

分类号

学校代码 10487

学号 M202072033

密级

华中科技大学

硕士学位论文

(学术型☒ 专业型☐)

智能家居场景中 BLE MESH 与 Wi-Fi  
接入优化研究

学位申请人：郭康

学科专业：信息与通信工程

指导教师：吴伟民 副教授

答辩日期：2023 年 5 月 22 日

**A Dissertation Submitted in Partial Fulfillment of the Requirements  
for the Master Degree in Engineering**

**Research on BLE MESH and Wi-Fi Access Optimization  
in Smart Home Scenarios**

**Candidate : Guo Kang**

**Major : Information and Communication Engineering**

**Supervisor : Assoc.Prof. Wu Weimin**

**Huazhong University of Science and Technology**

**Wuhan 430074, P. R. China**

**May, 2023**

## 独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：郭康

日期：2023 年 5 月 22 日

## 学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保 密 ☐，在 \_\_\_\_\_ 年解密后适用本授权书。

本论文属于 不保密 ☒。

（请在以上方框内打“√”）

学位论文作者签名：郭康

指导教师签名：郭康

日期：2023 年 5 月 22 日

日期：2023 年 5 月 22 日

## 摘要

通过各种无线通信技术构成的智能家居设备网络正在为人们提供舒适、便捷、节能的家居体验。BLE (Bluetooth Low Energy, 低功耗蓝牙) 技术因其低功耗、低成本等特点、Wi-Fi 技术因其高带宽、信号覆盖范围广等特点而被广泛应用于智能家居网络中。然而, 随着智能家居产业发展以及普及程度的加快, 大量家居设备加入网络引起了系统吞吐以及时延性能的下降, 进而对用户体验产生了负面影响。

在基于BLE技术的家居MESH网络中, 由于蓝牙设备的休眠机制, 导致与其相关的消息传输存在严重的实时性问题: 在蓝牙家具设备的点对多点通信过程中, 蓝牙家居设备可能会因处于休眠状态导致无法接收下行消息, 从而无法对家居设备进行实时控制或者实时控制效果不好。针对消息传输实时性问题, 本文首先分析了蓝牙终端现有的消息接入机制, 在此基础上, 提出了在现有蓝牙接入标准框架下引入TDMA (Time Division Multiple Access, 时分多址) 的消息接入机制, 提高对蓝牙家居设备的实时控制性能。论文利用离散时间马尔科夫模型对该接入机制进行了理论分析与优化, 并在开源协议栈上完成了软件实现以及测试验证, 验证了优化后机制能够在保证消息传输可靠性的前提下降低传输时延。

在基于Wi-Fi技术的家居网络中, 存在大量智能家居设备通过单个接入点接入情况。但WiFi系统中DCF (Distributed Coordination Function, 分布式协调功能) 信道接入机制在大量终端设备同时接入信道时会出现大量碰撞, 导致接入成功率和系统吞吐性能下降情况。针对这种情况, 本文提出了基于流量均衡受控的并行随机接入机制, 通过在AP (Access Point, 接入点) 侧引入负载均衡和多窗口并行接入机制, 来降低冲突概率, 从而提高WiFi在家庭场景下的接入性能, 在此基础上, 论文在不同操作系统平台的网卡驱动上进行了软件设计实现, 通过实际测试验证了方案能够有效降低冲突概率并提高系统吞吐率。

**关键词:** BLE MESH; 802.11 协议; 智能家居; 信道接入机制; 马尔科夫模型

## Abstract

The smart home device network composed of various wireless communication technologies is providing people with comfortable, convenient and energy-saving home experience. BLE (Bluetooth Low Energy) technology is widely used in smart home networks because of its characteristics of low power consumption and low cost, and Wi-Fi technology is widely used in smart home networks because of its characteristics of high bandwidth and wide signal coverage. However, with the rapid development and popularization of smart home industry, a large number of home devices are added to the network, resulting in the decline of system throughput and delay performance, which has a negative impact on user experience.

In the home MESH network based on BLE technology, due to the hibernation mechanism of Bluetooth device, the message transmission related to it has serious real-time problems: During point-to-multipoint communication of the Bluetooth furniture device, the Bluetooth home device may not receive downlink messages because it is in hibernation state. As a result, the home device cannot be controlled in real time or the real-time control effect is poor. Aiming at the real-time problem of message transmission, this thesis firstly analyzes the existing message Access mechanism of Bluetooth terminal. On this basis, it proposes to introduce the message access mechanism of TDMA (Time Division Multiple Access) under the existing Bluetooth access standard framework. Improve the real-time control performance of Bluetooth home devices. This thesis uses discrete time Markov model to carry out theoretical analysis and optimization of the access mechanism, and completes the software implementation and testing on the open source protocol stack, verifying that the optimized mechanism can reduce the transmission delay on the premise of ensuring the reliability of message transmission.

In a home network based on Wi-Fi technology, a large number of smart home devices are connected through a single access point. However, in the WiFi system, the DCF(Distributed Coordination Function) channel access mechanism will have a large number of collisions when a large number of terminal devices access the channel at the same time, resulting in the decline of access success rate and system throughput performance. In view of this situation, this thesis proposes a controlled parallel random access mechanism

based on traffic balance. By introducing load balancing and multi-window parallel access mechanism on AP(Access Point) side, it reduces the conflict probability, thus improving the access performance of WiFi in the home scene. On this basis, the software is designed and implemented on the network card drivers of different operating system platforms. The actual test verifies that the scheme can effectively reduce the conflict probability and improve the system throughput.

**Key words:** BLE MESH, 802.11, Smart Home, Channel access, Markov model

# 华中科技大学硕士学位论文

---

## 目 录

摘 要.....	I
Abstract.....	II
目 录.....	IV
1 绪论.....	1
1.1 研究背景与意义 .....	1
1.2 国内外研究概况 .....	2
1.3 论文的研究内容 .....	6
1.4 论文组织与结构 .....	7
2 BLE MESH 以及 Wi-Fi 背景技术概述 .....	8
2.1 BLE 技术 .....	8
2.2 BLE MESH 技术.....	14
2.3 Wi-Fi 信道接入技术 .....	19
2.4 本章小结.....	24
3 智能家居场景中 BLE MESH 接入优化研究 .....	25
3.1 BLE MESH 接入需求与目标.....	25
3.2 BLE MESH 现有接入机制及其缺陷.....	26
3.3 BLE MESH 接入机制优化技术路线.....	33
3.4 基于 TDMA 的 BLE MESH 群组可靠消息接入机制.....	34
3.5 不同接入机制的仿真与分析对比 .....	40
3.6 本章小结.....	44
4 智能家居场景中 BLE MESH 优化接入设计与实现 .....	45
4.1 基于 TDMA 的 BLE MESH 群组可靠消息接入机制的方案设计.....	45
4.2 基于 TDMA 的 BLE MESH 群组可靠消息接入机制的软件设计.....	53

# 华中科技大学硕士学位论文

---

4.3 实验结果与分析 .....	58
4.4 本章小结.....	62
<b>5 智能家居场景中 IoT Wi-Fi 优化接入的设计与实现.....</b>	<b>64</b>
5.1 IoT Wi-Fi 接入需求及现有机制问题分析 .....	64
5.2 IoT Wi-Fi 接入机制优化技术路线 .....	66
5.3 流量均衡受控并行随机接入机制 .....	67
5.4 对部分终端透明的流量均衡受控并行随机接入方案设计 .....	74
5.5 对部分终端透明的流量均衡受控并行随机接入方案软件实现 .....	79
5.6 实验结果与分析 .....	82
5.7 本章小结.....	86
<b>6 总结与展望 .....</b>	<b>87</b>
6.1 本文工作总结.....	87
6.2 不足与展望.....	88
<b>致 谢.....</b>	<b>90</b>
<b>参考文献.....</b>	<b>91</b>



## 1 绪论

### 1.1 研究背景与意义

随着科学技术以及人类生产力不断进步,智能家居技术应运而生。智能家居将科技融入日常生活,以较低的成本实现了生活品质提升。生活智能家居与普通家居相比更具舒适性、便携性以及节能性,现如今其应用已随处可见。

世界发达国家智能家居业的发展较早,技术相对成熟。我国的智能家居产业受技术普及、居民经济状况等问题的影响较大,发展速度一般。考虑到我国人口的基数,预计未来市场规模会在 2000 亿以上<sup>[1]</sup>。中国的智能家居市场吸引了大量的国内外公司加入市场。跨领域公司加入市场也带动了与科研机构之间的合作,推动了传统家居的“智能化”以及新型智能家居的研发,促进了智能家居产业的发展。

从智能家居的产品形态上来看,家居设备的“单品智能化”逐渐向“系统智能化”发展,也即家居设备之间依靠通信技术进行数据互通并形成了一个完整系统。考虑到智能家居内部其他设备与路由器、智能中继的连接灵活性问题,无线局域网通信方式成为智能家居发展的趋势<sup>[2]</sup>。智能家居通过互联网将所有不同种类的电子设备连接起来,这些设备通过不同协议来实现通信<sup>[3]</sup>。目前智能家居系统中没有一个统一的通信协议进行数据互通,通常依据不同的使用场景来使用不同的通信协议进行通信。对于需要高带宽应用的场景,常使用 Wi-Fi,对于需要低功耗的场景,则使用 BLE 或是 ZigBee。

基于 BLE 的 MESH 组网技术的提出使得 BLE 设备的传输距离大范围延伸。BLE 技术的互操作性好、功耗低等特点也使其在中距离 IoT (Internet of Things, 物联网) 通信中受到青睐<sup>[4]</sup>。为 BLE MESH 网络中的节点设备设置一定的休眠占空比能有效降低能耗、延长网络生命周期以及优化用户体验。但同时也使得网络中消息传输的端到端时延非常大,影响了网络中节点间的及时有效通信<sup>[5]</sup>。

在家庭 Wi-Fi 网络中,通常只会部署一个无线路由器来满足不同设备的联网需求。在当前使用的主流 Wi-Fi 标准下,大量设备通过单个无线路由器同时接入将导致

设备带宽受限、时延增大、掉线等问题。这些问题在影响了智能家居设备的同时还会对移动终端设备如手机、平板及笔记本电脑等的使用体验产生影响，出现浏览网页、看视频卡顿等问题。因此需要提出更好的解决方案来提高 Wi-Fi 网络的性能。

上述问题极大影响了智能家居用户的体验，阻碍了智能家居的发展。本文基于实验室与某知名企业的研究合作，针对上述智能家居场景中无线通信技术的实际应用问题提出了理论分析以及解决方案。

## 1.2 国内外研究概况

### 1.2.1 智能家居通信技术研究现状

随着人们对生活质量要求的不断提高，智能家居的普及将是时代发展的潮流，可以预期不久的未来，智能家居将会步入寻常百姓家中<sup>[6]</sup>。家居场景下的各种智能设备将依靠通信技术进行连接组网，组建起一个有效的日常管理系统，为用户提供安全便利的生活环境。

智能家居场景中的通信技术按照通信介质分类可被分为有线通信技术以及无线通信技术。使用无线通信技术的智能家居系统自带扩展功能而且成本较低。因此除开在某些特定场景下，为了最大程度保留设备连接的灵活性，智能家居组网一般采用无线通信方式。智能家居场景中常用的无线通信技术对比如表 1-1 所示。

表 1-1 几种无线通信技术对比表

规范	工作频段	速率	业务类型	通信距离	普及度
BLE	2.4GHz	1Mbps/2Mbps	数据、语音	300m	高
Wi-Fi	2.4GHz/5GHz	9.6Gbps	数据、多媒体	100m	高
ZigBee	2.4GHz	250Kbps	小数据量	75m	中

智能家居无线通信技术研究近年来备受关注。研究人员不仅关注传统的 Wi-Fi 无线通信技术，也将目光投向了 ZigBee、Z-Wave 等通信技术的应用<sup>[7]</sup>。通过各种通信协议、传感器和网关构建了智能家居联网通信系统，实现家庭设备的自动化管理。同时，研究人员还探索了如何保障智能家居的网络安全、数据隐私等问题，并提出了一些保护措施<sup>[8,9]</sup>。此外，许多研究还关注如何提高智能家居的能源使用效率，如通

过 MQTT 协议实现的能源管理系统<sup>[10]</sup>。总之，智能家居无线通信技术目前已经取得了很大的进展，但是仍需要进一步研究和发展，以实现更加智能、高效和安全的智能家居系统。

随着智能家居的加速发展，通过无线通信技术组网的智能设备数量迅速增加。大量设备接入网络使得设备间互相干扰的几率大大增加。智能家居场景中的无线通信技术应用问题也显露出来。

## 1.2.2 蓝牙通信技术研究现状

如今蓝牙技术日趋成熟，在不同领域、场合都有着广泛的应用<sup>[11]</sup>。蓝牙技术在日常生活工作中已无处不在，几乎所有的移动终端设备、汽车，甚至是家居都已经配备了蓝牙技术<sup>[12]</sup>。相关调查数据显示，至 2023 年全球蓝牙设备出货量将增长到 54 亿台。

蓝牙技术正在不断颠覆我们与设备互动、设备与设备互动的方式，提供了包括如定位、数据传输、音频传输、网络设备等服务。2007 年，随着 Wibree 并入蓝牙联盟，作为蓝牙技术的扩展，Wibree 成为蓝牙规范组成部分之低功耗蓝牙。2016 年 12 月，蓝牙 5 发布，兼容现有蓝牙版本，支持传统蓝牙、高速蓝牙和低功耗蓝牙三大技术，重点提升蓝牙低功耗技术在传输速度、传输距离、广播容量等方面的性能<sup>[13]</sup>。随着低功耗蓝牙技术的出现，蓝牙通信已经走入了更多应用。除了传统的点对点连接，蓝牙网络技术也得到了很大的发展。其中，BLE MESH 是一种创新的网络技术，使各种设备能够在网络范围内联接并进行信息交换。

基于 BLE 的首个多跳通信协议作为 BLE 服务被提出，在确认了可行性的同时保证了可移植性<sup>[14]</sup>。BLEmesh 采用机会路由代替泛洪路由，减少了不必要的广播，降低了整体能源消耗。BLEmesh 基于通告广播，是一种通过广播和机会路由将信息从一个节点传输到目的地的协议<sup>[15]</sup>。MRT-BLE 是基于 BLE 开发出的实时协议，允许在网状网络上实现有界数据包延迟，并提供优先级支持<sup>[16]</sup>。这些解决方案因缺乏标准化网络技术所特有的互操作性，从而导致了兼容性问题，限制了这些技术在市场上的渗透<sup>[4]</sup>。蓝牙首个正式 MESH 技术基于 CSR 剑桥大学研发团队于 2014 年提出的 CSR MESH<sup>[17]</sup>。基于该协议的正式 BLE MESH 协议自发布以来在不同场景下已被广

泛应用，并成为大型设备网络的必然选择。考虑到 MESH 协议实现还不够完善，实际应用需要在该协议基础上进行修改以适配不同设备、不同场景的应用。

近年来，不少研究关注开发基于 BLE MESH 的具体应用，如实时群组监控<sup>[18]</sup>、室内定位<sup>[19]</sup>等。同时，BLE MESH 网络的安全方面也得到了广泛关注，为保障网络和数据的安全，不少学者提出了多种保护措施。此外，通过仿真和性能分析，研究者确认了 BLE MESH 在医疗监测系统、物联网等方面的应用效果<sup>[20]</sup>。

总之，蓝牙技术和 BLE Mesh 目前的研究方向主要集中于网络实现、设备连接、安全保护、性能优化等方面，其应用前景也非常广阔。在对 BLE MESH 技术的研究中，并没有很多具体的研究来评估实际应用场景下的性能，显示出局限性，其中大部分是基于仿真环境<sup>[21]</sup>。

## 1.2.3 Wi-Fi 通信技术研究现状

IEEE 802.11 委员会于 1997 年发布了 802.11 的第一个版本<sup>[23]</sup>。该版本规定了 OSI 模型中的 PHY (Physical, 物理) 层和 MAC (Medium Access Control, 媒体接入控制) 层，允许设备之间以每秒 2Mbps 的速度无线传输数据。802.11 首个协议的提出为 Wi-Fi 及蓝牙技术的发展奠定了基础。

在 Wi-Fi 协议的发展过程中，为了满足 IoT 应用的需要，802.11ah 基于 802.11ac 被提出。802.11ah，又名 HaLow，具有低功耗、长距离、高可靠的特点。802.11ax 也是面向 IoT 场景进行设计的，支持大量设备进行高密度连接<sup>[24]</sup>。两者区别在于，802.11ah 是面向户外 IoT 场景传感器，追求更低功耗。802.11ax 则面向室内 IoT 场景，提供低延迟高可靠传输，并尽量减少功耗。

表 1-2 主流 Wi-Fi 版本相关信息

标准版本	802.11n(Wi-Fi 4)	802.11ac(Wi-Fi 5)	802.11ax(Wi-Fi 6)
工作频段	2.4GHz/5GHz	5GHz	2.4GHz/5GHz
最高传输速率	1.2Gbps	3.5Gbps	9.6Gbps
调制方式	64QAM OFDM	256QAM OFDM	1024QAM OFDMA
多用户	SU-MIMO	DL MU-MIMO	DL/UL MU-MIMO

目前主流的 Wi-Fi 协议版本的相关信息汇总见表 1-2。考虑到成本因素，智能家

居设备一般使用旧版本 Wi-Fi 采用传统 DCF 模式进行信道接入。因此针对旧版本的 Wi-Fi 协议，提高其在高密度接入场景下的性能对用户体验的改善有重要意义。虽然 802.11ax 和 802.11ah 无法直接应用，但协议中的相关内容可以用于借鉴和参考。

智能家居场景作为典型的设备高密度接入场景，家居网络中往往只会部署单个 AP。在这样的单 AP 高密度接入场景下，大量 STA（Station，站点）通过同一个 AP 接入家居网络。随着 WLAN 中随机接入的设备数量增多，设备在接入信道时将更易产生碰撞，甚至会发生“信道接入风暴”。引起这种现象的最直接原因在于传统 DCF 模式中核心接入机制 CSMA/CA（Carrier Sense Multiple Access with Collision Avoid，带有冲突避免的载波侦听多路访问）使用的 BEB（Binary Exponential Backoff，二进制指数退避）算法。在 WLAN 中设备数量较少时，BEB 算法能够有效避免设备间接入冲突工作，而一旦网络中设备数量超过一定阈值，系统性能将大幅下降。现有的解决方法大多是基于对 BEB 算法的修改实现的。

BEB 算法的引入是为了避免多次冲突。网络中的每个 STA 在接入信道前都需要从自身维护的 CW（Contention Window，竞争窗口）中随机选取一个值，在退避该值大小的单位时间后进行信道接入。CW 会在 STA 竞争信道失败时翻倍，在竞争成功时回到初始最小值。通过合理地调整 CW 的值，使得 STA 成功竞争信道时的退避时间更短可以有效提高系统性能。

针对 CW 的调整方法，有研究者提出了一些工作。EIED<sup>[25]</sup>退避算法的具体实现为在 STA 成功传输时将 CW 指数减少，在失败时指数增加。GDCF<sup>[26]</sup>、HBAB<sup>[27]</sup>退避算法使用 STA 在过去一段时间内成功传输的百分比做参考来调整 CW。一种基于 AI 的 CW 更新策略让 STA 间采用特定的概率分布函数来选择退避计数<sup>[28]</sup>。除了上述对 CW 进行调整的算法之外，还有一些研究借鉴了 OFDMA(Orthogonal Frequency Division Multiple Access，正交频分多址)和 MU-MIMO（Multi-User Multiple-Input Multiple-Output，多用户-多输入多输出）技术。VC-UORA<sup>[29]</sup>借鉴了 CSMA/CA 和 OFDMA 的协议思想，STA 间的空口信道竞争被转化为可以避免的虚拟竞争，从而提高了频谱资源的利用率。MPR 是一种通过 CDMA（Code Division Multiple Access，码分多址）来区分不同用户的上行多用户的 MAC 协议<sup>[30]</sup>。除此之外，还有一种共享

TXOP (Transmission Opportunity, 传输机会) 实现的多节点并行传输机制<sup>[31]</sup>。

上述算法实现在单 AP 多 STA 高密接入场景下的性能提升存在其局限性, 且实现相对复杂。

## 1.3 论文的研究内容

本文内容基于与国内知名通信企业的校企合作项目。本文先后对基于 BLE MESH 和 Wi-Fi 的智能家居网络有大量设备接入时的性能下降问题展开优化研究。本文对现有接入问题进行了分析, 指出了其问题所在, 并提出了相应的解决方案, 最后通过仿真以及实际测试验证了解决方案的可行性。

BLE MESH 网络中的周期性休眠设备被称为 LPN (Low Power Node, 低功耗节点)。在单跳 BLE MESH 网络中蓝牙网关与一组 LPN 节点进行群组可靠消息交互的场景中, 群组可靠消息以及确认消息的可靠性及实时性是主要的性能指标。现有消息接入机制存在缺陷: 1) 下行方向上, 蓝牙网关发送的群组可靠消息有极大概率因为 LPN 休眠而丢失, 影响了消息实时性。2) 上行方向上, 确认消息使用随机退避机制避免碰撞, 但同时也增大了确认消息的时延。3) 随着网络规模增大, 设备间的消息碰撞概率也随之增大, 现有机制无法保证消息传输的可靠性以及实时性。

在确定消息传输可靠性的前提下, 本文将分析现有消息接入机制下消息传输实时性差的原因。并提出一种新的基于 TDMA 思想的接入机制, 在保证消息传输可靠性的同时, 尽可能低降低消息传输的时延。

针对大量智能家居设备通过单 AP 接入 Wi-Fi 网络的场景, 随着接入设备数量的增多, 智能家居设备所使用的 CSMA/CA 信道接入机制的信道竞争成功率会随之降低, 系统的时延和吞吐性能也会随之恶化。同时在智能家居场景中, 不同终端设备的流量特性并不一致。这些设备在同时接入信道时, 流量较少的设备可能会出现一直无法接入信道的情况, 从而引发信道接入的公平性和及时性问题。

在上述前提下, 本文提出的基于流量均衡的受控并行随机接入机制按照不同终端设备的流量特性给设备进行分组, 不同分组间的流量大小相近。为不同分组分配不同的时隙进行信道接入, 通过降低直接竞争信道的设备数量来避免系统性能恶化, 从

而提高用户体验。

## 1.4 论文组织与结构

本文共分为 6 章内容，不同章节的大致内容描述如下。

第一章绪论部分，首先介绍了在智能家居场景中对 BLE MESH 接入机制以及 Wi-Fi 接入机制进行优化研究的背景以及意义，然后分析了智能家居通信技术、蓝牙通信技术、Wi-Fi 通信技术的发展研究现状，然后介绍了本文的研究内容，最后阐述了全文的整体组织与架构。

第二章介绍本文研究问题相关的背景技术，首先分别对 BLE 和 BLE MESH 组网技术进行了介绍。然后介绍了 Wi-Fi 接入技术，主要包括了 Wi-Fi 传统的 CSMA/CA 信道接入机制以及针对该机制的改进。

第三章首先对实际的 BLE MESH 网络接入需求和目标进行了分析，之后使用离散时间马尔科夫模型对现有的接入机制进行了分析，然后基于现有机制的缺陷，提出了优化的技术路线，引出了新的基于 TDMA 思想的改进接入机制，最后对改进前后的机制性能进行了数值仿真分析对比，验证了改进接入机制的理论可行性。

第四章在第三章提出的优化接入机制的基础上提出了相应的设计实现方案，然后在开源的 Nimble 协议栈上提出了具体的软件实现方案，最后对比了实际测试结果与理论分析值以及原有的接入机制性能，验证了新的接入机制能够在更短时间内实现相同的消息传输可靠性。

第五章首先对智能家居场景中 IoT WiFi 的接入需求进行了分析，之后对现有接入机制在 IoT Wi-Fi 场景中的缺陷进行分析，并在该基础上引出了流量均衡受控并行随机接入机制，然后结合实际应用场景，提出了该机制在路由器侧实现的设计方案，并阐述了在不同网卡驱动上进行软件实现的具体细节。最后通过实际测试结果与不同接入机制进行对比，验证了实现方案的可行性。

第六章对本文工作内容进行了总结，提出了不足，并对下一步工作做出展望。

## 2 BLE MESH 以及 Wi-Fi 背景技术概述

本章介绍在 BLE MESH 网络和 Wi-Fi 网络中进行接入的相关技术背景。考虑到 BLE MESH 组网基于 BLE 技术，本节分别对 BLE、BLE MESH 以及 Wi-Fi 的信道接入技术进行介绍。

### 2.1 BLE 技术

#### 2.1.1 BLE 协议架构

BLE 协议的层级架构如图 2-1 所示：

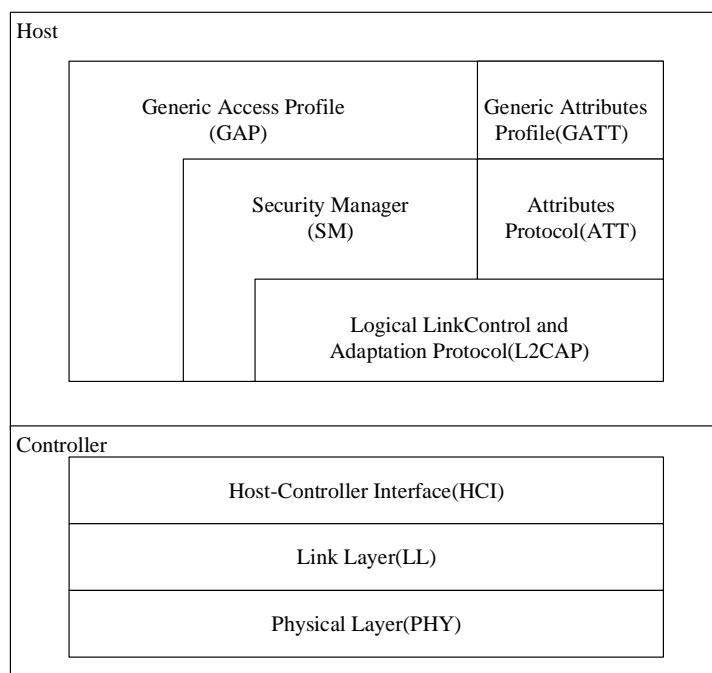


图 2-1 BLE 协议层级架构

实现协议的代码被称为协议栈，根据协议栈实现位置不同，可以将 BLE 协议栈划分为主机 Host 和控制器 Controller。Host 部分在主控芯片上执行，Controller 部分则在蓝牙芯片模块上被加载执行。不同层级的协议栈功能如下：

(1) PHY 层 (Physical Layer 物理层)：用来指定 BLE 所用的无线频段，调制解调方式和方法。

(2) LL 层 (Link Layer 链路层)：BLE 协议的核心。用来解决通信实体的频段



共享问题、构建两个通信实体的逻辑链路、提供数据传输过程中的校验、重传机制以确保数据传输可靠。

(3) HCI 层 (Host Controller Interface, 主机控制接口): 在 BLE 协议栈实现在双芯片方案下时出现。用于实现 Host 与 Controller 间的通信。

(4) L2CAP 层 (Logical Link Control and Adaptation Protocol, 逻辑链路适配协议): 对 LL 层来的数据进行区分, 分为加密通道或是普通通道。同时对连接间隔进行管理。

(5) SMP (Logical Link Control and Adaptation Protocol, 安全管理协议): 实现 BLE 的加密和安全管理, 实现连接安全性的同时保证用户体验。

(6) ATT (Attribute Protocol, 属性协议): 定义用户命令及命令操作的数据。

(7) GATT (Generic Attribute Profile, 通用属性配置): 规范 ATT 层定义的数据, 运用分组的概念进行分类管理。

(8) GAP 层 (Generic access profile, 通用访问配置): 简单的对 LL 层的数据进行部分规范和定义, 主要用于广播、扫描和发起连接等。

## 2.1.2 物理信道

蓝牙技术工作于 2.4~2.48GHz 频段, 该频段被 ISM 定义为 40 个射频信道。这 40 个射频信道都有特定的信道索引, 同时按被划分为两种功能信道:

(1) 主要广播信道: 3 个射频信道, 用于初始广播和遗留的广播活动。

(2) 通用信道: 37 个射频信道, 用于大多数的通信活动。

不同射频信道的功能划分见表 2-1, 上述两种功能信道用于四个 LE 物理信道: 广播、周期、等时及数据。其中广播将同时使用主要广播信道和通用信道用于发现设备、发起连接以及广播数据。这种情况下主要广播信道组成“主要广播物理信道”, 通用信道组成“次级广播物理信道”。其他的 LE 物理信道只使用通用信道。

两个设备通信的前提是使用共享一个物理信道。为了实现这个, 它们的收发机必须同时调到同一个射频信道。链路层在一个时刻只能使用一个物理信道。虽然蓝牙使用跳频收发, 但由于射频信道数量有限, 且在同一时空条件下, 不同的蓝牙设备独立操作, 它们调到同一射频信道的概率很大, 将导致物理信道冲突。为了减轻这样

的冲突，在物理信道上的每次传输将开始于一个访问地址（Access Address）。访问地址是设备调转到射频信道的相关代码，是物理信道的相关属性，每个被传输的消息前面都将包含一个访问地址。根据 Access Address 的不同，数据包被区分为广播包和数据包。

表 2-1 蓝牙物理信道划分

射频信道 (中心频率)	物理信道索引	主要广播信道	通用信道
2402MHz	37	●	
2404MHz	0		●
2406MHz	1		●
...	...	...	...
2424MHz	10		●
2426MHz	38	●	
2428MHz	11		●
2430MHz	12		●
...	...	...	...
2478MHz	36		●
2480MHz	39	●	

### 2.1.3 链路层状态机

作为 BLE 协议的核心部分，LL 层定义了使用 BLE 通信技术的设备如何通过物理信道建立起逻辑链路，并解释了设备在进行数据传输时如何使用信道以及如何保证数据传输可靠等问题。LL 层的动作可以由一系列状态描述，这些状态描述如下。

(1) 待命状态 (Stanby): 处于该状态的链路层不传输也不接收任何数据包。

(2) 广播状态 (Advertising): 处于该状态的链路层能够在广播信道上传输广播数据，并且能够监听并回复由这些广播数据触发引起的回应。处于广播状态下的链路层也被称为广播者 (Advertiser)。

(3) 扫描状态 (Scanning): 处于该状态的链路层能够监听由广播设备发出的广播数据。处于扫描状态下的链路层也被称为扫描者 (Scanner)。

(4) 启动状态 (Initiating): 能够监听固定设备发出的广播数据并回复这些数据, 由此来建立起与其他设备的连接。处于启动状态下的链路层也被称为启动者 (Initiator)。

(5) 连接状态 (Connection): 设备处于连接状态之下。围绕该状态, 定义了两个角色:

- a) 中心角色: 可与一个处于外围角色的设备进行通信, 并定义传输时序。
- b) 外围角色: 可与一个处于中心角色的设备进行通信。

(6) 同步状态 (Synchronization): 处于该状态的链路层将监听正在发送周期性广播的指定设备。该设备将发送周期性的物理信道数据包, 这些数据包组成了特定的周期性广播序列。

(7) 同步广播状态 (Isochronous Broadcasting): 处于该状态的链路层将在等时物理信道上传输等时数据包。处于一步广播状态的链路层被称为同步广播者 (Isochronous Broadcaster)。

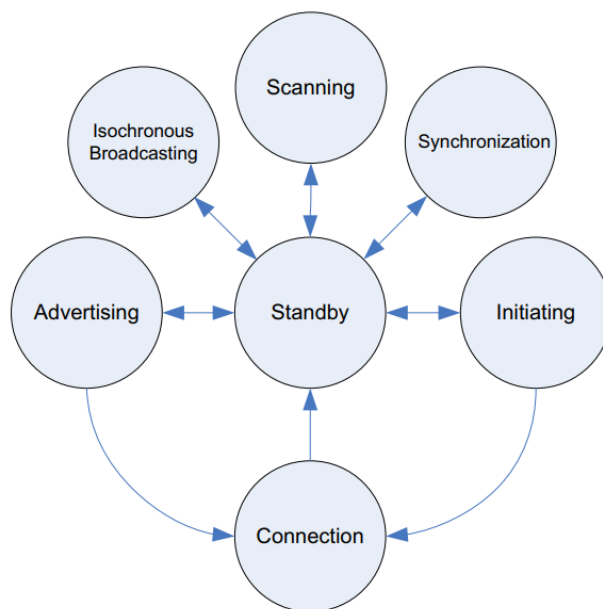


图 2-2 BLE 链路层状态转换关系

不同状态间的转换关系见图 2-2。在协议栈对链路层的实现中, 链路层可能会有多个链路层状态机的实例。其中至少存在一个支持广播态和扫描态。

## 2.1.4 广播消息的收发

BLE MESH 网络中的设备进行消息交互无需提前建立连接，并通过 BLE 广播消息的收发来完成消息交互，这里对广播消息的收发技术背景作出介绍。

### 1. 广播消息的发送

BLE 协议将每一次广播数据发送定义为一次事件，其中包含了一个或是多个主要广播信道上发送的广播数据包集合。广播数据包集合包含了在三个主要信道 37、38、39 上发送的 PDU（Protocol Data Unit，协议数据单元）。

广播事件按照是否支持某些功能进行分类，这些功能包括可连接/不可连接、可扫描/不可扫描、定向/不定向以及在蓝牙 5.0 之后引入的扩展广播。这里重点介绍 MESH 网络中节点进行消息交互的不可连接、不可扫描、不定向的广播事件类型（ADV\_NONCONN\_IND）。广播者在进行数据广播时的时序如图 2-3 所示。

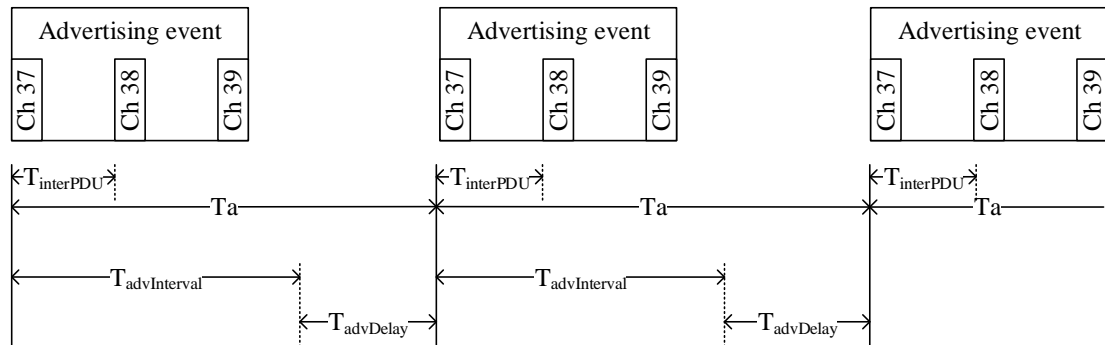


图 2-3 BLE ADV\_NONCONN\_IND 类型广播事件发送时序

对于这种广播事件，两个连续的广播事件起始点时间差值表示为

$$T_{advEvent} = T_{advInterval} + T_{advDelay} \quad (2-1)$$

其中  $T_{advInterval}$  应该为 0.625 的整数倍，且范围在 20ms 至 10485.759375s。  $T_{advDelay}$  是一个介于 0 到 10ms 之间的伪随机数，由链路层为每个广播事件生成。  $T_{interPDU}$  为同一个广播事件中两个 PDU 起始传输时间的间隔，范围为 0 至 10ms。

在链路层已经在其中一个主要广播信道发送完成之后，将直接转移至下一个主要广播信道发送数据，而不会监听信道。更短的  $T_{interPDU}$  能够在理论上减少事件的传输时延，也能够减少与其他事件的碰撞概率。

### 2. 广播消息的接收

接收广播消息的一方需要保持链路层状态机实例之一处于扫描者角色。链路层在三个主要广播信道上进行扫描以接收广播消息。扫描可分为主动扫描和被动扫描。

(1) 被动扫描：链路层将只接受数据包，不发送任何数据。

(2) 主动扫描：链路层将监听广播 PDU。基于广播类型，链路层可能会请求广播者发送额外信息。

扫描过程有两个主要参数：

(1) 扫描窗口 (ScanWindow)：链路层在一个主要广播信道上监听的时间。

(2) 扫描间隔 (ScanInterval)：两个连续 ScanInterval 间的开始时间间隔。

链路层应该按照主机的指示在每个扫描间隔监听完整的扫描窗口，除非有调度冲突。在每个扫描窗口中，链接层应该扫描与前一个扫描窗口中使用的不同的主要广播信道索引。扫描顺序总是三个主要广播信道 37、38、39，具体扫描过程见图 2-4。

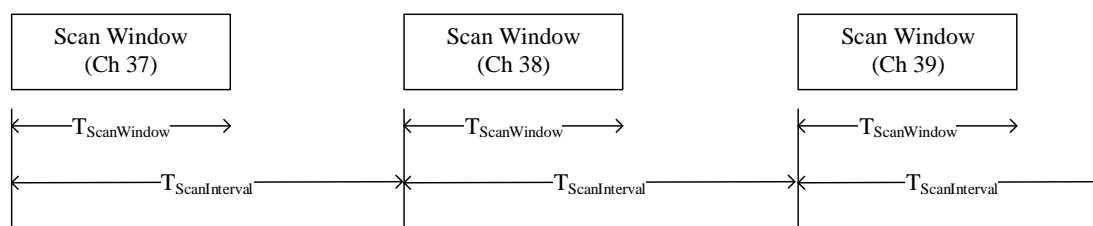


图 2-4 信道扫描时序

实际场景下，广播消息的接收过程如图 2-5 所示。

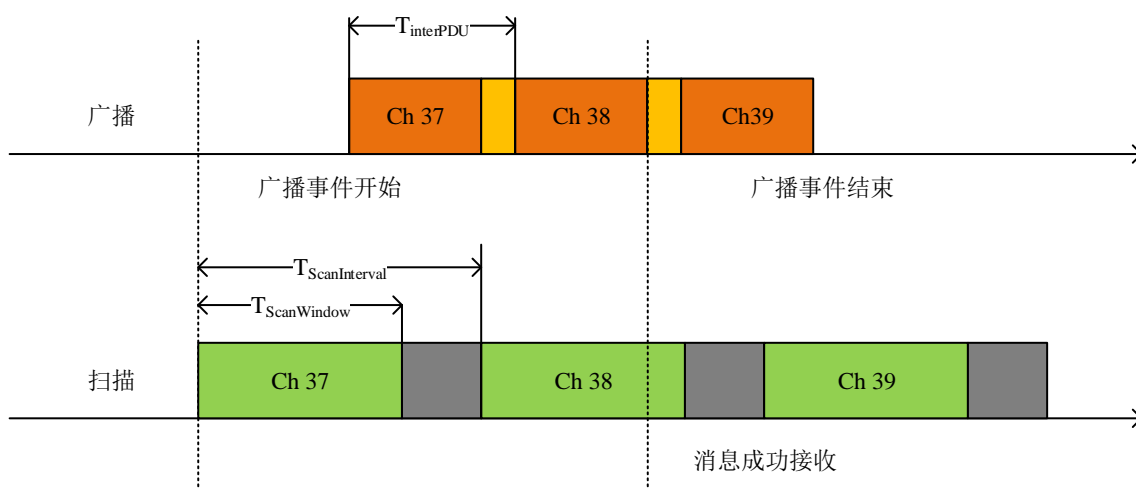


图 2-5 BLE ADV\_NONCONN\_IND 类型广播消息的收发过程

扫描窗口和扫描间隔应该小于 40.96 秒。扫描窗口应该小于或等于扫描间隔。特

殊情况下,如果主机设置的扫描窗口和扫描间隔参数相同,则链路层将在三个主要广播信道上连续扫描。若扫描事件的间隔设置的过小,频段切换得过于频繁,那么扫描不到三个广播事件的概率会增大。若扫描事件的间隔设置的较大,频段切换放缓,那么接收者更有可能收到三个数据包中的其中一个。但同时由于设备在一个信道中停留的时间过长,可能会导致传输失败。

## 2.2 BLE MESH 技术

### 2.2.1 BLE MESH 组网概述

传统局域网按照结构划分可分为星形结构、树形结构、总线结构等。目前典型的局域网最常采用的网络结构为星形结构。星形结构以中央节点作为核心,其他节点连接在中央节点之上。中心节点承担着为其通信节点建立物理连接的责任。这种网络结构有着延迟小、结构简单且便于管理的特点。但随着局域网中设备数量爆炸式增长,中央节点负担大大加重,一旦中央节点发生故障,整个局域网将受到影响<sup>[32]</sup>。

Mesh 网络是一种新型的多跳无线网络技术<sup>[33]</sup>。自诞生以来,Mesh 网络因为在有线网络中布线复杂等特点,而没有受到人们的重视。但随着无线时代的到来,Mesh 网络的布局变得相对容易起来。Mesh 网络也重新回到人们的视野中。在无线 Mesh 网络中的每个节点能够与其他设备节点建立对等的连接,同时发送和接收数据。传统星形网络的拓扑结构和 MESH 拓扑结构如图 2-6。

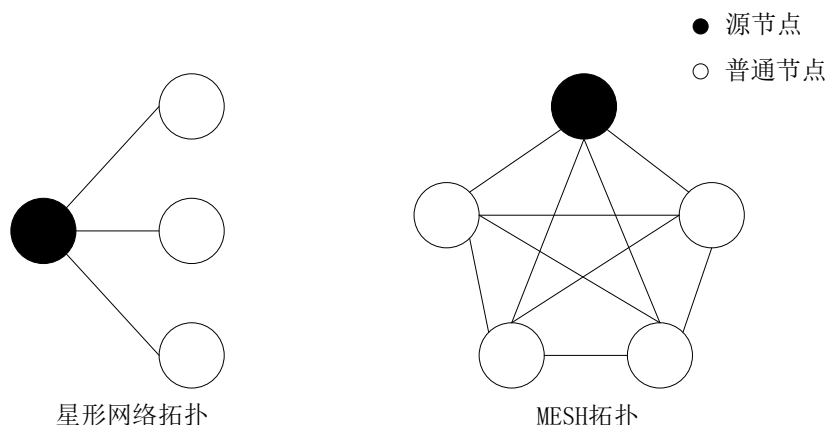


图 2-6 星形网络和 MESH 网络拓扑结构

相较于传统网络, MESH 网络具有如下特点:

- (1) 扩展性强, 新增节点在加入网络之后能够很快的进行配置并开始工作。
- (2) 稳定性高, 网络中的节点相对平等, 在某个节点出现故障时, 并不会影响到整个网络。
- (3) 覆盖范围大: MESH 网络能够实现数据的多跳传输, 通过数据的层层转发, 可以实现距离较远两个节点的相互通信。

在 BLE MESH 组网技术出现之前, 通过 BLE 建立的无线局域网大多是点对点的微微网。虽然蓝牙 5.0 在不增加功耗的前提下, 提高了传输距离、传输速度和广播数据传输量<sup>[34]</sup>。但有限的网络接入节点数目使得 BLE 技术在 IoT 领域发展受限。BLE MESH 是完全基于 BLE 协议栈的组网技术, 允许大量设备接入 MESH 网络, 并通过广播洪泛实现数据传输<sup>[35]</sup>。BLE MESH 组网技术与 BLE 4.0 之后的版本适配。所有经过认证的蓝牙智能设备都可以通过适当修改其软件/固件与 BLE MESH 网络进行交互。在 BLE MESH 网络中, 蓝牙设备可以不进行匹配连接就可以做到点对点通信。

## 2.2.2 BLE MESH 协议架构

BLE MESH 协议基于 BLE 协议定义了分层的协议架构, 如图 2-7 所示。

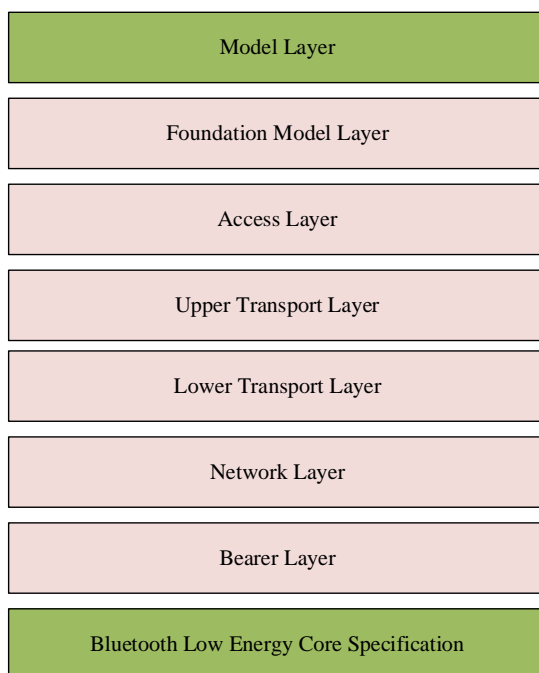


图 2-7 BLE MESH 协议层级架构

不同层次的协议负责实现 BLE MESH 组网的不同功能。

(1) 模型层 (Model Layer), 定义了标准化应用模型, 如照明和传感场景。

(2) 基础模型层 (Foundation Model Layer), 定义了用于配置和管理 Mesh 网络的模型。

(3) 接入层 (Access Layer), 定义了应用数据的格式, 控制数据加密/解密, 检测数据是否在正确的上下文和应用密钥。

(4) 上层传输层 (Upper Transport Layer), 实现对应用数据的加密、解密和认证。

(5) 下层传输层 (Lower Transport Layer), 实现消息分段和传输层 PDUs 的重组。

(6) 网络层 (Network Layer), 定义了传输层数据的寻址、消息中继/接收/拒接的规则、网络层数据格式以及网络层加密和认证。

(7) 承载层 (Bearer Layer), 定义了使用 ADV 或是 GATT 承载时交换数据的方式, 网络中的大多数数据通过 ADV 承载实现。

## 2.2.3 BLE MESH 组网基础概念

### 1. 设备与节点

在 BLE MESH 网络中, 根据是否已经加入到某个 MESH 网络中, 设备可以被分为未配置设备和节点。未配置设备将不能在 MESH 网络中收发数据, 但是可以向配网设备推送它的存在。配网设备将在未配置设备被认证之后邀请其加入 MESH 网络;

节点设备能够在 MESH 网络中收发数据。配网设备也能够将节点设备在 MESH 网络中移除, 使设备回到未配置设备的状态。

所有节点都具备数据收发的基础特性, 同时也可以通过配置来实现一个或使多个额外特性, 这些额外特性是:

(1) 中继, 接受和转发通过 ADV 承载发送的消息来形成更大的网络。

(2) 代理, 接受和转发在 GATT 承载和 ADV 承载间的消息。仅支持 GATT 承载的设备可以通过代理节点连入 MESH 网络。

(3) 低功耗, 功率受限的节点可能会利用低功耗特性来减少无线电接通时间并节省功耗。同时低功耗节点 (LPN) 可以与好友节点协同工作。



(4) 友谊, 功率不受限的节点很适合作为好友节点。好友节点能够存储发往 LPN 的消息和安全更新; 当低功耗节点需要时再将存储的信息传输至低功耗节点。

支持某个特性的节点可以打开或是关闭某个特性, 当打开时, 这些特性可能被使用也可能不被使用。对于支持低功耗的节点则必须打开这个特性, 并设置相应的好友节点。

有些节点功能较为复杂, 可能由一组相互独立的元素组成, 每个元素由定义节点功能和元素条件的实体组成。每个节点设备将至少具有一个元素。每个元素拥有单独的单播地址用于寻址。节点与元素的关系见图 2-8。

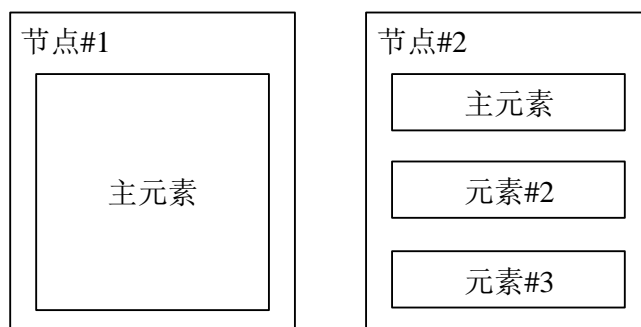


图 2-8 节点和元素的关系

## 2. 模型和状态

状态是用于表示元素条件的值, 例如最简单智能面板二值开关, 具有两个状态“开”和“关”; 更复杂的亮度状态, 具有 0~10 共 11 个离散状态。

用户需要通过网络来访问、修改或使用服务器的状态则需要通过 C-S (客户端-服务器) 的架构来完成。服务器暴露出元素状态的值, 而客户端则可以对元素状态进行操作。在 C-S 架构下, 产生了三种不同类型的模型:

(1) 服务器模型: 由一个或是多个跨越一个或多个元素的状态组成; 定义了一系列可以接收或是发送的消息; 定义与消息相关的元素行为。

(2) 客户端模型: 定义了一系列客户端用于请求、修改或是消耗服务器状态的消息; 客户端模型不包含状态。

(3) 控制模型: 包含了一个或是多个服务器模型; 包含了一个或是多个客户端模型; 包含一个或是多个用于协调模型间交互的控制逻辑。

根据设备功能需要, 同一个设备将可能同时包含服务器、客户端以及控制模型。

每个模型都有唯一标识符来确保其能被寻址到。蓝牙技术联盟（SIG）所定义模型标识符为 16 位，而供应商的模型标识符为 32 位。典型的 C-S 模型如图 2-9。

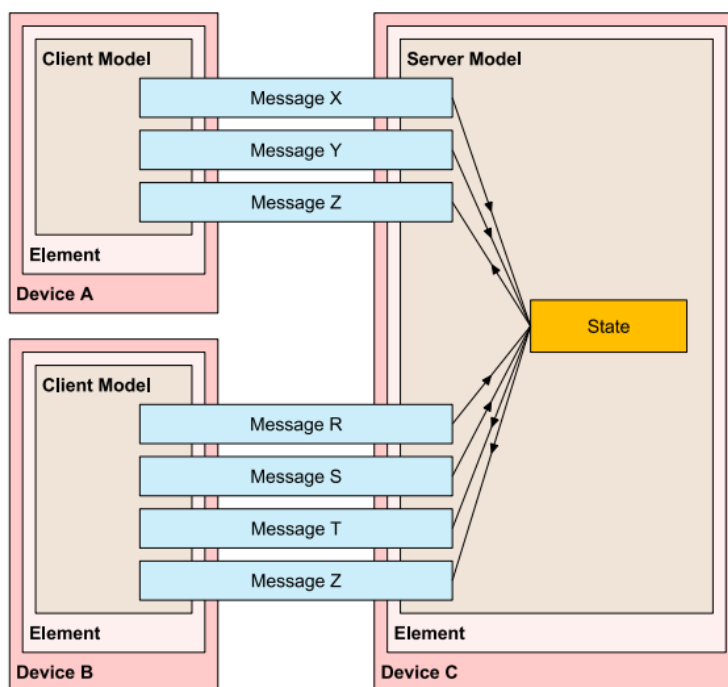


图 2-9 BLE MESH 网络中的典型 C-S 模型

设备 A 与设备 B 实现了客户端模型。设备 A 支持消息 X、Y、Z，设备 B 支持消息 R、S、T、Z。设备 C 实现了客户端模型，并定义了一系列状态。当设备 C 接收到来自设备 A、B 的消息时，将根据消息的内容对服务器状态进行相应的处理。

### 3. 发布/订阅和接入消息交互

BLE MESH 网络中的设备通过消息进行通信。消息可以分为控制消息和接入消息，其中控制消息为网络管理操作相关，接入消息用于实际的消息交互，该类消息允许客户端模型获取或设置服务器模型中的状态值，或被服务器用于报告状态值。接入消息按照是否需要确认又被分为可靠消息和不可靠消息。

(1) 可靠消息：每个接收元素需要在接收到可靠消息之后向发送者返回确认消息，内容通常为元素状态值。

(2) 不可靠消息：接收元素接收到不可靠消息无需做出任何响应。

BLE MESH 网络元素间的接入消息交互通过“发布/订阅”机制实现。当消息产生的一方需要发送时，若消息为某个可靠消息的确认消息，则使用可靠消息源地址作

为目的地址。若消息不是某个可靠消息的确认消息，而是元素主动发送的消息，则使用模型唯一的发布地址进行发送。该发布地址可能为单播、组播或是虚拟地址，其中组播和虚拟地址都可用于点对多点通信，除此之外还存在一种未定义地址表示未被分配的地址。

接收消息的一方中的每个模型实例会使用订阅列表来定义用于接收消息的有效地址。只要发送到模型订阅列表上的组地址或虚拟地址的消息到达，节点就会对其进行处理。当消息的目的地址是接收元素的单播地址，或者当消息的目的地址是该设备所属的固定组地址时，消息也会被处理。如果一个节点有多个元素，则在每个已寻址的元素上处理一次消息。接入消息的有效源地址和目的地址类型见表 2-2。

表 2-2 接入消息有效源地址和目的地址

地址类型	接入消息的有效源地址	接入消息的有效目的地址
未定义地址	No	No
单播地址	Yes	Yes
虚拟地址	No	Yes
组地址	No	Yes

## 2.3 Wi-Fi 信道接入技术

IEEE 802.11 系列协议定义了无线局域网标准，主要描述了无线局域网中的 MAC 层及 PHY 层标准。MAC 层定义了对传输媒介的接入方法，因此对 Wi-Fi 在无线局域网中的性能研究需要首先明确 MAC 层的接入机制。802.11 MAC 层的信道接入机制实现可以分为两种模式 DCF 与 PCF（Point Coordination Function，点协调功能），其中 DCF 使用名为 CSMA/CA 的信道接入机制实现对传输媒介的访问，PCF 作为 DCF 的补充，提供了一种无需竞争的接入方法。DCF 具有更加良好的分布式特性，应用更加广泛，PCF 的应用相对较少。本节重点介绍基于 CSMA/CA 的竞争接入及相关优化接入方法。

### 2.3.1 CSMA/CA 机制

当多个设备同时向网络发送数据时，数据包可能会发生碰撞，导致网络传输效率低下。CSMA/CA 通过在发送数据前进行预先的监听和等待，来尽可能地避免数据碰

撞，从而提高了无线网络的传输效率。基于 CSMA/CA 机制的 DCF 模式有分为基础模式及 RTS/CTS 模式两种不同子模式。

## 1. 基础模式

DCF 基础模式下的信道接入过程如图 2-10 所示，其过程可以简要描述如下：

1、需要发送数据的站点在发送前需要等待一个 DIFS（Distributed Inter-Frame Space，分布式帧间间隔）时间来监听信道状态。若信道在这段时间都为空闲的，站点从竞争窗口中随机取值作为退避值。

2、发送站点周期性检测空口是否空闲。退避值在空口空闲时递减，空口繁忙时不变。退避值减至 0 时，设备发送数据帧。

3、接收数据的站点在接收到数据之后需要等待一个 SIFS（Short Inter-Frame Space，短帧间隔），向发送站点回复一个 ACK（Acknowledgment，确认）帧。

4、发送站点接收到接收站点的 ACK 帧，本次数据帧传输成功。

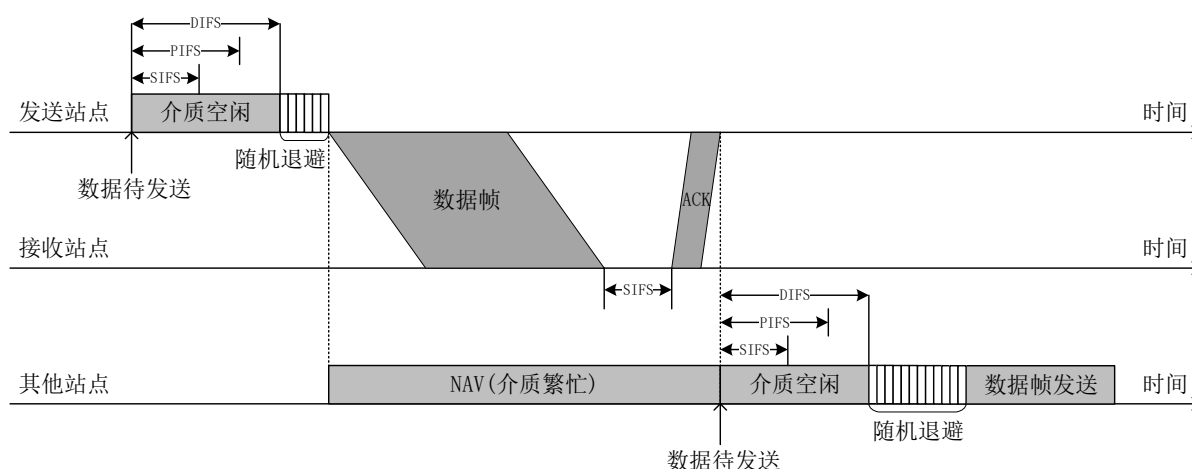


图 2-10 DCF 基础模式的信道竞争过程

若发送站点没有及时收到来自接收站点的 ACK 帧，站点认定数据帧丢失。发送站点在等待一个 EIFS（Extended Inter-Frame Space，扩展帧间间隔）之后，重新开始竞争信道重传数据帧。在接入设备较多的无线链路上，数据帧竞争接入信道的成功率也会相应降低，在多次重传竞争失败后，该数据帧的传输也被认定为失败。

## 2. RTS/CTS 模式

隐藏终端问题是指一个终端节点位于接收者的通信范围之内，而在发送者的通

信范围之外<sup>[37]</sup>。两个站点的发送范围无法互相覆盖，一个站点无法通过物理载波监听的方法判断另一个站点是否在发送数据，从而可能在基础模式下产生数据碰撞，影响信道利用率<sup>[38]</sup>。隐藏终端问题如图 2-11 所示。

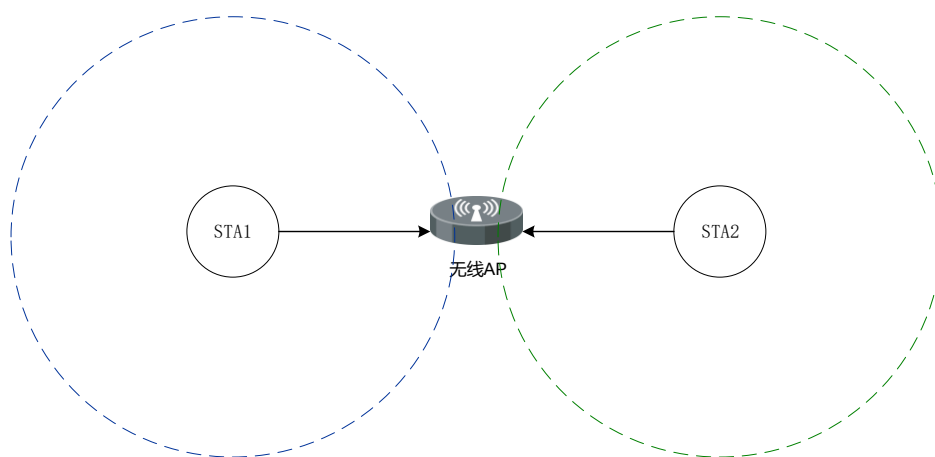


图 2-11 隐藏终端问题

为了解决隐藏终端问题，DCF 引入了 RTS/CTS 模式。实际场景中，设备通常根据需发送的数据量大小来决定使用何种模式。当需要传输的数据量较大时，DCF 使用 RTS/CTS 模式，否则使用基本模式。这是因为 RTS/CTS 模式能在大数据量的传输过程中降低发生冲突的概率，从而提升用户体验。DCF 下的 RTS/CTS 子模式信道接入流程如图 2-12 所示。

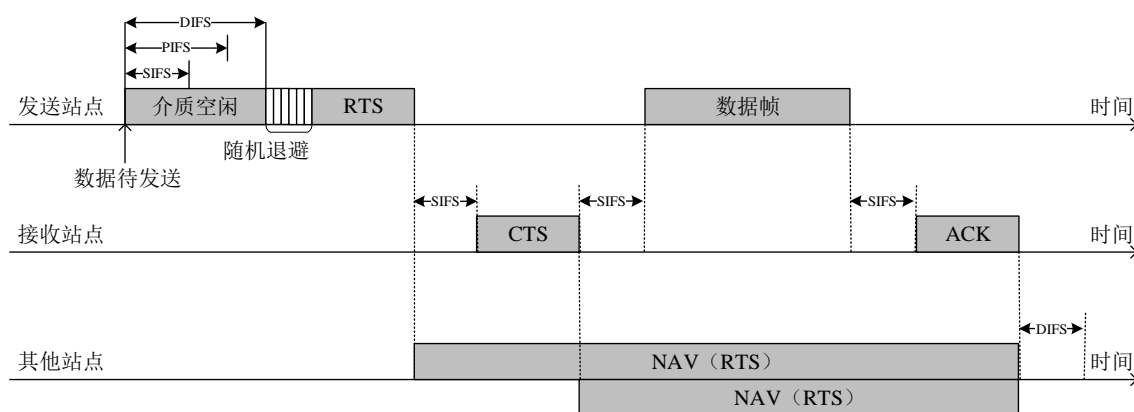


图 2-12 DCF RTS/CTS 模式信道接入过程

RTS 与 CTS 都属于不加密的管理帧，不同站点对接收到 RTS/CTS 的处理流程描述如下：

- 1、发送站点在发送之前监听信道一个 DIFS 的时长，当介质空闲时，从竞争窗

口中随机选取一个值作为退避值。

2、发送站点周期性检测空口是否空闲。退避值在空口空闲时递减，空口繁忙时不变。退避值减至 0 时，设备发送 RTS 帧。

3、接收站点接收到 RTS 之后等待一个 SIFS 的时间发送 CTS 帧。

4、其他站点收到 CTS 帧，发现这个帧目的地址与自身不符，会将 CTS 帧中的 duration 字段记录至本地开始进行退避。

5、发送站点收到 CTS 帧之后会等待一个 SIFS 的时间发送数据帧。

## 2.3.2 EDCA 机制

802.11 协议一开始并没有考虑到对不同业务的 QoS (Quality of Service, 服务质量保证)。随着 802.11 协议在商业场景中的广泛应用，Wi-Fi 被应用在了越来越多的场景当中。EDCA (Enhanced Distributed Channel Access, 增强的分布式信道接入) 机制在 802.11e 中被提出，实现了无线业务传输的 QoS。EDCA 与 CSMA/CA 接入机制的不同之处如图 2-13 所示。

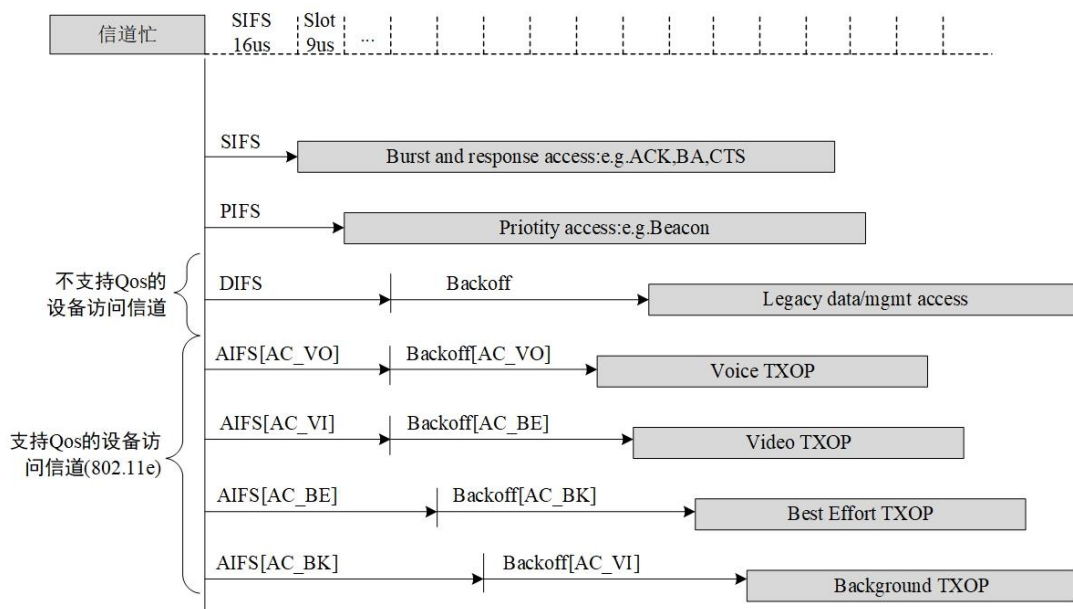


图 2-13 CSMA/CA 接入以及 EDCA 各 AC 接入的区别

EDCA 定义了四种不同类型的 AC (Access Category, 访问类别)，按照信道接入的优先级从高到低分别为 VO (Voice, 语音服务)、VI (Video, 视频服务)、BE (Best Effort, 尽力传输)、BK (Back Ground, 背景流量)。实际情况下，会根据应用需要来

选择使用的 AC。

(1) VO: 一般为 VoIP 流量类型, 对延迟最为敏感。

(2) VI: 对延迟较为敏感的流量类型。

(3) BE: 默认的无线流量类型, 如网页访问的流量。对延迟有一定要求, 但不会太敏感。

(4) BK: 对延迟要求最不敏感, 如文件传输, 打印作业流量。

EDCA 是在 CSMA/CA 的基础上进行修改来实现 QoS 的, 主要有以下几个方面的区别:

(1) 帧间间隔: EDCA 在 DCF 固定的帧间间隔的基础上做出了修改, 提出了 AIFS (Arbitration Interframe Space, 仲裁帧间间隔)。与 DIFS 不同, AIFS 的长度可以通过用户配置。具有更高接入优先级的业务将采用更短的 AIFS, 通过减少接入信道前的等待时间来保证业务传输性能<sup>[39]</sup>。

(2) 竞争窗口大小: CSMA/CA 使用了二进制指数退避机制来降低信道冲突的可能性。站点发送数据前检测到信道空闲了一段时间后, 会在 0 到 CW 之间的竞争窗口内选取一个随机数作为等待的时隙个数。每次冲突发生, CW 将会增大一倍, 但会存在一个上限值  $CW_{max}$ 。在成功传输时, CW 会回到最小值  $CW_{min}$ 。DCF 中的 CW 值相对固定, 而 EDCA 将  $CW_{min}$  和  $CW_{max}$  与 AC 绑定。AC 不同, 相应的 CW 范围也不同, 平均退避的时间也不同。由此实现了 QoS。

(3) 传输机会: TXOP 是站点通过 RTS/CTS 机制发送数据时, 可以与其他站点协商的可传输时间。这段时间内其他站点不能够接入信道<sup>[40]</sup>。与 DCF 机制中的固定  $TXOP_{Limit}$  (最大传输机会) 不同的是, EDCA 将其与 AC 关联起来。

(4) 重传次数: DCF 中的默认固定的数据帧最大重传次数为 7 次, 超过该限制将会做丢包处理。而在不同 AC 的最大重传次数在 EDCA 中被区分开来, 并可以进行修改。

### 2.3.3 RAW 机制

在 IoT 场景下, 设备数量多且集中导致了 Wi-Fi 性能的恶化。为此, 802.11ah 提出了 RAW 机制来缓解密集设备接入时的网络性能下降。RAW 机制采用了分组接入

的思想，限制了同时接入信道的设备数量来实现对网络性能的优化。

RAW 机制以 Beacon 帧间隔为一个周期，将一个周期划分为多个 RAW slot，给每个设备组单独分配一个 RAW slot 用于竞争接入。这种机制不仅降低了设备间冲突的概率，还能够有效降低能耗。因为设备除了在 RAW Slot 需要进行数据传输，其他时间可以进行休眠。

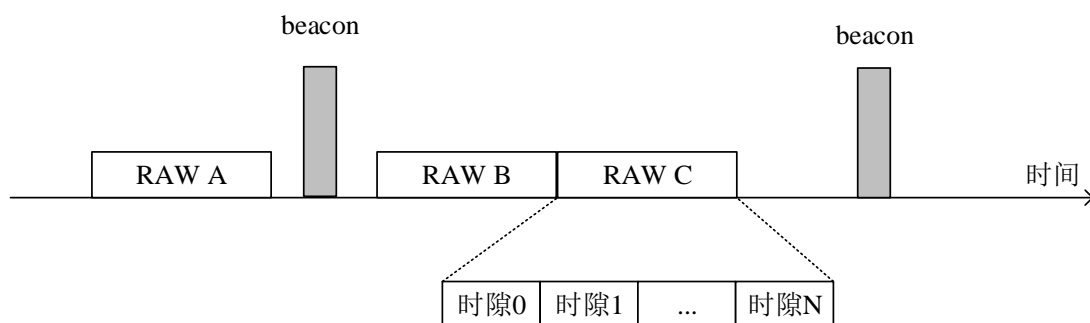


图 2-14 RAW 工作机制

RAW 机制分组时参考依据是设备数目，也就是等量地将设备均匀地分到不同的时隙上面。这种做法看似公平，实则有很大的不确定性。例如在智能家居场景中，设备的种类繁多会导致某些组内设备流量总和较小，而有些组内设备流量总和较大。这些流量相差大的组却使用相同长度的时隙，造成了空口资源分配的不公平性。在某些极端情况下，甚至会造成不如 CSMA/CA 机制的情况。

## 2.4 本章小结

本章介绍了在 BLE MESH 和 Wi-Fi 网络中进行信道接入的关键性技术。在 BLE MESH 网络中，节点间需要调用 BLE 广播进行消息交互。因此本文首先介绍了 BLE 协议的相关内容，包括 BLE 协议的基础架构、使用到的物理信道、链路层的状态机以及 BLE 广播消息的收发进行了介绍，之后对 BLE MESH 协议栈的架构和基本概念进行了介绍。针对 Wi-Fi 网络中信道接入问题，首先介绍了 802.11 DCF 模式中的核心信道接入机制 CSMA/CA 以及在后续协议中的改进信道接入机制 EDCA 以及 RAW。这些背景技术为后续接入机制的优化研究提供了技术支持。



## 3 智能家居场景中 BLE MESH 接入优化研究

BLE MESH 网关属于 Wi-Fi 网络的一部分, BLE MESH 网关则是作为 BLE MESH 网络的核心, 负责对 Wi-Fi 协议数据和 BLE MESH 协议数据进行数据转换。在规模日渐扩大的家居 BLE MESH 网络中, 网络节点设备需要周期休眠以降低功耗。休眠机制同时对现有接入机制下的消息传输可靠性和实时性产生了负面影响。本章针对上述网络中的点对多点消息接入展开优化研究。BLE MESH 网络中低功耗节点被称为 LPN, 这样的单跳 BLE MESH 网络的拓扑结构如图 3-1 所示。

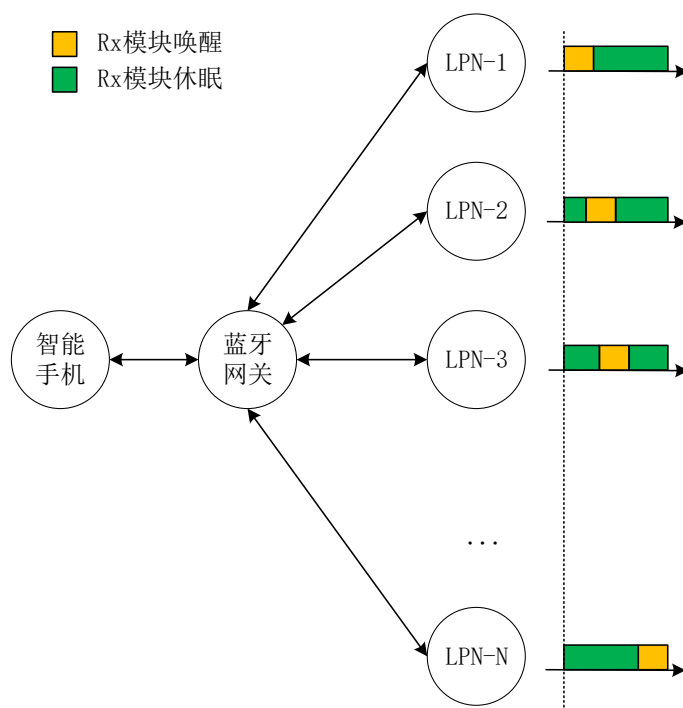


图 3-1 单跳 BLE MESH 拓扑结构

### 3.1 BLE MESH 接入需求与目标

基于 BLE MESH 的智能家居网络具有以下三个重要特征：

(1) 终端设备数量多。随着人们对家居生活的追求提高, BLE 的低成本以及互操作性好的特点使得通过 BLE MESH 接入网络的智能家居设备急剧增多, 网络中存在几十甚至上百个设备, 且数量还在增多。

(2) 终端设备具有休眠周期。为了控制网络的整体能耗, 网络中的家居设备常

常为通讯模块设置一定的休眠占空比。考虑到设备的主要功耗在持续扫描信道过程中引入，设备在休眠期间内关闭了信道扫描功能而无法正常接收消息。

(3) 消息传输的可靠性和实时性要求高。用户通过发送消息来控制智能家居设备。消息传输不可靠将导致用户控制设备失败。消息传输时延大将导致设备响应慢、用户侧状态显示慢等问题。实际场景中消息传输可靠性需要到达 99.9%，也即 1000 次消息传输最多失败一次，同时时延要求达到 200ms 之内。

在单跳 BLE MESH 网络中，对这样一组 LPN 进行群组可靠消息的传输应用逐渐增多。例如在基于 BLE MESH 的 WSN (Wireless Sensor Network, 无线传感器网络) 场景中，用户控制一组低功耗传感器设备对环境数据进行采集和上报，而在智能家居场景中的另一个常见应用是智能照明，用户对一组灯具进行开/关、亮度控制，灯具同时反馈自身的开/关、亮度状态信息给用户。

LPN 设备固有的休眠占空比属性对群组可靠消息以及确认消息传输的可靠性和实时性产生了负面影响。在不断扩大的 BLE MESH 网络规模中，在保证可靠性的前提下，如何降低群组可靠消息及其确认消息传输时延是一个亟待解决的问题。在单跳 BLE MESH 网络中 LPN 设备占空比一致的前提下，本文提出了群组可靠消息以及确认消息传输的优化接入机制。

## 3.2 BLE MESH 现有接入机制及其缺陷

### 3.2.1 现有 BLE MESH 接入机制

现有的对 LPN 的可靠消息接入机制主要分为两种，基于“友谊”特性的轮询机制以及基于网络层重传的消息接入。其中基于“友谊”特性的轮询机制通过 LPN 向好友节点轮询数据缓存实现对 LPN 的消息的可靠传递。这种机制在单跳 MESH 网络中 LPN 设备数量较多时会出现设备间“友谊”断连以及传输时延大等问题，因此常采用后一种接入方法。

网络层重传机制将本地生成的消息重传一定次数来实现可靠。通过网络层重传实现消息可靠传输的过程可以简单描述为：

- 1、网关在 LPN 休眠/唤醒全周期内持续发送一定次数，以保证 LPN 能正确接收

群组可靠消息。

2、LPN 接收到网关消息之后首先退避固定的网关发送时间，之后随机退避一段时间发送多次确认消息以保证网关正确接收。这两段退避时间的作用可以分别描述为：

(1) 固定退避时间：避免 LPN 设备的确认信息与网关发送的消息互相碰撞，导致未收到的 LPN 设备无法正确接收群组可靠消息。

(2) 随机退避时间：避免 LPN 设备间的确认消息互相碰撞。

同时，考虑到节点的中继功使得消息能够通过多条路径到达目的节点，能在一定程度上增加消息成功传输的概率，但同时也会使得不同节点消息碰撞的几率增加。在单跳 BLE MESH 网络中，随着节点数目增多，网络的传输性能将会大大恶化。因此在实际的单跳 MESH 网络实现中，常常关闭节点的中继功能。实际场景中网关设备与 LPN 进行群组可靠消息以及确认消息的交互过程如图 3-2 所示。

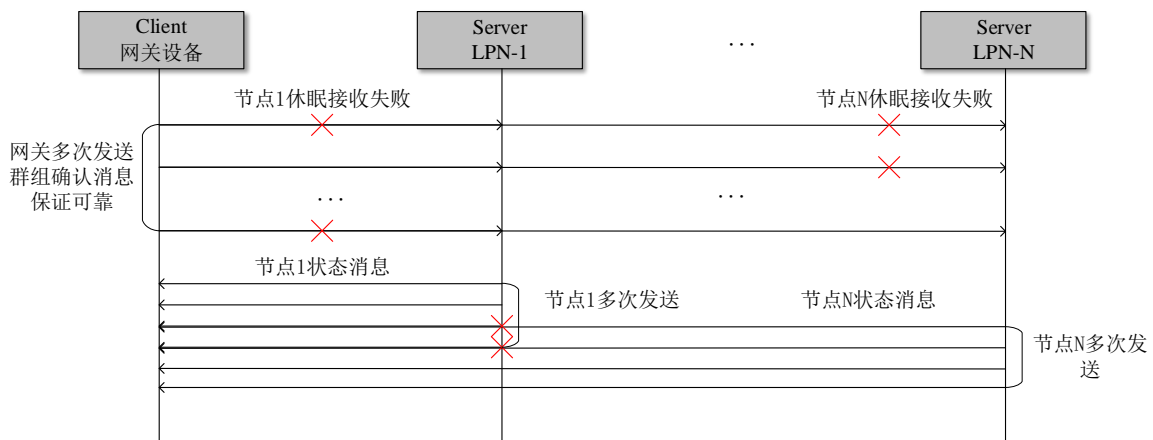


图 3-2 实际场景中的群组可靠消息交互流程

## 3.2.2 现有接入机制性能分析

本节对现有的基于网络层重传的接入机制进行性能的理论分析，在保证可靠性的前提下对群组可靠消息以及确认消息传输时延进行了分析，并对确认消息的碰撞概率进行了分析，数值仿真结果见 3.5 小节。

### 1. 群组可靠消息传输性能

考虑到 LPN 设备至少会退避固定网关设备发送的持续时长，可以认为网关在传输群组可靠消息时不会受到网络中其他设备的干扰，与其他设备发生碰撞的概率为

0。在不考虑 BLE 广播和扫描引起的消息丢失的情况下，对群组可靠消息的传输过程使用离散时间马尔科夫模型进行建模。

定义群组可靠消息的传输时延为网关第一次开始发送到所有 LPN 成功接收的时间。实现对群组可靠消息的可靠传输时间与消息的传输次数高度相关，通过分析群组可靠消息的传输次数可以得到消息的传输时延。在之前描述的消息传输系统中，所有 LPNs 将在消息重传的时间间隔内确定状态，因此 LPNs 的当前状态仅依赖于 LPNs 在一个传输间隔范围内的操作，符合马尔科夫性。

首先考虑 LPN 接收模块常开的情况。对系统做出如下假设

1. 网关设备在向 LPN 设备广播消息时，三个蓝牙主要广播信道上的信道 PER (Packet Error Rate, 误包率) 分别为  $P_{37}$ 、 $P_{38}$ 、 $P_{39}$ ，且相互独立。不考虑广播、扫描参数对消息收发的影响，消息传输成功与否只取决于数据传输的过程中是否发生误码。

2. 蓝牙网关持续发送群组可靠消息，直至所有 LPNs 正确接收。所有 LPN 具有  $T_{wake}$  和  $T_{sleep}$  的唤醒和休眠时长，两者大小具有如下关系。

$$\frac{T_{wake}}{T_{sleep}} = \delta \quad (3-1)$$

3. 群组可靠消息的可靠传输指的是消息传输一定次数后全部 LPNs 接收到消息的成功率能够到达  $\mu_{mul}$ 。

为了满足马尔科夫性，可以将马尔科夫链的状态使用不同节点设备是否接收到当前群组可靠消息的状态。假设系统中共  $N$  个 LPN，那么相应地，每个设备状态可以表示为  $\{s_1(t), s_2(t), s_3(t), \dots, s_N(t)\}$ ，其中  $s_i(t) \in \{0, 1\}$ ， $i \in [1, N]$ ， $s_i(t) = 0$  代表设备  $i$  未收到群组可靠消息， $s_i(t) = 1$  代表设备  $i$  在  $t$  时刻已成功接收群组确认接收消息。系统状态将随设备数量指数级增长，在网络规模较大时，一步状态转移概率矩阵依赖于不同设备的信道 PER 难以生成。因此作出不同设备在相同信道上 PER 一致的假设以简化模型。

$$P_{ch,1} = \dots = P_{ch,i} = P_{ch} \quad (3-2)$$

其中  $P_{ch,i}$  表示向序号为  $i$  的 LPN 设备发送群组可靠消息时在  $ch$  信道上的 PER。三个

信道上的 PER 相互独立，那么一次传输的失败概率表示为

$$P_F = P_{37}P_{38}P_{39} \quad (3-3)$$

定义离散随机过程 $X(t)$ 为经过多次消息传输之后，接收到消息的 LPN 节点个数。系统状态数为  $N+1$  个，时间步长为广播事件传输间隔 $T_{advEvent}$ ，建立同质化马尔科夫链，系统状态转移见图 3-3。

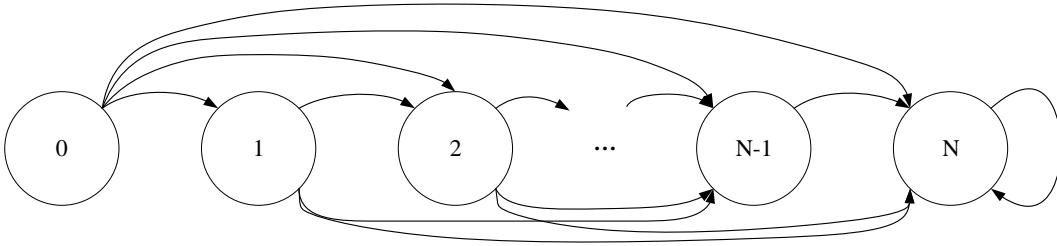


图 3-3 群组可靠消息传输系统状态转移图

各状态一步转移概率可归纳为：

$$P(n, m) = \begin{cases} P_F^{N-m}, n = m \\ C_{N-m}^{n-m} (1 - P_F)^{n-m} P_F^{N-n}, n > m \\ 0, n < m \end{cases} \quad (3-4)$$

在式(3-4)中，式（1）代表了在一次群组可靠消息传输时，暂未接收消息的所有 LPN 接收失败的情况，式（2）代表了暂未接收消息的 LPN 部分接收的情况，式（3）代表了系统中成功接收群组可靠消息的 LPN 数量减少的情况。由此可以得到一步状态转移矩阵 $P$ ，初始分布向量可以表示为  $N+1$  维的列向量，代表在初始状态下，系统中不存在已经成功接收到群组可靠消息的 LPN。

$$\pi(0) = [1 \quad 0 \quad \dots \quad 0]^t \quad (3-5)$$

已知一步状态转移矩阵 $P$ 和初始分布向量，可以得到第  $k$  次事件传输时，系统的状态分布。

$$\pi(k) = P^k \pi(0) \quad (3-6)$$

考虑所有 LPN 成功接收的情况 $\pi_N(k)$

$$\pi_N(k) = \sum_{i=0}^N p_{N,i}^{(k)} * \pi_0(0) \quad (3-7)$$

考虑到下行群组可靠消息需要到达成功率 $\mu_{mul}$ ，通过以下式子求得群组可靠消息的

最小传输次数 $MR_{mul}$ 。

$$\pi_N(MR_{mul}) = \mu_{mul} \quad (3-8)$$

对结果向上取整可以得到使得群组可靠消息可靠时的最小传输次数。系统存在有  $N$  个 LPN 节点时的最小可靠传输次数 $MR_{mul}$ 。已知在设备处于接收扫描常开状态下，网关需要传输 $MR_{mul}$ 次群组可靠消息保证可靠，那么在 $\frac{T_{wake}}{T_{sleep}} = \delta$ 的前提下，网关需要传输至少 $LPMR_{mul}$ 次才能保证消息传输可靠

$$LPMR_{mul} = \left(1 + \frac{1}{\delta}\right) MR_{mul} \quad (3-9)$$

传输时间 $T_{tx}$ 与传输次数  $x$  具有如下关系，其中 $T_{advEvent}$ 为 BLE MESH 协议栈固定的两次消息传输的时间间隔， $T_{trans}$ 为完成一次事件传输的时间。

$$T_{tx} = T_{advEvent} * (x - 1) + T_{trans} \quad (3-10)$$

在 LPN 开启接收占空比时，群组可靠消息的最大传输时延 $T_{mul}$ 为

$$T_{mul} = T_{advEvent} * (LPMR_{mul} - 1) + T_{trans} \quad (3-11)$$

由式(3-7)、(3-8)、(3-9)、(3-11)联立可以得到群组可靠消息的最大传输时延。

## 2. 确认消息传输性能

LPN 回复确认消息的流程如图 3-4 所示。

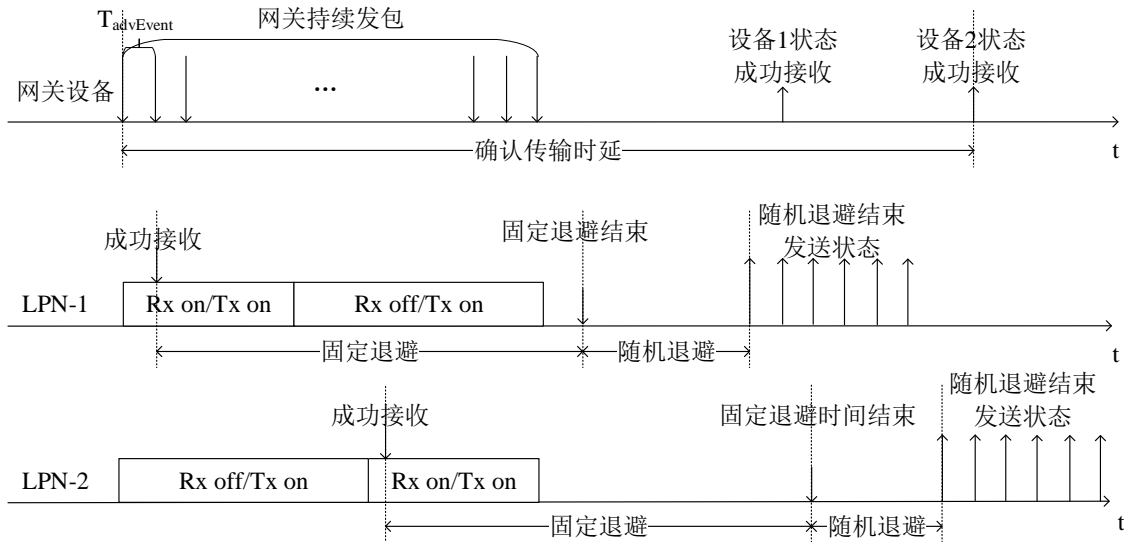


图 3-4 LPN 确认消息传输流程

网关在 LPN 设备休眠/唤醒全周期内持续发送群组可靠消息，以保证 LPN 能可靠接收。LPN 接收到网关消息之后首先退避固定的网关发送时间，之后随机退避一

段时间后发送多次确认消息。

## (1) 确认消息碰撞概率

虽然 LPN 设备在发送确认消息之前引入了随机退避时间来避免 LPN 设备间碰撞，但在网络规模逐渐扩大的 BLE MESH 网络中，设备间确认消息的碰撞概率将逐渐增大。

某个 LPN 在某次群组可靠消息传输完成后在时刻  $t$  进行确认消息的传输。在主要广播信道上完成一次 PDU 传输的时间为  $T_{PDU}$ 。以该节点作为参考节点，考虑两个节点完全碰撞的情况，其他节点该参考发生碰撞时，其发送时刻需要满足在  $[t - T_{PDU}, t + T_{PDU}]$  的范围内。图 3-5 描述了任节点间确消息碰撞的过程。

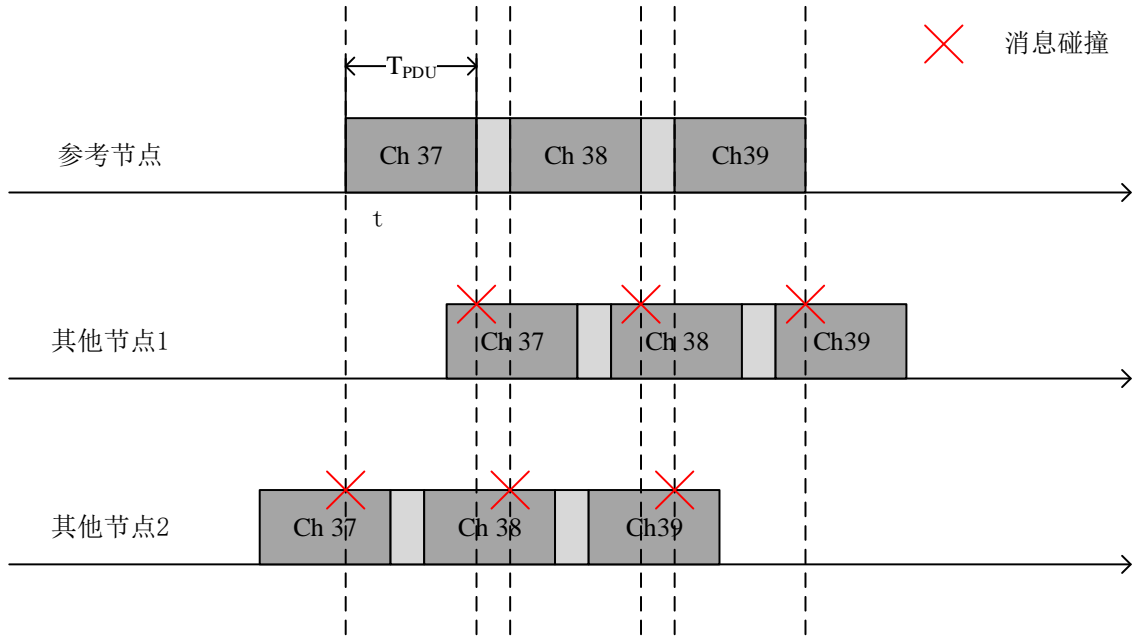


图 3-5 确认消息碰撞

在同一次群组确认消息传输中，LPN 设备的固定退避时间一致，是否碰撞只取决于随机退避时延的选取。假定随机退避时长在  $[T_{Rmin}, T_{Rmax}]$  的范围内完全独立随机选取，定义两个随机变量  $X$ 、 $Y$  代表了两个节点随机退避的时长，则有  $X, Y \sim U(T_{Rmin}, T_{Rmax})$ ，在  $|X - Y| < T_{PDU}$  时发生碰撞。可以得到任意两个 LPN 设备的碰撞概率为

$$p = -\frac{T_{PDU}}{(T_{Rmax} - T_{Rmin})^2} + \frac{2 * T_{PDU}}{T_{Rmax} - T_{Rmin}} \quad (3-12)$$

假设网络中有  $K$  个 LPN 需要传输确认消息，那么发生冲突的概率为

$$P_{col} = 1 - (1 - p)^{K-1} = 1 - \left[ 1 - \left( -\frac{T_{PDU}}{(T_{Rmax} - T_{Rmin})^2} + \frac{2 * T_{PDU}}{T_{Rmax} - T_{Rmin}} \right) \right]^{K-1} \quad (3-13)$$

定义随机变量  $Z$  代表一次群组可靠消息传输成功接收并进行确认消息发送的设备数量，则  $Z$  符合参数为  $(1 - P_F)$  的二项分布，可以得到一次群组消息传输后需要发送确认消息 LPN 个数的期望  $K$ 。其中  $P_F$  由式(3-3)得到。

$$K = N(1 - P_F) \quad (3-14)$$

由式(3-13)、(3-14)联立可以得到确认消息的冲突概率。

$$P_{col} = 1 - \left[ 1 - \left( -\frac{T_{PDU}}{(T_{Rmax} - T_{Rmin})^2} + \frac{2 * T_{PDU}}{T_{Rmax} - T_{Rmin}} \right) \right]^{N(1-P_F)-1} \quad (3-15)$$

在  $T_{PDU}$  确定的前提下，可以看到碰撞概率随着网络规模  $N$  的增大而增大，随事件传输失败率  $P_F$  以及随机时延选取区间大小  $(T_{Rmax} - T_{Rmin})$  的增大而降低。考虑到 LPN 广播事件以固定的间隔执行，上述碰撞过程还会引起多次广播事件碰撞的发生。

## (2) 确认消息传输时延

定义确认消息的传输时间为开始群组可靠消息传输至所有 LPN 成功反馈确认信息的时间。已知群组可靠消息的发送时间为  $T_{tx}$ ，设设备固定退避时间表示为  $T_S$ ，设备随机退避时间表示为  $T_R \in [T_{Rmin}, T_{Rmax}]$ ，同时假定网关设备与 LPNs 具有一样的发送时间间隔  $T_{advEvent}$  和一次广播事件的时长  $T_{trans}$ 。则有

$$T_S = T_{tx} = T_{advEvent} * (LPMR_{mul} - 1) + T_{trans} \quad (3-16)$$

假定协议栈设定的随机退避时间范围足够大能够有效避免设备间冲突。定义随机变量  $\epsilon$  表示 LPN 持续向网关发送确认消息直至成功时的传输次数，则随机变量  $\epsilon$  符合参数为  $(1 - P'_F)$  的几何分布，也即

$$P(\epsilon = k) = P_F'^{k-1} (1 - P_F') \quad (3-17)$$

其中  $P'_F$  为 LPN 节点向网关设备执行一次消息传输的失败概率。相应地，作出不同设备在向网关广播消息时三个主要信道上误包率一致的假设，有

$$P'_F = P'_{37} P'_{38} P'_{39} \quad (3-18)$$

$P'_{37}$ 、 $P'_{38}$ 、 $P'_{39}$  表示 LPN 设备在 37、38、39 信道上的传输的误包率。

设定确认信息可靠传输要求 LPN 在连续发送  $k$  次消息之后，网关成功接收概率



需要达到 $\mu_{ack}$ ，通过几何分布的累计分布函数，通过计算下面的式子可以得到确认消息可靠传输的最小重传次数 $MR_{ack}$ 。

$$1 - P_F'^{MR_{ack}} = \mu_{ack} \quad (3-19)$$

可以得到确认消息最大传输时间 $T_{ack}$ 为

$$T_{ack} = T_{tx} + T_S + MAX(T_R) + T_{advEvent} * (MR_{ack} - 1) + T_{trans} \quad (3-20)$$

由(3-16)、(3-20)联立可以得到确认消息的最大传输时延。当随机退避时间选取范围设置得很大时，能够有效降低 LPN 间确认消息的碰撞概率，但也会引起确认消息传输时延的增大。

### 3.2.3 现有接入机制问题分析

基于网络层重传的消息可靠传输的优点在于简单且易于实现。但随着网络规模的扩大，在保证消息可靠传输的前提下，其在传输时延性能上表现较差，其问题在于：

(1) LPN 休眠引起的群组可靠消息丢失。在下行方向上，MESH 网关设备在传输群组可靠消息时 LPN 处于休眠状态从而导致了消息传输失败。多次重传的做法虽然最终也能够保证一定的传输可靠性，但会导致了消息的端到端时延很大。而且多数传输是无用的，造成了能量的浪费。

(2) LPN 退避引起的时延和碰撞问题。确认消息在上行方向上，LPN 设备退避时间过长能够在一定程度上避免与其他设备的消息发生冲突，从而保证确认消息的成功传输，但同时也增大了确认消息的可靠传输时延。随着网络中 LPN 数目增多，LPN 间的冲突也会更加频繁。固定的退避时间选取并不能很好地适应网络的变化。

(3) 无法适应网络规模和信道环境变化。群组可靠消息和确认消息的重传次数是由开发者提前确定的，在 LPN 数目增多时，设备间的冲突也会更加频繁，原本设定好的固定重传次数也不再能够保证消息传输的可靠。

## 3.3 BLE MESH 接入机制优化技术路线

为了实现 BLE MESH 网络接入的需求和目标，基于 3.2 小节中对现有接入机制分析和缺陷指出，提出了基于 TDMA 的可靠消息接入机制。机制在网络中设备时间同步的前提下，有如下优化技术路线：

(1) 在时钟同步的基础上, 控制网关在设备唤醒期间发送群组可靠消息, 从而在很大程度上避免群组可靠消息因 LPN 休眠丢失。将 LPN 的唤醒时间同步起来在提供统一的时间段开启信道扫描, 蓝牙网关将消息缓存起来并被控制在 LPN 的信道扫描功能开启时段进行消息传输, 做到有的放矢地进行数据传输。

(2) 在时钟同步的基础上, 将 LPN 信道扫描关闭的时间段划分为多个小的时隙, 不同时隙被分配给不同的 LPN 用于确认消息的传输, 从而大大降低设备间确认消息的碰撞。考虑到随着消息的传输, 网络中未成功传输确认消息的设备数量减少。将可用时隙动态均匀地调整分配给未成功传输确认消息的设备, 分配的传输时间增多使得网络中这样的设备数目能够更快地收敛至 0, 从而大大降低确认消息的传输时延。

(3) 网关根据群组可靠消息的确认情况来判断是否需要继续重传, 从而适应网络规模和信道环境的变化。在所有 LPN 确认消息成功传输之前, 蓝牙网关将保持重传直至所有 LPN 反馈成功。

## 3.4 基于 TDMA 的 BLE MESH 群组可靠消息接入机制

### 3.4.1 TDMA 接入机制简介

TDMA 与 CSMA 是无线传感器网络中常用的媒体接入方法<sup>[41]</sup>。TDMA 由于其免碰撞和节能的特性, 更适合作为无线传感器网络中的 MAC 协议。

TDMA 帧是 TDMA 系统的基本单元, 每个 TDMA 帧又被划分为多个时隙。根据时隙分配方法来分类又可以分为静态分配和动态分配<sup>[42]</sup>。在静态分配的 TDMA 中, 用户没有数据发送也会被分配传输时间。这种方法可能导致介质利用率的不足, 不能很好地用于具有较长空闲时间的混合流量。而在动态分配的 TDMA 中, 将根据用户需要为用户分配时隙, 这种做法使得延迟和带宽分配变得可控, 提高了信道利用率, 更容易支持 QoS 要求。不同时隙分配方法的 TDMA 机制如图 3-6 所示。

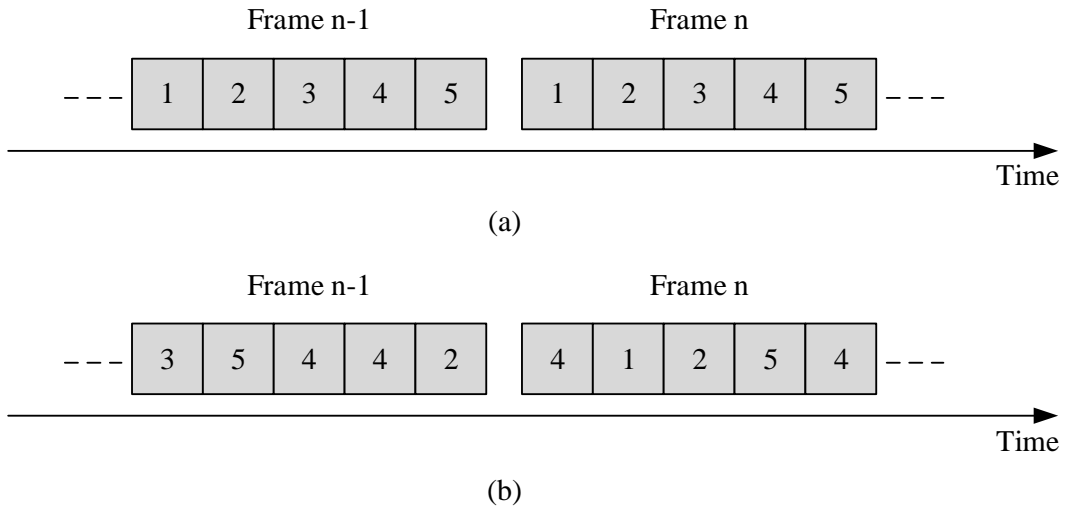


图 3-6 (a)静态分配 TDMA (b)随机分配 TDMA

基于动态 TDMA 的思想，本文提出了一种在 BLE MESH 网络中由网关设备动态分配时隙的 TDMA 接入方法。与此同时，TDMA 系统实现的一大前提是系统的同步，在 BLE 协议栈没有提供同步服务的前提下，本文在 4.1.1 小节提出了一种基于 BLE 广播消息传输的同步实现方法。

### 3.4.2 基于 TDMA 的优化接入机制设计

在网络中各设备同步的前提下，将 TDMA 系统中的一个 TDMA 帧大小定义为  $T_{frame}$ ，并将一个 TDMA 帧分为两个 CFP（Contention Free Period，免竞争时段），分别使用 CFP1 和 CFP2 表示， $T_{CFP1}$  和  $T_{CFP2}$  分别表示 CFP1 和 CFP2 时间段的大小，有

$$T_{frame} = T_{CFP1} + T_{CFP2} \quad (3-21)$$

不同 CFP 分配给不同的设备用于消息的传输，且 CFP1 和 CFP2 通过同步机制使得与所有 LPN 的接收唤醒时间和休眠时间相重合。用  $T_p$  表示 LPN 的休眠周期大小，有

$$T_{CFP1} = T_{wake} \quad (3-22)$$

$$T_{CFP2} = T_{sleep} \quad (3-23)$$

$$T_{frame} = T_p = T_{wake} + T_{sleep} \quad (3-24)$$

经过同步后的 TDMA 帧和 LPNs 的唤醒和休眠时段如图 3-7 所示。

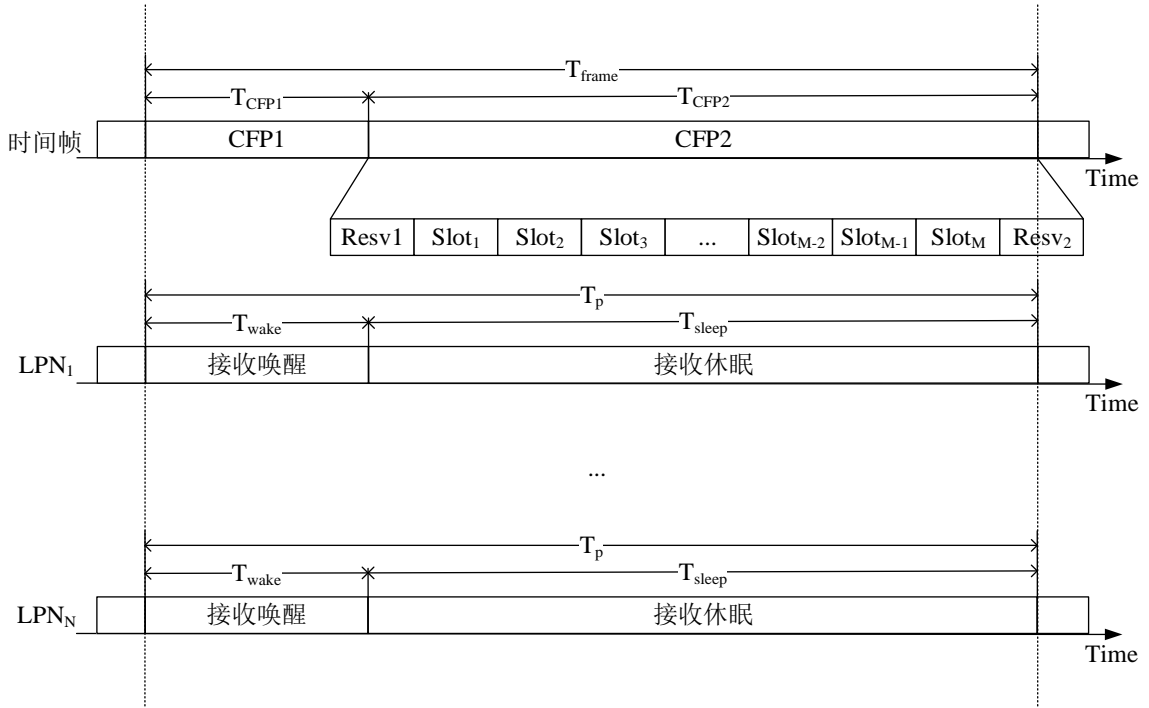


图 3-7 TDMA 帧中的设备行为

(1) CFP1: 网关在 CFP1 期间向三个主要信道广播群组可靠消息，并在每个 CFP1 开始前为 LPNs 分配时隙，通过群组可靠消息携带通知给 LPNs；LPNs 在 CFP1 期间打开信道扫描接收网关发送的群组可靠消息。

(2) CFP2: 网关在 CFP2 期间处于信道扫描状态接收 LPNs 回复的确认信息；在 CFP1 期间接收到群组确认消息的 LPNs 在 CFP2 期间向三个主要信道广播确认消息，这个时间段被划分为多个小的时隙分配给暂未返回确认消息的设备。同时考虑到网关的发送和接收切换时间，在 CFP2 的开头和结尾预留了一段时间。

网关在 CFP1 期间集中持续发送群组可靠消息，已知 BLE 协议栈广播事件固定执行间隔  $T_{advEvent}$ ，每个间隔执行三个广播事件，可以得到消息发送次数  $R_{mul}$  为

$$R_{mul} = \left\lceil \frac{T_{CFP1}}{T_{advEvent}} \right\rceil * 3 \quad (3-25)$$

假设在 CFP2 期间共分配  $M$  个时隙，时隙大小为实现一次三个主要信道广播的时间  $T_{trans}$ 。每个时隙最多只有一个 LPN 进行确认信息的传输，这  $M$  个时隙被表示为  $Slot_M\{slot_1, slot_2, slot_3, \dots, slot_m\}$ ，有如下数学形式的表示：

$$T_{slot1} = T_{slot2} = \dots = T_{slotm} = T_{slot} = T_{trans} \quad (3-26)$$

$$\sum_{i=1}^M T_{sloti} + T_{resv1} + T_{resv2} = T_{CFP2} \quad (3-27)$$

其中 $T_{resv1}$ 、 $T_{resv2}$ 是考虑到网关设备在扫描和广播状态转换需要时间，已知 $T_{resv1}$ 、 $T_{resv2}$ 的前提下可以得到时隙个数  $M$ 。

$$M = \frac{T_{CFP2} - T_{resv1} - T_{resv2}}{T_{trans}} \quad (3-28)$$

$M$  个时隙被均匀分配给网络中暂未成功传输确认消息的 $N_{res}$ 个 LPN。同样地考虑到 BLE 协议的广播事件固定间隔。每个设备的传输时隙个数 $R_{ack}$ 可以简化表达如下。

$$\begin{cases} R_{ack} = \left\lfloor \frac{M}{N_{res}} \right\rfloor, N_{res} > N_{thresh} \\ R_{ack} = \left\lfloor \frac{M}{N_{thresh}} \right\rfloor, N_{res} \leq N_{thresh} \end{cases} \quad (3-29)$$

其中 $N_{thresh}$ 为 LPN 设备的一个广播事件间隔内被分配的时隙个数，表示为

$$N_{thresh} = \left\lfloor \frac{T_{advEvent}}{T_{trans}} \right\rfloor \quad (3-30)$$

考虑到 $M < N_{res}$ 的情况，会导致部分接收到群组可靠消息的 LPN 设备没有时隙发送数据，这些设备的确认消息被推迟到下一个 TDMA 帧中进行传输。

### 3.4.3 优化接入机制性能分析

本节延续 1 小节的基于离散时间马尔科夫链方法，在保证消息可靠性前提下，对优化接入机制中的群组可靠消息以及确认消息传输次数对进行分析，数值仿真结果见 3.5 小节。

#### 1. 群组可靠消息传输性能

本节延续 1 小节中基于离散时间马尔科夫链方法对群组可靠消息可靠传输次数对优化接入机制进行分析。在 1 小节中系统假设的前提下，定义离散随机过程 $Y(t)$ 为经过多轮 TDMA 帧传输之后接收到群组可靠消息的 LPN 节点个数，系统状态数为  $N+1$  个，时间步长修改为 TDMA 帧大小 $T_{frame}$ 。为了简化模型分析，仍然作出不同设备在相同信道上的误包率相等的假设。

网关设备在 CFP1 期间发送群组可靠消息，在一个 CFP1 期间完成 $R_{mul}$ 个群组可靠消息的传输。网关设备在一个 TDMA 帧中向某个设备传输群组可靠消息的传输失

败概率 $P_{FPR}$ 为

$$P_{FPR} = P_F^{R_{mul}} \quad (3-31)$$

其中 $R_{mul}$ 由式(3-25)得到, 各状态一步转移概率可归纳为:

$$P(n, m) = \begin{cases} P_{FPR}^{N-m}, n = m \\ C_{N-m}^{n-m} (1 - P_{FPR})^{n-m} P_{FPR}^{N-n}, n > m \\ 0, n < m \end{cases} \quad (3-32)$$

式(3-32)中, 式(1)代表一轮 TDMA 帧传输过程中暂未接收消息的 LPN 全部接收失败的情况, 式(2)代表一轮 TDMA 帧传输过程中暂未接收消息的 LPN 部分接收失败的情况, 式(3)代表经过一轮 TDMA 帧后系统中接收到消息数量变少的情况。

与式(3-7)、(3-8)的计算方法类似, 系统同样具有相同的  $N+1$  维初始分布向量 $\pi(0) = [1 \ 0 \ \dots \ 0]^t$ , 通过下面的式(3-33)可以计算出使得群组可靠消息成功传输的最小 TDMA 帧数 $MF_{mul}$ 。

$$\pi_N(MF_{mul}) = \mu_{mul} \quad (3-33)$$

其中 $\mu_{mul}$ 是要实现的可靠性。在基于 TDMA 的消息接入机制下, 传输时延 TDMA 帧数正相关, 传输时间 $T_{tx}$ 与 TDMA 帧个数  $y$  的关系为

$$T_{tx} = T_{frame} * (y - 1) + T_{CFP1} \quad (3-34)$$

在设备具有工作占空比的前提下, 群组可靠消息可靠传输的最大时延为

$$T_{mul} = T_{frame} * (MF_{mul} - 1) + T_{CFP1} \quad (3-35)$$

## 2. 确认消息传输性能

在基于 TDMA 思想的群组可靠消息接入机制下, LPNs 确认消息是否成功传输的状态可以在一个 TDMA 帧内确定。当前确认消息传输状态仅依赖于上一个 TDMA 帧内的操作, 符合马尔科夫性。本节延续 1 小节的方法使用马尔科夫链对确认消息的可靠传输过程进行分析。首先, 对确认消息的传输做出如下假设:

1. 所有 LPNs 在三个蓝牙主要广播信道上对网关设备进行广播的信道误包率 (PER) 分别为 $P'_{37}$ 、 $P'_{38}$ 、 $P'_{39}$ , 且相互独立。不考虑广播、扫描参数对消息收发的影响, 消息传输成功与否只取决于数据传输的过程中是否发生误码。

2. 网关持续发送群组可靠消息触发 LPNs 的确认消息发送, 直至所有 LPNs 的

确认消息被正确接收。

3. 确认消息的可靠传输指的是在群组可靠消息传输一定 TDMA 帧后全部 LPNs 确认消息接收的成功率能够达到  $\mu_{ack}$ 。

已知，LPN 设备在三个主要广播信道上向网关设备传输数据，网关作为接收的一方持续扫描信道，在一个或一个以上信道上成功接收时，则本次确认消息传输成功。同样作出不同设备在相同信道上的误包率相等的假设，有

$$P'_F = P'_{37}P'_{38}P'_{39} \quad (3-36)$$

定义离散随机过程  $Z(t)$  为经过多轮 TDMA 帧传输之后，成功传输确认消息的 LPN 个数。系统状态数为  $N+1$  个，时间步长为 TDMA 帧大小  $T_{frame}$ ，建立同质化马尔科夫链。网关在每个 TDMA 帧开始得到剩余未成功传输确认消息的节点个数为

$$N_{res} = N - Z(t) \quad (3-37)$$

假定时隙总数  $M > N_{res}$ ，结合式(3-29)可以得到 LPN 节点每轮 TDMA 帧的传输时隙个数  $R_{ack}$ 。在 LPN 在该轮 TDMA 帧中已经接收到群组可靠消息的前提下，可以得到传输确认消息的失败概率。

$$P_{ack} = P'_F R_{ack} \quad (3-38)$$

各状态一步转移概率为

$$P(n, m) = \begin{cases} (P_{FPR} + (1 - P_{FPR})P_{ack})^{N-m}, & n = m \\ C_{N-m}^{n-m} [(1 - P_{FPR})(1 - P_{ack})]^{n-m} (P_{FPR} + (1 - P_{FPR})P_{ack})^{N-n}, & n > m \\ 0, & n < m \end{cases} \quad (3-39)$$

由此可以得到状态转移矩阵  $P$ ，式(3-39)中，式(1)代表网络中未成功传输确认消息的所有 LPN 因为群组可靠消息丢失或是确认消息丢失的情况，式(2)代表部分 LPN 因群组可靠消息丢失或是确认消息丢失，部分在成功接收群组可靠消息后成功传输确认消息的情况，式(3)代表网络中成功确认 LPN 设备数量减少的情况。同样地，已知初始分布向量  $\pi(0) = [1 \ 0 \dots 0]^t$  为  $N+1$  维列向量，与式(3-7)、(3-8)的计算方法类似，可以得到使得所有 LPNs 确认消息可靠传输时的最小 TDMA 帧数  $MF_{ack}$ 。

$$\pi_N(MF_{ack}) = \mu_{ack} \quad (3-40)$$

已知确认消息传输时延 $T_{ack}$ 与传输完成时 TDMA 帧数  $z$  的关系，可以得到网络中确认消息可靠传输的最大时延。

$$T_{ack} = T_{frame} * z \quad (3-41)$$

$$T_{ack} = T_{frame} * MF_{ack} \quad (3-42)$$

## 3.5 不同接入机制的仿真与分析对比

本节基于 3.2.2、3.4.3 小节的分析，对现有机制以及优化机制的性能进行数值仿真分析。相关参数设置参照 BLE MESH 协议栈，使用到的参数如表 3-1 所示。

表 3-1 BLE MESH 接入机制仿真参数

参数	取值
$T_{wake}(ms)$	20
$T_{sleep}(ms)$	60
$T_{PDU}(ms)$	0.5
$T_{trans}(ms)$	3
$T_{advEvent}(ms)$	20
$T_R(ms)$	[20,500]

$T_{wake}$ 和 $T_{sleep}$ 分别代表了 LPN 设备的唤醒和休眠时间。 $T_{PDU}$ 为单信道 PDU 传输完成所需要的时间。 $T_{trans}$ 为执行完一次三信道广播的时间，包括了在三个信道上发送 PDU 的时间以及信道切换的时间，也是 TDMA 机制中的确认消息传输时隙大小。 $T_{advEvent}$ 为设备连续发送两次消息的时间间隔，这里取消了随机间隔，取固定值 20ms。 $T_R$ 为协议推荐规定的针对群组可靠消息的确认消息的随机退避时间范围。

### 3.5.1 确认消息碰撞概率分析

在网络中设备同步的前提下，优化机制中的 LPN 设备发送确认消息不存在相互碰撞。这里对现有机制在不同信道误包率和网络规模下的确认消息碰撞的情况进行仿真，结果如图 3-8 所示。

从趋势上看，在信道误包率不变的情况下，可以看到确认消息碰撞概率随网络规模增大而增大。网络规模一样的情况下，碰撞概率随误包率降低而增大。这是因为随着规模增大或是误包率降低，同时执行随机退避发送消息的设备数量越多，碰撞概率



越大。可以看到网络规模到达 256 时，设备间的碰撞概率超过 20%，将对确认消息传输的可靠性和实时性极大的负面影响。设置更大的退避范围能够降低碰撞概率，但会一定程度上增大消息的传输时延。

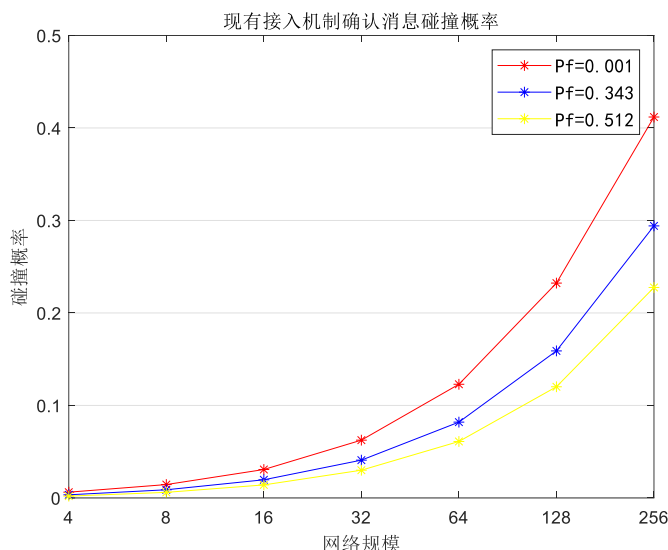


图 3-8 现有机制确认消息随网络规模和信道误包率的变化

### 3.5.2 不同信道误包率下的时延性能分析

假设网络不同设备广播时三个主要广播信道上的信道误包率 $P_{37}$ 、 $P_{38}$ 、 $P_{39}$ 是一致的，并会随着信道干扰情况发生变化。根据 BLE MESH 协议中规定的数值范围，将系统仿真参数分为 6 个不同的场景，网络规模统一为 32，具体参数如表 3-2 所示。

表 3-2 不同接入机制的仿真参数表

参数	(a)	(b)	(c)	(d)	(e)	(f)
$P_{37}$	0.10	0.20	0.25	0.30	0.35	0.40
$P_{38}$	0.10	0.20	0.25	0.30	0.35	0.40
$P_{39}$	0.10	0.20	0.25	0.30	0.35	0.40
$\mu_{mul}$	0.999					
$\mu_{ack}$	0.999					

上述参数中 N 为网络中的 LPN 设备数目。群组可靠消息以及确认消息的传输可靠性设置为 0.999，每 1000 次消息传输只允许一次失败。仿真时不考虑时隙总数小于需要传输的 LPN 设备数量导致的确认消息推迟。

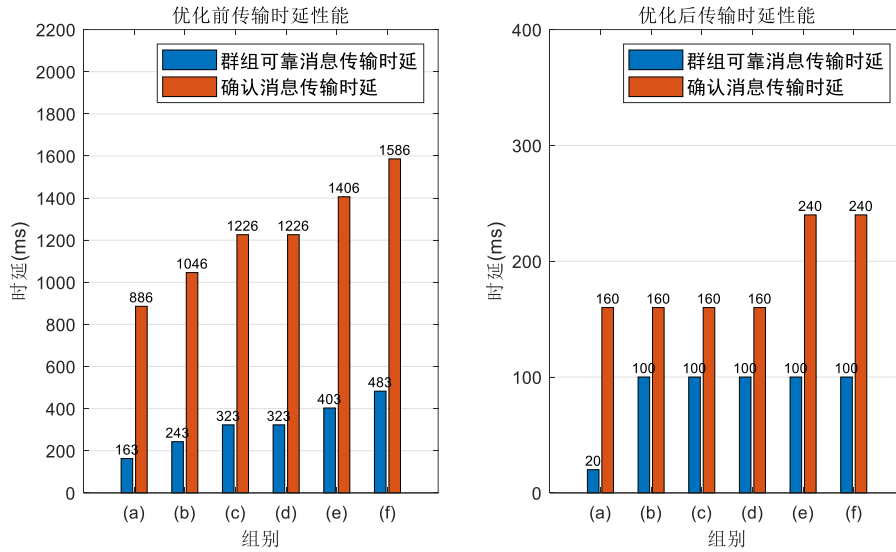


图 3-9 不同信道误包率下的消息传输时延对比

两种接入机制在不同的信道误包率下的消息传输时延性能如图 3-9 所示。图中蓝色柱状图代表了群组可靠消息传输的理论最大时延，红色柱状图代表确认消息可靠传输时的理论最大时延。

对上面 6 组仿真进行对比，可以得到基于 TDMA 的优化接入机制的群组可靠消息最大传输时延相比现有接入机制平均下降了 70.42%，而确认消息的最大传输时延对 6 组仿真取平均相较于现有机制下降了 82.48%。

6 组仿真从左至右，误包率逐渐增大，一次广播事件传输的失败率 $P_F$ 也随之增大。随着 $P_F$ 增大，群组可靠消息需要更多次的重传来保证可靠，相应地，群组可靠消息的传输时延也会增大。在信道误包率较高的(e)、(f)组的仿真结果中，群组可靠消息的传输时延超过了 400ms。优化接入机制因集中在设备唤醒的时间端进行集中发送，在传输时延更小的同时也避免了在消息发送时 LPN 设备处于休眠状态而导致的消息丢失。其消息的最大传输时延一直处于较低的水平，不超过 200ms。

随着 $P_F$ 增大，现有机制确认消息的传输时延逐渐增加，平均超过了 1s，一方面因为需要退避群组可靠消息传输时间避免与其碰撞，另一方面需要设置较大的随机退避时间避免 LPN 间消息碰撞。而优化接入机制通过划分时隙避免了设备间传输消息时的“碰撞”，另一方面是因为机制为未成功确认的设备分配了更多的时隙进行确认消息传输，确认消息的传输时延总是与群组可靠消息的传输时延相近。

## 3.5.3 不同网络规模下的时延性能分析

除了对上述几种不同误包率的情况进行仿真之外，还通过仿真验证了固定误包率下，不同网络规模对时延的影响。将三信道的误包率 $P_{37}$ 、 $P_{38}$ 、 $P_{39}$ 分别设置为 0.25、0.25、0.25，控制网络中 LPN 设备数量为 8、16、32、64、128 对现有接入机制以及优化接入机制分别做了仿真。同样地，仿真时不考虑时隙总数小于需要传输的 LPN 设备数量导致的确认消息推迟。仿真结果如图 3-10 所示。

可以看到随着网络规模的指数增长，消息传输时延会呈现缓慢的线性增长趋势。但总的来说，基于 TDMA 的优化接入机制比现有的接入机制的时延特性都会好上许多。平均来看，优化机制的群组可靠消息传输时延相比现有机制减少了 50.10%，而确认消息的传输时延减少了 79.44%。

需要注意的是，现有机制中确认消息的碰撞概率会随着网络规模增大而增大，为保证可靠需要设置更大的传输次数，实际的消息传输时延应该更大。

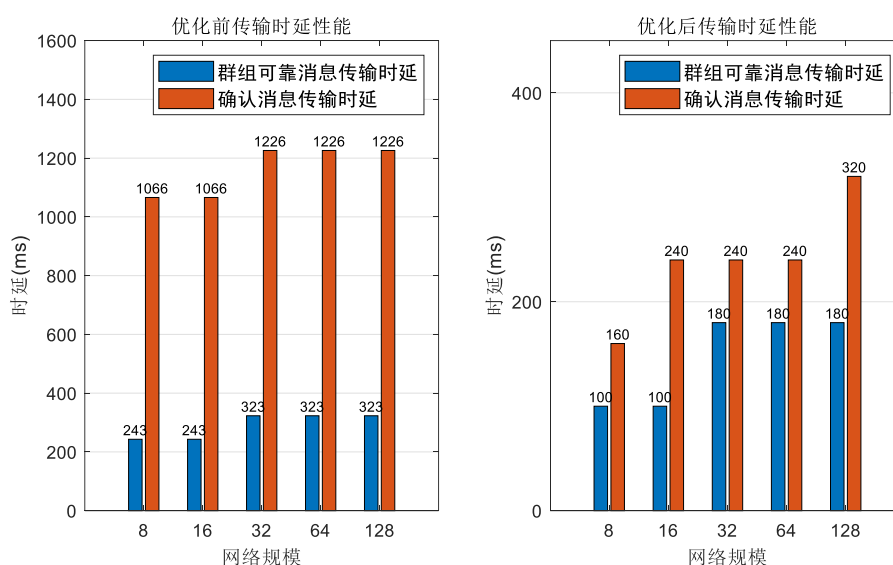


图 3-10 不同网络规模下的消息传输时延对比

通过以上的仿真结果可以得知，优化机制通过同步有效避免了设备间碰撞。在相同消息传输可靠性的前提下，优化机制能够在不同网络规模 and 不同信道误包率的网络中对时延性能作出优化。

## 3.6 本章小结

本章的主要内容是 BLE MESH 网络中群组可靠消息接入机制的优化策略的构建与分析。

首先分析了在低功耗设备节点按一定休眠占空比进行工作的 BLE MESH 网络中实现群组可靠消息接入机制的需求以及目标。之后对现有的基于网络层重传方法的接入机制进行了描述,并分析了现有机制在应用于群组可靠消息接入时的缺陷,包括了在传输下行群组可靠消息时设备休眠引起的丢包、上行传输确认消息需要引入较大的时延来避免设备间消息传输发送“碰撞”以及重传次数固定无法应对信道环境变化等问题。

为了避免上述问题,提出了基于 TDMA 方法的接入机制。机制首先在 BLE MESH 协议栈实现缺少同步服务的情况下,提出了基于广播消息传输的 MESH 节点间同步方法。在网络中节点同步的前提下,通过控制网关设备在低功耗设备处于唤醒状态时进行群组可靠消息的传输。同时将设备休眠但发送模块开启的时间段划分为小的时隙,采用了动态的时隙分配方法为暂时未成功确认的设备分配更多的传输时间,在避免了设备间冲突的同时也能让群组中暂时未成功确认的设备快速收敛至 0。

本文使用了离散时间马尔科夫链对现有机制以及优化机制中的最大群组可靠消息传输时延以及确认消息时延进行了理论分析,并通过 Matlab 数值仿真验证了优化后的接入机制比现有机制更好,为后续的软件实现提供了理论支撑。

## 4 智能家居场景中 BLE MESH 优化接入设计与实现

第三章分析了 BLE MESH 网络中实现群组可靠消息传输的现有接入机制的性能不足，之后提出了 BLE MESH 网络中的基于 TDMA 的群组可靠消息接入机制，通过仿真分析验证了新的接入机制能够有效避免网络中 LPN 设备发送消息时的碰撞，且相较于原有的群组可靠消息接入机制能够在更短的时间内实现下行群组可靠消息和上行确认消息的可靠传输。

本章基于之前提出的优化接入机制，提出具体的实现方案，并在开源的 BLE MESH 协议栈上进行了软件实现。通过实际测试结果对比进一步验证基于 TDMA 的群组可靠消息接入机制的可行性。

### 4.1 基于 TDMA 的 BLE MESH 群组可靠消息接入机制的方案设计

#### 4.1.1 基于 BLE 广播的低成本同步方法

##### 1. 基于 BLE 广播的 DMTS 同步方法

TDMA 的实现基于网络设备间的时间同步有较高的要求。BLE MESH 协议基于 BLE 协议栈，在 BLE 协议中解决同步问题的工作有限。部分基于点对点连接在网络规模增大时难以实现<sup>[43]</sup>。部分基于能量测量和频偏估计的研究工作需要引入额外的硬件资源<sup>[44-46]</sup>。

在无线传感器网络中有一些基于消息传递的同步方法<sup>[47-50]</sup>。其中，DMTS (Delay Measurement Time Synchronization, 时延测量时间同步) 是一种灵活轻便的同步方法。对于有  $N$  个节点的单跳无线传感器网络中，无论  $N$  值大小，只需要一次广播即可实现网络中设备的同步。一次广播实现同步也意味增加的网络开销更小，能够有效的控制能耗，方便低成本的同步实现。

本文基于 DMTS 的思想，在 BLE MESH 应用层实现了一种低成本同步方法如图 4-1。其具体实现流程为：

- 1、发送者在应用层调用发送时在同步消息中写入本地时间戳  $t$ 。
- 2、发送者记录每一次调用发送到发送处理完成之前的时间间隔  $\Delta t$ 。发送者本地保存一份发送端处理时延的预期值，由下面的式子迭代计算，初始情况下  $t_{txproc} = 0$ 。

$$t_{txproc} = t_{txproc} * \frac{1}{8} + \Delta t * \frac{7}{8} \quad (4-2)$$

- 3、发送者后续发送同步消息时填入的时间戳为  $t_{tx} = t + t_{txproc}$ 。
- 4、参照 2. 的做法, 接收者记录每个消息从接收再到送至应用层的处理时延  $t_{rxproc}$ 。
- 5、接收者接收到同步消息后取出消息中的时间戳, 并进行相应的矫正得到修正后的时间戳  $t_{rx} = t_{tx} + t_{rxproc} + n\tau$ 。接收者将  $t_{rx}$  作为自己的本地时钟值, 并进行定时更新, 后续的事务按照该时间进行。

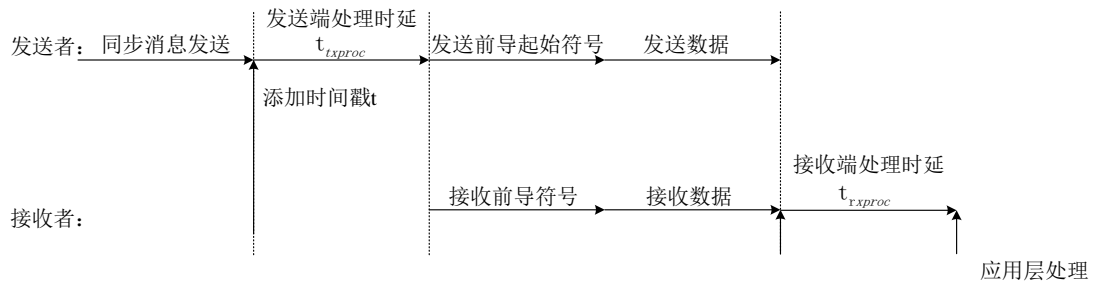


图 4-1 基于 BLE 广播的 DMTS 同步实现

在经过上述流程之后, 最后接收者设置到本地时钟的值为

$$t_{rx} = t + t_{txproc} + n\tau + t_{rxproc} \quad (4-3)$$

$n$  为同步消息的比特数,  $\tau$  为发射机传输一个比特所需要的时长。这种方法能够实现网络中设备大致同步。

## 2. 基于 BLE 广播的同步实现

蓝牙网关作为 MESH 网络中的同步源定时发送同步消息, LPN 作为被同步设备需要定时打开扫描窗口来接收蓝牙网关发送的同步消息。该过程可以描述如下:

- 1、网关设备在上电后按本地时间戳所记录的时间以每  $T_{sync}$  的周期定时广播同步消息, 消息携带了网关的时间戳信息。

- 2、LPN 设备在加入 MESH 网络之后处于常开状态。设备在第一次接收到网关发送的同步消息之后, 对消息中所携带的时间戳信息进行修正并作为本地的时间戳。修正方法见 1 小节, 该方法能够有效减少因传输时延和处理时延引起的同步误差。

- 3、LPN 设备使用本地定时器定时更新修正后的本地时间戳, 并按照本地时间戳所记录的时间以  $T_{sync}$  的时间间隔定时打开接收同步消息的窗口用于接收同步消息。在接收同步消息之后按同样的方法更新本地时间戳, 之后重复执行 3。

上述方法能够保证 LPN 在漏掉某个同步消息的情况下仍然能够按本地时钟与网关设备实现大致同步。同步消息的发送和接收过程如图 4-2 所示。

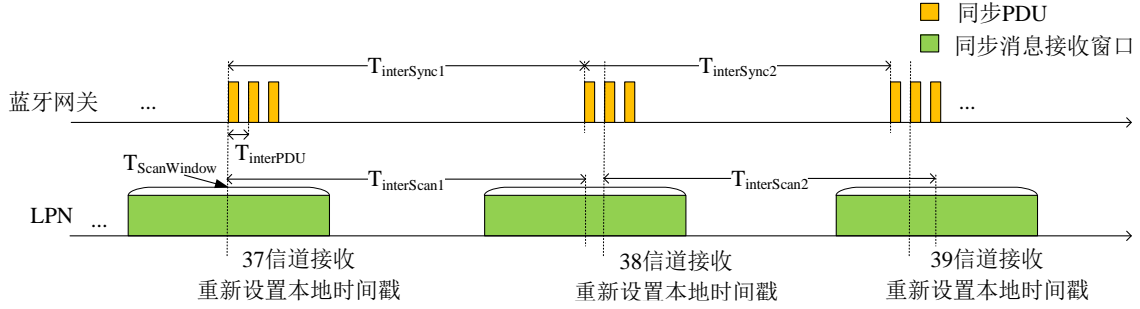


图 4-2 BLE 广播实际同步流程

接收同步消息的窗口设置为 LPN 本地时间戳每经过  $T_{sync}$  会前后打开时长为  $T_{ScanWindow}/2$  的窗口。按照这种方式设置同步消息接收窗口的原因有两点：

(1) 同步消息以 BLE 广播的形式在三个主要广播信道上传输 PDU，而 PDU 传输的时间间隔为  $T_{interPDU}$ 。作为接收者的 LPN 将有可能在三个信道上接收数据，在应用层开发的优化接入机制无法拿到接收到消息时的信道信息，因此会造成一定的同步误差。设置前后窗口能够确保 LPN 在三个信道上的同步 PDU 传输时都处于扫描状态。

(2) 网关和 LPN 设备的本地时钟可能会存在一定的定时误差，本地时间戳所记录的  $T_{sync}$  时长不一定符合实际的时长。设置前后窗口能够有效对抗这种时钟误差的影响。

## 4.1.2 休眠占空比时间设置

LPN 设备在第一次接收到蓝牙网关设备发送的同步消息之后完成了与网关设备的同步。为了保证网关设备在发送时，所有 LPN 节点处于唤醒状态。LPN 在发送时的时隙是严格区分开来的。需要保证所有 LPN 的休眠唤醒时间是一致的。网关本地时间戳为  $t_{GATE}$ 。假设网络中 LPN 设备按照自然数编号，序号为  $i$  的 LPN 设备在同步后的本地时间戳为  $t_{LPN}^i$ ，网络中 LPN 设备的序号集合为  $\Omega$ ，在同步之后有

$$t_{GATE} \approx t_{LPN}^i, i \in \Omega \quad (4-4)$$

LPN 设备的休眠时间和唤醒时间分别为  $T_{sleep}$  及  $T_{wake}$ ，整个周期长为  $T_p = T_{sleep} + T_{wake}$ 。方案实现将  $T_{sync}$  设置为  $T_p$  的整数倍。LPN 设备按照如下规则设置休

眠占空比时间：

(1)  $t_{LPN}^i \bmod T_p \leq T_{wake}$ ，LPN 设备为唤醒状态。

(2)  $t_{LPN}^i \bmod T_p > T_{wake}$ ，LPN 设备为休眠状态。

休眠唤醒时间安排如图 4-3 所示。

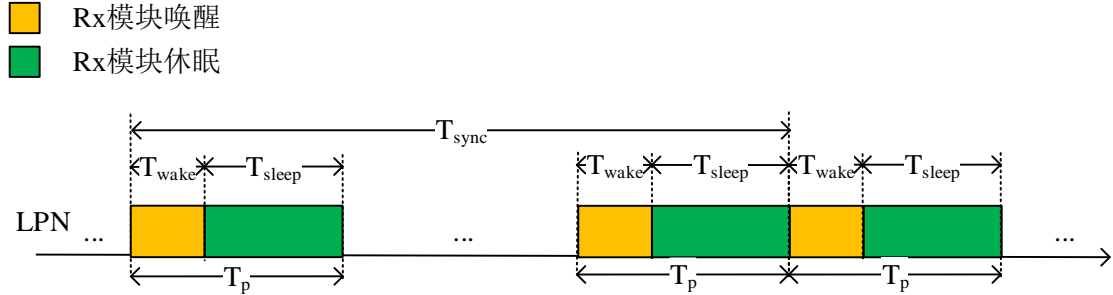


图 4-3 LPN 的休眠唤醒设置

### 4.1.3 群组信息管理及重传

网关记录每个组地址的订阅元素情况，并为这些元素分配 ID。这个 ID 在组地址对应的群组内生效。假设某个群组中共  $N$  个 LPNs 订阅，网关为这些 LPNs 从 1 到  $N$  进行编号，并通过私有广播消息通告给 LPNs。群组的 ID 分配基于 BLE MESH 的订阅机制，其流程如图 4-4 所示。

MESH 协议栈并未在应用层实现对群组及群组内设备信息的记录。考虑到用户会创建多个群组、每个群组中会有多个设备的情况，为了降低对群组和设备信息增删改查的时间复杂度，选择使用哈希表对群组和设备进行的管理。

网关侧的优化模块将全局保存一个群组哈希表用于记录群组信息，记录信息主要包括群组地址、当前群组 ID 分配情况、群组最近一个可靠消息的确认情况以及群组中的设备信息。群组哈希表的节点还将保存一个设备哈希表记录群组中的设备信息，设备信息包括设备地址、设备在该群组中分配到的 ID。在设备加入或是退出群组时，对哈希表执行添加和删除操作。



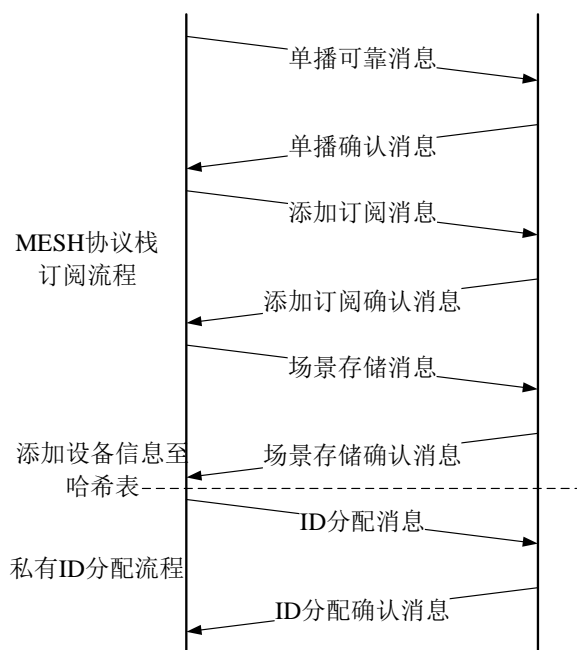


图 4-4 LPN 的订阅流程

终端设备的优化模块的群组信息存储是对上述过程的简化版本，只记录设备加入到的群组以及在群组的 ID。方案中使用的哈希表结构如图 4-5 所示。

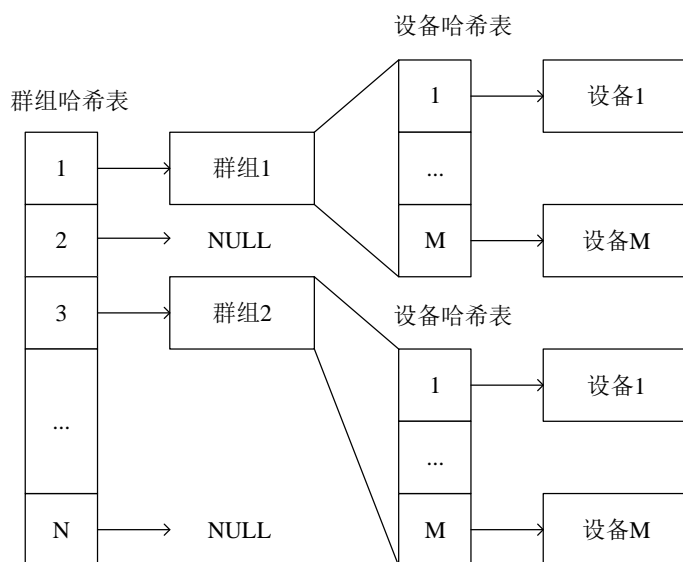


图 4-5 群组和设备哈希表

网关在每个 CFP1 发送群组可靠消息，并在 CFP2 对 LPNs 的确认消息进行接收。在下个 TDMA 帧开始之前，网关判断当前群组中是否还存在有 LPNs 没有回复确认消息。若存在，网关在下个 TDMA 帧继续重传群组可靠消息，若不存在，则此次群

组可靠消息传输完成。同时网关为每个群组可靠消息设定了最大重传次数，若群组可靠消息传输超过了该次数，则此次群组可靠消息传输失败。具体流程见图 4-6。

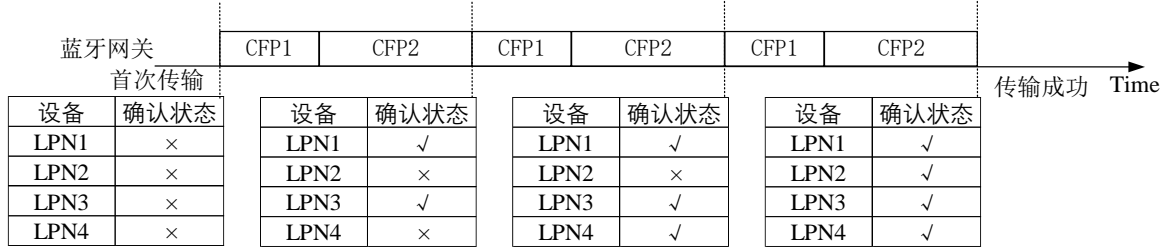


图 4-6 优化接入机制重传过程

## 4.1.4 基于位图的动态时隙分配

### 1. 动态时隙分配方法

考虑到在每一个 TDMA 帧开始之前，会存在一部分的 LPNs 在上个 TDMA 帧成功传输确认消息的情况。若继续按照原来的时隙分配方法进行传输，本可以进行传输的时隙不被利用将减少信道利用率。本文提出的接入机制采用动态 TDMA 方法。其思想为通过在每个 TDMA 帧开始之前统计不同 LPNs 的确认消息传输情况，为暂未成功传输确认消息的 LPN 分配更多时隙进行传输，使得系统的确认状态更快地进行收敛。

群组中共  $N$  个 LPNs 按照  $1, 2, \dots, N$  进行编号， $M$  个时隙按照  $1, 2, \dots, M$  进行编号，时隙个数  $M$  通过式(3-28)得到。网关在每一帧开始前统计暂时未收到确认消息的 LPN 个数  $N_{res}$  以及在这一帧中暂未成功传输确认消息 LPN 的序号排序。序号靠前的时隙将分配给序号靠前的 LPN，将  $M$  个时隙大致均匀分配给  $N_{res}$  个 LPN。考虑到 BLE 广播事件固有事件间隔， $N_{thresh}$  为间隔时间可被分配的时隙个数。假设序号为  $j$  的时隙被分配给序号排序为  $k$  的 LPN，动态时隙分配规则如下：

- (1) 若  $N_{res} > N_{thresh}$ ，则在分配时无需考虑广播间隔

$$j = k + N_{res} * R \quad (4-5)$$

- (2) 若  $N_{res} \leq N_{thresh}$ ，则在分配时必须考虑广播间隔

$$j = k + N_{thresh} * R \quad (4-6)$$

其中  $R$  为自然数，且  $j \leq M$ 。

上述分配规则中，一般情况下有  $N_{res} \leq M$ ，对于  $N_{res} > M$  的情况，序号靠后的设

备将放弃在本次 TDMA 帧中传输确认消息，而在下个 TDMA 帧中进行传输。网关根据网络中还未上报确认消息的 LPN 分配更多的时隙用于传输。使得没有传输成功的 LPN 能够更快地上传确认消息。具体分配流程见图 4-7。

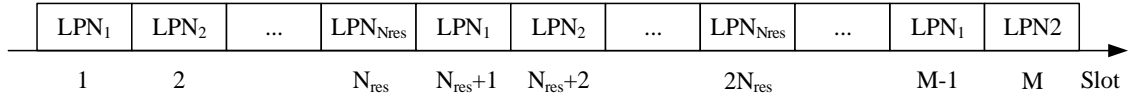


图 4-7 动态时隙分布

LPN 节点在明确上个 TDMA 帧是否有成功确认、这个 TDMA 帧中的未确认 LPN 序号排序 $k$ 以及还需要进行确认消息传输的设备个数 $N_{res}$ 的前提下，可以确定该 TDMA 帧中是否需要发送数据以及在哪个时隙发送数据，这些信息将通过位图表示，并在每一轮重传的群组可靠信息中进行携带。

## 2. 位图的生成与解析

BLE MESH 网络的 PDU 可携带的负载大小是及其有限的，应用层可携带数据只有 11 个字节，本文提出使用位图来对不同设备的接入时隙进行表示，通过消息携带给 LPN 设备。位图中的每一位与群组中的每一个 LPN 相对应，接收到群组可靠消息的 LPN 根据自身的群组 ID 判断确认信息是否被网关成功接收。若确认消息已被成功接收，设备无需再次传输确认消息。若确认消息未被成功接收，设备根据当前群组中未成功传输确认信息的 LPN 个数计算自身传输时隙。位图大小可以自适应调整以适应不同群组大小的情况。位图具体表示见图 4-8。

群组ID	0	1	2	3	...	N-1
确认消息接收情况	×	√	√	×	...	√
位图信息	0	1	1	0	...	1

图 4-8 位图表示

为了实现对 LPN 设备的动态时隙分配，网关在传输群组可靠消息时在消息后面追加当前群组中不同 LPN 的确认情况。考虑到 MESH 网络 PDU 的载荷大小限制，采用位图的形式来表示不同 LPN 的确认情况。位图信息可以根据当前群组中的 LPN 数量进行动态调整。假设群组中 LPN 数量为  $N$ ，考虑到 MESH 网络中传输的信息以

字节为单位，那么完全表示当前群组中确认情况的位图大小  $L$  可以表示为

$$L = \left\lceil \frac{N}{8} \right\rceil \quad (4-7)$$

假设群组中设备数量为 20，那么可以得到对应的位图大小为 2.5 向上取整为 3。位图中的最低位表示序号为 0 的 LPN 设备的确认情况，如图 4-9 所示。

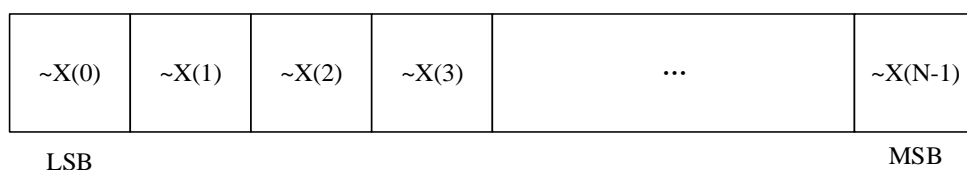


图 4-9 位图生成

上面的位图最左边表示最低位，它代表了序号为 0 的 LPN 设备的确认情况，最右边则是序号为  $N$  的 LPN 设备的确认情况。其中  $X(i)$  为群组中序号为  $i$  的 LPN 设备的确认情况，这里取反表示若设备确认消息传输失败的设备被表示为 1，成功表示为 0。上述位图在处于传输状态的每轮 TDMA 帧结束时进行实时更新并在下一轮发送给 LPN 设备。

相应地，LPN 设备在接收到群组可靠消息之后对位图做出解析。首先要判断设备在之前的确认消息传输是否成功。假设设备序号为  $i$ ，通过判断位图的第  $i$  位来确定。若为 0 则代表消息成功传输，这一轮之后的群组可靠消息都无需再作确认；若为 1 则代表消息传输失败，LPN 需要取得当前未确认的设备数量  $N_{res}$  以及设备序号在当前未确认的 LPN 设备中的排序  $k$ 。前者可以通过计算位图中取值为 1 的比特位数得到，后者则可以通过计算位图第  $k$  位前为 1 的比特位数得到。之后根据 1 小节的动态时隙分配方法得到设备所分配的时隙进行数据传输。

## 4.1.5 消息缓存及 TDMA 接入

蓝牙网关设备和 LPN 设备的消息需要在特定的时间接入信道，因此需要设计消息的缓存机制在设备发送时间段到来之前将需要优化接入的消息缓存起来。消息的缓存采用顺序循环队列进行实现，其结构如图 4-10 所示。

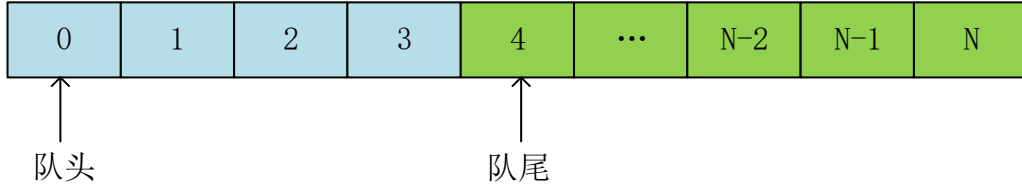


图 4-10 消息缓存队列

这种形式的顺序循环队列使用数组结构实现，队头是下一个出队消息的下标，队尾是下一个入队元素的下标，并做出如下约定：

- (1) 队头=队尾，队列为空。
- (2) 队头=(队尾+1) % 队列长度  $N$ ，队列为满，此时无法再入队新的消息。
- (3) 入队或是出队时，相应的队尾和队头加一并对队列长度  $N$  取模。

上述队列中的每个单元记录了消息相关的信息，包括了对应的群组信息、消息发送的所需要的上下文以及传输次数等信息。设备在需要发送的时隙对队头消息进行发送直至消息传输成功或是超过最大重传次数，之后删除当前队头消息并移动至下一个消息进行传输，直至队列被取空。

对于本地时间戳为  $t_{GATE}$  的蓝牙网关设备，随着时间戳的不断更新，网关设备也对该时间戳进行判断，在  $t_{GATE} \bmod T_p \leq T_{wake}$  时进行群组可靠消息的传输，且在一个休眠周期  $T_p$  之中只传输一次群组可靠消息。对于序号为  $i$  的 LPN 设备，本地时间戳为  $t_{LPN}^i$ ，在  $t_{LPN}^i \bmod T_p \geq T_{wake}$  时关闭信道扫描，在被分配的时隙内进行确认消息的传输，时隙分配的规则见 3.2.5 小节。考虑到确认消息的产生由群组可靠消息触发，因此 LPN 设备在每个周期结束时删除当前传输的确认消息。

实际场景中的蓝牙网关发送的不仅仅只有群组可靠消息，为保证其他类型可靠消息传输的可靠性，通常也会将其他类型消息缓存至队列中进行 TDMA 接入。

## 4.2 基于 TDMA 的 BLE MESH 群组可靠消息接入机制的软件设计

### 4.2.1 开发平台与架构

为了验证提出的 BLE MESH 接入机制，在国内某知名通信公司提供的 BLE MESH 协议栈上做了软件实现。该协议栈的实现基于开源的低功耗协议栈 NimBLE。现在市面上智能家居产品的 BLE 协议栈实现大多是对 NimBLE 修改和适配。

NimBLE 是 Apache Mynewt 提供的开源蓝牙 5.0 协议栈实现，具备完整的 Host 及 Controller 层，且支持 BLE MESH，包括广播和 GATT 承载层、PB-ATT 和 PB-ADV 配置、基础模型、中继支持以及 GATT 代理等功能。NimBLE 对系统资源占用少，并允许应用程序开发人员配置系统以获得最佳性能。开发人员可以通过修改 NimBLE 中的大量选项以实现大吞吐和确保数据传输的低时延。

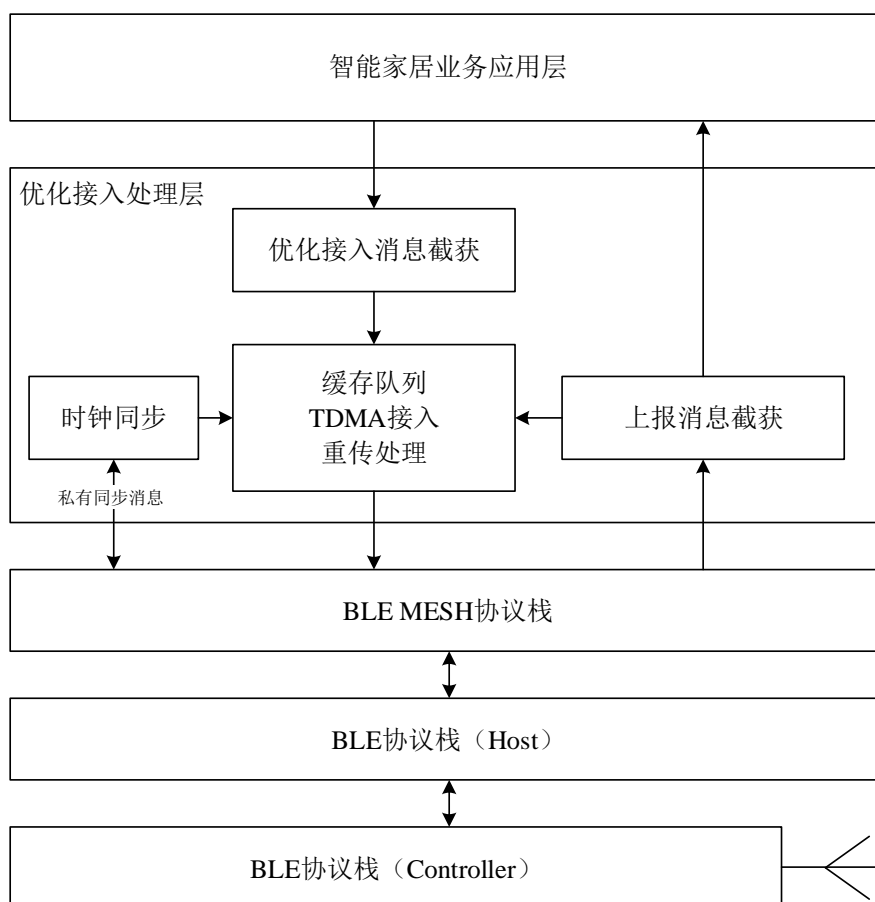


图 4-11 接入机制优化模块与 BLE MESH 协议栈的关系

本文提出优化接入机制的主要部分在应用层实现，同时也会对 BLE MESH 的协议栈部分做出部分修改。接入机制优化模块与协议栈的层次关系如图 4-11 所示。测试使用到的蓝牙网关以及终端设备 MESH 协议栈实现都是基于 NimBLE 协议栈进行修改的。不同设备的软件实现的大致框架是一致的，但具体实现也略有区别。

蓝牙网关侧的软件实现的主要模块包括私有同步消息的构造和发送、优化接入消息的缓存、消息的 TDMA 接入和重传以及对 LPN 确认消息的截获处理。其中优化

接入消息包括群组可靠消息、订阅消息以及单播可靠消息等。LPN 侧的软件实现的主要模块包括私有同步消息的接收和处理、确认消息的缓存、确认消息的 TDMA 接入和重传以及群组可靠消息的截获处理。

## 4.2.2 蓝牙网关侧软件实现

蓝牙网关侧基于 BLE MESH 协议栈的软件实现框架如图 4-12 所示。

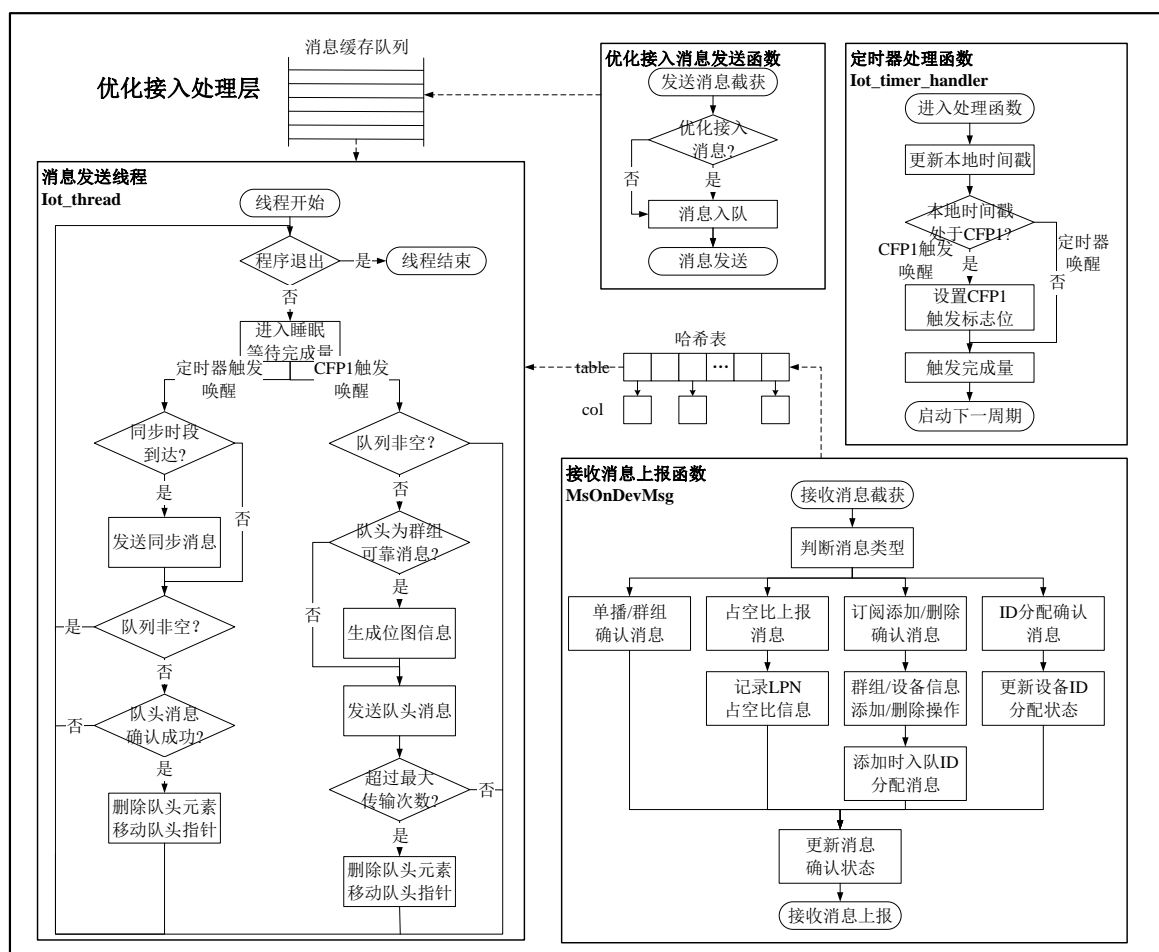


图 4-12 BLE MESH 优化接入机制蓝牙网关侧实现框架

软件运行需要一个初始化流程，该初始化流程包括对哈希表、消息缓存队列、线程及定时器的初始化操作。这些操作被封装在一个函数中，在协议栈初始化完成之后调用。除开优化接入模块的初始化过程，整个软件实现按照功能实现位置进行划分，可被分为四个模块，包括了消息发送线程、定时器处理函数、优化接入消息发送函数以及接收消息上报函数。

在优化接入消息发送函数中,考虑到网关需要对不同类型的消息做优化接入,实际实现的 BLE MESH 协议栈中不同消息调用发送的接口也不一样,单播可靠消息、群组可靠消息、群组订阅/删除消息的函数接口分别为 `SendMsgHandler`、`SendGroupMsgHandler`、`AddModSubHandler` 以及 `DelModSubHandler`。接收消息的上报函数则统一在函数 `MsOnDevMsg` 中,该函数是最接近应用层上报消息的部分。

4.1.1 小节中提到的基于 BLE 广播的时间同步的实现是通过定时器和消息发送线程两个部分联合作用实现。定时器不断触发执行更新本地时间戳并触发线程执行,线程中判断本地时间戳按照一定的周期执行同步消息发送处理。

4.1.3 小节中提到的群组消息的管理部分主要在接收消息上报函数中进行处理。网关在对群组订阅/删除消息的确认消息进行处理时,同时对哈希表记录的群组和设备信息进行添加和删除处理。

4.1.5 小节中提到的消息缓存及 TDMA 接入的实现则需要优化接入消息发送函数、定时器以及线程三个部分联合作用实现。首先,用户通过智能手机操作,手机与 MESH 网关进行通信,使得 MESH 网关应用层产生了 MESH 消息。MESH 消息的发送需要经过优化接入消息发送函数,在优化接入消息发送函数中判断消息类型,并对需要进行优化接入的消息入队缓存。定时器执行的同时对当前时间戳进行判断。如果本地时间戳到达了 CFP1 时间段,设置 CFP1 发送的标志位,之后触发完成量在消息发送线程中对缓存队列中的消息进行出队发送。

4.1.4 小节中提到的基于位图的动态时隙分配将确认消息上报的情况记录在哈希表中,发送线程会在下一轮 CFP1 到来时根据消息确认情况对队列中群组可靠消息的位图进行更新。在发送线程中会判断队头消息的确认情况,对于已经成功确认的消息或是超过最大重传次数的消息,将消息出队代表该消息的传输完成。

### 4.2.3 LPN 侧软件实现

同样地,LPN 侧的软件实现需要首先完成模块的初始化,模块的初始化操作是在协议栈初始化之后调用的。LPN 侧的软件实现按照功能模块区分,同样可以分为同样的四个模块,包括了工作处理线程、定时器处理函数、优化接入消息发送函数以及接收消息上报函数。发送消息的截获处理是在 Model 层的发送处理函数



bt\_mesh\_model\_publish 中实现的，而接收消息的上报处理是在 bt\_mesh\_model\_recv 中实现的。LPN 侧的软件框架实现如图 4-13 所示。

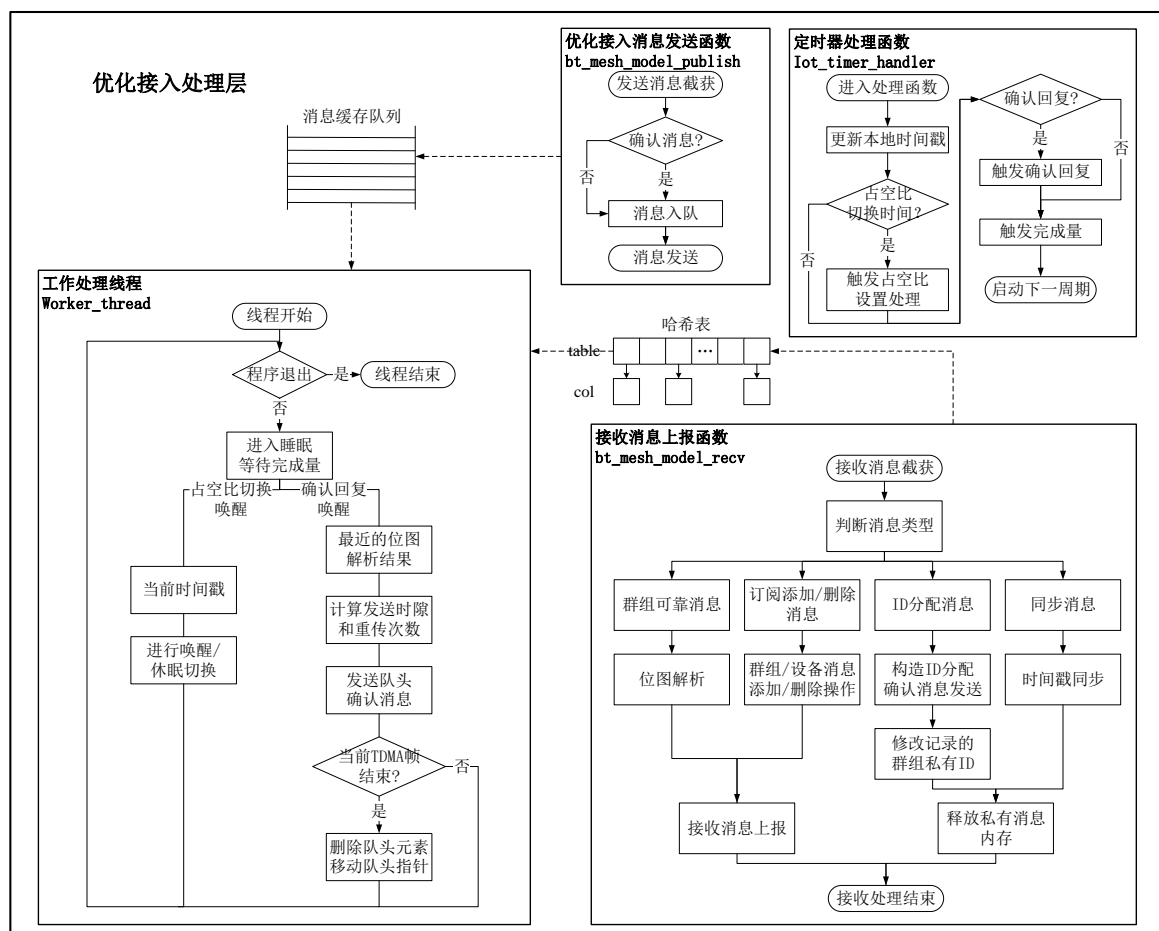


图 4-13 BLE MESH 优化接入机制 LPN 侧实现框架

LPN 设备作为同步消息的接收者，在接收消息上报函数中对接收到的同步消息时间戳按照 4.1.1 小节中的方法进行修正，并将修正后的时间戳按照本地时钟进行更新。时间戳的更新在定时器处理函数中实现，更新之后对当前时间戳进行判断是否需要执行休眠/唤醒的状态切换，定时器处理在需要切换时触发工作处理线程进行设备状态的切换，具体的判断方法见 4.1.2 小节。LPN 的群组消息管理和私有 ID 的分配处理都在接收上报函数中进行处理。在接收到订阅添加/删除消息时，对记录的群组信息继续添加/删除处理处理方法见 4.1.3 小节。软件实现在消息发送函数中对即将发送的确认消息进行缓存，同时在定时器中检测到确认消息的到达，会触发线程的处理流程。确认消息的生成始终与群组确认消息的接收有关。在发送线程中对确认消息进

行处理时会根据最新解析到的群组确认消息位图来选择时隙发送，具体的处理方法见 4.1.4 小节的位图解析部分。

## 4.3 实验结果与分析

### 4.3.1 实验测试环境概述

为了验证提出的优化机制能否在实际场景中提高群组可靠消息以及确认消息的传输时延性能，使用了国内知名芯片厂商的蓝牙网关设备以及蓝牙智能开关设备进行测试，测试环境的网络拓扑如图 4-14 所示。

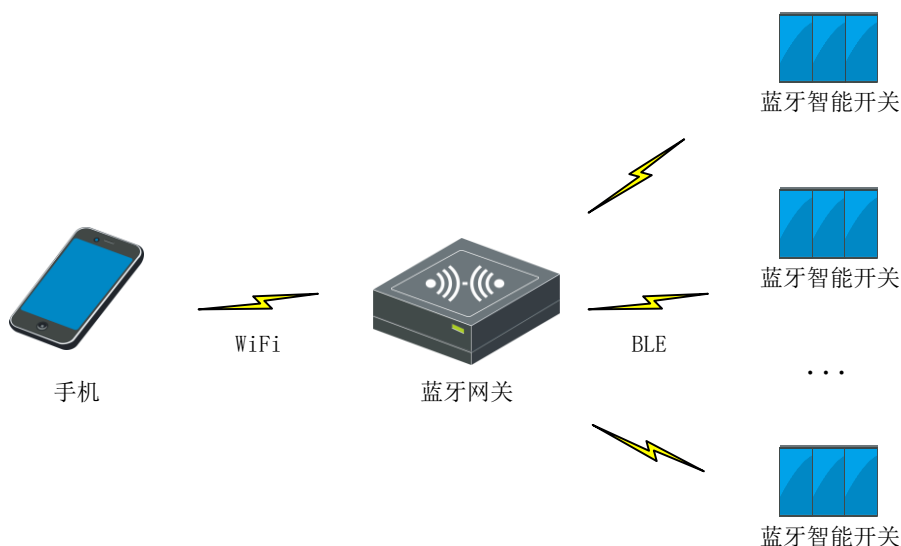


图 4-14 BLE MESH 测试网络拓扑

网络中的智能手机通过 Wi-Fi 实现与蓝牙网关设备的信息交互。在手机上添加过蓝牙网关设备之后，手机控制蓝牙网关对网络中具有一定休眠占空比的蓝牙智能开关进行配网，网关和智能开关面板通过 BLE 实现进行信息交互。

测试过程中智能手机通过点击屏幕对所有的蓝牙智能开关设备进行打开/关闭控制。通过调整网络中的节点数目分别对基于网络层重传的接入机制以及基于 TDMA 的优化接入机制进行了测试，并在蓝牙网关以及开关设备代码中统计了群组可靠消息及其确认消息的传输时延情况。

图 4-15 为测试的整体环境，红色方框选中的部分为蓝牙智能开关、蓝色方框选中的部分为蓝牙网关，黄色方框为发送指令的手机设备。手机中创建了对蓝牙智能开

关进行集体打开或是关闭的场景，脚本点击屏幕交替打开和关闭。结合设备状态变化以及统计到的消息传输情况确认消息传输的可靠性以及实时性。



图 4-15 BLE MESH 实际测试环境

## 4.3.2 实验测试结果分析

实际测试中考虑网络规模大小的影响，在网络中低功耗节点数目为 8、16、32 时对现有机制和优化后机制分别进行了测试，每组测试交替执行打开和关闭的群组可靠消息，共执行约 1000 次，分别记录了群组可靠消息和确认消息传输的平均时延以及最大时延。网络中所有的蓝牙智能开关一致设置为 20ms/60ms 的唤醒休眠占空比。

为了将现有机制和优化机制进行对比，两种机制保证消息传输可靠性是一致的，实测结果均能保证 99.9% 的传输可靠性，也即 1000 次测试中平均失败次数小于等于 1。

### 1. 群组可靠消息时延性能

表 4-1 为经过测试得到的群组可靠消息的最大传输时延。蓝牙智能开关在本地记录了第一次收到某个群组可靠消息时该消息正在执行第几次传输。通过统计不同蓝牙开关面板接收到群组可靠消息时消息传输的当前次数最大值，计算其均值，再通

过式(3-10)以及式(3-34)中发送次数与时延关系得到最大时延。

由表中结果计算可知,平均来看,优化机制的群组可靠消息传输最大时延性能要比现有机制的好 59.63%。

表 4-1 群组可靠消息最大传输时延测试结果

面板数量	接入机制	第一组	第二组	第三组	第四组	第五组	平均
8	现有机制	422.5	425	420	420	417.5	421
	优化机制	80	100	70	80	70	80
16	现有机制	440	445	437.5	435	435	438.5
	优化机制	180	200	205	190	180	191
32	现有机制	465	463.12	461.25	475	462.5	465.37
	优化机制	255	265	297.5	280	265	272.5

从整体趋势上看,无论是基于网络层重传的接入机制还是基于 TDMA 的优化机制,当蓝牙智能开关个数增多时,群组可靠消息的传输时延也相应增大。这是因为设备数量较少时,网络中的大部分设备能够在网关传输较少次数消息时成功接收,而随着设备数量的增多,出现部分设备在网关传输更多次数时才能全部接收成功,从而影响了整体时延性能。从结果上看,不同接入机制在不同网络规模下的时延变化趋势与仿真结果是一致的。

现有接入机制的时延往往更大,且在不同规模的网络上相近的,这是因为消息发送开始时间是随机的,在大量测试中的某些极端情况下,需要网关一直重传至理论上的最小可靠传输次数才能保证设备唤醒且成功接收。而在对优化机制的测试中,因为群组可靠消息被控制在设备唤醒时进行传输,设备在前几轮的传输中就能够成功接收数据。

表 4-2 为经过测试得到的群组可靠消息的平均时延。统计所有蓝牙智能开关在收到群组可靠消息时传输次数的分布情况,计算次数的加权均值,再根据发送次数与时延的关系可以得到平均时延。

从平均时延的变化情况上来看,平均时延并没有最大时延的增大而增大。这是因为平均值的计算对某些极端的大时延值进行了滤波。总的来说,优化机制要比现有机制的群组可靠消息传输平均时延性能表现要好 89.98%。

表 4-2 群组可靠消息平均传输时延测试结果

面板数量	接入机制	第一组	第二组	第三组	第四组	第五组	平均
8	现有机制	245.66	243.98	238.34	241.61	235.69	241.05
	优化机制	20.66	19.77	20.23	19.92	20.50	20.22
16	现有机制	246.88	231.11	231.72	233.73	217.19	232.13
	优化机制	23.43	22.32	21.24	23.37	20.29	22.13
32	现有机制	253.65	243.08	232.84	253.00	223.76	241.27
	优化机制	28.77	30.11	28.28	28.43	29.99	29.12

## 2. 确认消息时延性能

网关记录了每个群组可靠消息从第一次发送到其所有确认消息成功接收的时间差作为确认消息的时延，同样地记录了平均时延以及最大时延。最大时延结果如表 4-3 所示。

确认消息的时延是通过网关本地记录每次群组消息的发送时间以及该消息所有确认消息的接收完成时间，之后对这两个值取差值求得。总的来说，优化机制的确认消息传输最大时延表现要比现有机制平均要好 77.95%。

表 4-3 确认消息最大传输时延（ms）测试结果

面板数量	接入机制	第一组	第二组	第三组	第四组	第五组	平均
8	现有机制	1515	1480	1481	1486	1449	1482.2
	优化机制	199	197	210	218	219	208.6
16	现有机制	1544	1525	1506	1543	1490	1521.6
	优化机制	381	356	354	386	332	361.8
32	现有机制	1558	1555	1565	1567	1545	1558
	优化机制	455	458	434	444	413	440.8

从整体数值水平上来看，现有机制的确认消息成功传输时延要远远大于优化机制，这是因为现有机制设置了大量的退避时间来避免设备间冲突。而优化机制取消了大量的退避时间，并通过同步以及为各个设备划分时隙来传输确认消息避免碰撞，从而降低了时延。

从时延随网络规模的变化情况上看，随着设备数量的增多，两种机制的传输时延都相应地增大了。这是因为在现有机制中，设备数量增多，节点间碰撞的概率也会增

大,设备需要更多的传输次数保证可靠,某些极端情况还出现了确认消息完全丢失的情况。而在优化机制中,设备数量的增多一方面造成了群组可靠消息需要更多的 TDMA 帧来传输来保证所有设备接收,另一方面,每个设备平均每轮的传输时隙的减少,需要更多 TDMA 帧来进行反馈。

表 4-4 确认消息平均时延测试结果

面板数量	接入机制	第一组	第二组	第三组	第四组	第五组	平均
8	现有机制	1338.9	1339.3	1348.2	1347.7	1347.8	1344.4
	优化机制	112.43	111.56	109.25	106.82	113.77	110.77
16	现有机制	1346.4	1328.5	1346.8	1355.5	1344.6	1344.4
	优化机制	181.07	180.80	178.95	180.44	179.09	180.07
32	现有机制	1351.3	1348.0	1336.7	1343.2	1331.4	1342.1
	优化机制	222.69	233.59	236.08	245.81	226.75	232.99

平均时延的测试结果如表 4-4 所示,平均来看,优化机制的确认消息传输平均时延相比现有机制下降了 87.00%。可以看到现有机制受退避时间影响平均时延普遍偏大,但随着设备数量的增多,时延性能没有明显变化,这说明现有的退避机制总的来说能够有效避免设备间碰撞。优化机制在设备数量增多时需要更多的 TDMA 帧来保证群组可靠消息及确认消息成功传输,因此确认消息的时延也相应增大。

## 4.4 本章小结

本章针对第三章中提出基于 TDMA 的群组可靠消息接入机制,结合智能家居中的实际应用场景,提出了新的接入机制的具体实现方案,并在实际设备上进行了测试。测试结果证明新的接入机制能够有效提升群组可靠消息及其确认消息的接入时延性能。首先以功能模块为单位先后介绍了基于 BLE 广播的低成本同步方法、休眠占空比时间设置、群组信息管理与重传、基于位图的动态时隙分配以及消息缓存及 TDMA 接入的实现原理,然后整体介绍了在开源 BLE MESH 协议栈上进行软件实现的流程,最后对方案进行了测试验证。

测试结果表明,在不同的单跳 BLE MESH 网络规模以及消息传输保证 99.9%可靠的前提下,基于 TDMA 的群组可靠消息接入机制相较于现有接入机制能够有效提

升消息接入的最大时延以及平均时延性能。

然而，在实际场景中，随着 BLE MESH 网络中的群组以及群组成员个数增多，基于 TDMA 方式的群组可靠消息接入将出现如下问题：1) 在极端条件下，BLE MESH 网络中的节点数目可以到达 32767 个。网关需要消耗大量的内存和处理资源来管理群组信息。2) 群组中成员增多，相应地需要在消息中增加位图长度。在网络规模较为庞大时，位图负载大小将会超过原本群组控制消息的负载大小。3) 目前 LPN 设备的分时接入以高精度定时器为基础，网络整体能量消耗将会有一定程度上升，还需要进一步优化。

## 5 智能家居场景中 IoT Wi-Fi 优化接入的设计与实现

在智能家居场景下基于 Wi-Fi 技术的 WLAN 中，大量不同流量特性的终端设备通过单个 AP 接入信道导致业务传输时延高、系统带宽下降，严重影响用户体验。在上述背景下，本文对智能家居场景中 IoT Wi-Fi 信道接入的优化问题展开研究。

### 5.1 IoT Wi-Fi 接入需求及现有机制问题分析

基于 Wi-Fi 的智能家居场景下的网络部署通常是 WLAN 中的所有终端设备通过单个路由器接入网络，属于典型的单 AP 多 STA 的密集接入场景。这样的接入场景具有如下特点：

(1) 终端设备数量较多。目前一些智能家居场景中设备数量能够达到 30 到 50 个终端设备，未来还会增加。

(2) 终端设备类型较多。不同终端设备具有不同的流量特性，具有异构流量特性。例如，智能网络电视的主要流量为下行流量，智能网络摄像头以上行流量为主。不同设备的流量大小也不尽相同，具有不饱和的流量特性。

(3) 终端设备使用的 Wi-Fi 协议版本较低。Wi-Fi 6 提供了对智能家居场景下终端设备接入的解决方案，但出于成本的考虑，大部分终端设备仍然使用较老的 Wi-Fi 协议，且暂时无法更新。

智能家居场景下的终端设备采用 CSMA/CA 接入机制。通过 Wi-Fi 接入介质的终端设备通常采用 EDCA 或是 CSMA/CA 机制终端设备数量庞大会引起设备接入信道时碰撞概率高，数据传输成功率降低。在设备数量比较极端的情况下，站点信道竞争成功率将会降低至 10% 以下。与此同时，终端设备的异构流量特性可能会引起小流量设备难以成功接入信道，产生接入的不公平问题。因此需要提出新的接入机制来保证智能家居业务能够可靠、及时、公平地接入网络。

针对 CSMA/CA 机制的性能分析研究目前已相对成熟。其中较为经典的分析模型为单数据包和饱和流量模型。在单数据包模型中，所有站点有且仅有一个数据包待发送，而在饱和流量模型中，站点将始终有一个数据包待发送。两种模型的信道争用成功概率随网络中站点数量变化的数值仿真结果如图 5-1 所示。可以看到无论何种



模型，随着站点数量的增加，信道争用成功概率随之降低。同时随着信道争用概率的恶化，各站点会持续进行 BEB 算法，最终超过最大重传次数导致数据包被丢弃。

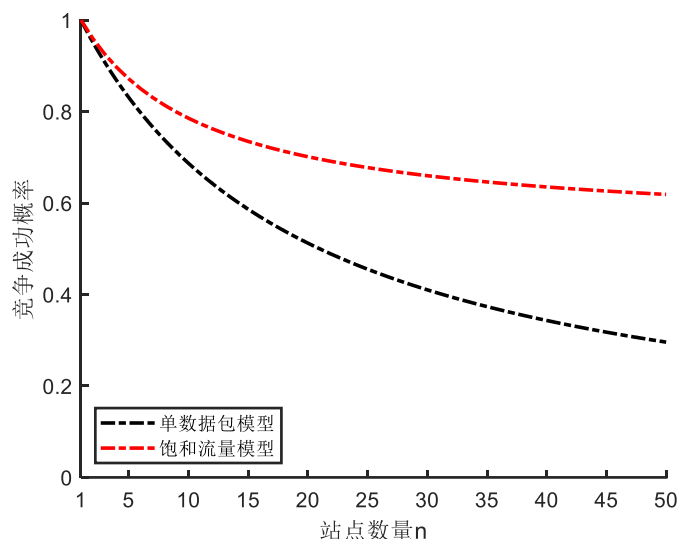


图 5-1 CSMA/CA 接入机制不同流量模型下的信道竞争成功率

虽然智能家居业务属于异构流量模型，但已经有研究表明可以通过对上述两种流量模型的估计来拟合任何流量模式的信道争用成功概率的回归函数。异构流量模型下的争用成功概率变化曲线处于单数据包模型和饱和流量模型之间，且具有相同的走势。由此可见，流量因素是影响系统性能的一个重要因素。

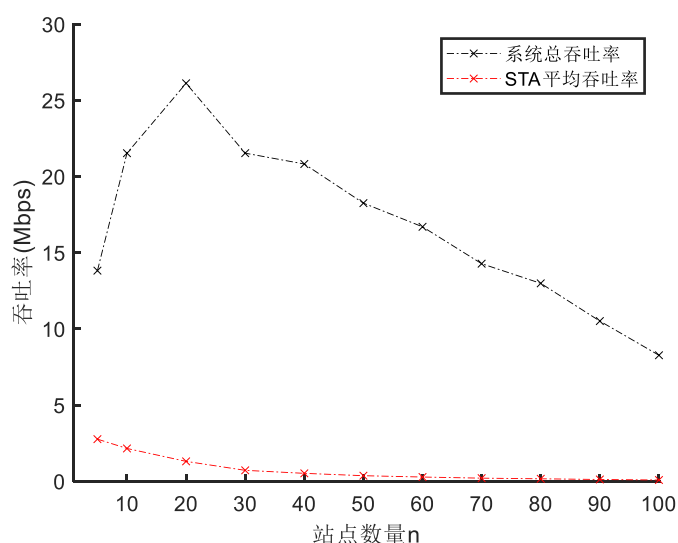


图 5-2 异构流量模型下系统吞吐随站点个数的变化

图 5-2 为基于 CSMA/CA 接入机制的 Wi-Fi 网络中系统吞吐率以及 STA 吞吐率随站点数量变化的情况。随着站点数量增加，站点信道竞争成功概率不断降低。在站点数量较少时，系统吞吐还处于上升趋势。但当站点数量超过一定的数量时，系统吞吐急剧恶化，单个设备吞吐甚至趋近于 0。由此可知，站点间冲突概率是影响系统吞吐的重要因素之一，降低站点间冲突概率能够对系统性能起到优化作用。

## 5.2 IoT Wi-Fi 接入机制优化技术路线

针对智能家居场景中的 IoT Wi-Fi 接入需求，结合现有信道接入机制的特点以及 802.11 ah RAW 机制的接入思想，提出了新的接入机制，其核心思想包括以下两点：

(1) 协调 WLAN 中的 Wi-Fi 终端设备进行分时分组随机接入，通过降低直接接入信道的设备数量来降低冲突概率，从而提高系统性能。将终端设备进行分组，一组设备共享相同的时隙进行随机接入，组内随机接入竞争信道，组间分时接入信道。

(2) 按照设备流量特征进行分组使得不同分组流量特性基本一致，在有效避免终端设备接入不公平问题的同时最大限度地利用时隙资源。

按照机制特点，将机制命名为基于流量均衡的受控并行随机接入机制。机制在具体实现时需要在 WLAN 中设置一个协调者来协调不同终端分时接入，协调者可以设置是 AP（路由器）也可以设置为 STA（终端设备）。协调者需要负责协调其他终端设备分时分时接入，同时统计不同设备的流量特性，并以此作为依据进行分组。根据实验室已有成果，以 STA 作为协调者具有天然的短板，这是因为：

(1) 终端设备随时可能会退出 WLAN，系统需要执行周期性地分布式选举保证鲁棒性，提高了系统复杂度。

(2) 终端设备需要通过报文交互统计 WLAN 中设备的流量特性，在设备数量较大时会占据一定量的空口带宽。

(3) 网络中将会存在不可控的设备业务流量随机接入。实际 WLAN 中的终端设备只有部分可以进行软件修改，其他设备无法执行受控接入。

相比之下，以 AP 作为协调者具有如下优势：

(1) 路由器在作为终端设备的接入点将随网络持续存在，因此网络无需执行分

布式选举，降低了系统复杂度。

(2) 通过修改路由器侧代码可以方便地对网络中终端设备流量特性进行统计，避免了消息交互的过程。

(3) 路由器可以利用智能家居 TCP 业务的特点对不可控设备低优先级背景流量进行控制，从而降低不可控流量随机接入对系统性能的影响。

基于上述分析以及实验室已有实现的以 STA 作为协调者的方案实现缺陷，本文提出一种以 AP 作为协调者的流量均衡受控并行随机接入的实现方案。

## 5.3 流量均衡受控并行随机接入机制

在保证接入性能的前提下，实现 Wi-Fi 流量的均衡分配至关重要。流量特征是影响接入性能的一个重要因素，因此每个周期内需要对每个终端设备的流量特征进行统计和预测，以此作为终端设备分组的依据。为了解决大量终端设备同时接入信道时的冲突问题，提出的新接入机制采用“分而治之”的策略，只允许部分终端设备在某个时间段内抢占信道资源，从而达到受控的目的。此外，在一个分组周期内，原本发生碰撞的多个节点可以在将其划分到不同的时间段后在同一周期内完成数据传输，从而实现并行。另外，在一个周期中，将所有设备划分到某个时间段内，通过 DCF 机制抢占信道，使得所有设备可以随机接入信道，从而实现随机接入。

### 5.3.1 系统模型概述

考虑在智能家居场景中有  $N$  个终端设备通过同一个 AP 接入 WLAN，这  $N$  个终端设备按照  $i = 0, 1, 2, \dots, N - 1$  进行编号。终端设备向 AP 发送数据，只考虑网络中的上行流量。终端以泊松分布产生数据流，通过 AP 转发给远端服务器。由于业务类型不一样，WLAN 中的各终端设备以不同速率产生流量。

在受控并行随机接入机制中，时间按周期  $T_p$  进行分段，每个周期中包含了  $M$  个小的时隙。每个时隙中安排一组设备进行随机接入，分组依据流量特性。理想状态下的各组设备流量大小之和是相同的。假设第  $j$  个分组包含了  $n_j$  个设备。 $M$  个时隙对应了  $M$  个组，每个时隙大小被表示为  $slot_j$ ，有

$$\sum_{j=0}^{M-1} slot_j = T_p \quad (5-1)$$

$$\sum_{j=0}^{M-1} n_j = N \quad (5-2)$$

分组规则被表示为

$$X_{i,j} = \{0,1\} \quad (5-3)$$

$$\sum_{j=1}^K X_{i,j} = 1, i \in [0, N-1] \quad (5-4)$$

$X_{i,j}$  为二值变量, 表示了终端设备  $i$  是否加入了第  $j$  个分组, 加入为 1, 否则为 0。每个设备必须且只能加入到一个时隙中。系统网络拓扑如图 5-3 所示。

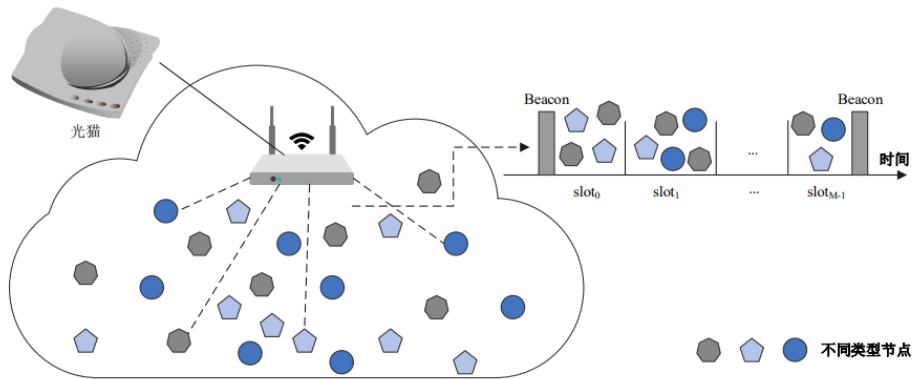


图 5-3 受控接入机制网络拓扑图

### 5.3.2 系统性能分析

#### 1. 非饱和马尔科夫模型

Bianchi 提出的针对 CSMA/CA 机制的马尔科夫分析模型相较于之前的其他模型更加准确<sup>[51]</sup>。但该模型与实际接入机制有所差别, 在 Bianchi 提出的模型的基础上, 考虑不饱和流量模型的系统性能。首先做出了如下假设

1. WLAN 中的 STA 数据包达到模型为到达率为  $\lambda$  的泊松流, 缓存队列大小为  $K$ 。
2. 每个 STA 可以通过物理或虚拟载波监听的方法“听”到其它的 STA。
3. 某个 STA 发送数据包时碰撞概率只与本次发送相关, 而与之前的发送情况无关。

4. 信道为理想信道，不会在传输过程中产生错包。

定义两个离散时间、离散值的随机过程

(1)  $s(t)$ : 给定 STA 的退避阶数随机过程，最大退避阶数为  $R$ 。

(2)  $b(t)$ : 给定 STA 的退避计数器随机过程，该值在每个时间步长开始时更新，若侦听到空口繁忙则不做修改，否则减去 1。

对单个数据包的最大重传次数设置为  $R + f$ ，STA 间碰撞概率为  $p$ ，缓存中存在至少一个数据帧需要发送的概率为  $q$ 。以退避时隙作为步长，之后将  $\{s(t), b(t)\}$  建模为离散时间马尔科夫链。系统状态转移如图 5-4 所示。

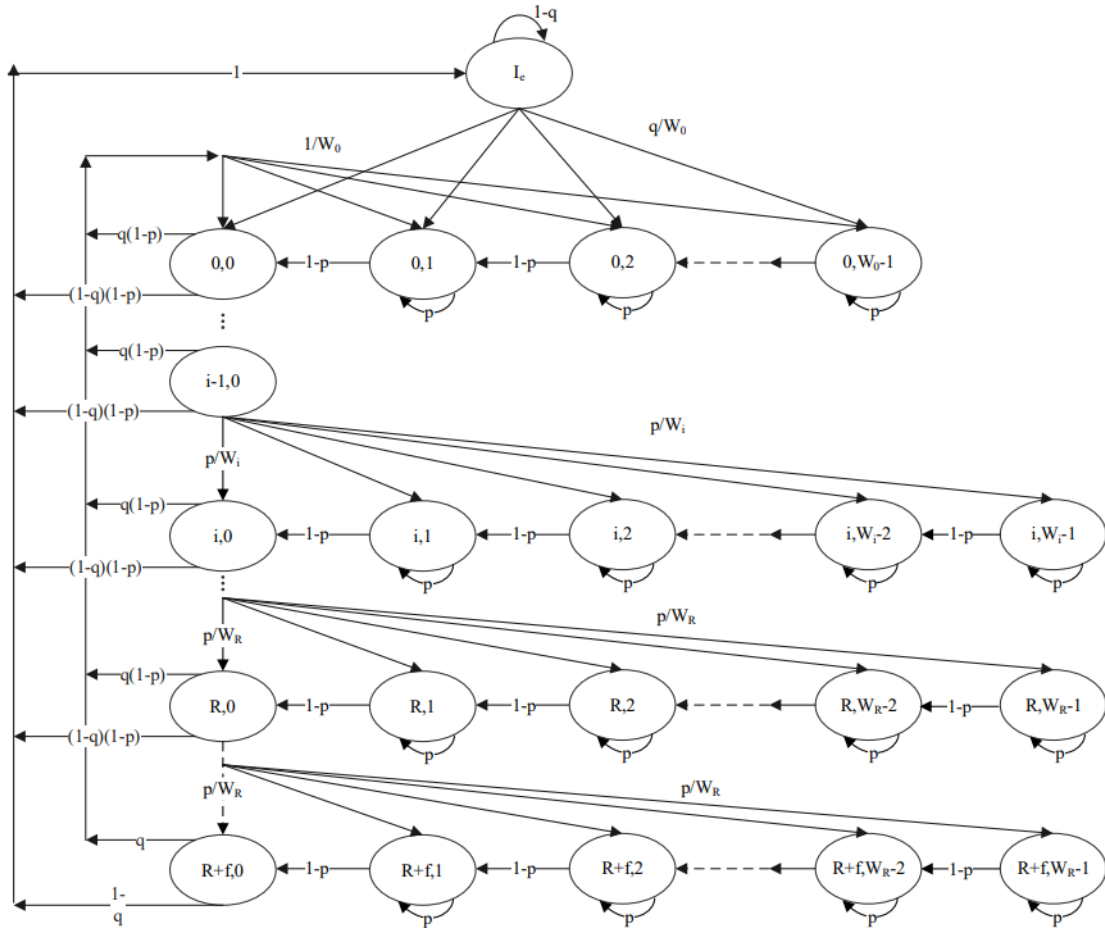


图 5-4 系统状态转移图

已知最大退避阶数可以得到，不同退避阶数下的退避窗口变化情况，假设设备在第  $i$  阶退避窗口的值为  $W_i$ ，有

$$W_i = \begin{cases} W_0 = W_{min}, i = 0 \\ 2^i W_0, i \in [1, R] \\ 2^R W_0 = W_{max}, i \in [R + 1, R + f] \end{cases} \quad (5-5)$$

需要注意的是系统考虑了不饱和流量的情况，引入了一个 STA 队列中无缓存数据的状态  $I_e$ ，表示非饱和状态下的等待状态。在不饱和流量的情况中，终端设备在某个步长时间过后仍保持  $I_e$  状态的概率为  $(1 - q)$ ， $q$  可以通过 M/M/1/K 排队模型计算得到。

各状态一步转移概率可以表示为

$$\begin{cases} P(i, k-1|i, k) = 1 - p, i \in [0, R + f], k \in [1, W_i - 1] \\ P(i, k|i-1, 0) = \frac{p}{W_i}, i \in [1, R + f], k \in [0, W_i - 1] \\ P(0, k|i, 0) = \frac{q(1-p)}{W_0}, i \in [0, R + f - 1], k \in [0, W_0 - 1] \\ P(0, k|R + f, 0) = q, k \in [0, W_0 - 1] \\ P(I_e|i, 0) = (1-p)(1-q), i \in [0, R + f - 1] \\ P(I_e|R + f, 0) = (1-q) \\ P(0, k|I_e) = \frac{q}{W_0}, k \in [0, W_0 - 1] \end{cases} \quad (5-6)$$

式 (1) 表示在一个退避时隙内，STA 检测到空口持续空闲，退避计数器减去 1；式 (2) 表示 STA 在第  $(i-1)$  阶的退避计数减为 0 后，尝试发送数据帧时发生碰撞，从而执行第  $i$  阶退避，需要注意的是，退避窗口只在阶数未超过最大退避阶数时增大，其他情况维持  $W_{max}$  不变；式 (3) 表示 STA 在第  $i$  阶退避完成后有数据且发送成功，同时退避窗口减小为  $W_{min}$ ；式 (4) 表示 STA 的某个数据帧在超过最大重传次数后，退避阶数归零；式 (5) 表示当 STA 处于非最大退避阶数且退避完成时无数据发送，状态回到  $I_e$ ；式 (6) 表示 STA 处于最大退避阶数且退避完成时无数据发送，状态回到  $I_e$ ；式 (7) 表示 STA 在数据包到来，状态从  $I_e$  进入第 0 阶退避。可以求得非饱和马尔科夫模型的封闭解为

$$b_{0,0} = \left[ \frac{1-p}{q} + \frac{1}{2(1-p)} \left( \frac{W_0(1-(2p)^{R+1})}{1-2p} + \frac{1-p^{R+1}}{1-p} + \frac{(2^R W_0 + 1)(p^{R+1} - p^{R+f+1})}{1-p} \right) \right]^{-1} \quad (5-7)$$

任意一个 STA 进行数据传输的概率  $\tau$  为

$$\tau = \sum_{i=0}^{R+f} b_{i,0} = \frac{1 - p^{R+f+1}}{1 - p} b_{0,0} \quad (5-8)$$

假设网络中存在  $n$  个 STA，一个退避时隙中的冲突概率  $p$  为

$$p = (1 - (1 - \tau)^{n-1}) \quad (5-9)$$

式(5-9)为非饱和流量模型下传统 CSMA/CA 机制的平均冲突概率，在  $R$ 、 $f$ 、 $W_0$  给定的情况下，冲突概率为站点数量  $n$  的函数，有

$$p = g(n) \quad (5-10)$$

系统吞吐可通过下式计算

$$S = \frac{P_S P_{tr} E[P]}{(1 - P_{tr})\sigma + P_{tr} P_S T_S + P_{tr} (1 - P_S) T_c} \quad (5-11)$$

$P_{tr}$  和  $P_S$  分别代表了在任意一个时隙中，存在至少一个站点发送数据和数据成功传输的概率； $T_S$  和  $T_c$  表示在成功传输时和发生传输冲突时的退避时隙长度； $\sigma$  表示在信道空闲情况下的退避时隙长度； $E[P]$  表示 STA 传输数据帧长度的期望值。其中  $P_{tr}$  和  $P_S$  分别表示为：

$$P_{tr} = 1 - (1 - \tau)^n \quad (5-12)$$

$$P_S = \frac{n\tau(1 - \tau)^n}{P_{tr}} = \frac{n\tau(1 - \tau)^n}{1 - (1 - \tau)^n} \quad (5-13)$$

当采用 CSMA/CA 基础接入方式有  $T_S$  和  $T_c$  如下：

$$\begin{cases} T_S = H + E[P] + SIFS + \delta + ACK + DIFS + \delta \\ T_c = H + E[P^*] + DIFS + \delta \end{cases} \quad (5-14)$$

其中  $\delta$  表示传播时延， $E[P^*]$  为冲突发生时涉及的最大负载长度期望，为了简化分析，令  $E[P^*] = E[P]$ ，其他参数通过 802.11 协议取得，同样地，系统吞吐率为站点数量  $n$  的函数，有

$$S = h(n) \quad (5-15)$$

## 2. 流量受控并行接入机制性能

对于流量受控并行随机接入机制，假定不同站点流量特性一致是同质的，第  $j$  个时隙大小  $slot_j$  和时隙中的设备个数  $n_j$  满足

$$\begin{cases} slot_j = \frac{T_p}{M}, j \in [0, M-1] \\ n_j = \frac{N}{M} \end{cases} \quad (5-16)$$

定义一个周期内平均碰撞概率 $p'$ 和平均吞吐率 $S_{total}'$ 为

$$p' = \frac{\sum_j p_j}{T_p} \quad (5-17)$$

$$S_{total} = \frac{\sum_j (S_j \times slot_j)}{T_p} \quad (5-18)$$

其中 $p_j$ 、 $S_j$ 分别代表了第 $j$ 个时隙具有 $n_j$ 个站点的冲突概率和吞吐率,由式(5-10)、(5-15)、(5-17)、(5-18)其周期内平均冲突概率 $p'$ 和周期内平均吞吐率 $S_{total}'$ 为

$$p' = g\left(\frac{n}{M}\right) \quad (5-19)$$

$$S_{total}' = h\left(\frac{n}{M}\right) \quad (5-20)$$

### 5.3.3 数值仿真结果

在 Matlab 中采用表 5-1 中的参数对传统 CSMA/CA 接入机制以及受控并行随机接入的平均冲突概率以及系统吞吐率进行了数值仿真,冲突概率结果见图 5-5。

表 5-1 数值仿真参数设置

参数	取值
数据帧速率	54Mbps
控制帧速率	24Mbps
传播时延	2us
时隙时长	9us
SIFS	16us
DIFS	34us
最小竞争窗口	15
最大竞争窗口	1023
最大重传限制	6

其中  $M$  为分组个数,可以看到在相同 STA 数量的前提下,改进后机制的冲突概率始终比传统 CSMA/CA 机制要小,在 STA 数量较大时,冲突概率降低的效果更加明显。



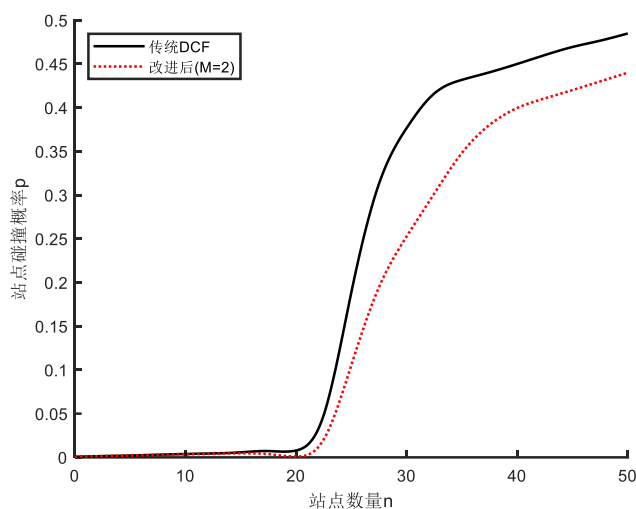


图 5-5 改进前后系统平均冲突概率对比

改进前后的系统吞吐率对比如图 5-6。

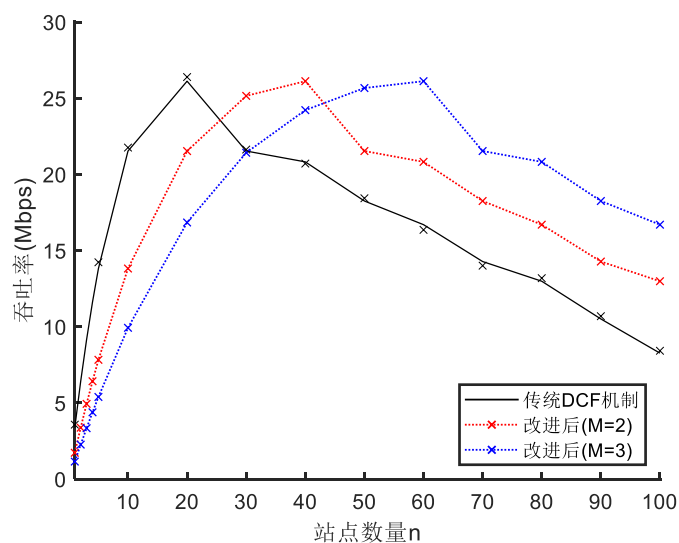


图 5-6 改进前后系统吞吐率对比

在站点数量较少时，此时系统处于非饱和状态，站点分时接入对信道利用率产生了负面影响，使得改进机制的系统吞吐率低于 CSMA/CA 机制。但在站点数量较多时，改进机制的吞吐始终比传统机制要高。

通过上述仿真结果，可以看到在网络中设备数量较多时，受控并行随机接入机制能够有效起到降低冲突概率并提升系统吞吐的作用。

## 5.4 对部分终端透明的流量均衡受控并行随机接入方案设计

### 5.4.1 方案设计概述

对终端透明指的是通过修改 AP 侧代码来让 AP 作为协调者来协调其他终端设备进行受控并行随机接入，不对终端设备代码进行修改。但是考虑到实际网络中也会存在部分终端设备可以修改软件，因此方案对部分终端透明。WLAN 中同时存在可修改软件的可控终端设备以及不可修改软件的不可控终端设备。网络拓扑如图 5-7 所示。对可控终端设备，AP 采取协调帧通告的方式来控制接入；对不可控设备，AP 根据其 TCP 业务的特性来大致控制接入时间。

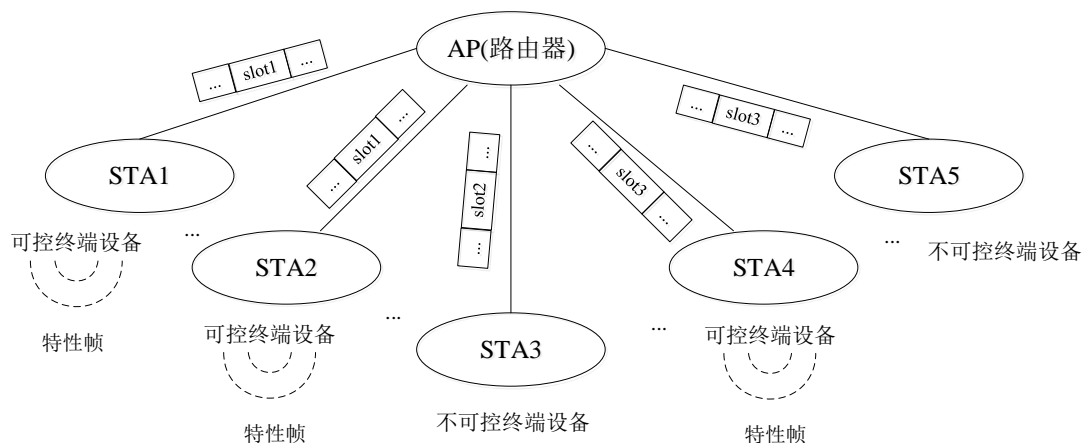


图 5-7 对部分终端透明方案网络拓扑图

网络中涉及到两种私有帧，协调帧以及特性帧。协调帧是路由器以固定时间间隔发送的广播帧，其中包含了各个终端设备的分组信息。特性帧是可控终端设备发送的单播帧，其中包含了可控终端设备的类型等信息，具有向路由器通知自身属性的功能。

### 5.4.2 同步机制

终端设备间的随机受控并行接入是以设备间时间同步为前提实现的，时间同步的精确程度在一定程度上影响了接入性能。在 IEEE 802.11 中提供了两种不同的同步机制（TSF）。这两种不同的同步机制分别用于 BSS 以及 IBSS 网络模式中。本文讨论的 WLAN 采用 BSS 网络模式建立起星形拓扑，因此使用 BSS 网络模式下的同步机制。

在 BSS 网络模式下实现时间同步的过程为

- 1、AP 和 STA 在本地都会维护一个 64 位 TSF 定时器，定时器精度为毫秒级。
- 2、AP 作为网络中的同步源，周期性广播信标帧。信标帧中的 Timestamp 字段包含了本地 TSF 定时器时戳。
- 3、STA 在接收到 AP 的信标帧之后将其中的 TSF 时戳取出，考虑收发处理时延以及传输时延对该时戳进行修正处理。之后将修正后的值与自身维护的 TSF 时戳值进行对比，若不同，则将本地时间戳更改为修正后的 TSF 时间戳。
- 4、在 STA 没有成功接收到某个信标帧的情况下，本地的 TSF 时间戳通过自身定时器触发进行更新，也能够与网络整体进行同步。

终端设备通过获取本地的 TSF 时间戳就可以与网络中的其他设备进行同步。

### 5.4.3 接入周期动态调整

在 Wi-Fi 网络中所有终端设备时间同步的基础上，接入周期 $T_p$ 被划分为多个时隙，接入周期结构如图 5-8 所示。AP 按照接入周期定时发送协调帧，其中包含了不同终端设备接入的时隙信息。STA 接收到协调帧之后根据当前时间点按照协调帧中给出的接入时隙执行接入。

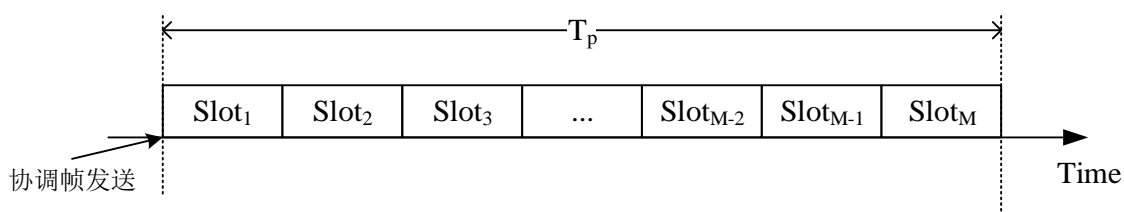


图 5-8 Wi-Fi 接入周期

现在主流的路由器最多可同时支持接入 256 个终端设备。在这种终端设备数量非常多的情况下，使用固定的分组个数可能会导致单个时隙内进行随机接入的终端个数增加，引起系统性能下降。而在接入终端数量减少时，固定分组引起单个时隙接入设备数量降低，导致信道利用率降低。

在终端设备数量  $n$  确定的前提下，分组数量  $M$  决定每个时隙中的接入设备数量，对式(5-20)求极值可以求得最佳分组个数 $M_{opt}$ 。已知单个时隙大小为slot，周期时长调整为

$$T_p = M_{opt} * slot \quad (5-22)$$

更新后的分组和周期信息通过协调帧发送给可控终端设备，其他类型的设备的受控接入则通过路由器本地控制。

#### 5.4.4 流量统计和设备类型区分

流量均衡的受控随机接入要求协调者对 WLAN 中其他终端设备的流量情况进行统计。AP 也即路由器能够很方便地对终端流量进行统计，方案采用定时统计和平滑处理的方式，来记录不同终端设备的流量。为了降低增删改查复杂度，采用哈希表对设备类型以及流量情况进行统计。

假设 WLAN 中的所有终端设备按照自然数从小到大进行编号，用  $T_p$  表示受控并行随机接入机制的一个周期大小，路由器以周期为单位统计终端设备的流量情况，分别用  $flow_d^i(j)$ 、 $flow_u^i(j)$  来表示编号为  $i$  的设备在第  $j$  个周期的下行流量、上行流量。按照如下方式对统计到近  $m$  个周期的流量进行平滑处理。

$$SmoothFlow_d^i(j) = \sum_{k=0}^m \alpha_k \times flow_d^i(j-k) \quad (5-23)$$

$$SmoothFlow_u^i(j) = \sum_{k=0}^m \beta_k \times flow_u^i(j-k) \quad (5-24)$$

其中  $\alpha_k, \beta_k \in [0,1]$ ，且  $\sum_{k=0}^m \alpha_k = 1$ 、 $\sum_{k=0}^m \beta_k = 1$ ， $\alpha_k, \beta_k$  的值随着  $k$  的增大而增加， $m$  的值越大考虑的周期个数更多， $m$  和  $\alpha_k, \beta_k$  值通过实际情况进行动态调整。

若在某一个周期统计到某个终端的平滑下行流量超过一定的阈值  $SmoothFlow_d^{thresh}$ ，则判断设备为移动终端设备，否则为智能家居设备。设备的具体分类以及处理方法为

(1) 智能家居设备：流量主要为上行流量，下行流量较少。智能家居设备的所有流量都需要执行受控并行随机接入。根据设备是否可以通过修改软件适应方案设计可以被区分为可控和不可控的智能家居设备。

a) 可控智能家居设备：可控智能家居设备通过软件实现对自身所有流量进行受控接入处理。

b) 不可控智能家居设备：不可控智能家居设备的受控接入通过路由器处理

实现。

(2)移动终端设备：移动终端设备的软件都是不可修改的。其受控接入需要在路由器实现。考虑到移动终端设备高优先级业务如语音、视频业务等对时延敏感，只对这种设备的低优先级背景业务进行受控接入处理。

考虑到移动终端设备只需要对低优先级业务的流量进行受控接入，对设备的访问类型 AC 属于 BE、BK 的流量进行统计，使用  $flow_{LowPri}^i(j)$  表示序号为  $i$  的移动终端设备在第  $j$  个周期的上行低优先级流量，同样地，采用与式(5-24)一样的方法得到第  $j$  个周期的上行低优先级平滑流量  $SmoothFlow_{LowPri}^i(j)$ 。

已确认的可控设备和移动终端设备通过 MAC 地址进行唯一标识。路由器将这些设备的 MAC 地址和类型写入文件系统，在下次启动时读出用于后续设备区分处理。

## 5.4.5 不同类型终端设备的受控接入实现

根据 5.4.4 小节中的设备类型分类方法，对不同类型终端设备采用不同的受控接入方式。

(1)可控智能家居设备：通过解析路由器定时广播的时隙分组信息，确定发送时隙。设备在传输时隙处于“苏醒”状态进行数据传输。而在其他时隙处于“休眠”状态，将需要发送的数据帧缓存至本地队列中，等到传输时隙再进行数据传输。

(2)不可控智能家居设备：不可控设备的软件无法被修改，因此不能在设备本地对数据发送时间进行控制。考虑到智能家居设备业务采用 TCP 协议进行传输，而 TCP 报文段的发送需要 TCP ACK 报文进行驱动，因此可以通过下行 TCP ACK 的受控接入对上行 TCP 报文段的发送进行控制。

(3)不可控移动终端设备：与不可控智能家居设备类似。移动终端设备采用的接入机制通常会支持 QoS。为了保障语音、视频高优先级业务的传输时延，仅对低优先级业务进行受控接入处理。

路由器通过为不可控的终端设备建立缓存队列，将数据缓存至对应终端设备所划分的时隙进行发送，从而实现不可控终端设备的受控并行随机接入。对于不可控终端设备的受控接入示意如图 5-9 所示。

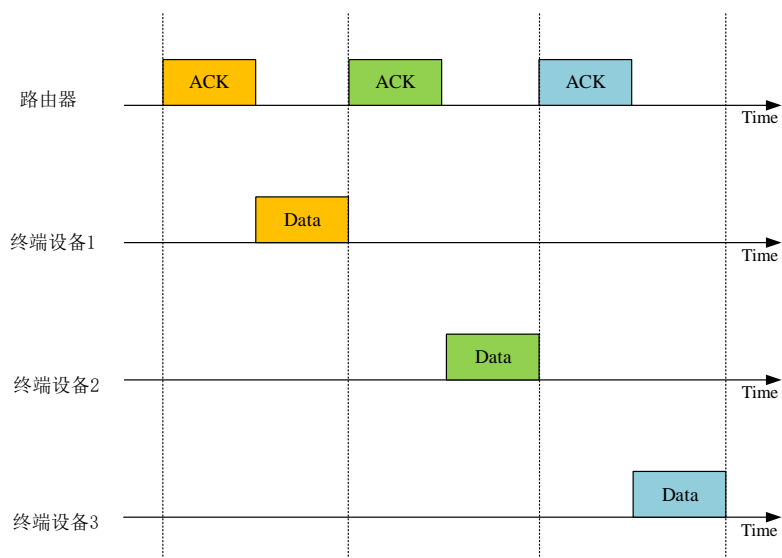


图 5-9 不可控终端设备的受控随机接入

#### 5.4.6 基于终端类型的流量预测

路由器需要根据设备之前的流量特性对设备接下来的流量特性进行预测。流量预测的结果直接影响到了分组结果。考虑到不同类型终端的流量特点不一样，基于终端类型以及过去的流量特性实现联合流量预测。

表 5-2 终端类型权值

编号	终端类型	参考权值
0	移动终端设备	220
1	未知类型智能家居设备	200
2	网络摄像头	150
3	智能音箱	100
4	冰箱	50
5	空调	40
6	扫地机器人	30
7	烟雾报警器	20
...	...	...

终端设备的流量特性使用权值表示，权值越高平均流量越大，序号为  $i$  的终端设备权值表示为

$$w^i(j) = \eta \times w_r^i(j) + (1 - \eta) \times w_p^i(j) \quad (5-25)$$

$w^i(j)$ 、 $w_r^i(j)$ 、 $w_p^i(j)$ 分别表示序号为  $i$  的终端设备在第  $j$  个周期的联合权值、设备类型权值以及流量权值。 $\eta$  表示为设备类型权值在联合权值中所占据的比例。 $w_r^i(j) \in [0, 255]$ ，如表 5-2 所示，理论上平均流量更高的终端设备具有更大的权值。 $w_p^i(j)$ 通过对统计到的终端上行平滑流量得出，通过如下公式计算

$$w_p^i(j) = \frac{SmoothFlow_u^i(j)}{flow_{max}} \times 255 \quad (5-26)$$

$SmoothFlow_u^i(j)$ 由式(5-24)得到，代表了设备  $i$  在第  $j$  个周期计算得到的上行平滑流量情况。特别地，对于移动终端设备，根据低优先级业务的流量大小计算权值。

$$w_p^i(j) = \frac{SmoothFlow_{LowPri}^i(j)}{flow_{max}} \times 255 \quad (5-27)$$

$flow_{max}$ 是第  $j+1$  周期内的终端最大流量值。路由器侧已知所有终端在的物理层协商速率  $v_{phy}$ ，可以通过如下式子计算得到  $flow_{max}$

$$flow_{max} = \max(\{v_{phy}^i, i \in \Omega\}) \times T_p \quad (5-28)$$

其中  $v_{phy}^i$ 表示序号为  $i$  的终端设备当前的物理发送速率， $\Omega$  为所有终端设备序号的集合， $T_p$ 为周期大小。

## 5.5 对部分终端透明的流量均衡受控并行随机接入方案软件实现

### 5.5.1 开发平台与架构

路由器和可控终端侧的软件实现都是基于 Wi-Fi 网卡驱动。路由器和终端设备所选取的网卡设备及其操作系统平台略有不同。路由器的网卡驱动运行于 Linux 操作系统，而终端设备的网卡驱动运行于 LiteOS 操作系统。

Linux 操作系统是一个多用途的操作系统，支持大量的硬件平台和软件应用。而 LiteOS 是一个轻量级的操作系统，专门针对物联网设备开发，具有低功耗、低内存占用的特点。虽然这两种操作系统在实现和用途上各有区别，但基于两者的 Wi-Fi 网卡驱动实现架构是一致的。网卡驱动实现被分成了 UMAC 层以及 LMAC 层两个部分。UMAC 和 LMAC 分别实现了不同的功能模块。路由器侧和终端侧实现模块与协议栈的层次关系如图 5-10 所示。出于实现需求的考虑，本文提出方案的软件实现在 UMAC 层。从协议栈实现的角度上看，该部分位于 TCP/IP 协议栈和 802.11 MAC 层

之间。

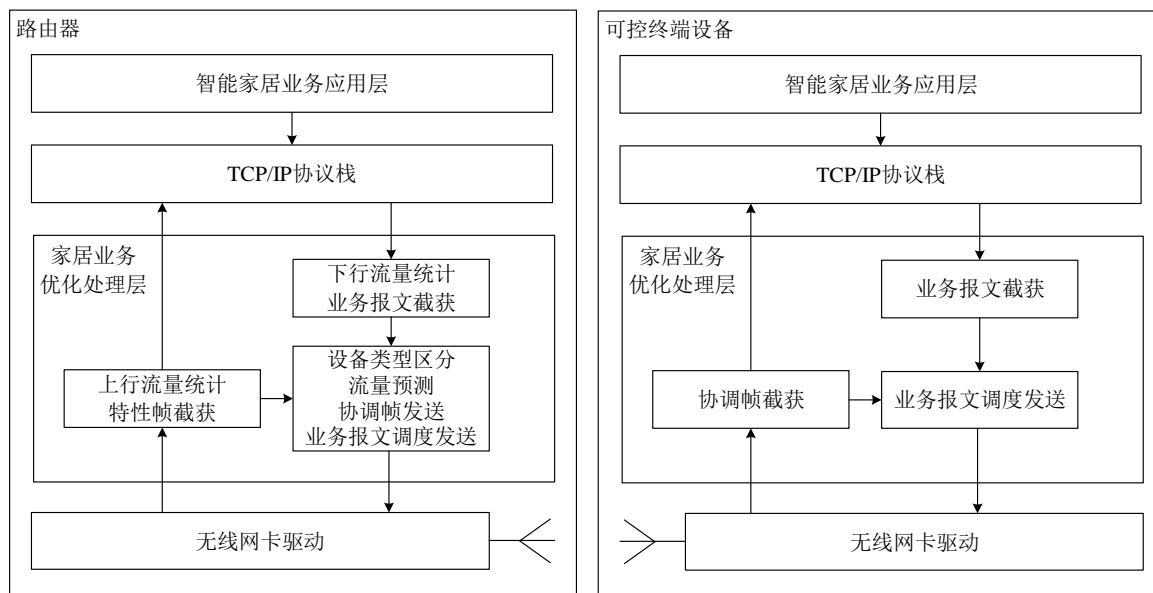


图 5-10 路由器及终端侧优化处理模块与协议栈关系

## 5.5.2 路由器侧设计实现

业务优化处理层是在已有的网卡驱动上实现的，它包含于 UMAC 的软件实现中。TCP/IP 协议栈的数据报文发送将调用 UMAC 的入口函数 `hmac_vap_start_xmit`，这也是最接近 TCP/IP 协议栈的驱动函数接口。而消息的接收处理是在无线数据包转为有线数据包的处理函数 `hmac_send_data_to_bridge` 中实现的。除开这两个与已有驱动的接口函数，优化处理层的实现还需要实现通过初始化和去初始化的实现，这些实现是放在 UMAC 模块的加载和卸载中的。

按照实现功能位置对软件实现进行区分，可以分为发送线程、定时器、报文的发送和接收处理。其中报文的发送和接收处理是在已有的 UMAC 驱动接口中实现。

在 5.4.4 小节中提到的流量统计与设备类型区分是在报文的发送、接收函数进行实现的。首先，在接收函数中记录有发送特性帧的设备作为可控终端设备，同时也统计上行流量情况。之后在发送函数中统计不同设备的下行流量情况，之后再按照 5.2.2 小节中的方法区分出移动终端设备以及不可控的智能家居设备。上述的终端类型信息都通过相应的接口写入文件系统中，方便后续读出进行处理。

流量预测、动态时隙分配以及流量均衡分组都是由定时器触发信号量给发送线



程开始进行处理的。线程在被定时器触发唤醒时将首先根据 5.4.6 小节中的方法对网络中终端设备在下一个周期中的流量情况进行预测，之后按照网络中设备个数的情况按照 5.4.3 小节中提出的方法对时隙个数和周期大小进行调整，最后按照预测的流量情况以及当前时隙个数进行分组。

不可控终端设备的受控并行随机接入是在线程中进行实现的。在发送函数中每截获到一个下行的 TCP 业务报文就会对线程进行触发，线程中判断到有不可控终端设备的 TCPACK 时，按照提前分配好的时隙在指定的时隙发送，从而实现不可控终端设备的受控接入。

整体软件实现框架如图 5-11 所示。

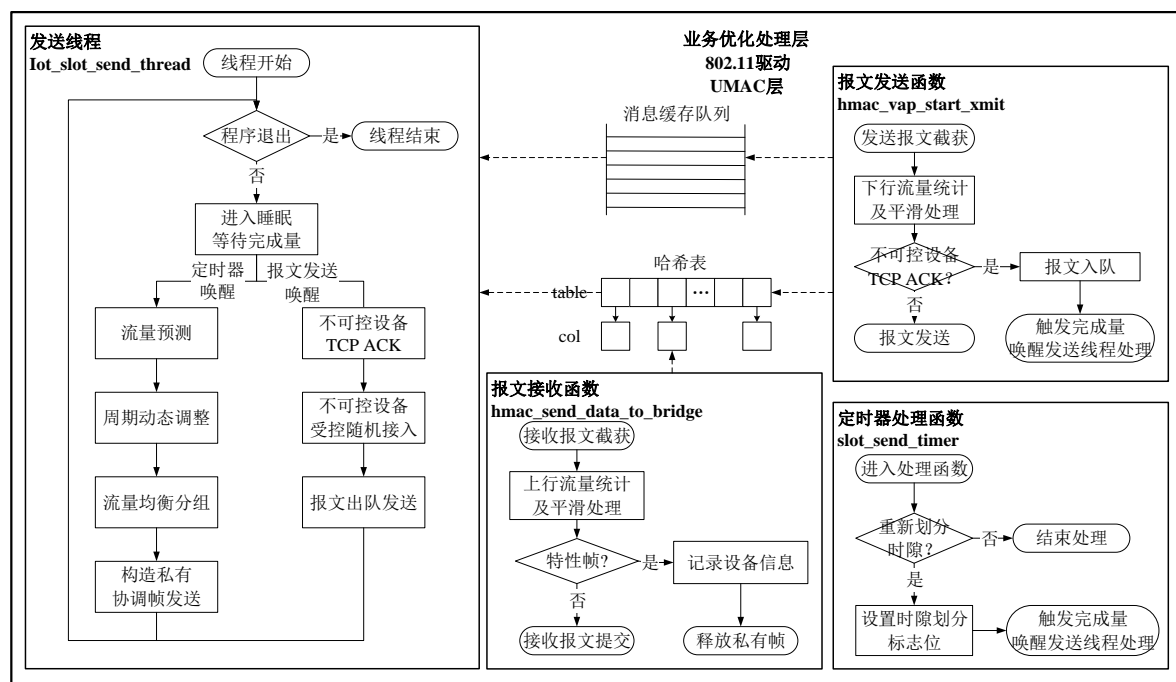


图 5-11 路由器侧软件实现框架

## 5.5.3 可控终端侧设计实现

由于可控终端侧软件实现的操作系统平台不一样，所以在 UMAC 层的接口也有所区别。在 UMAC 层驱动与 TCP/IP 协议栈最接近的接口为 ieee\_80211\_subif\_start\_xmit。而对无线数据包的接收处理是在 ieee\_80211\_deliver\_skb 函数中实现。与路由器侧的实现相比，可控终端侧的实现更为简单，这是因为可控终端设备需要实现的功能模块更少。其软件实现框架如图 5-12 所示。

按照功能实现位置进行模块划分,可以分为发送线程、定时器处理、报文发送和报文接收处理函数四个部分。在接收截获函数中对接收到的协调帧进行解析处理,得到设备当前分组和对应的发送时隙。在发送函数中对所有待发送的上行报文进行缓存入队。定时器定期触发发送线程执行判断当前时间是否可以发送缓存入队的报文。在可以发送时发送数据报文,由此实现了可控设备的受控并行随机接入。设备与路由器的同步利用了已有的同步机制。

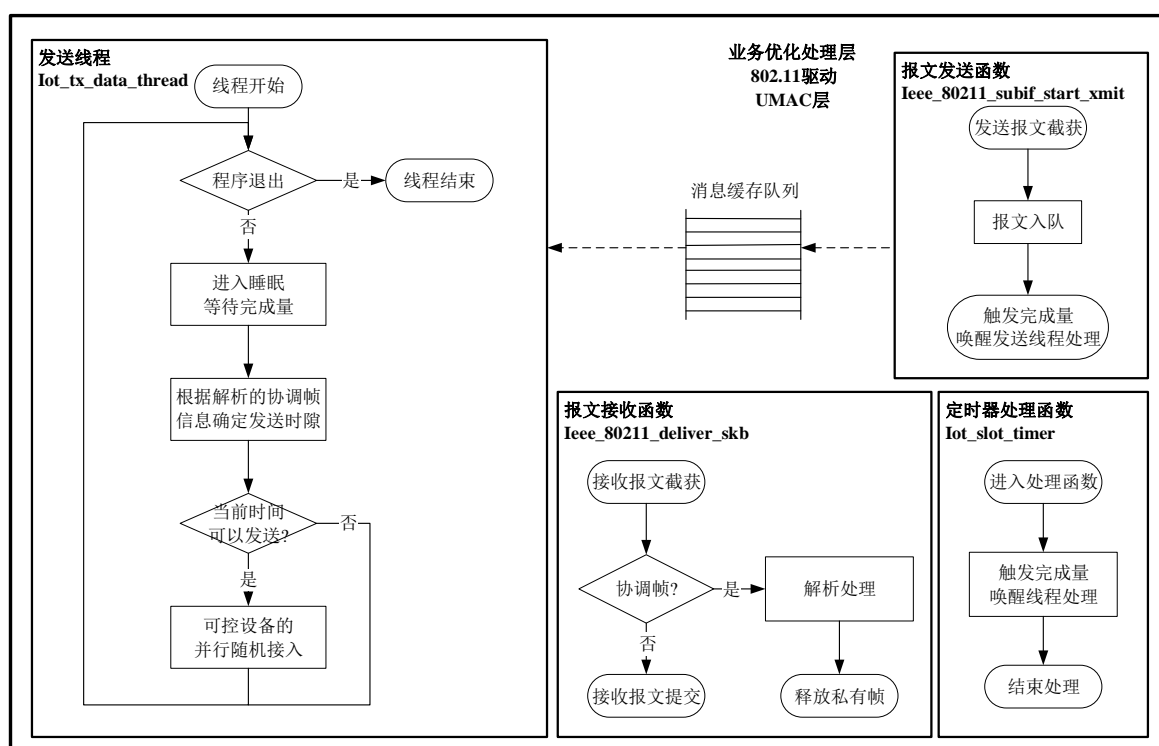


图 5-12 可控终端软件实现方案

## 5.6 实验结果与分析

本节对提出的对部分终端透明的流量均衡受控并行随机接入方案进行实际测试结果的分析。除了与传统 CSMA/CA 接入机制进行对比外,还引入了实验室已实现的以 STA 作为协调者的对路由器透明方案实际测试结果作为对比。

### 5.6.1 实验测试环境概述

依托国内知名通信公司的平台支持,在实际的场景中对提出的接入优化方案进行了测试。路由器、智能家居 IoT 设备以及移动终端设备均由公司提供。在单个无线

路由器场景下，分别测试了网络规模从 5 到 60 个终端设备时系统吞吐以及平均发送时延的变化情况。网络拓扑结构如图 5-13 所示。

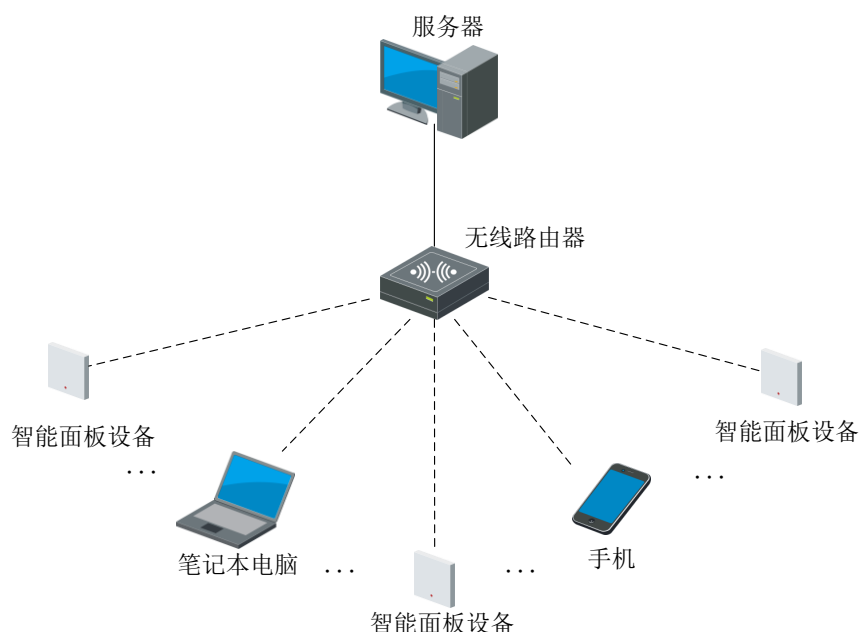


图 5-13 测试场景网络拓扑结构

智能家居 IoT 设备使用的智能面板，移动终端设备包括了手机以及笔记本电脑，除此之外网络中存在一台台式机作为服务器通过网线连接路由器 LAN 口，其他设备均通过无线方式连接无线路由器。设备间无隐藏终端问题，并关闭 RTS/CTS 机制，使用 CSMA/CA 机制的基本模式竞争信道资源。实验测试环境中的部分设备如图 5-14 所示。



图 5-14 IoT Wi-Fi 部分测试设备

红色方框选中的部分为智能面板设备，蓝色方框选中的部分为使用到的移动终

端设备，黄色方框选中的部分为路由器设备。为了对路由器透明的方案进行对比，网络中 20%的站点为不可控的移动终端设备，20%的站点为不可控的智能家居 IoT 设备，其余设备为可控的智能家居 IoT 设备。同时设置 10%的设备处于大流量传输状态，90%的设备处于小流量传输状态，均使用 TCP 业务向服务器打流测试。分别测试对部分终端透明方案以及对路由器透明方案的性能。

## 5.6.2 实验测试结果分析

### 1. 系统吞吐率分析

实际测试的系统吞吐率随网络规模的变化趋势如图 5-15 所示。

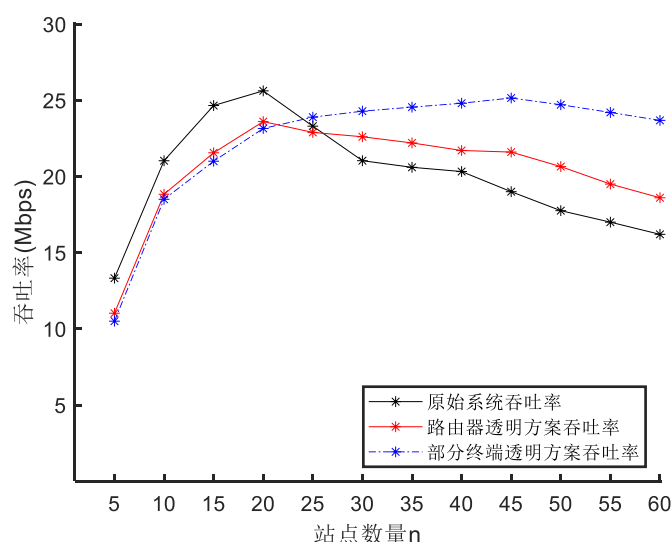


图 5-15 不同接入机制的吞吐率随站点数量的变化

在网络中站点数量少于 25 时，对部分终端透明方案的吞吐率要低于原始系统吞吐以及对路由器透明的方案。这是因为对部分终端透明方案为所有站点划分了时隙接入，在设备数量较少时进行时隙划分反而降低了信道利用率。对路由器透明方案以及原始系统分别有部分或是全部站点使用 CSMA/CA 机制进行随机接入的方法，信道利用率相对来说更高，因此系统吞吐也比对部分终端透明方案的高一些。

而随着设备数量增多至 25 到 45 区间内时，对部分终端透明方案的吞吐率持续上升，同时原始系统以及对路由器透明方案的吞吐表现都有不同程度地降低。这是因为原始系统以及对路由器透明方案中通过 CSMA/CA 进行随机接入的设备数量增多，站点竞争信道成功率下降。

需要注意的是,当设备数量大于 45 时,对路由器透明方案开始急剧下降,一方面是因为大量第三方设备随机接入使得站点竞争信道成功率下降,另一方面是因为方案引入大量额外网络资源。而对部分终端透明的方案能够有效避免这些问题,吞吐性能表现也只是略微下降。

## 2. 发送时延分析

测试出的时延性能随站点数量的变化情况如图 5-16 所示。

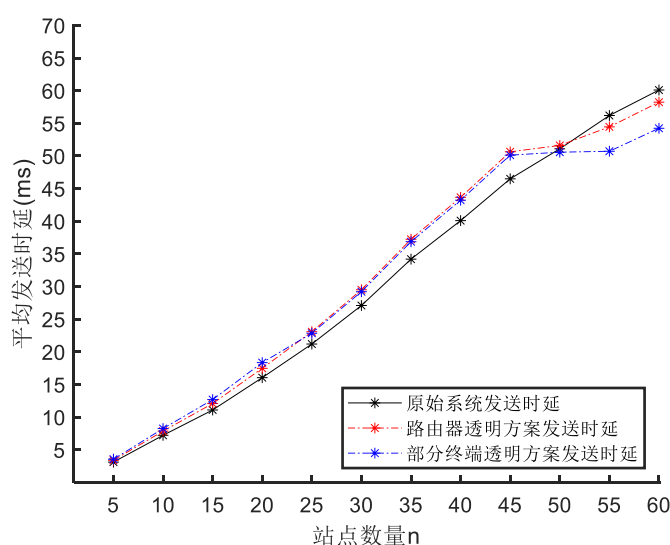


图 5-16 不同接入机制的发送时延随站点数量的变化

在站点数量较少时,对路由器透明方案以及对部分终端透明方案的划分时隙接入增加了站点的平均发送时延,导致两种优化方案的时延比原始系统要高。其中对部分终端透明方案为更多的设备划分了时隙接入,因此时延相对来说会更高一些。

随着设备数量的增多,两种优化方案通过时隙划分接入能够降低冲突概率,使得站点退避时间减少,从而减少了平均发送时延。其中对部分终端透明方案因为对网络中所有设备进行了受控接入,避免了部分不可控设备随机接入引起的信道接入成功率下降,进一步优化了时延性能。

综上所述,在网络中设备数量较多时,优化接入机制都能够有效提升智能家居场景中 IoT WiFi 系统的时延和吞吐性能。其中本文提出的对部分终端透明的流量均衡受控并行接入机制能够较好地应对网络中存在不可控流量随机接入的情况。在所测试的网络规模范围内,时延性能相较原始 CSMA/CA 接入机制降低了 9.75%,吞吐性

能相较原始 CSMA/CA 接入机制提升了 42.35%。

## 5.7 本章小结

本章首先对智能家居场景下 IoT Wi-Fi 接入的特征及其需求进行了分析，之后对在该场景下使用 CSMA/CA 接入机制的问题进行了分析。大量不同类型设备同时接入使得信道竞争成功率大大降低，引起了站点接入信道不公平、不及时以及系统吞吐下降等问题。之后在 CSMA/CA 的基础上，结合 RAW 机制分时接入的特性，引出了受控并行随机接入机制。结合实际场景情况，提出了对部分终端透明的流量均衡受控并行随机接入方案。根据理论分析以及实际测试结果验证了提出的接入机制实现方案能够在较大网络规模下有效提高 IoT Wi-Fi 系统吞吐以及站点发送时延表现，时延性能相较原始 CSMA/CA 基础接入机制降低了 9.75%，吞吐性能相较原始 CSMA/CA 基础接入机制提升了 42.35%。

同时所提出机制只在网络具有一定规模时才对系统吞吐和时延性能具有优化作用。在网络规模较小时，接入机制的引入在提升了系统复杂度的同时反而对性能产生了负面影响。

## 6 总结与展望

### 6.1 本文工作总结

家居场景中的智能设备多通过无线通信技术接入互联网。BLE 具有低功耗、操作性好等特点，而 Wi-Fi 在系统吞吐率、成本方面有优势。这两种技术按照实际需求也被应用在不同的智能家居场景中。但随着智能家居产业的发展，智能家居中终端设备数量和种类急剧增多，导致了这些技术在实际应用时的性能表现下降，影响了用户体验。

为了保证上述无线通信技术的性能表现，本文分别对两种技术在实际应用时性能下降的原因进行了分析，提出了解决方案，并通过软件实现和测试验证了方案的可行性。

针对智能家居场景下的基于 BLE MESH 组网的点对多点通信的实时性问题，做出了如下工作：

(1) 指出了当前场景对点对多点通信的实时性和可靠性需求。在 BLE MESH 网络节点接收功能间隙性休眠的前提下，分析了现有接入机制在应用于一对多的群组可靠消息及其确认消息时在时延和可靠上表现不足的原因。针对当前机制的不足之处提出了新的接入机制方案。

(2) 提出了基于 TDMA 思想的群组可靠消息接入机制。在网络中设备时钟同步的前提下，该机制将网络中的不同设备划分至不同时隙进行信道接入。通过分时有序接入以及动态时隙分配能够有效避免设备休眠引起的丢包以及设备间碰撞问题，从而起到增加消息传输可靠性以及降低消息传输时延的作用。本文还在 BLE MESH 协议栈缺少同步服务的情况下，提出了基于广播消息的低成本同步方法。

(3) 针对不同的群组可靠消息接入机制，基于离散时间马尔科夫模型对不同机制在确定可靠性下的时延性能作出了分析，并进行了数值仿真对比。证实了在相同可靠性保证的前提下，基于 TDMA 的群组可靠消息接入机制能够有效降低消息传输时延，提升实时性表现。之后在此基础上提出了具体的实现方案，并在开源 BLE MESH 协议栈上进行了软件实现以及测试。测试数据表明，在相同的可靠性要求下，

优化接入机制相对于现有机制在不同网络规模中都能够有效提升时延性能表现。

针对智能家居 IoT Wi-Fi 场景中大量终端设备通过单 AP 密集接入引起的吞吐和时延性能下降问题，做出了如下工作：

(1) 指出了智能家居 IoT Wi-Fi 场景业务特点以及需求，并分析了该场景使用 DCF 接入机制吞吐率下降的原因。针对 DCF 机制在 WLAN 中接入终端数量较多时的不足之处，引出了流量均衡的受控并行随机接入机制。机制一方面基于流量特性实现分组接入能够充分利用信道资源，另一方面通过减少直接接入信道的设备数量降低碰撞概率，进而到达了提升系统吞吐率的目的。

(2) 针对流量均衡的受控并行随机接入机制，使用基于 Bianchi 模型改进的马尔科夫模型进行了理论分析，并通过理论数值分析对比，得出了改进机制在网络规模较大时相较于传统 CSMA/CA 接入机制更有优势的结论。结合实际应用场景特点，分析了在路由器侧实现集中式控制接入的优势并在路由器侧基于 TCP 业务特点提出了对智能家居场景中背景业务流的受控接入方法，通过实际测试验证了方案能在较大规模的单 AP 接入网络下有效优化性能。

## 6.2 不足与展望

本文针对实际智能家居场景下无线通信技术的应用问题进行了理论分析，并根据实际应用场景提出了优化方案，在实际测试中验证取得了一定的效果。但因为实际应用场景复杂且多样，本人能力有限，本文工作还存在一些不足之处需要加以改进和进一步研究。

(1) 对于 BLE MESH 接入机制的性能分析主要针对碰撞概率和消息传输的理论最大时延展开，本人分析思路相对简单，并没有考虑信道扫描对数据传输的影响，后续希望能够从更多的指标对接入机制的性能进行分析。

(2) 本文提出的基于 TDMA 的优化接入机制需要引入一定的资源消耗。例如，在某些极端网络规模下，网关需要耗费大量内存保存不同群组的成员信息；位图引入的网络负载大小甚至会超过群组可靠消息本身。同时，低功耗节点设备实现 TDMA 接入机制依靠高精度定时器会增加一部分能耗。因此需要考虑在较大网络规模的情



况下对机制作出优化。

(3) 本文提出的在 BLE MESH 网络中基于 TDMA 的优化接入机制并没有考虑到网络中存在多个网关的情况，还可以对多网关场景进行进一步建模分析以及改进。同时可以考虑在某些特殊的工业场景中进行应用。

(4) 本文提出的在 IoT Wi-Fi 网络中受控并行随机接入机制在设备数量较小时反而会对系统的时延产生较大影响，还需要考虑对现有机制做进一步优化以适应网络规模小的场景。

(5) 受到实际测试平台的限制，针对 BLE MESH 以及 IoT Wi-Fi 网络提出的优化接入机制还需要补充更大网络规模以及不同场景的测试来验证机制有效性。

## 致 谢

回望我的研究生生涯，它开始于我的导师——吴伟民老师的一通电话，后来第一次去了国家光电实验室，第一次见到了吴老师以及实验室的师兄师姐，第一次参与到实际的工程项目中，第一次完成一个实际的工程项目…。在我即将完成我的研究生生涯之际，我心怀感激之情写下这篇致谢。

首先，我要感谢指导过我的老师们。感谢我的导师吴伟民老师，吴老师在指导我进行理论学习的同时也让我有了参与大量实践项目的机会。项目在进行的过程并不总是一帆风顺的，吴老师却总是能够为我指引方向，有时稍显严厉，有时也会激励我，也正是这些鞭策和鼓励支撑我走到今天。感谢刘应状老师，刘老师在我的毕业论文写作上提出了许多宝贵意见。感谢喻洪涛老师，喻老师在实践项目上的逻辑思辨使我受益匪浅。

然后，我要感谢同门的兄弟姐妹们。记得第一次在学校见到了刘寺杰学长，他的专注给我留下了深刻印象。后来，我第一次参与实践项目，在郭志斌学长和饶林虎学长的带领下，我们实现了 Windows 蓝牙驱动的基础功能。我惊叹于学长们的知识储备、工程经验，这在一定程度上也激励了我努力学习和积极实践。现在毕业论文又回到了蓝牙 MESH 的内容，让我不禁感叹时间之快。在此我对学习、生活上帮助和陪伴过我的其他同门也一一提出感谢，有你们的研究生生活将成为我最宝贵的记忆之一。

我还要感谢我的父母和家人。你们无条件地支持着我，让我能够不断前行去找到自己的路。

感谢我的女朋友周冀楚。她从走进实验室的那一天起就给我的研究生生活注入了一抹亮色。感谢她在学习、项目还有生活上的陪伴，让我能够顺利度过最后的研究生生活。

最后，衷心感谢所有参加论文评审、答辩的专家和老师！衷心感谢所有帮助和支持过我的人！

## 参考文献

- [1] 张洪雷,李春华,高建.无线通信技术在智能家居中的发展与研究[J].数字技术与应用,2020,38(09):40-42.DOI:10.19695/j.cnki.cn12-1369.2020.09.16.
- [2] 刘廷.基于有线、无线通信技术的物联网智能家居系统[J].长江信息通信,2022,35(05):186-188.
- [3] S. S. I. Samuel, "A review of connectivity challenges in IoT-smart home," 2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC), Muscat, Oman, 2016, pp. 1-4, doi: 10.1109/ICBDSC.2016.7460395.
- [4] R. Rondón, A. Mahmood, S. Grimaldi and M. Gidlund, "Understanding the Performance of Bluetooth Mesh: Reliability, Delay, and Scalability Analysis," in IEEE Internet of Things Journal, vol. 7, no. 3, pp. 2089-2101, March 2020, doi: 10.1109/JIOT.2019.2960248.
- [5] 陈良银,王金磊,张靖宇,王强,刘燕,殷锋,罗谦.低占空比 WSN 中一种节点休眠调度算法[J].软件学报,2014,25(03):631-641.DOI:10.13328/j.cnki.jos.004401.
- [6] 廉小亲,安飒,王俐伟,文欣雨,周栋.智能家居发展及关键技术综述[J].测控技术,2018,37(11):1-4+15.DOI:10.19708/j.ckjs.2018.11.001.
- [7] Wei Shi, Keqiu Li, Jiafu Wan. "Smart Home Communication Technologies: A Comprehensive Review." IEEE Communications Surveys & Tutorials, 2018.
- [8] Hui Li, Xiaotong Wang, Tingzhe Ning, Limin Sun, Hanlin Zhang. "A lightweight authentication protocol for smart home in internet of things." Journal of Ambient Intelligence and Humanized Computing, Vol. 9, No. 4, August 2018.
- [9] Meysam Sadeghi, Qinghua Li, Mingsheng Wang, Jianying Zhou. "Secure and Privacy-Preserving Data Collection Protocols in Smart Home Environments." IEEE Internet of Things Journal, Vol. 5, No. 4, August 2018.
- [10] Chen-Fu Chiang, Kuo-Qin Yan, & Yu-Shan Chen. "A Smart Home Energy Management System Using IoT and MQTT Protocol." IEEE Access, Vol. 6, June 2018.
- [11] 刘洋.Wibree 前景正在超越 Bluetooth[J].电子设计技术,2007(10):130+133.
- [12] 林满山.浅谈蓝牙技术的发展现状和前景[J].黑龙江科技信息,2016(14):165.

- [13] 杨广贺. 蓝牙 5 的性能提升要点[J]. 安全与电磁兼容, 2017(06):25+54.
- [14] K. Mikhaylov and J. Tervonen, "Multihop data transfer service for Bluetooth Low Energy," 2013 13th International Conference on ITS Telecommunications (ITST), Tampere, Finland, 2013, pp. 319-324, doi: 10.1109/ITST.2013.6685566.
- [15] H.-S. Kim, J. Lee and J. W. Jang, "BLEmesh: A Wireless Mesh Network Protocol for Bluetooth Low Energy Devices", 2015 3rd International Conference on Future Internet of Things and Cloud, pp. 558-563, aug 2015.
- [16] L. Leonardi, G. Patti, and L. Lo Bello, "Multi-hop real-time communications over Bluetooth low energy industrial wireless mesh networks," IEEE Access, vol. 6, pp. 26505–26519, 2018.
- [17] J. León, A. Dueñas, Y. Iano, C. A. Makluf, and G. Kemper, "A Bluetooth low energy mesh network auto-configuring proactive source routing protocol," in Proc. IEEE Int. Conf. Consum. Electron. (ICCE), Las Vegas, NV, USA, 2017, pp. 348–349.
- [18] Yining Zhang, Junfeng Zhang, and Lifeng Qi. "A Bluetooth Mesh Network for Real-Time Group Monitoring and Management." Wireless Communications and Mobile Computing, Vol. 2019, Article ID 2323513.
- [19] Vivek Patwardhan and Vijay Gadepally. "Bluetooth Mesh-Based Indoor Localization: A Comprehensive Survey." IEEE Access, Vol. 7, 2019.
- [20] Abdulkadir Gokceli and Sibel Karagoz Kantarci. "Simulation and Performance Analysis of Bluetooth Low Energy Mesh Network for Healthcare Monitoring System." Journal of Medical Systems, Vol. 42, No. 9, August 2018.
- [21] S.-D. Wang and K.-J. Chiang, "BLE tree networks for sensor devices in Internet of Things," in Proc. IEEE 15th Int. Conf. Depend. Auton. Secure Comput. 15th Int. Conf. Pervasive Intell. Comput. 3rd Int. Conf. Big Data Intell. Comput. Cyber Sci. Technol. Congr. (DASC/PiCom/DataCom/CyberSciTech), Orlando, FL, USA, 2017, pp. 1304–1309.
- [22] P. Pierleoni, A. Gentili, M. Mercuri, A. Belli, R. Garelli and L. Palma, "Performance Improvement on Reception Confirmation Messages in Bluetooth Mesh Networks," in IEEE Internet of Things Journal, vol. 9, no. 3, pp. 2056-2070, 1 Feb.1, 2022, doi: 10.1109/JIOT.2021.3090656.

- [23] Sharma J. Wi-Fi 技术的演进[J]. 电子产品世界, 2020, 27(09): 84-87.
- [24] 李项军, 张水利, 赵海波. Wi-Fi 6 技术介绍及在能源互联网中的应用[J]. 延安大学学报:自然科学版, 2021, 40(3): 5.
- [25] Song N O, Kwak B J, Song J, et al. Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease backoff algorithm[C]. Vehicular Technology Conference, 2003.
- [26] Wang C, Bo L, Li L. A new collision resolution mechanism to enhance the performance of IEEE 802.11 DCF[J]. IEEE Transactions on Vehicular Technology, 2004, 53(4): 1235-1246.
- [27] Nasir Q, Albalt M. History Based Adaptive Backoff (HBAB) IEEE 802.11 MAC Protocol[C]. Communication Networks & Services Research Conference, 2008.
- [28] 张睿. 基于 AI 的 Wi-Fi 系统接入控制研究[D]. 西南交通大学, 2019.
- [29] 王春剑. IEEE 802.11ax 标准 MAC 层多用户随机接入技术研究[D]. 西南交通大学, 2020.
- [30] Mukhopadhyay A, Mehta N B, Srinivasan V. Acknowledgement-aware MPR MAC protocol for distributed WLANs: Design and analysis[C]. 2012 IEEE Global Communications Conference (GLOBECOM), 2013.
- [31] 胡智群. 多层次网络中高效 WLAN 的性能分析与优化研究[D]. 北京邮电大学, 2018.
- [32] 魏柏舟. 网络拓扑结构类型简论[J]. 才智, 2012(25): 54.
- [33] Steve Methley. 无线 Mesh 网络基础[M]. 西安: 西安交通大学出版社, 2012.
- [34] 王莹. 蓝牙 5 与 Mesh: 助力终端节点遨游 IoT 蓝海 [J]. 电子产品世界, 2016, 23(10): 81-82.
- [35] 汪亚琼. 面向 BLE MESH 的机会路由协议设计与实现 [D]. 东南大学, 2020. DOI:10.27014/d.cnki.gdnau.2020.004177.
- [36] 吴彬, 何莉. 基于 ZigBee 继电器式单火线智能开关的研发和设计 [J]. 电工技术, 2021, No.542(08): 28-30+98. DOI:10.19768/j.cnki.dgjs.2021.08.011.
- [37] 许毅. 无线传感器网络原理及方法 [M]. 北京: 清华大学出版社, 2012: 126

—130.

- [38] 姜绍君,王颖.基于 CSMA/CA 协议的隐藏终端问题的改进[J].计算机技术与发展,2015,25(01):33-36.
- [39] 邓皓文. 基于 EDCA 机制的无线局域网 QoS 性能研究[D].西南科技大学,2020
- [40] Mehmet Ali Erturk, Luca Vollero, Muhammed Ali Aydin. Optimal joint load balancing and EDCA configuration of IEEE 802.11 wireless hotspots[J]. International Journal of Communication Systems, 2018, 31(2).
- [41] Bhatia, Ashutosh and Ramesh C. Hansdah. "TRM-MAC: A TDMA-based reliable multicast MAC protocol for WSNs with flexibility to trade-off between latency and reliability." Comput. Networks 104 (2016): 79-93.
- [42] F. Gebali, Analysis of Computer and Communication Networks. New York, NY, USA: Springer, 2008.
- [43] A. Bideaux, B. Zimmermann, S. Hey, and W. Stork, "Synchronization in wireless biomedical-sensor networks with Bluetooth low energy," Current Directions Biomed. Eng., vol. 1, no. 1, pp. 73–76, 2015.
- [44] F. J. Dian, A. Yousefi, and K. Somaratne, "Performance evaluation of time synchronization using current consumption pattern of BLE devices," in Proc. IEEE 8th Annu. Comput. Commun. Workshop Conf. (CCWC), Las Vegas, NV, USA, 2018, pp. 906–910.
- [45] K. Somaratne, F. J. Dian, and A. Yousefi, "Accuracy analysis of time synchronization using current consumption pattern of BLE devices," in Proc. IEEE 8th Annu. Comput. Commun. Workshop Conf. (CCWC), Las Vegas, NV, USA, 2018, pp. 841–844.
- [46] F. Asgarian and K. Najafi, "BlueSync: Time Synchronization in Bluetooth Low Energy With Energy-Efficient Calculations," in IEEE Internet of Things Journal, vol. 9, no. 11, pp. 8633-8645, 1 June1, 2022, doi: 10.1109/JIOT.2021.3116921.
- [47] P. Su, "Delay measurement time synchronization for wireless sensor networks," IEEE Micro, June 2002, vol. 122, pp. 12-24.
- [48] S. Chen, J. Tang, H. Wang, Y. Dong, and J. Su, "A Practical Time Synchronization Method Based on Message Exchange in Wireless Sensor Networks," IEEE Sensors Journal, vol. 18, no. 11, pp. 4442-4451, Jun. 2018.

- [49] Z. Zhang, Z. Zhang, and C. Li, "A Flooding-Based Synchronization Protocol for Wireless Sensor Networks," IEEE Internet of Things Journal, vol. 6, no. 3, pp. 5027-5037, Jun. 2019.
- [50] B. Sethi and A. Kumar, "A Multi-Reference-Time Synchronization Scheme for Wireless Sensor Networks," IEEE Transactions on Industrial Electronics, vol. 65, no. 3, pp. 2312-2321, Mar. 2018.
- [51] Bianchi G. Performance analysis,of the IEEE 802.11 distributed coordination function[J]. Ieee Journal on Selected Areas in Communications, 2000, 18(3): 535-547.