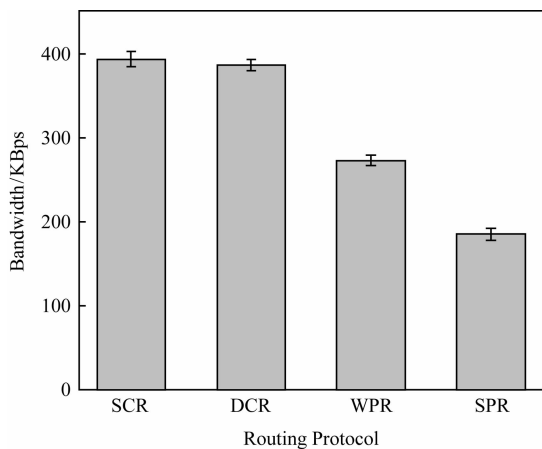


Fig. 5 Comparison of bandwidth

图5 路由带宽比较

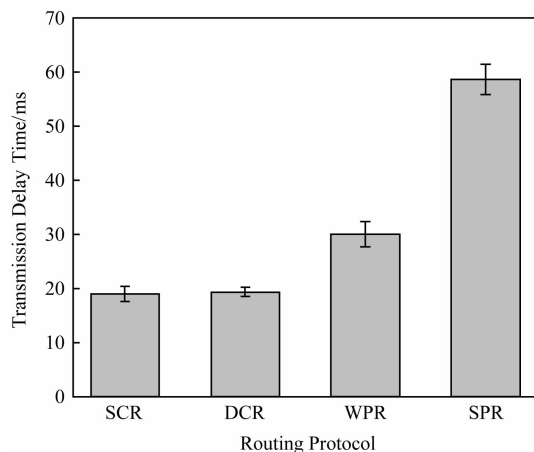


Fig. 6 Comparison of transmission delay

图6 路由传输延迟比较

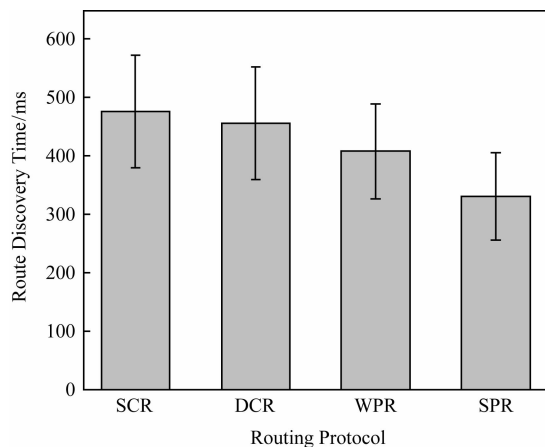


Fig. 7 Comparison of route discovery time

图7 路由发现时间比较

170%。这同样表明 SCR 协议可以显著降低多跳无线网络的传输延迟。

路由发现时间是指从源点发起报文传输请求到发现满足带宽要求路由的时间间隔。从图 7 中可以看到,SCR 协议和 DCR 协议都引入了额外的路由发现时间开销。这种开销主要来自于 3 点:和选取辅助节点相关信息的传输、路由有效性验证、干扰区域的计算和更新。在这里,SCR 协议比 DCR 协议略高主要是因为控制器和 WiFi AP 之间的信息传递所引入的延迟,而控制器计算能力强大在实验平台中由于节点较少而无法体现。虽然协作路由增加了路由发现时间,但路由发现对于同样目的地址的数据流只需要进行一次,之后的数据流都会沿着同样的路径进行传输,所以数据报文传输的时间并不会因此显著增加。

在软件定义无线网络上的结果表明,其选取的辅助节点和分布式协作路由有部分重合,其主要原因是在研究中为了部署方便,限制每节链路只能有一个辅助节点,而且该节点必然在该链路的干扰半径之内,这样的选择有些时候是唯一的。

在实验中,发现 SCR 协议和 DCR 协议无论在路径发现还是辅助节点选取上都获得了比较接近的结果,其原因可能是受限于实验平台的规模和场景,WiFi AP 节点较少且干扰范围一般不超过 3 跳,所以不容易体现软件定义架构的优势。当然,为了进一步发挥软件定义无线网络集中式架构的优越性,可以将路由发现也由控制器完成,这样不但可以做到单个路径的优化,也可以做到当前路径和历史路径的综合优化。也就是说,如果在没有任何一条满足 QoS 路径存在的情况下,可以对已有的路径进行调整,达到对全网所有节点的最优化利用,这也是未来资源优化研究的一个方向。

从图 5 与图 6 可以看出,基于软件定义无线网络的协作式路由能够提高带宽和降低延迟,不可否认的是控制器的引入也带来了额外的通信开销和安全隐患。然而,通信开销在分布式协作路由中同样无法避免,而且因为其点对点的传输属性,其开销比基于 SDWN 的协作式路由更大。在安全方面,由于控制器可以通过指令决定节点是否需要转发不属于自己的报文,给系统带来了单点失效的风险。但在一个大型网络中,控制器和路由器之间有专用的加密通信频段,而且控制器由多个设备组成集群,有效提高了控制信息传输的效率和降低了系统级别的风险,

DCR 协议,是由于在部分场景中 SCR 协议选取出的辅助节点更优。SCR 协议相比较 WPR 协议和 SPR 协议,网络延迟可以分别降低约 18% 和

这就是软件定义无线网络更适合大型网络的原因,也是下一步研究和实验的重点。

6 总 结

在软件定义无线架构中,控制器可以实时监控网络设备的运行状况和调度相关的网络资源。由于无线传输广播的特性,使得利用空闲节点辅助报文转发成为可能,从而能够提高全局路径的最小带宽。本文给出了协作式路由的模型,并提出了基于SDWN协作式路由算法,在WiFi AP上实现并验证了基于SDWN协作通信的可行性,给出了和传统路由的性能比较结果。实验表明:基于软件定义无线网络的协作式路由可以显著提高多跳无线网络的传输性能。

参 考 文 献

- [1] Dangovas V, Kuliesius, F. SDN-driven authentication and access control system [C] //Proc of the Int Conf on Digital Information, Networking, and Wireless Communications, Ostrava, Czech Republic; The Society of Digital Information and Wireless Communication, 2014; 20-23
- [2] Hakiri A, Sellami B, Patil P, et al. Managing wireless fog networks using software-defined networking [C] //Proc of the 14th Int Conf on Computer Systems and Applications, Hammamet, Tunisia; IEEE AICCSA, 2017; 1149-1156
- [3] Zuo Qingyun, Chen Ming, Ding Ke, et al. Eliminating redundant control messages in OpenFlow networks [J]. Journal of Computer Research and Development, 2014, 51(11): 2448-2457
(左青云, 陈鸣, 丁科, 等. OpenFlow网络冗余控制报文消除机制研究[J]. 计算机研究与发展, 2014, 51(11): 2448-2457)
- [4] Bhatnagar M R, Mallik R K, Tirkkonen O. Performance evaluation of best-path selection in a multihop decode-and-forward cooperative system [J]. IEEE Transactions on Vehicular Technology, 2016, 65(4): 2722-2728
- [5] Liu Pei, Tao Zhifeng, Narayanan S, et al. CoopMAC: A cooperative MAC for wireless LANs [J]. IEEE Journal on Selected Areas in Communications, 2007, 25(2): 340-354
- [6] Sharma S, Shi Yi, Hou Y T, et al. Cooperative communications in multi-hop wireless networks: Joint flow routing and relay node assignment [C] //Proc of the 29th IEEE Int Conf on Computer Communications (IEEE INFOCOM 2010). Piscataway, NJ; IEEE, 2010; 1-9
- [7] Lin Yuxia, Song J H, Wong V W S. Cooperative protocols design for wireless ad-hoc networks with multi-hop routing [J]. Mobile Networks & Applications, 2009, 14(2): 143-153
- [8] Shi Yi, Sharma S, Hou Y T, et al. Optimal relay assignment for cooperative communications [C] //Proc of the 9th ACM Int Symp on Mobile Ad Hoc Networking and Computing. New York; ACM, 2008; 3-12
- [9] Behrouz M, Debbah M, Jorungnes A. Energy-efficient Cooperative routing in BER constrained multihop networks [J]. Frontiers of Computer Science in China, 2009, 3(2): 263-271
- [10] Rad A H M, Wong V W S. Joint channel allocation, interface assignment and mac design for multi-channel wireless mesh networks [C] //Proc of the 26th IEEE Int Conf on Computer Communications (IEEE INFOCOM 2007). Piscataway, NJ; IEEE, 2007; 1469-1477
- [11] Sheikh S M, Wolhuter R, Engelbrecht H A. A survey of cross-layer protocols for IEEE 802.11 wireless multi-hop mesh networks [J]. International Journal of Communication Systems, 2016, 30(6): e3129
- [12] Wu Dalei, Luo Haiyan, Ci Song, et al. Quality-driven optimization for content-aware real-time video streaming in wireless mesh networks [C] //Proc of the 2008 IEEE Global Telecommunications Conf. Louisiana, USA; GLOBECOM, 2008; 1-5
- [13] Wang Hongpeng, Lin Cui. An enhanced AODV for mobile ad hoc network [C] //Proc of the 2008 IEEE Int Conf on Machine Learning and Cybernetics. Piscataway, NJ; IEEE, 2008; 1135-1140
- [14] Chen Jiwei, Lee Y Z, Maniezzo D, et al. Performance comparison of AODV and OFLSR in wireless mesh networks [C] //Proc of the IFIP Med-Hoc-Net Conf. Laxenburg, Austria, 2006; 14-17
- [15] Daggitt M L, Griffin T G. Rate of convergence of increasing path-vector routing protocols [C] //Proc of the 2018 IEEE 26th Int Conf on Network Protocols. Cambridge, UK; ICNP, 2018; 335-345
- [16] Feng Wei, Liu Jian, Cao Jiannong, et al. Interference-aware cooperative QoS routing in multi-hop wireless networks [J]. Computers & Electrical Engineering, 2013, 39(6): 1650-1666
- [17] Fei Ning, Feng Wei, Chen Chunlin, et al. Design, implementation and evaluation of cross-layer cooperative routing in multi-hop wireless networks [J]. Application Research of Computers, 2013, 30(9): 2839-2842 (in Chinese)
(费宁, 冯伟, 陈春玲, 等. 多跳无线网络跨层协作式路由协议实现和性能分析[J]. 计算机应用研究, 2013, 30(9): 2839-2842)

- [18] Tang Jian, Guoliang Xue, Weiyi Zhang. Interference-aware topology control and QoS routing in multi-channel wireless mesh networks [C] //Proc of the 6th ACM Int Symp on Mobile Ad Hoc Networking and Computing. New York: ACM, 2005: 68-77
- [19] Jia Juncheng, Jin Zhang, Qian Zhang. Cooperative relay for cognitive radio networks [C] //Proc of the 28th IEEE Int Conf on Computer Communications (IEEE INFOCOM 2009). Piscataway, NJ: IEEE, 2009: 2304-2312
- [20] Pawan K, Dhaka K. Performance analysis of a decode-and-forward relay system in-and-fading channels [J]. IEEE Transactions on Vehicular Technology. 2016, 65(4): 2768-2775
- [21] Chakeres I D, Belding-Royer E M. AODV routing protocol implementation design [C] //Proc of the 24th Int Conf on Distributed Computing Systems Workshops. Piscataway, NJ: IEEE, 2004: 698-703
- [22] Abolhasan M, Wysocki T, Dutkiewicz E. A review of routing protocols for mobile ad hoc networks [J]. Ad Hoc Networks, 2004, 2(1): 1-22



Fei Ning, born in 1977. PhD, associate professor. Her main research interests include software defined network, information retrieval and analysis, machine learning.



Xu Lijie, born in 1983. PhD, lecturer. His main research interests include wireless sensor networks, social networks, mobile and distributed computing.



Cheng Xiaohui, born in 1982. MSc, lecturer. Her main research interests include Web mining, information retrieval, and machine learning.