北京郵電大學

本 科 毕 业 设 计（论文）

题目：认知无线电系统组网研究

姓　　名　李孟辉

学　　院　信息与通信工程学院

专　　业　通信工程

班　　级　2014211118

学　　号　2014210506

班内序号　04

指导教师　郭文彬

2018 年 5 月

北 京 邮 电 大 学

本科毕业设计（论文）任务书

学院	信息与通信工程学院	专业	通信工程	班级	2014211118
学生姓名	李孟辉	学号	2014210506	班内序号	04
指导教师姓名	郭文彬	所在单位	信息与通信工程学院	职称	教授
设计(论文)题目	认知无线电系统组网研究				
	Research on cognitive radio adhoc networking technology				
题目分类	工程实践类☐ 研究设计类☑ 理论分析类☐				
题目来源	题目是否来源于科研项目 是☑ 否☐				

主要任务及目标：

- 认知无线电通过感知频谱环境，自适应调整通信的方式，达到有效利用空闲频谱资源的目的，从而缓解无线资源紧张的矛盾。由于认知用户是非授权用户，如何实现认知多用户之间的组网问题是认知自组网的关键问题。本课题研究认知自组网的组网策略及协议设计，给出相关性能分析或仿真。

主要内容：

文献阅读与课题综述报告，通过文献检索与阅读，给出认知无线自组网中多用户组网的研究现状分析报告。

针对认知自组网中多用户组网建模，研究不同场景下多用户组网的策略，给出理论或仿真分析。

主要参考文献：

- 洪一诺，北京邮电大学硕士论文，2017.
- 吴锐，北京邮电大学硕士论文， 2016.
- 王宇昆，北京邮电大学硕士论文，2018

进度安排：

- 2017 年 12 月-2018 年 3 月，文献阅读与综述报告完成；
- 2018 年 3 月-2018 年 5 月，完成认知无线自组网建模与仿真分析。
- 2018 年 5 月-2018 年 6 月，完成论文撰写

指导教师签字		日期	年 月 日

认知无线电系统组网研究

摘　要

随着无线频谱资源的不断使用，为了满足人类社会对于通信的需求，认知无线电组网技术应运而生，成为了缓解频谱压力的一大研究热点方向。传统的认知网络使用中心控制结构，依靠控制中心，例如基站，基站可以辅助认知无线电设备发现临近的用户与接入节点，不断的更新邻近用户的信息，在本地网中以无线自组网的方式建立与接收节点的通信路径，能够通过基站建立与其它本地网接收节点的单点与多点通信。但是这种方式受制于控制中心地理位置，不符合日后移动认知网络的需求，如果采用固定信道作为控制信道则会浪费大量频谱资源这是与认知无线电的原则相违背的。

为了满足非固定式的认知自组网的建立需求，本文研究了与组网线管的以下内容：信道交汇，对授权用户与认知用户进行建模，实现环境感知与基于 CGB 算法的自动跳频完成相邻节点交汇；拓扑发现: 研究了不同指标下的两种拓扑发现方法，给出了相关协议与发现过程的代码仿真分析；路由选择与数据传输：对于不同的组网要求给出了两种路由选择与传输的数据类型，分析了各个模式下的网络性能。

关键词　认知无线电　自组网　拓扑发现　路由协议

Research on cognitive radio adhoc networking technology

ABSTRACT

This is ABSTRACT.
You can write more than one paragraph here.

KEY WORDS BUPT undergraduate thesis template example

目　录

第一章　绪论

1.1　选题背景及意义

无线电频谱是非再生资源并且是国家的战略资源，许多无线通信方式需要在政府划分的频带下工作，这些频段也被称为授权频段，但是随着无线通信技术的发展，以及也有像 700Mhz 的通信黄金频段分配给广电总局进行无线电视的建设，这种分配方式使用效率极为低下，不符合与时俱进的无线通信的需要。无线局域网，蓝牙，zigbee 等技术应用更为广泛导致了在 ISM 频段上也十分拥挤，这使得频带的整体使用率并不高而且在不同频段使用效率十分不平衡，频谱资源和卫星轨位需求愈加旺盛，需求增长与频谱的静态分配带来的频谱稀缺问题日益浮现，规避拥堵的非授权频段，而在使用效率低下的授权频段内进行通信成为了解决这一问题的可行方案。

认知无线电技术一经提出就被广泛讨论研究，认知无线电采用动态频谱分配策略（DSA），未授权用户或者辅助用户（SU）可以在授权频段内的自主寻找和使用未被授权用户（PU）占用的空闲的频谱资源，在时间和空间上同时增加频谱的使用效率。通过对周围环境的感知，调整自身工作最适当的频段与调制类型等参数，从而实现认知设备间的通信。

认知无线电网络即认知设备间的网络，是一种多信道多节点无线网络，具有优秀的频谱复用性能与巨大的覆盖面积，拥有与传统无线通信网络不同的特点：认知无线电网络可以与普通无线通信网络共存并且不会产生干扰，其独特的认知功能可以智能分配无线资源，使得认知设备进行正常通信的同时不干扰授权频段内的其他系统的通信；系统支持多信道通信且不再具有传统网络的控制中心（例如基站），整个认知网络将会处于不同系统同时工作的嘈杂频谱背景中，这便要求网络不再使用统一固定的信道从而避免了盲节点的产生，而应该随着周围环境的改变做出适应性变化，来保证信道可用的基础上满足通信的 QoS 需求，同时由于网络中每个认知设备所处的环境并不完全一致，若是想要达到最佳的通信效果应当舍弃控制中心进行集中管理，而是采用分布式的管理方式来配合移动终端状态快速变化的特点，控制中心不再以实体形式存在，而是用控制中心频率进行节点间信息的交换，每个认知设备都将具有完备的认知通信功能，可以同时作为管理节点与通信节点，这是目前的民用移动无线网络所不具备的特点。

得益于认知自组网优秀的认知能力，认知自组网在不同的频谱占用情况下都可以寻找到合适的通信频段，在多种系统同时工作的环境中，认知设备的认知能力可以自主选择信道并满足自身的 QoS 需求，并且可以节省人工配置通信参数的成本，对于在医院，商场等活动设备密度高的场所建设内网有着天然的优势，由于认知设备周期性进行频谱感知，使得整个认知网络具有优秀的重构性能，在当前信道由于主用户活动不再可用时不会发生长时间的链路断裂而是直接跳跃到其他信道进行通信，CR 设备可以根据无线环境动态编程，可以重构工作频率、调制方式、发射功率和通信协议等重要参数。同时

由于是分布式管理，网络结构不会因为管理节点的瘫痪而无法通信，这也使得认知网络的重构性能更为优秀。相比于传统通信网络，认知无线网络的多信道通信模式可以更好的适应当前的无线环境，并且可以满足某些特殊的通信需求，在军用、民用方向都有着不同的优势。

1.2 面临的问题与研究内容

认知无线电原则上是为了提高频谱的利用率，原则上讲不应该一直占用信道进行控制信息的传送。但是这也是大多交汇算法的形式，它需要协商某频率作为控制信道来让想要加入的用户可以直接在该信道上发布信息，这种方法要求我们必须使用某一个频带，而这与将会降低该频带上的频谱利用率，所以应当舍弃公共控制信道的想法，寻找一种不需要固定的公共控制信道就能完成网络配置的组网策略。

组网并非是两个节点之间的连接，比如蓝牙之类的主从设备配对，而应该是多节点之间互相通信，虽然双节点之间的信道交汇已经有很多策略，但是多点的组网策略与拓扑发现仍然还是初步发展，应当在节点交汇的基础上完成网络的构建。组网过程很重要的就是拓扑发现，此过程决定了网络的连通性，这也会是后期的路由协议设计以及数据传输的基础，起到了而在拓扑发现的过程中也是会存在网络主用户的干扰，这将会对信道可用率产生很大的影响，因此这也是多点组网的难点所在。

自组网不能再任何时候都进行主用户的检测同时维护整个网络拓扑，也不能在任何阶段都允许新节点的加入。虽然为了使得所有节点都尽可能同时加入某一个网络，可以延长整个拓扑建立的时间（周期数），但是组网的一个重要指标便是组建网络的时间消耗，为了整个网络的快速建立应当减少每个阶段的耗时，这两者的要求便会产生矛盾，如何平衡两者之间的需求来达到最佳的认知自组网络的性能，也是需要解决的问题。

论文的主要研究内容如下:

(1) 认知节点与主用户节点建模

(2) 研究节点交汇算法与拓扑发现

(3) 给出了两种信道中传输数据类型与相应的路由模式，分析不同情况下网络性能

1.3 认知无线电自组网发展现状

1.3.1 认知无线电网络架构

所谓组网，是指多节点之间根据某一协议完成互相之间信息的传输，形成一个具有完备的路由功能的通信整体。不同的网络架构拥有不同的组网算法与功能，认知无线电的网络架构主要有两种：控制中心型网络架构，分布式网络架构。

控制中心型网络架构即集中式网络架构，也是传统认知无线通信网络的典型架构，在这种架构中，如图所示，认知设备与主用户处于相邻环境中，也就是意味着会有管理

主用户的中心控制器，此控制器可同时对于通信范围内的认知设备进行管理，所有认知设备可以不具有完备的认知网络组件功能，由中心控制器负责无限资源的管理，需要组网时每个认知设备向控制中心申请空闲频谱资源，从而可以不需要认知设备提前完成相邻节点之间的信道交汇，只需要从控制中心获取自身相关的传输参数就可以快速完成与其他节点的交汇，在此基础上就可以完成组网。此种网络架构的负载压力主要由中心控制器承担，所以中心控制器的认知性能与处理速度直接决定了整个认知网络的性能，整个网络性能将会受制于控制中心的地理位置，通信范围等通信指标，并不适合野外军事，医用的环境。

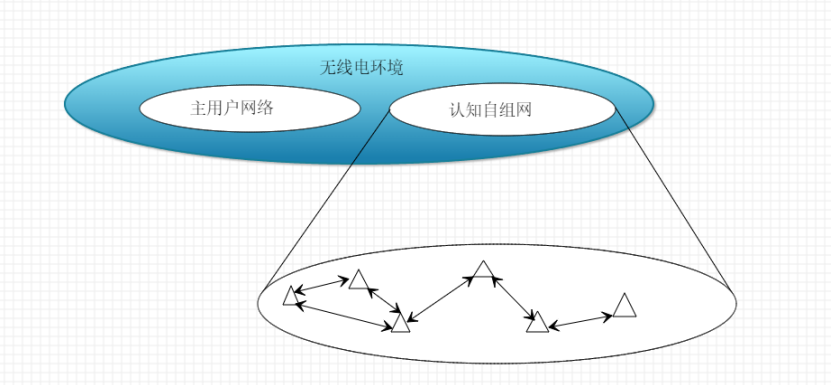


图 1-1 分布式网络架构

分布式网络架构，每个认知用户随机分布在不同位置，采用 Ad Hoc 模式，组成一个完整的认知网络。此种网络中每个认知设备地位平等，具有完备的认知自组网功能，同时负责路由器与终端的任务，每个终端自主学习周围环境并设计自身的通信参数。这种网络架构对于终端设备的要求提高，不过因为节点之间地位平等，不会因为某一个节点瘫痪造成整个网络的死亡，也正是由于分布式架构功能更为全面，整个组网过程的协议设计也相比更为困难。

本文针对第二种网络架构进行研究，网络中每个节点都会具备认知自组网的所有功能组件，相比集中式网络架构将会更为稳健与可靠。

1.3.2 感知学习过程

认知自组网系统的基础环节就是认知设备的感知学习过程，认知过程主要包括三个环节：频谱感知与分析，频谱决策，频谱共享。

- 频谱感知与分析：认知设备需要扫描通信范围内的所有可用频段以及主用户占用的频段，分析空闲频段的可用概率，信道质量等信息。
- 频谱决策：认知设备应当根据需要的服务类型定制数据传输速率，传输方式以及使用频段，此过程保证了最佳的点对点通信质量。

- 频谱共享：多个认知用户接入时都会进行频谱感知与数据传输，多个认知用户之间应当避免同时占用一个频段，频谱共享可以使得多认知用户分享频谱资源而不产生互相之间的干扰。

1.3.3 节点交会过程

认知自组网是一个多信道网络，每个无线电设备都会检测到大量频谱空穴，在需要组网时，每个认知设备会与邻居节点产生一条信道进行信息的交换，而节点间同时在某一个信道相遇则被称为节点交会。

1.3.4 拓扑发现过程

拓扑发现是在完成节点交汇的基础上获取整个网络结构的发现过程，网络研究界对捕获一个准确的网络拓扑结构图有极大兴趣，因为它有许多用途，是设计和评估新的协议以及服务的脆弱性分析是网络的基础。拓扑发现同样是认知自组网关键环节，是后续的路由设计的基础。

1.3.5 路由发现与数据传输过程

认知自组网是多跳网络，在拓扑发现完成之后需要进行路由的发现以完成完备的网络通信功能，由于区别于普通的集中式无线网络，认知无线电的路由发现需要考虑能耗，跳频时延等问题，需要不同的策略来获得最佳的对认知设备适应性以及累积时延。

1.4 认知自组网关键算法发展

1.4.1 频谱感知

频谱感知分为辅助频谱感知技术与独立频谱感知技术，辅助频谱感知技术不对认知设备自身做频谱感知要求，但是需要由中心管理设备检测可用频谱，认知设备得到授权之后即可以直接使用此频段进行交汇，但是由于辅助频谱感知技术属于集中式网络架构，不符合认知自组网理念，所以主要介绍独立频谱感知技术的发展状况。

- **主用户检测**：独立频谱感知技术中，每个 CR 用户通过不同的频谱检测机制检测主用户的活动，避开主用户的工作频段，目前主流的频谱感知算法都是基于主用户发射机的检测算法，其中包括：1）匹配滤波器算法 2）能量检测算法[1] 3）特征检测算法[2] 4）广义似然比检测，以上的算法可以满足不同的实际需求，所需求的信息环境也不同，并且不会产生 PU 与 SU 之间的通信，不过在阴影衰落较为严重的情境下，检测效果不佳；协作检测算法：协作检测需要多台 CR 设备交换各自的感知结果，弥补了单个节点由于 PU 用户信号的阴影衰落而无法被检测从而 CR 设备在此频段通信会对 PU 产生干扰的缺点，算法融合本地的感知信息可以使 SU

获取更为全面的主用户信息，不过此算法需要中继完成转发功能，在 CR 设备位置变化较快的时候检测难度也会很大；干扰温度检测：FCC 提出了一种干扰温度模型，并给出了干扰温度界，也就是 PU 用户所能忍受的干扰大小，不过实际中无法给出确切的干扰温度，在噪声与主用户行为同时存在的情况下（绝大部分室外环境）CR 设备并无法分辨两者之间的区别，这也是该算法的缺点。

- **合作频谱感知：**合作频谱感知技术并非辅助频谱感知，由于每个认知设备独立进行频谱感知不一定可以获得完整的频谱信息，所以基于合作感知的 mac 层协议不断被提出，合作频谱感知技术意为认知用户间以某种策略将独立得到的感知结果进行分析叠加，从而尽力获得全面准确的频谱感知结果，文献[3] 中提出了一种基于簇的能量采集的协同频谱感知，认知节点基于它们的接收功率水平聚集在一起，以提高感知性能，文献[4] 提出的基于簇的协同频谱感知，在成簇的基础上，根据每个簇内成员的采集到的信道的方差信息优化自身的大小。文献[5] 中提出了在簇成员发现某一个频谱不可用时，会立刻通过洪泛路由发送给簇内其他成员终止自身的信息传输。

1.4.2 频谱决策

通过频谱感知获取到可用频带的信息后，如何根据需要的服务类型与服务质量选择最适合进行传输作业的信道，是认知自组网的重要研究内容之一，是接入控制与频谱分配的前提。非合作的本地频谱决策，最简单的方式是随机选择一个可用频谱，不对频谱的可用概率，链路质量等传输参数进行考虑，这样可以节省大量的算力，但是并不能保证通信质量。频谱决策应当考虑信道通信质量的同时考虑信道的数学统计特征，此特征根据主用户的动作进行实时改变，适合在主用户活动剧烈的条件下寻找适合的信道，这样可以尽量减少由于主用户的频繁改变导致的信道可用性差的问题，虽然一些频段的通信质量高，但是有可能主用户的活动同样强烈，这也会使得认知设备频繁执行跳频过程影响整个网络的性能，文献[6] 提出了基于信道主用户活动概率的频谱决策方案, 在次用户负荷较低时选择可用概率最高的信道可以有效节约系统时间。

1.4.3 频谱共享

在某个认知设备的通信范围内是可能存在多个认知设备的，在这种条件下，由于自私策略，每个认知设备在感知到类似的频谱结果后，必然会优先使用某个最佳的信道，这会使得此信道由于认知设备的争夺而不再具有通信的可能性，如果这种条件下每个设备都可以与通信范围内的认知节点分享频谱感知结果，通过协商不再争夺同一个信道，这样可以有效提升信道的使用率与网络的性能，所以频谱分享对于认知设备密度高的场景十分有必要，目前的频谱共享依然分为集中式与分布式两种，文献[7] 对比了两种策略，表明了集中式分享策略虽然可以达到最优，但是分布式可以使用较少的能耗。为了平衡两者之间的优缺点，目前的经典的方法就是在成簇的基础上完成频谱共享，同时具

有分布式与集中式的优势，当然也是对两者的一种妥协。

1.4.4 频谱移动

认知设备在通信过程中使用的信道有可能会被主用户占用或者自身的移动导致某一个频带不再可用，此时认知设备必须中断数据的传输，这在射频设备的表现为频率的改变，在这个网络中表现为某个节点暂时丢失（时间很短），在多跳网络中，频谱切换会导致间歇性连通性，甚至是网络分区。在链路断裂之后进入其他信道重新进行组网或者重启数据传输过程，这个过程被称为频谱移动，频谱移动属于组网的基础技术，频谱移动对于射频物理层与链路层和网络层协议都提出了新的要求，所有通信层都应该保证在频谱切换的过程中对数据通信的影响最小，文献[8] 设计了集中式与分布式两种频谱切换与路由设计方案，其设计的互斥算法可以在路由迁移到新的链路上是不干扰其他链路同时降低整体链路迁移的时延。

频谱切换带来的主要结果是数据传输的延时。频谱移动的主要原因来源于主用户的活动，所以制定最佳的频谱移动策略在于处理主用户的活动上，对主用户活动进行合理的建模，分析主用户的时间统计特性，从而保证频谱移动的次数尽量减少。

1.4.5 路由协议

认知无线电网络是电池供电的，且相对密度较大，面对不同的应用程序，认知网络的 QoS 要求也不尽相同，文献[9] 中提出使用基于簇的节点收缩的簇头选择方案与基于可用概率的路由选择方案，在这种簇头选举的条件下，路由选择具有最高可用性概率的路由，即途径路由信道可用概率的乘积。在文献[10] 中提出了一种基于簇的路由协议，将簇分为簇头、成员节点、中继节点和网关节点这四类节点，此路由算法中，节点不需要开机后立刻确认频谱的使用情况，而是随着时间的推移不断确定信道可用性，一组地理相邻的认知设备倾向于共享相似的可用通道集，因此较小的集群规模会增加集群中公共通道的数量，算法中根据度大小，移动特征与可用信道数目确定簇头，之后由簇头广播 RREQ 消息维护整个网络的路由，该种方法使得路由协议的开销，吞吐量，延时都达到了最优。

1.5 论文组织结构

论文一共包括四个章节，具体安排如下：第二章对主用户以及认知用户的行为进行了建模，给出了一种从节点间的信道交汇到路由发现的组网算法，第三章通过代码实现给出仿真结果，针对仿真结果进行了分析；第四章指出了论文中的缺点与未来应该研究改善的方向。

第二章　认知自组网算法

分布式认知自组网算法分为三个部分：节点交会，拓扑发现与路由设计。节点交会是整个组网过程的基础，认知用户使用跳频策略完成邻居节点的交会，文中对节点交会过程的主用户，次用户行为进行了建模，并给出了基于 CGB 算法的算法仿真结果。拓扑发现则是根据分簇思想给出了发现过程，以及算法的仿真结果。路由协议针对网络最低时延与最长寿命分别进行了设计，给出了两者的仿真结果与对比分析。

2.1　系统建模

主用户模型需要考虑主用户到达离开模型，占用的信道；次用户行为则包括通信范围内频谱感知避让主用户其主用户与次用户在同一个信道中的模型如图 2-1 所示

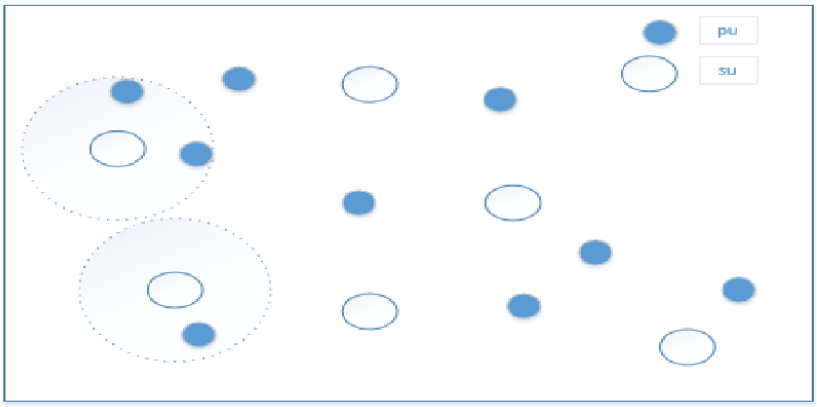


图 2-1 同一信道内模型

2.1.1　主用户行为建模

在 100×100 的空间随机分布一定数量的主用户，每个主用户占用一个频段，互相之间占用的信道不重叠，每个主用户服从到达率为 λ 离开率为 μ 的泊松分布。

2.1.2　认知用户行为建模

在 100×100 的空间内随机分布一定数量的认知用户，每个认知用户可以扫描自身通信范围内的主用户活动即主用户占用的频谱，并可以将可用频段信息存储在可用频段列表中，每个认知用户具有一个唯一的 macid（$1, 2, 3...n$）。

2.2 点对点节点交会算法

文中使用的是 CGB 交汇算法[11]，CGB 算法首先将空间中可用的 N 条信道分为多组，即 N=C_{GROUP}×C_{NUMBER}，信道如下式所示

$$C=\begin{bmatrix} C_{11} & C_{12} & C_{13} & \cdots & C_{1C_{NUMBER}} \\ C_{21} & C_{22} & C_{23} & \cdots & C_{2C_{NUMBER}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ C_{C_{GROUP}1} & C_{C_{GROUP}2} & C_{C_{GROUP}3} & \cdots & C_{C_{GROUP}C_{NUMBER}} \end{bmatrix} \quad \text{式（2-1）}$$

每个认知用户都已概率 P_i 和 1-P_i 分别进入主跳频模式和从跳频模式，两种模式的跳频策略如下：

- **主跳频模式**

  主跳频模式中跳频序列周期性生成，每个周期包含 N 个时隙，为生成主跳频序列，首先需要在集合 C 中随机选取一个子信道集合 C_m 作为新的集合的第一个元素，生成一个经过循环移位的新的信道集合 $M_A^{'}$ = {C_m,C_{m+1},$\cdots$,C_{n-2},$C_{C_{GROUP}-1}$ } 然后按照子信道集合的顺序，分别从每个子信道集合中抽取一个空闲信道 $c_{i_m}^{(m)}$，构成一个序列集合:

  $$M_A=\{c_{i_m}^{(m)},c_{i_{m+1}}^{(m+1)},\cdots,c_{i_{n-2}}^{(n-2)},c_{i_{n-1}}^{(n-1)}\} \quad \text{式（2-2）}$$

  主跳频序列的 $M_A^{'}$ 在 N 个周期中并不会发生改变，只是在大周期中不断改变 M_A 。主跳频序列产生后会出现长度为 C_{GROUP}× C_{NUMBER} 的序列，其中每个跳频元素将会重复 C_{NUMBER} 次，详细的实现方法如图 2-2 。

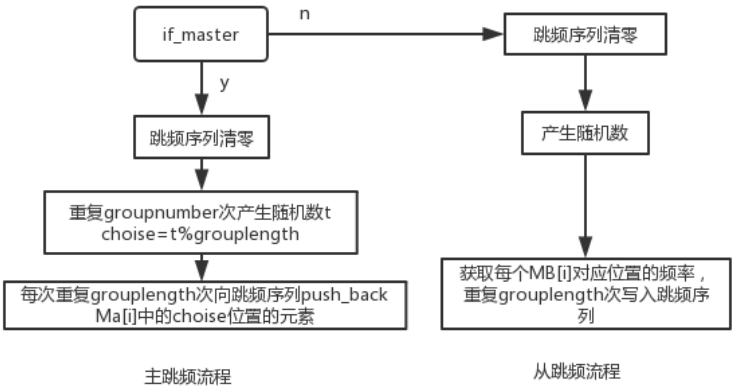


图 2-2 主跳频算法

- **从跳频模式**

从跳频模式与主调频模式相似，首先需要在集合 C 中随机选取一个子信道集合 C_m 作为新的集合的第一个元素，生成一个经过循环移位的新的信道集合 $M_B^{'}$ = $\{C_m,C_{m+1},\cdots,C_{n-2},C_{C_{GROUP}-1}\}$，但是并不要求两者的 C_m 相同，而且只需要进行一次随机数的选举，之后按照顺序将选中的子信道集合中的元素排列进跳频序列即可。跳频模式的算法流程如图 2-2 。

- **运行实例**

下面介绍程序代码中的实例实现 GROUP_NUMBER=27，GROUP_LENGTH=6，在此条件下举例说明运行过程，假设信道分组

$$\begin{aligned} C_1 &= \{1,2,3,4,5,6\} \\ C_2 &= \{7,8,9,10,11,12\} \\ C_3 &= \{13,14,15,16,17,18\} \\ &\vdots \end{aligned} \qquad \text{式（2-3）}$$

假设此时生成的主跳频序列

$$M_A = \{2,10,13,\cdots\} \qquad \text{式（2-4）}$$

生成的次跳频序列

$$M_B = \{1,2,3,4,5,6,1,2,3,4,\cdots\} \qquad \text{式（2-5）}$$

主跳频序列每隔 27 次进行一次跳频，从调频模式下则会每隔时隙进行一次跳频，代码实现中时隙为小循环，两种模式下的交会如图 2-3 所示 当两个节点分别进入

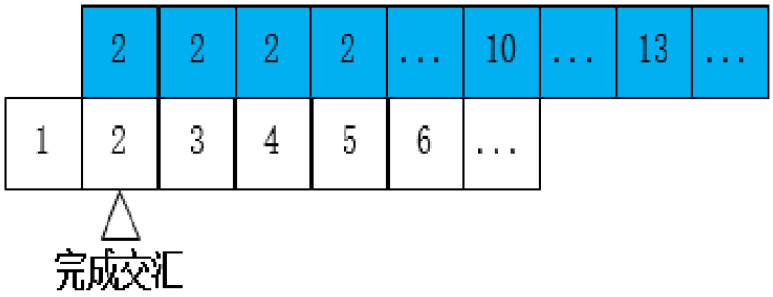


图 2-3 交汇过程

主跳频模式与从跳频模式，在节点时间同步的基础上，由于主跳频序列包括所有

子信道集合中的一个频率，即 C_{GROUP} 个频率点，而从跳频序列会包含某一个子信道集合中的所有元素，即 C_{NUMBER} 个频率点，在同步的条件下，必然可以在 N 个时隙（小循环）中完成，在异步的条件下，则需要 2N 个时隙（小循环），因此 CGB 算法属于有界算法，其 MTTR 为 2N，在代码实现时也选择设为 2N 个循环进行节点交会。

2.3 拓扑发现过程

拓扑发现过程是节点完成通信范围内的节点交会过程之后进行的操作，为了保证每个节点都加入拓扑，需要让每个节点至少拥有一个邻居节点，所以程序运行时如果出现通信范围内没有邻居节点时，组网必然无法包含所有点数，所以所有节点会重新进行随机位置算法。此拓扑发现算法基于成簇的拓扑发现，由于纯分布式网络拓扑发现过程中每个节点都需要得到完备的拓扑信息才能确定网络位置，维护成本随着节点数目增多而增加，所以选用不完全分布式组网方法，选择成簇算法可以减少网络的维护开销，只需要簇头承担拓扑维护工作以及数据转发工作，减轻了网络带宽的消耗。每个簇包含簇头(根节点 R)，一层叶子节点（L），二层叶子节点（E），多层节点（F,G...），可连通的簇间会存在网关节点（N）。拓扑发现算法最终可以将在散落在 100 × 100 内一定数目的认知设备连接成图，形成多个簇组成的完整的拓扑，整个拓扑发现算法需要以底层节点交会为基础，所以下面将会分别介绍拓扑发现的几个过程。

2.3.1 用户初始化

用户初始化过程会对主用户以及认知设备设定地理位置，分配主用户占用的频带，清空用户的路由表，子节点，父节点，信道可用列表等数据。认知设备具有完备的频谱感知功能，可以完整的感知到通信范围内的主用户占用的频率同时建立自身的可用信道列表，列表格式如 2-4 ：列表使用 vector 二维向量进行保存，自向量首位存储信道频率，

Vector<vector<float> > channel

channel	S1	S2
channe2	S1	S2
...	...	...

图 2-4 信道列表

第二位存储累积可用概率，第三位存储当前可用情况，每次进行频谱感知都会得到当前使用频谱是否存在主用户或者次用户数据活动，并将可用信道结果写入信道列表中，如果出现所有信道都被占用，则需要等待一个检测周期不做组网动作，在此之后重新进行频谱感知。频谱感知在算法中使用的是简单的对于主用户能量检测算法，对每个信道进行检测，每次检测都会获取范围内的主用户的 get_exist 函数结果，，此时主用户将会根据到达离开模型给出存在结果返回给此函数，认知设备根据主用户占用的信道获得自身的可用信道列表，此算法要求每个节点每次进入拓扑发现循环都会执行一次。

2.3.2 节点交会

由于节点交会时拓扑发现的基础，所以拓扑发现过程需要与节点交会相连接，而且拓扑发现的性能会对节点交汇过程产生一定的要求。文中使用的节点交会算法基于 CGB 算法，代码实现过程如下：根据 2-2 为每个 CR 设备产生主跳频序列或者从跳频序列，每个认知节点都有相同的概率 P 和 $1-P$ 进入两种跳频模式，在通信范围内的两个节点分别进入两种模式中，则会发生信道交汇，如果进入同一种模式则很大可能无法检测，成功交汇概率即选择不同模式的概率：$P_i = 1 - P^2 - (1-P)^2$ ，以跳频序列为基础，两个设备进入不同的跳频模式，主跳频用户进行如下操作：在所有的可用频段广播寻找帧，如果收到回复则成功在某一信道完成交会，此时设备会交换可用信道列表，并对比选择一条 $S1_i \times 0.5 + S1_j \times 0.5$ 取值最大的频段作为通信信道，此时的频谱决策应当以最大连通的统计特征作为参考，同时此频段为了避免干扰将会被其他认知用户感知到正在活动，这样就完成了频谱共享与频谱决策工作；从跳频用户需要在所有的可用信道监听广播寻找帧，如果接收到主跳频用户活动则同样进行频谱共享与频谱决策。

节点交会阶段的耗时决定了拓扑的完整性，由于并非是所有节点间都能恰好处于主跳频与从跳频同时存在的状态，所以一次的结果并不能代表正确结果，整个阶段应当进行循环检测以求尽力得到完整信息。

2.3.3 成簇

本文使用的是成簇算法，为了确定簇的结构，需要为每个簇确定簇头以及簇内成员。在此处使用获得最大连通性的簇头的选举方法。

在初始化阶段，每个节点的节点状态（R,L,E...）都将会被初始化为 R，即簇头节点，不过子节点数目初始化为 0。进入簇头的选举阶段后应当开始向邻居节点发送簇头选举帧，帧结构如下：收到帧后每个认知用户 i 将会进行如下判断操作：

- SU[i].rootsituation==SU[j].rootsituation==’R’

  两个根节点相遇，则需要对子节点数目进行判断：

  - SU[i].sonnode.size()<=SU[j].sonnode.size()

    节点 i 的子节点数目如果小于等于节点 j 的子节点数目，将 i 的节点状态设置为 L 叶子节点，i 的子节点状态设置为 E 第二层叶子节点，重复执行以确定

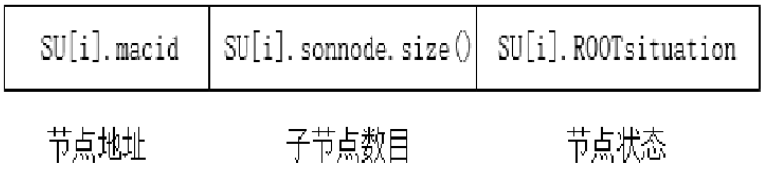


图 2-5 簇头选举帧

节点所位于的叶子层数，同时把 i 的父节点设置为 j，j 的节点状态依然保持 R 即根节点状态，节点 i 将节点 j 加入到自身的子节点列表中。

– SU[i].sonnode.size()>=SU[j].sonnode.size()

节点 i 的子节点数目大于 j 的子节点数目，此时 j 的节点状态更换为 L，j 的子节点变为 E，重复执行同上，j 节点把 i 节点设置为父节点，同时节点 i 将会把 j 加入自身的子节点向量中。

- SU[i].rootsituation==’R’&& SU[j].rootsituation!=’R’

  根节点与非根节点相遇，节点 i 直接将 j 加入子节点，j 的节点状态设置为 L，并确定 j 的子节点的节点状态，j 将 i 设置为父节点。

- SU[i].rootsituation!=’R’

  由于自身为非根节点，与其他非根节点相遇不做任何操作。

此阶段结束后会产生不同的节点状态以及多个簇，此时由于在簇头选举过程中节点可能多次更换父节点，但是却不能确定是到达簇头的最短状态，而且此时并不存在网关节点 N，所以需要进行其他过程来确定最短到簇头状态以及网关节点。当此状态结束后，所有的子节点逐级上传信道列表的信息，由簇头选择簇内通信应该使用的信道，所有的簇头节点需要进入主跳频模式发送寻找帧，如果此时新接入节点接收到寻找帧可以直接作为 L 状态进入下一过程，如果新接入节点在 2N 时隙之后依然没有接收到簇头的消息则作为 R 状态直接进入下一个过程。

2.3.4 最短拓扑发现

由于在成簇阶段的簇头选举过程不能获得最佳拓扑，所以需要在现成的簇的基础上选择节点的最佳状态。此过程需要所有的簇头节点向子节点列表中的所有子节点广播消息，帧格式如下：

当节点 i 收到广播帧后，将会做如 2-6 处理：

每个节点都会产生一个层数值，与节点状态的对应关系为 R->0，L->1....，接收到广播帧后，如果自身状态为簇头，则不做任何处理，如果帧中的跳数低于自身的层数 T，

表 2-1 广播帧

$su[ROOT].macid$ 广播节点 macid	$su[Last].macid$ 上一跳节点 macid	T 跳数

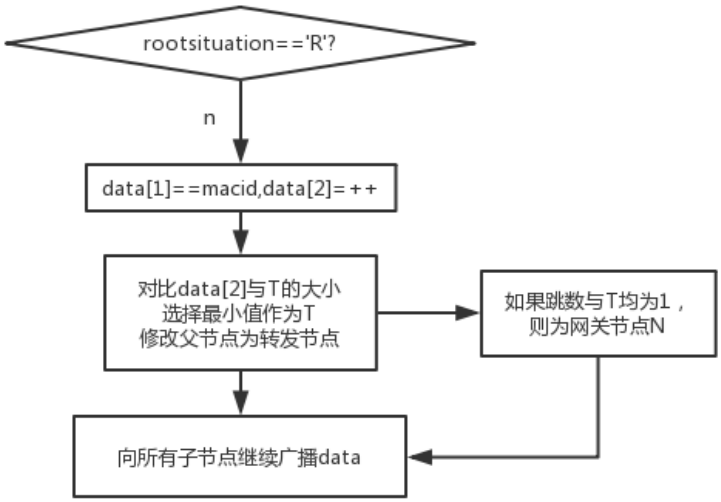


图 2-6 处理广播消息帧

说明有一条更短的到达其他簇头的路径，此时应当把该簇头作为自己记录的簇头节点并与之交换频谱消息，并将转发节点设置为父节点，如果收到两个簇头的消息，证明此节点与两个簇头相连，则节点状态修改为网关节点 N，之后继续进行转发。

此过程完成后整个拓扑发现过程就已经完成，此时网络中将会出现网关节点，网关节点负责两个簇之间的通信需求，簇头负责维护整个簇的信道状况，每个簇头节点将会周期性发送寻找帧，以便新加入节点可以直接加入网络，每层叶子节点周期性向簇头节点汇报可用信道消息，以便在一个主用户产生时候将所有簇内节点搬移到其他频段发送消息。发生拓扑断裂的情况分三种：

- 可用频率断裂。如果簇头节点在当前频段的通信断裂，但是其他簇内成员不会受到影响，依然可以在此频率内进行通信，此时可以直接选举新的簇头，为了通信的需求不能等待簇头的重新连接，而应该尽快建立新的簇完成通信；如果其它层节点出现断裂，则影响较小，由于在频谱共享阶段存储了邻居节点的可用信道，所以可以选择可用概率次大的频率点监听，如果子节点数目不为 0，会在次大概率信道发送寻找帧，而其父节点检测到子节点消失会在次大频率内发送寻找帧，如果重连成功则可直接加入原来拓扑并重新选举簇内的通信频率，一段时间后无法重连则再次进入成簇阶段，初始化自身通信参数。

- 能量耗尽。簇头节点能量必然是最先耗尽的，这也会导致网络的平均寿命不长，这个问题将会在下一章进行讨论。叶子节点能量耗尽会直接退出拓扑，其子节点与

父节点依然同可用频率断裂进行处理。

- 超过通信范围断裂。算法中由于认知节点初始化之后不会再次发生位置移动，所以这部分并未出现在代码中，提出的想法是即使超过通信范围，也应该尝试在次大频率内等待重连一段时间，如果重新进入了簇的通信范围那么可以直接加入，如果再也不回到原始簇范围，将会重新回归成簇阶段，在新的通范围内建立通信。

2.4 路由算法

目前无线网络大多是由 Ad Hoc 路由算法发展而来，主要分为三类：按需路由协议，先验式路由协议，混合式路由协议，路由协议应该考虑的因素有公平性，时延，吞吐量，开销等。本文中的路由算法是在拓扑发现阶段已经完成的情况下进行的，路由设计的优先考虑因素应当是时延，对于时延，在认知无线电中应该考虑跳频时延，这在簇间通信中出现，所以在经过网关节点 N 的时候应当考虑跳频时延，其次则是地理位置所决定的节点距离产生的传输时延，对于处理时延，由于帧长一定，所以每个节点的处理时延大致相同，可以不予考虑。下面对于不用的网络需求给出了按需路由协议与先验路由协议的算法。

2.4.1 按需路由协议

按需路由协议，顾名思义是在节点需要进行通信的情况下申请路由通路，发起路由寻找协议，在寻找到路由协议之后进行数据传输，也仅仅在数据传输过程中进行路由的维护，在通信结束之后不再进行路由的发现与维护，按需路由协议 DSR{$Dynamic\ Source\ Routing$} 是最早应用按需路由协议的，所以文中使用的按需路由协议仿照 DSR 协议进行了设计。在这种方法中，每个节点都会维护自身的路由表，路由表中存储着到达不同节点的路由信息，路由表组织与实际运行结果如图 2-7 与 2-8 所示。

Vector<Vector<int> >routinglist

目的地址	跳数	时延	路由消息
...	...	...	...

图 2-7 路由表

在节点需要通信时，会发起路由寻找，路由寻找帧的设计如下：为了区别网络中传送的路由帧，需要对帧类型进行区分，路由寻找帧 type=0。文中使用的路由属于洪泛式路由，当某个节点发起路由寻找的时候，将会向自身的子节点以及父节点发送此路由帧，节点收到路由帧之后将会进行如下处理：

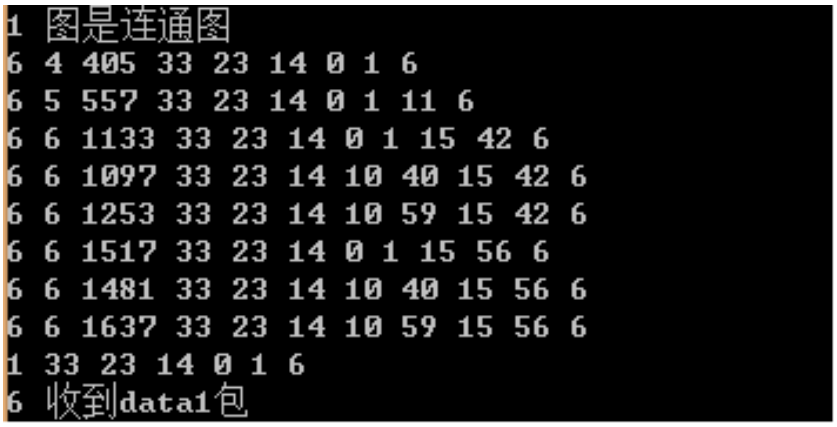


图 2-8 路由表

Vector<int> REQ

Type	源地址	目的地址	跳数	时延	上一跳地址	路由消息

图 2-9 路由寻找

- 当自身不为目的节点时，不会把路由写入自身的路由表，自身的 macid 写入到帧最后的路由表中，帧中的跳数加 1，根据上一跳的地址增加传播时延，如果自身状态值为 N，则需要增加跳频时延（50），根据上一跳中保存的地址，计算出距离加成，为了对信道的统计特征进行计算，应当在传输时延基础上乘以一个与信道可用概率相关的系数，这样可以表现出认知无线电中信道中间的区别，使用概率越高时延应当越小，所以文中取了 $k \times \frac{1}{P}$。
- 当自身即为目的节点时，则会将收到的帧中的跳数，时延，源地址，路由消息录入自身的路由表中，同时向源节点按照最少跳数路由发送 ACK 帧，ACK 帧只是在 REQ 帧的基础上增加了路由表的消息，在源节点收到该 ACK 消息后将会把路由消息进行处理后存入自身的路由表中。

由于网络的每个节点在数据传输过程中消耗的能量不同，所以传输过程中会出现节点死亡的状况，如果只保留最短路径的路由，很有可能在数据传输的过程中发生断裂，所以需要记录多条路由消息以尽力完成信息的传输。每次路由表的写入过程将会将此路由消息与当前路由表进行对比，如果产生重复将更新时延消息后不再重复写入。

为了维护路由，在数据帧传送前需要对当前使用的路由进行能量统计，尽量少的收到其他节点回复的能量耗尽错误，此过程也是路由维护过程，在路由维护过程中将不会再次对路由通路的性能进行重新计算，只对最少能量节点的能量进行估计，估计可以连续传送的数据帧的数量。

2.4.2 先验路由协议

先验式路由协议需要每个节点都保存一张到达同一个网络中其他节点的路由消息表格，不同与按需路由，每当节点需要发送数据包的时候只需要直接在路由表查找现成的路由即可。在文中的实现方案是每隔一个周期进行一次查找作为路由维护过程，而不考虑是否需要进行通信，在需要通信的时候直接发送数据帧，不再执行路由发现过程。

两种路由协议使用的路由发现帧与 ACK 帧相同，很明显两者对应的业务不同，路由协议的开销也不同，下一章节则对两者的不同环境下的性能进行了仿真分析。

2.5 数据传送

数据发送是在路由建立成功的基础上进行的，数据传送的性能主要考虑的因素就是收发时延，在不考虑网络寿命的条件下，所有节点都应使用最小时延的信道进行数据的传输，这样可以获得最好的效果。但是网络的容量有限，所有数据包都使用最短路由，必将会使得路径的节点带宽占用率过高，不利于网络的维护与延长网络的寿命，所以对于不同的业务类型，应当进行不同的处理，文中主要是对于最低时延数据包和最长网络寿命对数据包的路由选择进行了选择。每种数据包的帧构造相同，构造如下 2-10 ：

Vector<int> data1

Type	目的地址	路由消息	负载

图 2-10 数据帧结构

数据包分为以下两种：

- 对时延要求高的业务，对应的 type=1。在路由选择时应当选择最短时延路径，只需要在当前路由表中对比所有到达目的节点的路由，寻找到其中时延最短的节点，不考虑其能量的剩余，路径上的节点收到该数据包的时候，如果自身能量不足以进行下一步转发，则会沿着路径回馈能量不足以转发消息的错误，源节点接收到之后会在路由表中删除当前最短时延通路，之后重新执行最小时延路由选择，重复进行发送，由于路由维护只能估计可以发送数据包的数量，所以如果当前的路由都不再具有转发条件，就会重新寻找路由。

- 对时延要求低的业务，其 type=2。此类型的业务对于时延要求不高，可以不再按照最低时延方案进行寻找，而可以为了获得最长的网络寿命，尽力不耗尽路由上某个节点的能量，所有路由的死亡时间尽量相同。在进行路由选择前，会沿着所有路由向目的节点发送一条能量查询帧，获取每个节点的能量并写入能量路由表

的对应位置，此过程收集完一条可用路径的全部节点能量信息后，如果该条路由支持数据传送，就会直接按照此条通路进行一次数据通信，当所有的节点信息统计结束后，会把路由表中的每个路由条目中节点最低的节点作为该条路由的能量，在进行路由选择的时候，直接对比路由条目的能量，选取其中的最大者进行数据传输。路由表某个条目最低能量为 1 的时候，说明该条路由表不再可用，舍弃该条路由表。

第三章　仿真结果分析

仿真过程中，对于未给出的参量都是用默认的值

表 3-1 默认参数

认知用户数目	70
主用户数目	55
通信半径	35
离开到达率	0.2
时延系数	1
最大跳数	5

3.1　组网时间

组网时间主要由节点交会，成簇阶段用时有关，节点交会过程中，每次进入主跳频模式和从跳频模式都会进行 2N 个循环，而可用信道的变化，也会直接导致认知设备在跳频模式中的交会时间，所以通信半径，主用户数目，主用户到达离开频率，认知设备数目都会对拓扑组网时间产生影响，所以下面分别对三种变量进行了仿真，分析了平均组网时间（CGB 循环执行次数）。

3.1.1　通信半径的影响

图 3-1 中给出了通信半径对组网时间的影响，由于半径的增加，每个认知节点将会监测更多的主用户活动引起可用频段的减少，同时需要进行交汇的节点数也会增多，这会直接导致交会时间的延长，所以整个组网的时间也会延长，根据仿真也可以确定，随着通信半径的增加，组网时间将会不断延长。

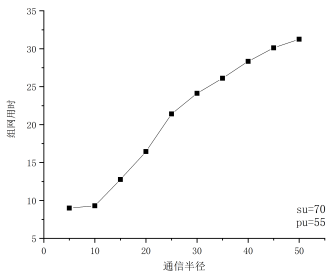


图 3-1 通信半径对组网时间的影响

3.1.2 认知设备数目与主用户数目的影响

认知设备数目的增加会直接引起节点密度的增加，在通信范围一定的情况下每个认知设备将会与更多的节点进行交会，分析如上，主要受到影响的依然是在节点交会阶段的耗时。主用户的影响主要体现在对频率的占用情况上，由于每个主用户都会占用一个不同的频段，这直接会导致认知设备可用信道的减少，当主用户节点数目占据了所有的信道分组之后所有节点都会无法进行通信，组网时间将无限延长。图中所示即为认知设备与主用户数目对于组网时间的影响，图中可知，认知设备与主用户数目的增多，的确会增加组网消耗的时间。

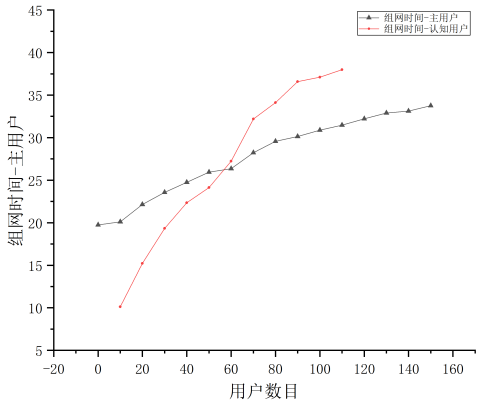


图 3-2 用户数目对组网时间的影响

3.1.3 主用户变化概率的影响

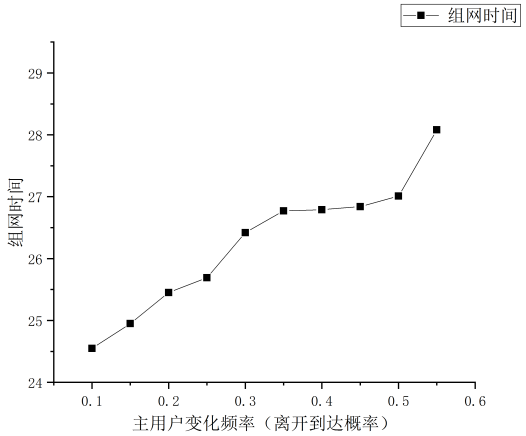


图 3-3 主用户变化概率对组网时间的影响

在仿真中，每个主用户都是具有到达离开模型的，而其中的到达率与离开率直接决定了主用户变化的频率，如果主用户一直不变化，那么认知设备可以更好地在稳定的可

用信道上完成交汇，而随着主用户变化频率的增多，必将使得信道的可用概率变化更为突出，这也会延长组网时间，其仿真结果如图所示，根据图 3-3 中所示我们可以发现，在主用户变化频率低的时候，组网消耗的时间很短，而随着变化频率的增加，信道可用性变差，也使得组网时间不断延长。

3.2 组网成功率

根据数据结构课程内容可知，一个无向图的连通性可以利用它的邻接矩阵特性进行评定，我们在代码实现过程中，建立所有节点的邻接矩阵，根据自身的子节点向量与父节点，建立矩阵 L：

$$L[i][j]=L[j][i]=\begin{cases}1 & if(SU[i].fathernode==SU[j] or SU[j]\in SU[i].sonnode)\\ 0 & else\end{cases} \quad \text{式（3-1）}$$

要判断一个无向图是否是连通图，可以对图做深度遍历（DFS）[12]，遍历即从图的某个初始节点开始，对图中的每个节点进行访问，每个节点访问且仅访问一次。深度遍历步骤如下：

- 指定某个节点作为初始节点
- 若当前的访问节点的某个邻接节点未被访问，则开始对其深度遍历，若邻接节点都被访问则退回最近访问的节点，直到所有节点都被访问
- 若某个节点未被访问，重新选择某个节点进行深度遍历。

其伪代码如下

```
count=0
for 0 to CR_NUMBER
 for 0 to CR_NUMBER
 if L[i][j]&&!known(j)
 known[j]=1
 DFS(j)
 count++
```

如果节点密度不一致，不同区域分布的节点数相差较大，如果通信半径较小，则会产生不相交的簇，簇之间不拥有任何网关节点，因此簇之间不再具有通信的可能性，而解决这个问题的首要办法就是平均节点的分布或者提高网络的通信半径，理论上通信半径越高，可以获得的节点消息越完整，越不容易出现孤立的节点或簇。图**??**给出了通信半径对组网成功率的影响（由于在 R<15 时多次出现节点无邻居的情况，成功率很低，所以从 15 开始进行对比），图中可知上述理论是正确的，随着通信半径的提高，出现孤立节点的次数减少，更容易获得连通图，但是在通信半径高于 35 以后，成功率将会出现些

许的下降，这是因为随着通信半径的增加，成簇的数目将会减少，而每个通信簇内的节点数目将会增多，所以平均到每个簇内会有更多的主用户占用信道，组网成功率就会有所下降。

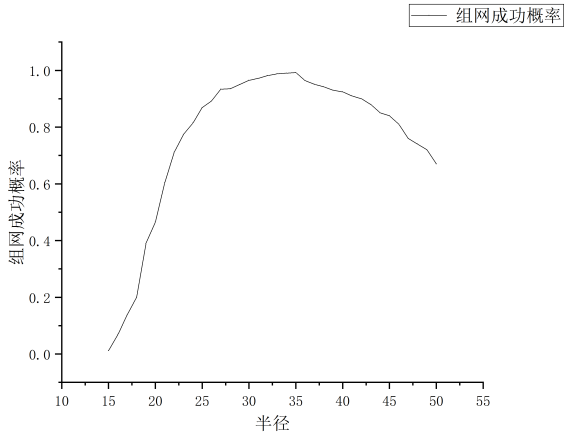


图 3-4 通信半径对组网成功率的影响

认知节点数目决定了整个范围内部的整体密度水平，也就是在通信范围不变的情况下也会影响组网成功率，认知节点数目越多，通信范围内可以连通的点越多，越不容易得到孤立的簇或者节点，认知节点数目过多的时候，会使得每个簇占用的空间区域较大，可以使用的信道数目变少，使得节点交会变得困难，所以组网成功率会不断下降，图 3-5 给出了不同的节点数目对于组网成功率的影响

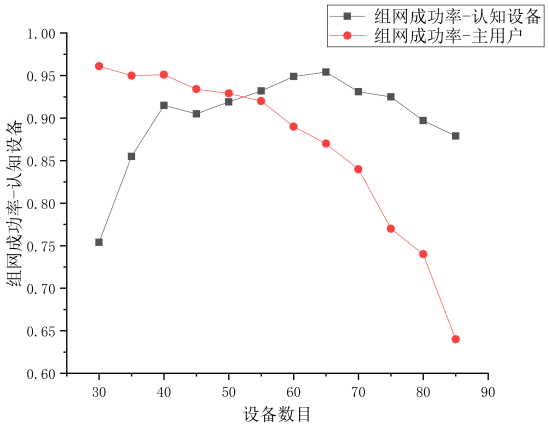


图 3-5 节点数目对组网成功率的影响

3.3 网络健壮性

评价网络的好坏不仅仅只有连通性，还有网络的健壮性，当节点间的通路越多的时候，整个网络的连通性越强，根据图论的知识，可以根据一个图的 *Laplacian* 矩阵的次

小特征值判断图的连通性，$Laplacian$ 矩阵特征值的性质：最小特征值为 0，特征值中 0 的个数即连通子图的个数，第二特征值越大，代数连通度越高，图的健壮性越好，代码实现过程中参考了 QR 分解法求解特征值 $Matrix_EigenValue$。$Laplacian$ 矩阵的定义为 $L = D - W$，W 为上述的邻接矩阵，D 为度矩阵，度矩阵即

$$D[i][j] = \begin{cases} \sum_{m=0}^{CR_{Number}-1} W[m][j] & if(i == j) \\ -1 & else \end{cases} \quad \text{式（3-2）}$$

文中的认知网络由分簇的模型构成，不同簇之间可能拥有不同数目的网关节点，每个簇之间的网关节点越多，簇内的叶子层数越少，整个网络的健壮性越高，图 3-6 中给出了不同认知数目与主用户数目下对网络健壮性的影响以及通信半径 (3-7) 对网络健壮性的影响。

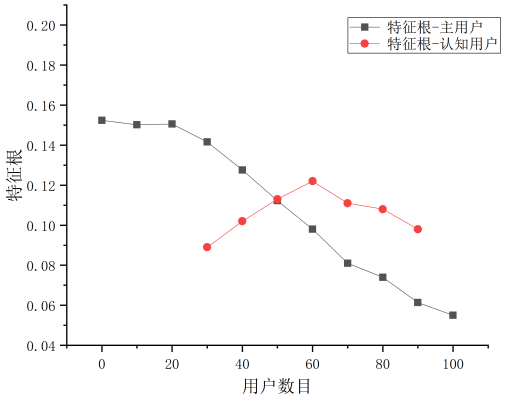


图 3-6 节点数目对网络健壮性的影响

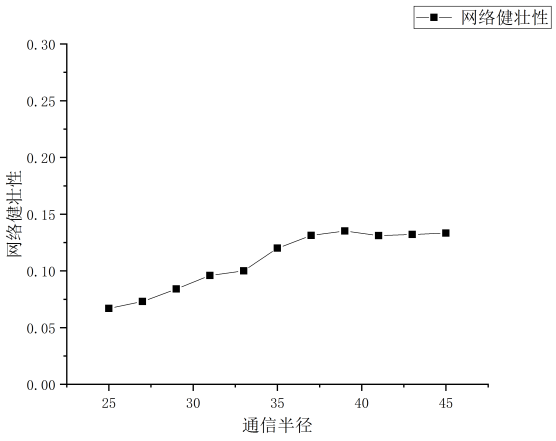


图 3-7 通信半径对网络健壮性的影响

图中可知，主用户作为干扰因素，其数量越多，组网后网络的健壮性越低，认知用

户作为通信节点，初始阶段会因为节点数的增加使得整个网络的簇更容易相连，出现多个连通子图的情况减少，在认知节点不断增多后，网络健壮性又会下降，认知节点增多会使得子簇的深度加深，虽然也会增加网关节点的数目，但是深度影响增加还是使得网络的健壮性下降，不同节点之间点点连通的概率下降，而通信半径也会增加网络的健壮性，减少子图出现概率，在增加到一定数值之后不再对健壮性产生过大影响，甚至使健壮性略微下降，这种情况与信道减少以及深度增加相关。

3.4 路由与数据传送性能

3.4.1 可用路由数目

可用路由条目的数目是保证数据能够成功发送的根本，在分簇网络中，节点间通信的最远方式为：簇内成员 $\rightarrow$ 簇头 $\rightarrow$ 网关节点 $\rightarrow \cdots$ 簇头 $\rightarrow$ 簇内成员，由于分簇网络作为多跳网络，节点间的通信需要借助多个簇头与网关节点，而路由条目的数目保证了在某个簇头或者网关节点死亡的条件下依然可以选择其他通路进行通信。

文中使用的路由寻找方法属于洪泛式路由，在路由寻找过程中一个比较严重的问题就是网络中将会出现过多的路由寻找数据包，这要求必须对每个路由寻找帧的寿命进行限制，以免在真实应用场景中造成网络拥塞，所以对于这个问题下面给出了路由寿命（路由寻找帧的跳数上限）与可用路由数目的关系，选择合适的网络路由寿命有利于整个网络的运行。路由寻找的过程，跳数越多，可以获得的路由将会越多，但是也会延长网络的整体路由发现时间，并且真实无线网络中多跳路由的生存类似于通信网课程中的生灭过程[13]，其过程的状态转移图如图 3-8 所示

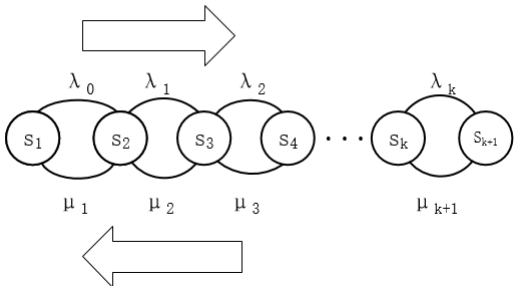


图 3-8 路由状态转移表

令 $\theta_0 = 1$，$\theta_k = \frac{\lambda_0\lambda_1\lambda_2\cdots\lambda_{k-1}}{\mu_1\mu_2\mu_3\cdots\mu_k}$，其中每个状态的 $P_k = \theta_k P_0$，代表路由条目中簇头正常工作的数目。稳态分布下 $P_0 = \left(1 + \sum_{i=1}^{\infty}\theta_k\right)^{-1}$，所以随着路由条目中经过的节点的数目增加将会同时增加路由条目失效的风险，因此在实际应用场景中对于跳数以及路由失效风险应该进行权衡的选择。

仿真中给出了不同跳数下可以获得的路由条目数目以及程序运行的平均时间（c++ 中利用 clock tick 表示）的关系图 3-9，由于在 T>6 之后程序运行时间实在过长，所以

随机选取了两个节点作为通信双方，并对他们的路由过程进行了平均计算，为了便于观察，对 clock tick 进行了取对数后缩倍处理，据此来观察路由协议的收敛时间，由于计算机模拟在超过 8 跳之后即使是两个节点采平均数的时间也是十分漫长，所以只观察 8 跳之前的路由状况。根据仿真结果中可知，整体网络的路由过程运行时间与最大跳数成

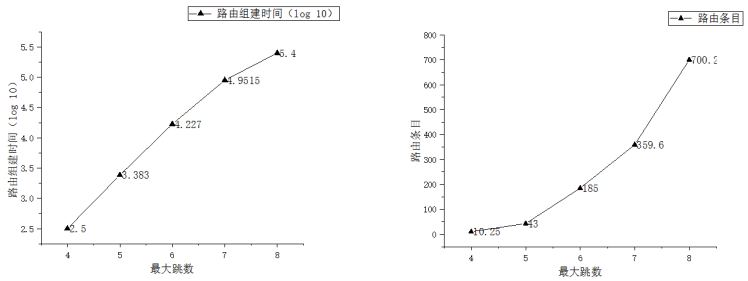
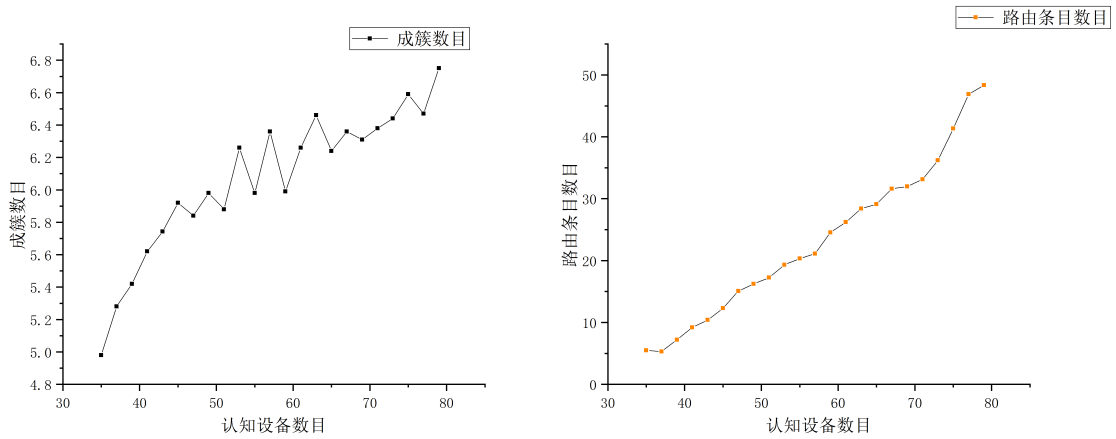


图 3-9 跳数与路由性能的关系

正比关系，且随着跳数增长，消耗的时间也会呈现指数型增长趋势，而路由条目同样也与最大跳数的幂函数成正比，这是因为跳数的增加意味着可以经过的簇头数目与网关节点数目增加，而并非简单地加法，是与簇头及网关节点的次幂函数相关。图中我们也可发现，这两者的优势不能同时具备，要向获得更多的路由条目意味需要消耗更多的组网时间。

分簇网络中的路由与分簇数目有着密切的关系，由于节点通信过程中会使用簇头和网关节点作为中间跳点，所以成簇的个数一定程度上影响了路由的跳数以及路由的可靠性，在文中成簇的个数主要受通信范围以及认知设备的密度（数目）的影响，下图 3-10 以及图 3-11 中给出了路由条目的数目与通信半径和认知节点数目的关系以及两者与成簇数目的关系，考虑到通信半径过大时组网成功率将会到达最大概率后降低，所以只考虑单调增区段的通信半径与路由数目的关系。图中可知，随着认知节点的增加，

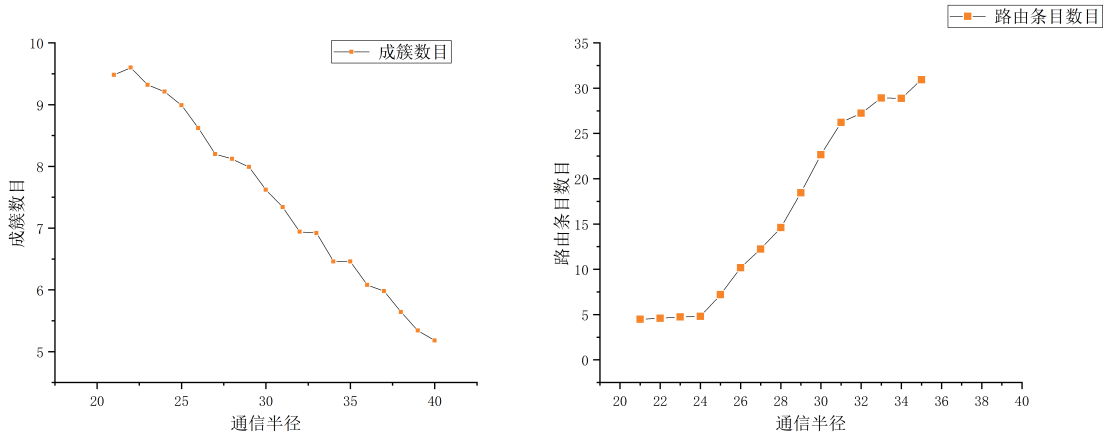


(a) 认知节点与成簇数目关系　　(b) 认知节点与路由数目关系

图 3-10 认知节点数目对路由的影响

整个成簇的数量减少，意味着成簇深度的增加，而路由条目数目也随着认知节点数目增

加而增加，证明网关节点的增多，产生了更多的路由条目，而通信半径的增加也同样增加了网关节点的数目，虽然网络健壮性下降但是路由数目却不断增加。



(a) 通信半径与成簇数目关系　　(b) 通信半径与路由条目关系

图 3-11 通信半径对路由的影响

3.4.2 网络平均寿命

文中给出了两种路由策略，一种先验式路由协议，一种按需路由协议，这两种路由策略的应用场景有所区别，在数据传送活动较少的区域，按需路由协议理因为其实时性的特点，应当有着更加优秀的能耗，在数据传送活动较多的区域，先验式路由协议可以节省大量路由数据包，从而获得更佳的能耗比。图 3-12 中给出了不同数据活动比（活动时间占总系统运行时间的比例）下的两种路由模式的能耗比例 由图可知，按需路由

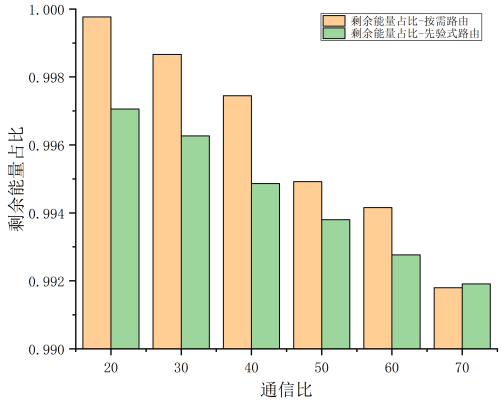


图 3-12 路由类型与剩余能量比例

协议在进行少量通信的时候可以节省更多能量，大于 70 的活动时间比时能耗将会高于先验路由协议，所以在数据活动剧烈的区域应当优先使用先验式路由协议，相反则优先使用按需路由协议。

3.4.3 传送数据类型

对于网络中不同的数据类型要求，适当的路由选择方式可以有效延长整个网络的节点运行时间。所以给出了两种数据包类型，一种低时延数据包和一种最长网络寿命数据包。低时延数据包由于会长时间使用最短时延线路，将会先出现死亡节点，通过观察第一个死亡节点的出现时间，可以估计整个网络的寿命，所以代码实现中，会对所有节点初始能量设置为随机数。图 3-13 中给出了首次出现节点死亡的时间对比。

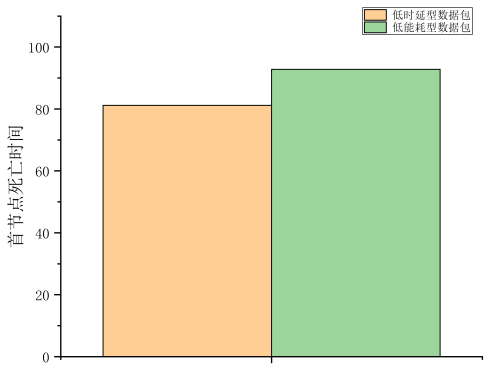


图 3-13 数据包类型与首节点死亡时间对比

图中可以看出，使用第二种路由方案可以有效延长首个节点死亡时间，同理可以增加整个网络的平均寿命，但是这是以不能取得最短时延作为代价的，在真正的网络中，每个节点都是自私的，不会有设备愿意承担超过发送自身需要的数据之外的功能，因为这会导致自身的能量优先消耗完，所以对于大量数据要进行传输，同时时延要求不高，建议优先使用最长寿命的路由方案，减少对其他设备的负担同时增加网络的流量。

3.5 拓扑发现的另一种方案

当我们把延长网络寿命的任务不仅仅放置在路由协议的设计上，而同时插入在拓扑发现协议中，应该可以获得更长的网络寿命。在成簇阶段，上面的算法是将子节点数目最多者设置为簇头，但是簇头的能量却有可能是最小的，这样的条件下就会多次重复选举簇头，为了减少重复选举过程，我们可以在选举过程中将能量作为优先考虑要素，抱着这种想法在同样的环境下进行了模拟。下图中给出了在这种情况，首个节点出现死亡的时间的对比，以及相同条件下的组网成功率对比。如图中所示，虽然选举能量最大者作为簇头可以有效延长首节点死亡时间，但是组网成功率低于最大连通性的簇头选举方法，网络寿命与最佳连通性两者之间如果在提出的选举簇头的方案上做改变并不能同时满足。

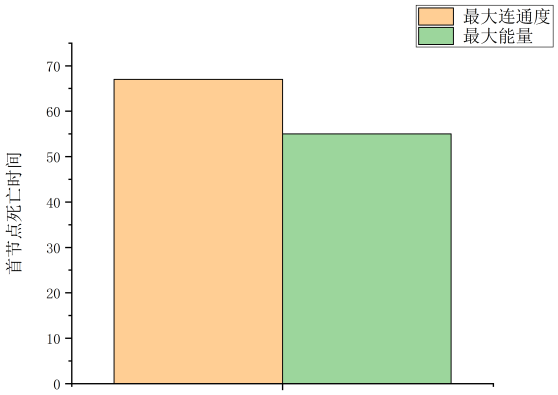


图 3-14 首节点死亡时间对比

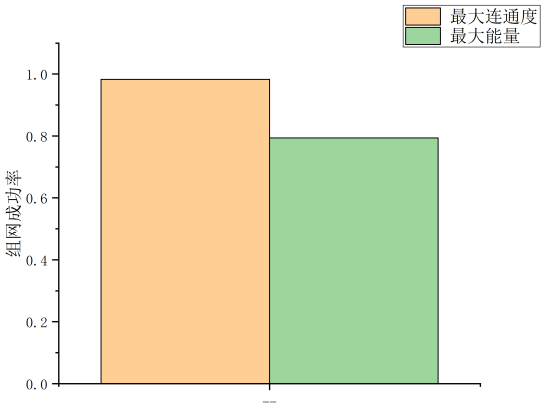


图 3-15 组网成功率对比

第四章　总结与展望

4.1　论文总结

本文给出了认识无线电组网的一种策略，方案中包括了从节点交汇到路由发现和数据传输的整个阶段，第二章中详细地对算法的执行过程进行了描述，并在第三章中给出了仿真结果与分析，并尝试了一种改进策略。

4.2　论文展望

上文中模拟了从节点交会到正常通信的全过程，但是文章的缺点也是显而易见的，成簇的算法作为一种分布式与集中式两者的妥协产生的结果，在获得了两者的部分优点的条件下，也同样具备了两者的缺点，簇头作为网络中权值较重的节点，担任着本不应该由普通节点担任的责任，这样实际上破坏了节点间的公平性，所以若是想要解决这个问题，需要提出一种周期性簇头轮换策略，但是又应该保证轮换过程中簇结构不会发生较大改变，避免对正常通信产生过大的干扰，这需要一个能量与连通性之间的权值，综合考虑两种因素之后决定应该由那个节点担当簇头，另外一种策略就是抛弃分簇思想直接从分布式结构入手，这样符合了节点间的公平性，但是也不再拥有低难度的设计参考，对系统方案的设计者提出了更高的要求，完全平均权值的无线网络目前依然是不存在的，既要满足节点的自私策略又要获得最佳的通信效果与网络寿命。

参考文献

[1] Digham F F, Alouini M S, Simon M K. On the Energy Detection of Unknown Signals Over Fading Channels [J]. IEEE Transactions on Communications. 55 (1). 2007, 1: 21–24.

[2] D Cabric R W B, A Tkachenko. Spectrum Sensing Measurements of Pilot Energy and Collaborative Detection [C]. In MILCOM 2006 - 2006 IEEE Military Communications conference, 2006: 1–7.

[3] Yao F, Wu H, Chen Y et al. Cluster-Based Collaborative Spectrum Sensing for Energy Harvesting Cognitive Wireless Communication Network [J]. IEEE Access. 5 (3). 2017, 5: 9266–9276.

[4] A Ghasemi E S S. Collaborative spectrum sensing for opportunistic access in fading environments [C]. In First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005: 131–136.

[5] X Liu Z D. A Channel Evacuation Protocol for Spectrum-Agile Networks [C]. In 2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2007: 292–302.

[6] Wang L C, Wang C W, Adachi F. Load-Balancing Spectrum Decision for Cognitive Radio Networks [J]. IEEE Journal on Selected Areas in Communications. 29 (4). 2011, 4: 757–769.

[7] Chunyi Peng B Y Z, Haitao Zheng. Utilization and fairness in spectrum assignment for opportunistic spectrum access [J]. Mobile Networks and Applications. 11 (4). 2006, 8: 555–576.

[8] Feng W, Cao J, Zhang C et al. Joint Optimization of Spectrum Handoff Scheduling and Routing in Multi-hop Multi-radio Cognitive Networks [C]. In 2009 29th IEEE International Conference on Distributed Computing Systems, 2009: 85–92.

[9] Huang X L, Wang G, Hu F et al. Stability Capacity Adaptive Routing for High Mobility Multihop Cognitive Radio Networks [J]. IEEE Transactions on Vehicular Technology. 60 (6). 2011, 6: 2714–2729.

[10] Saleem Y, Yau K-L A, Mohamad H et al. SMART: A SpectruM-Aware ClusteR-based rouTing scheme for distributed cognitive radio networks [J]. Computer Networks. 01. 2015, 11: 196 – 224.

[11] 吴锐．认知无线自组网中组网技术研究 [学位论文]．北京邮电大学，2016．

[12] 徐雅静．数据结构与 STL [M]．北京邮电大学出版社，2014．

[13] 苏驷希．通信网性能分析基础 [M]．北京邮电大学出版社，2005．

致　　谢

normalize