

**研究生学位论文开题报告**

**报告题目 区块链共识算法的研究**

**学生姓名 张利 学号 201818020629027**

**指导教师 姚郑/张宝贤 职称 教授**

**学位类别 工学博士**

**学科专业 计算机应用技术**

**研究方向 区块链共识算法**

**培养单位 人工智能学院**

**填表日期 2021-9-1**

**中国科学院大学制**

**填 表 说 明**

1. 本表内容须真实、完整、准确。
2. “学位类别”名称填写：哲学博士、教育学博士、理学博士、工学博士、农学博士、医学博士、管理学博士，哲学硕士、经济学硕士、法学硕士、教育学硕士、文学硕士、理学硕士、工学硕士、农学硕士、医学硕士、管理学硕士等。
3. “学科专业”名称填写： “二级学科”全称。

目录

[1选题的背景及意义 4](#_Toc93415122)

[2国内外本学科领域的发展现状与趋势 5](#_Toc93415123)

[2.1 单一主链区块链共识算法 5](#_Toc93415124)

[2.1.1 基于工作量证明共识算法 6](#_Toc93415125)

[2.1.2 基于权益证明的共识算法 7](#_Toc93415126)

[2.1.3单一委员会混合共识算法 7](#_Toc93415127)

[2.1.4多委员会混合共识算法 8](#_Toc93415128)

[2.1.5其他共识算法 10](#_Toc93415129)

[2.2 DAG区块链共识算法 10](#_Toc93415130)

[2.2.1 Tangle共识算法 11](#_Toc93415131)

[2.2.2 基于见证人的共识算法 11](#_Toc93415132)

[2.2.3 基于DPoS的共识算法 12](#_Toc93415133)

[2.2.4基于虚拟投票的共识算法 12](#_Toc93415134)

[3课题主要研究内容、预期目标 13](#_Toc93415135)

[4拟采用的研究方法、技术路线、实验方案及其可行性分析 14](#_Toc93415136)

[4.1技术路线方案 14](#_Toc93415137)

[4.1.1基于节点稳定度的无线区块链网络共识算法 14](#_Toc93415138)

[4.1.2基于稳定性的PBFT无线区块链网络共识算法 16](#_Toc93415139)

[4.1.3基于稳定度的分片PBFT区块链共识算法 17](#_Toc93415140)

[4.1.4基于稳定度的DAG区块链共识算法 19](#_Toc93415141)

[4.2可行性分析 21](#_Toc93415142)

[5研究工作计划与进度安排 21](#_Toc93415143)

[参考文献： 22](#_Toc93415144)

# 1选题的背景及意义

随着互联网的蓬勃发展，多方协作的安全性和可信性要求越来越高。随着多方协作的规模越来越大，传统的第三方信任机构会伴随着较高的人为决策风险和成本。因此中心化机构已经无法满足大规模协作的需求，区块链技术作为新的信用体系应运而生。

自2008年中本聪[1]提出比特币后，其底层技术—区块链技术受到了工业和学术领域的高度关注。区块链本质上是一个去中心化的数据库，结合了P2P网络、分布式数据库、共识算法、非对称加密等多种技术。区块链应用的P2P网络中各节点的计算机地位平等，每个节点有相同的网络权力，不存在中心化的服务器。所有节点间通过特定的软件协议共享部分计算资源、软件或者信息内容。区块链主要是解决信任和安全问题，提供了一种在不可信场景中安全可靠地进行信息与价值传递交换的协作模式。

区块链大致经历了三个发展阶段：分为区块链1.0、区块链2.0和区块链3.0时代。区块链1.0时代，区块链技术主要是作为数字货币的底层支撑技术。在这个时期,区块链技术的发展与数字货币密切相关,应用普遍集中在货币转移、兑换和支付等方面。随后区块链2.0时代主要是解决市场的去中心化，这个时期的发展与合约技术的发展密切相关。V.Buterin[2]首次提出实现智能合约编写的以太坊区块链平台，将智能合约和区块链技术结合使得区块链技术得以提升。区块链为智能合约提供可信执行环境,而智能合约便于区块链扩展应用。随着时间的推移,这项技术的应用超越了货币,延伸至更广阔的领域,并具备了更大的兼容性。区块链3.0时代也是区块链全面应用的时代,由此构建一个大规模协作社会。除金融、经济等方面,此时的区块链在社会生活中的应用更为广泛,特别是在政府、健康、科学、文化和艺术等领域。区块链技术无论是在构建价值自由流通的互联网，还是在企业基于建立联合多中心的数据共享方面，都有着广阔的应用前景。

随着5G和物联网的普及，在无线节点之间管理设备、共享消息和执行计算任务等面临的问题越来越多，在具有小世界和超密集特征的无线网络中，安全和信任问题甚至变得难以处理。无线网络具有高灵活性、便捷性、易于网络规划和调整、故障定位容易以及易于扩展等优点。但是无线网络面临着严重的网络安全问题，包括女巫攻击、无线信道劫持攻击、日蚀攻击、窃取网络资源等，这些都会对建立在无线网络上的应用产生致命伤害。

区块链的安全可信任特性可以为无线网络的应用提供安全和信任保障。如果将无线网络应用中数据的上传和在应用中的操作都当作区块链交易来处理，则可以在传输层面保护数据和通信安全。区块链可以将设备映射为公私密钥，数据的上传视为由智能合约验证的交易事务，区块链使用非常先进的加密技术来确保事务信息被锁定在区块链中，安全性相比不使用区块链技术的无线网络应用得到了很大的加强。

区块链技术可以有效的解决建立在无线网络上的应用中的安全信任问题，确保数据的安全性、可信性和完整性，但是区块链技术应用于无线网络场景还面临着一些挑战。一方面，区块链的区块链系统的性能很大程度上是受到共识算法的限制的。目前主流的区块链共识算法主要有PoW共识算法,PoS共识算法以及PBFT共识算法等。这些共识算算法具有能耗巨大，效率低，依赖代币，网络通信复杂度高等缺点。另一方面，无线网络可用资源的波动性和移动终端设备可用资源的有限性，无法满足目前需要耗费大量计算资源以及需要稳定的网络传输数据的区块链系统的需求。因此，亟需根据无线网络的特性设计高效节能的区块链共识算法，使得引入区块链技术的无线网络应用具有很好的扩展性和安全性。

# 2国内外本学科领域的发展现状与趋势

共识算法是区块链技术的基础和核心，影响着区块链系统的可扩展性和安全性。在设计区块链共识算法时需要重点考虑系统模型和共识过程。系统模型包括网络模型、（同步、部分同步、异步）、节点模型（动态节点、静态节点）、故障模型（拜占庭故障、非拜占庭故障）、网络可用资源（功率、信道、存储资源等）；共识过程则包括共识节点的选择、交易排序、验证和区块上链。通过考虑不同的应用场景，为了降低共识延迟、提高系统交易吞吐量，学术界和工业界对区块链共识算法开展了大量的研究。下面将阐述与本报告研究内容相关的典型工作：单一主链区块链共识算法和DAG区块链共识算法。

## 2.1 单一主链区块链共识算法

在区块链系统中，大致可以将共识算法分为：基于工作量证明共识算法、基于股份证明的共识算法、委员会共识算法（单一委员会混合共识算法、多委员会混合共识算法）以及其他共识算法。

### 2.1.1 基于工作量证明共识算法

工作量证明最早C.Dwork和M.Naor[3]在1992年提出，用来解决垃圾邮件问题.该机制要求邮件在被发送之前必须完成某个数学难题的答案来证明确实执行了一定量的工作。A.Back在1997年提出,并在2002年正式发表的Hashcash [4]通过寻找哈希函数原像实现工作量证明。在1999年，M.Jakobsso[5]正式提出了工作量证明的概念。这些工作都为比特币的共识机制奠定了坚实的基础。

为了解决交易无第三方的信任问题，中本聪[1]采用工作量证明的共识算法为电子货币的交易达成安全共识。用户通过使用算力进行哈希计算来证明自己生成区块的合法性，并通过一个链式结构来确保达成共识的账本不可篡改，保证了账本的安全性。在区块链技术中，工作量证明主要用来防止敌手制造假身份发动女巫攻击。基于工作量证明共识算法中，所有参与共识的节点都可以通过消耗计算算力解决一个数学难题证明自己完成了一部分工作，从而获得生成区块的权限，出块成功后可以获得系统的奖励以及区块中所有交易费用。

工作量证明共识算法提出之后，一些研究者针对比特币的安全性进入深入研究。J.Garay等人[6]提取和分析了比特币协议的核心，针对区块链的共同前缀、链质量和链增长特性，证明了在满足哈希算力和同步网络下区块链的安全性，为加密货币的共识算法提供了首个“可证明安全”模式。R.Pass等人[7]在此基础上进一步分析了区块链在部分同步的网络环境下区块链的安全性。此外，工作量证明共识算法也存在被敌手攻击的可能性，常见的有日蚀攻击[10]、双花攻击以及自私挖矿攻击[11,12]等。其中，I.Eya[12]通过分析比特币系统中自私挖矿的成功概率，最终证明敌手可以通过控制全网的小部分算力攻击系统成功。

采用工作量证明的区块链系统中矿工可以集中计算资源发起自私挖矿攻击而获益，为了解决比特币中矿工算力中心化和挖矿资源集中化的问题，以太坊[2]设计抵抗ASIC且支持轻量级客户端快速验证的工作量证明算法Ethash缓解挖矿中心化的问题。通过降低找到工作量证明的难度使得即使拥有较少算力资源的用户也能够快速找到工作量证明获得奖励。但是降低挖矿难度会使得系统出块速率提升。由于出块时间短，以太坊出现分叉时算力小的矿池或节点的分支将被裁减。以太坊引入叔块(Uncle block)使得分叉区块也会被后出现的区块引用，从而解决了低算力矿池和节点缺乏分支竞争力的问题。因此，基于工作量证明算法Ethash在确保系统安全性的时候提升交易吞吐量降低交易确认延时。除了在电子加密货币上的应用，区块链也可以作为智能合约的底层技术。以太坊作为能够运行智能合约的公共区块链平台，允许双方在没有可信第三方的情况下实现可信交易。

为了解决比特币区块链系统中由于自私挖矿攻击导致的诚实用户链质量的下降的公平性问题，R.Pass和E.Shi[9]提出了水果链（FruitChains）。其共识算法也是基于工作量证明的，水果链引入了两种数据结构：区块和水果（fruits）。区块和水果的产生都是通过寻找工作量证明来完成,但是生一个区块需要完成工作量证明的难度要远远高于生成水果的难度。其中水果是挂靠在区块下的，而交易则是与水果绑定的。新挖到水果被放入有效水果集(Fruit set)中，有效水果集随着水果的挖出不断更新。新挖到的区块将有效水果集中的水果放入到区块中，由于系统需要确保水果的新鲜度（fresh），因此自私挖矿生成的区块和水果不会被诚实矿工接受。最终几个区块和水果的奖励会平均奖励给所属节点。在FruitChains中,节点不再需要参与矿池就能频繁获得挖矿收益,提升了系统参与者获取奖励的公平性，降低了矿池带来的算力集中化。

采用工作量证明作为共识算法的区块链系统面临着交易吞吐量低的问题，有研究者对提升区块链的交易处理速率做了深入研究。I.Eyal等人[8]提出的Bitcoin-NG也是一种基于工作量证明的共识算法，提升了比特币中完成一次工作量证明能够处理交易的数量。同时，Bitcoin-NG也改变了比特币的激励机制，提升了诚实矿工愿意在诚实链后添加区块的积极性，降低了区块链分叉的动机。Bitcoin-NG中引入关键块和微块，使得系统能够在增加交易区块频率、提升交易吞吐率的同时, 保证协议的安全性和公平性。这个共识算法将交易区块与出块者选举的过程分离开来, 体现了协议设计的模块化思想。

针对工作量证明共识算法中基于ASIC设备算力集中化的问题，M.H.Ashilk等人[51]提出过滤工作证明共识算法 (FPoW)来解决。Filtered Proof-of-Work采用回合制，在每一回合开始时，都会有一个一定难度级别的虚拟数学谜题来计算采矿设备的计算能力。这将使系统能够找出基于 ASIC 的挖矿节点，所有基于 ASIC 的采矿设备都将在列表的顶部。根据该列表，系统会从网络中移除一些算力比其他节点更大的挖矿节点。完成所有过滤后，原有的工作量证明机制将在剩余的挖矿节点之间运行，具有一定的难度。能够首先解决该问题的挖矿节点将创建一个新区块并获得奖励。之后将开始新一轮，并重复相同的程序。在每一轮之后，网络将允许连接新节点或移除旧节点。这个共识算法可以较好的防止由于ASCI设备带来的算力中心化，实现比较公平的共识算法

工作量证明的共识算法面临着严重的能耗问题和双花攻击问题。现有的基于工作量证明的共识算法主要是解决自私挖矿攻击引起算力集中化而导致的不公平问题，以及提升完成一次工作量证明能处理交易的速率。但是工作量证明的寻找需要投入大量的算力，而算力的维持则需要消耗巨额的电力。这对于资源有限的用户或者设备是非常不友好的。同时这个算力除了用于工作量证明的寻找并没有其他的效益，这使得能源浪费十分严重。由于基于工作量证明的共识算法只具有弱一致性，因此敌手可以通过在同一个区块的不同分叉上挖矿发起双花攻击来使自己受益。工作量证明共识算法种这种攻击只能通过降低敌手的攻击意愿，并没有比较好的方式能够完全杜绝。

### 2.1.2 基于权益证明的共识算法

基于权益证明的共识算法的出现主要是为了解决工作量证明中能源消耗巨大的问题。基于权益证明的共识算法中节点或用户根据拥有资产的比例来决定成为下一个出块者的概率。节点生成区块不需要消耗大量的计算算力，也就没有较大的能耗，对环境比较友好。

由S.King和S.Nadal[13]提出的点点币首次引入了权益证明的概念。首先通过一定阶段的PoW生产出一定数量的代币，随后根据代币的币龄选择出块的节点。节点代币越多、持币时间越长则获取出块权限的概率则越大。这类区块链共识算法可以快速生成区块，因此具有较大的交易吞吐量和较低的交易确认延时。但是，基于权益的共识算法面临着节点长期离线和敌手攻击的威胁，主要有无利害关系攻击[21]、打磨攻击[22]、长程攻击[23]、权益窃取攻击[24]等。此外，由于PoS中节点的权益信息是可以被获知的，敌手可以通过这些信息来偏置出块节点的选取使得自己获益。

基于PoS的共识算法面临着中的无利害关系攻击、离线攻击和长程攻击的威胁。针对两个问题，V.Buterin等人[14]提出的Casper FFG是以太坊的基于PoS共识算法。其中区块产生仍然依靠以太坊的Ethash工作量证明算法,但是每隔50个区块出现一个检查点,验证者通过PoS的方式来对检查点完成最终确定。共识算法中的检查点机制解决了PoS共识算法中的长程攻击。对于无利害关系攻击，Casper FGG采用质押机制来解决。验证者需要将持有的以太币质押到系统的智能合约中，节点被选中的概率与其质押的以太币成正比。如果发现参与者想要发起发利害关系攻击，就没收其保证金。对于PoS共识算法中节点长期离线的问题，也采用罚没一定数量的代币来确保节点愿意保持在线，确保系统的安全性与活性。，Casper FFG采用惩罚机制使得节点有长期保持在线的理由，从而确保安全性。

针对PoS共识算法中节点长期离线问题和交易效率低的问题，委托权益证明(delegated proof of stake, DPoS)[20]采用委托人的方式对确保区块生成的可信性以及达成共识的高效性。权益持有者投票给信任的委托人,票数与其权益大小成比例关系,最终选出101名委托人以平等的权利轮流作为区块生产者行使权利。如果委托人在委任期间未按规则产生正确区块,该委托人将会被除名,所有权益持有者选出新的超级节点将其替代。这个共识算法消除了交易需要等待一定数量的区块被非信任节点验证的时间，降低了交易确认延时。同时，委员会机制也避免了参与者在不同分支上生成区块的无利益关系攻击和长程攻击。DPoS以选举委托人的形式实现共识,但是带来了严重的中心化问题。

为了提高PoS共识算法处理交易的效率和提高共识算法对于打磨攻击的抗性，由A.Kiayias等人[15]提出的Ouroboros,利用形式化的方法建立了PoS共识机制的模型,并证明了Ouroboros 能够满足安全性。该共识算法将时间分为多个时期，每个时期分为多轮，每轮最多产生一个区块，如果出块节点不在线则在这一轮不产生区块。在每个时期开始，系统根据节点所持有的股份选出这个时期的出块候选人，并且随机的选举出下一个出块人。Ouroboros 通过使用公开可验证秘密分享算法的安全多方计算协议生成的随机数在存在敌手的情况下具有高抗偏置性（抗打磨攻击性）。通过选出的多个背书节点对交易进行验证并打包给出块者可以提高共识算法的无利益攻击的抗性。Ouroboros的激励机制是将多个区块的交易费根据参与节点的贡献度按比例分配，这个机制激励权益持有者愿意保持在线获得收益，从而解决节点长期离线问题。

为了防止Ouroboros共识算法中敌手对出块者发起的贿赂攻击或DDoS攻击，B.David 等人[16]提出的Ouroboros Praos改进了Ouroboros 中出块者的选举方式。在这个共识算法中，节点私下确定是否自己被选为出块者，节点之间不能提前判断其他节点是否被选中,直到出块者成功将区块生成，这样有效防止了敌手做坏的可能。对于Ouroboros中PoS共识算法中新加入节点的安全自启问题和长程攻击问题，Badertscher 等人 [18] 提出了Ouroboros Genesis。该共识算法中新加入的节点通过从不同的节点获得多个链做对比来最终确定与其他链的共同前缀且是最长链。Ouroboros Genesis利用公共前缀的性质在没有采用检查点机制的前提下能够抵抗长程攻击,并且在通用可组合(universally composability)模型下, 形式化证明了协议的安全性。

对于PoS算法中节点的离线、敌手选举出块候选者偏置以及敌手腐蚀候选者等问题，P.Daian等人[19]提出的Snow White是适用于PoS的可重配置共识算法,重配置的间隔时间短暂,能够满足节点随机加入和退出网络的需求。通过重配置过程选出系统中最近的权益拥有者作为候选出块者集合,然后按集合中成员权益占比随机选择出块者。这个候选人重配置机制即使节点离线也能够确保共识的安全性。Snow White中出块候选者集合随着系统中权益的变化而重新选择,防止敌手的后来腐化 (posterior corruption) 攻击。重配置过程的随机性也确保了系统的抗敌手偏置性。通过将几个区块的奖励和交易费用平均分给出块者确保了节点参与共识的活性和公平性。同时，采用了类似于水果链（Fruitchains）中区块和水果生成机制提高区块链的交易处理效率，提升区块链系统的交易吞吐量。

基于PoS的共识算法虽然解决了PoW共识算法能耗高的问题，但是却由于生成区块不需要大量耗能而面临着敌手攻击的危险。由于PoS也是一个具有弱一致性的共识算法，因此双花攻击是共识算法中最常见的攻击。PoS中的双花攻击通常可以利用无利益关系攻击、长程攻击和打磨攻击来实现。当前的工作主要通过押金机制来降低无利益关系攻击，通过检查点机制和单链区块链公共前缀来降低长程攻击，通过随机数机制来提高系统的抗打磨攻击性。新节点自启机制则可以根据单链区块链公共前缀来保证安全。虽然这些机制可以一定程度上确保PoS共识算法的安全性，但是该算法的弱一致性还是让系统面临着分叉的安全威胁。为了提高共识算法的交易处理速率，通过增加生成单个区块处理交易的数量，也可以通过降低交易确认时延。但是这样的方式能够提升的交易吞吐量是非常有限的。

### 2.1.3单一委员会混合共识算法

混合共识算法的含义是将经典分布式共识算法与区块链共识算法相结合，即采用PoW或PoS的方式选举特定的委员会，在委员会内部运行经典分布式共识算法生成区块。混合委员会共识算法又分为单一委员会共识算法和多委员会共识算法。采用单一委员会的混合共识算法选举一个委员会负责全网所有交易的处理， 而采用多委员会的混合共识共识算法将选举多个并行运作的委员会，将全网划分为多个片区，分片处理网络中的交易。混合共识算法的一般流程是：选举委员会成员、选举委员会首领、执行委员会内一致性算法、广播区块、更换委员会视图。

C.Decker等人[25]提出的PeerCensus首次将经典分布式一致性算法PBFT 与PoW共识算法结合，利用比特币作为底层链，选出一定数量的节点完成身份认证后，通过PBFT来实现链一致算法完成最终区块的生成。该共识算法能够通过身份认证来抵抗女巫攻击，并且最终能够达成强一致性共识。由于节点的进入和离开都需要委员会成员达成共识，恶意用户会利用PeerCensus的离开检测机制通过不断制造离开提议的方式降低整个系统的运行效率。单一委员会混合共识算法需要解决以下问题

1. 新节点安全自启问题 在不依赖可信启动来初始化委员会配置的问题。新节点加入时时需要解决节点快速安全的自启问题，节点获取交易信息真实有效以及识别虚假信息问题。
2. 委员会选举 在选择委员会成员的时候需要解决委员会中诚实成员达到一定比例问题。委员会内共识的运行需要由领导者发起，并且采取一定的机制防止领导者不作为或恶意行为。
3. 委员会一致性算法 委员会内需要解决客户端提交交易快速响应问题，提高委员会内一致性算法运行效率问题，降低委员会通信复杂度和计算复杂度问题，以及并行提议实现问题。
4. 委员会重置 一个委员会工作时间对应一个时期，共识算法需要合理设置每个时期的长度，防止敌手腐蚀委员会成员。在一个时期结束后，委员会重配置需要解决委员会成员交替时交易的处理问题和敌手干扰委员会新成员选择问题。重置委员会通常是只更换部分成员，随后进入新时期。

为了解决敌手恶意影响委员会重置和腐蚀委员会成员来降低委员会效率，最终影响系统的安全性和性能，E.Kokoris-Kogias等人[26]提出的ByzCoin采用滑窗机制来实现委员会重置。该共识算法采用PoW的方式防止敌手发起女巫攻击，同时区块链的数据结构采用Bitcoin-NG的思想来提升交易处理效率。委员会成员采用滑动窗口的方式重置委员会确保委员会中成员不会快速变化。通过找到最近的144或1008个关键区块的节点进入委员会，节点的投票权由关键块的比例决定。委员会采用群体签名的方式利用默克尔树结构对每个消息的节点签名排列，最终对微块达成共识。但是这个机制没有解决初始委员会配置的问题。相较于PeerCensus共识算法，该算法中节点的进出不用委员会成员协商，提高了效率。但是这个算法没有解决节点自启的问题，也没有解决初始委员会配置问题。同时委员会成员的更新速率取决于工作量证明，难度高会更新比较慢且容易被敌手腐蚀。这个算法也没有解决委员会成员更换时交易的处理问题。

为了解决了委员会重配置时正在进行交易处理的问题，I.Abraham等人[27]提出的Solida共识算法。该算法采用Paxos思想，首先为每个领导者赋予不同等级，新节点加入委员会重配置进入委员会之后等级增加，并成为新的领导。Solida共识算法委员重配置是利用PoW机制进行随机更新。PoW共识机制可以防止系统中节点女巫攻击的优越性。当节点找到一个PoW时，该节点的生存时期增加并成为领导。一旦出现新领导，当前领导任期被打断，经过委员会共识之后，新领导接管委员会，并且对之前打断的提案重新发起共识。当一个节点长期成为领导时，在委员会中选出新的领导更换视图。为了防止自私挖矿带来链质量的下降，Solida在PoW中嵌入了随机数，将每一轮提议最终得到的多个签名嵌入到 PoW 中，保证诚实用户的链质量。但是这个共识算法中委员会成员的更新速率比较慢，这会使得委员会成员容易被敌手腐蚀。对于初始委员会的配置问题也没有解决，新节点的安全自启问题也还未解决。

为了解决初始委员会配置问题和交易的响应速度慢的问题，R.Pass和E.Shi[28]提出Hybrid consensus将PoW与BFT算法结合的混合共识机制,采用滑窗机制更换委员会成员。Hybrid consensus 共识算法首先利用Snailchain选出一个初始静态委员会，诚实节点会根据其局部链去掉尾部中不稳定的个区块，并选择前个区块的出块者作为BFT委员会成员。为了解决自适应腐败攻击，采用滑窗机制轮换委员会成员的方式。当诚实节点的链达到长度时第R个委员会成员也是去掉尾部k个不稳定区块，选择最后个区块的出块者成为委员会成员。这使得敌手无法干扰委员会成员的选择。第个委员运行BFT协议来提交交易直到收到停止命令。因此委员会成员会逐渐输出已提交的交易。提交的交易会被记录在每日日志中。该算法首次利用形式化的安全模型和模块化的设计建模混合共识机制，并证明了其能够满足一致性和活性等安全特性。这个共识算法中委员会成员的更新速率比较慢，这会使得委员会成员容易被敌手腐蚀。新节点的安全自启问题也还未解决。采用该共识算法的区块链也容易出现分叉，最终导致系统性能低。

为了解决采用PoW和一致性算法结合的共识算法中能耗高、委员会更新时间长、交易处理效率低的问题， Y.Gilad等人[29]提出了Algorand将PoS与经典分布式一致性算法结合的混合共识算法。Algorand共识算法采用随机更换机制来重置委员会成员，并且采用随机可验证机制选择委员会首领。首先根据上一轮的区块信息计算本轮的随机数，节点根据随机数判断自己是否被选中。设置两个不同的值判定自己是否选中成为首领或委员会成员。委员会成员内部执行新的双层共识的拜占庭一致性算法⋆ 来解决敌手腐蚀问题。每一轮投票的成员是利用PoS 共识算法在所有权益持有者中随机选择一定数量的委员会成员, 相比于 PoW，PoS选择成员的方式更加快速、高效, 因此可以被 Algorand 用于委员会的快速更新。Algrand具有较高的交易吞吐量和扩展性，可用于规模较大的应用场景。该共识算法具有强一致性特性,区块链分叉概率很小,交易能够得到较快速度的确认。

针对PBFT一致性算法固有的二次消息复杂性而在网络中的副本数量方面存在可扩展性问题，M.M. Jalalzai等人[48]提出了一种新的基于 BFT 的共识协议Proteus。该共识算法选择一个大小为的节点子集作为根委员会保证稳定的性能。无论网络中的故障数量如何，它都将典型的基于 BFT 的协议的二次消息复杂度提高到 ，Proteus 确保了与网络故障数量无关的一致性能，并提高了二次消息复杂性。

针对单一委员会中领导者选取的偏置性高，并且对首领的恶意行为缺乏验证的问题，P.Berrang等人[49]提出一种新的基于推测BFT一致性算法和PoS的共识算法Albatross。Albatross 是一种混合权益证明 (PoS) 算法和投机性 BFT 一致性算法的共识算法。Albatross通过选择与权益成比例的验证者列表使其无需许可；通过仅使用 随机可验证函数选择前一个块上的块生产者来增加对自适应对手的抵抗力； 靠块的投机执行来提高达成一致性性能。共识算法通过大块和微块机制来提升处理交易的效率，提高区块链的交易吞吐量。通过验证者机制检测节点的的恶意行为，通过更换视图和惩罚机制来确保共识算法的安全性。

单委员会混合共识算法将经典分布式一致性算法与非授权共识相结合,通常区块链通常具有强一致性，能够极大概率的降低区块链的分叉风险。采用基于PoW的滑窗机制或者随机机制重置委员会具有更新缓慢，委员会成员容易被腐蚀。采用基于PoS机制的随机重置委员会更新迅速，能够抵抗敌手的腐蚀。但是对于共识算法如何安全的启动初始委员会还是需要解决的问题，可以采用安全多方计算等算法。新节点进入系统如何安全快速的自启也是需要解决的问题。如何确保委员会诚实成员达到一定比例也是需要重点考虑的问题。

### 2.1.4多委员会混合共识算法

为了解决区块链处理交易的可扩展性,利用多个并行的委员会来处理网络中不同分片的交易的混合共识算法被提出，也被称为分片共识算法。目前的分片分为通信分片、计算分片以及存储分片。

1. 通信分片 将全网分为不同的片区，每个片区由一个对应的委员会处理。每个委员会内部成员大部分时间只需内部通信，每个分片内部的其他节点可以通过与该分片内委员会通信获得目前区块链的状态。
2. 计算分片 每个分片委员会只负责处理其对应的交易，对交易运行委员会内一致性算法,验证该交易的合法性,最终决定该交易是否被添加到区块链中。计算分片使不同的交易以并行的形式被不同的委员会处理,当网络中节点数量增多时,可以增加更多的委员会,这样交易处理性能随着网络中节点数量的增多而增加, 进而实现了交易处理的可扩展性。
3. 存储分片 在不同分片委员会将处理后的交易分片存储,每个分片委员会只负责处理本分片对应的交易,将交易放到本分片专属的交易区块链上。存储分片将整个区块链系统的交易数据或未花费的交易输出数据分片存储,降低了节点的存储负担。如果采用存储分片,那么可能每个分片各自生成和广播其区块链,不存在全局的区块链。

为了提高交易处理的效率，L.Luu等人[30]提出ELASTICO通过分片并行处理交易的方式提升交易处理的效率。该共识算法将参与共识的节点分为多组，每组委员会在一轮共识中会输出一个区块，通过最终共识委员会整合输出一个总区块。ELASTICO并行处理交易的方式提升了区块链系统的性能。该算法通过工作量证明和随机数的方式选择出一个时期内的共识节点，从而避免女巫攻击。通过目录委员会机制来为多个委员会分配成员，委员会内部都运行PBFT一致性算法，每个分片委员将其内部共识结果附上签名发送给最终委员会，最终委员会内部也运行PBFT一致性协议对最终总区块达成共识。最终委员会通过随机数生成协议为下一轮的共识节点的选择增加可验证性。但是这个共识算法抗偏置性差、并且无法实现跨分片交易。分片共识在实现交易处理可扩展性的同时,引入了一些新安全性问题的问题。

1. 分片委员会配置 委员会选举如何确保一定比例的诚实成员，以及分片委员会的合理高效的分配问题。解决新加入成员分配到不同委员会的过程受到敌手的偏置的问题,避免委员会冲恶意成员个数超过限制使得敌手可能控制委员会。因此,为了防止敌手在此过程中影响成员分配,需要设置合理的分配策略。针对委员会中的恶意行为是很难发现的，因此需要解决快速检测恶意委员并迅速恢复或者重置委员会的问题。
2. 委员会一致性算法 需要解决分片委员会最终结果的整合问题，以及委员会共识的高效性问题。同时也需要设置安全的委员会首领的选举算法，当值首领不作为或恶意行为。委员会分片共识中,当不要求每个分片委员会数目相同时,当两个委员会之间交流过于频繁时,可以考虑将其合并为一个委员会,提高交易处理速度等。
3. 跨片交易 需要解决多个分片共同协作安全和高效地处理。在过程中需要解决双花攻击、重放攻击[37]和交易锁死等问题,并且需要一定的方式提升交易的处理速度。在运行委员会内一致性算法步骤中,当一个交易包含多个输入且其属于不同分片时,需要多个分片协作完成对该交易的处理, 防止交易双花攻击。

针对ELASTICO中分片共识中抗偏置性低以及无法跨片交易的问题，E.Kokoris-Kogias等人[31]提出的Omniledger。该算法提出了跨片交易防锁死的解决方案。该算法通过密码学抽签的方式确定首领，并通过RoundHound算法生成本轮随机数作为随即置换函数的种子对委员会成员进行分片，提高了委员会分配的抗偏置性。委员会内执行一致性算法处理分片内部的交易。对于委员会重置问题，Omniledger设置合理的挖矿难度来部分替换委员会中的节点，被替换节点采用本轮随机数来进行选择，这样可以提高系统容错率和抗偏置性。针对跨分片的交易，Omniledger采用UTXO模型，网络中不同分片的节点只需处理和存储该分片对应的UTXO数据，利用原子性跨片交易防止跨片交易锁死。为了更好的防止敌手使得委员会选举出现偏置问题，文献[32]对Omniledger使用的随机数生成算法改进,采用公开可验证秘密分享，确保了随机数的不可预测性、抗偏置性和公开可验证性,并且保证了随机数的持续生成。但是这个共识算法的身份确认采用的工作量证明算法，使得每一轮委员会的重置会耗时较长，具有被敌手腐蚀的危险。同时，基于工作量证明的身份证明需要耗费大量算力，能耗巨大对环境不友好。

分片共识算法除了应用在电子加密货币上，也可以扩展到智能合约应用上。M.Al-Bassam等人[33]提出的Chainspace，在分片的基础上构建了智能合约应用平台。针对交易和智能合约跨片处理的问题，Chainspace通过将智能合约的执行和验证分离，提出能够解决跨片交易的BFT协议和原子提交协议结合的原子提交协议。Chainspace实现了交易和智能合约的通信分片和计算分片，在确保其安全性的同时提供较高的交易吞吐量。但是这个并没有考虑委员会的选择和分配问题，也没有考虑委员会重置问题，更没有解决节点的身份验证问题，也没有解决交易存储分片的问题。

针对节点存储有限的问题，M.Zamani等人[34]提出的RapidChain很好实现了计算分片、通信分片和存储分片大大的提高了性能。共识算法主要包括启动 (bootstrap)、共识和重配置。对于初始委员的选举和创世块的生成RapidChain采用随机图（Random Graph）的方式选出根群组(Root Group)，随后利用根群组运行分布式随机数生成协议生成目录委员会，目录委员会再负责生成其他的分片委员会。该共识算法分时期运行，每个时期又分为多个轮，委员会内部每一轮采用使用同步拜占庭容错协议。对于每个时期委员会重置，为了防止自私挖矿攻击，节点通过加入随机数的工作量证明加入委员会。系统采用有界布谷鸟规则(Bounded Cuckoo Rule)尽量确保新节点替换掉委员会中不活跃的旧节点。该共识算法在网络层中采用消息扩散算法(Gossip Protocal)能够将区块传播速度提升。RapidChain 对于跨分片交易采用将交易拆分的方式，并再对应交易所属的分片中达成共识，最终根据共识结果分片存储。这个方法实现了分片通信、分片计算和分片存储。但是由于每个时期中委员会内部的共识采用的是同步网络模型，这将导致交易确认不能满足快速响应的特性。并且

针对委员会成员选择和分配过程的抗偏置性，分片委员会内共识算法效率低下，跨分片事务处理会花费昂贵的系统开销等问题，Y.Liu等人[46]提出了一种新颖的安全可扩展混合共识算法SSHC。该算法为遴选委员会成员提出了一个公平的分片选择方案；针对分片内共识设计了流水线拜占庭容错，将流水线技术与阈值签名相结合，提高共识效率；提出了一种响应式分片事务批处理机制来处理跨分片事务，从而减少了对拜占庭容错算法的调用次数；设计了一种安全的委员会重新配置方法来有效地更新分片成员。SSHC采用正式的安全框架来设计和分析分片区块链，证明SSHC可以实现一致性和活性。

针对随机分片系统的安全性和吞吐量不够高，合作的激励机制的开销大的问题，C.Huang等人[47] 提出了基于声誉的安全快速的分片区块链共识算法RepChain，为激发节点合作提供了高激励。通过一种新的双链架构——交易链和信誉链。对于交易链，提出了一种高效的基于 Raft 的同步共识。对于信誉链，结合集体签名的同步拜占庭容错共识被用来防止对信誉评分和相关交易块的攻击。该共识算法支持具有中等生成速度的高吞吐量交易链。RepChain提出了一种基于声誉的分片和领导者选择方案，可以有效提高分片中诚实成员的比例，并且防止领导者不作为或者恶意行为。

针对PBFT一致性算法通信复杂，扩展性低的问题，W.Li和M.He[50]提出了一种分层和基于组的 BFT 共识算法——高效 BFT（EBFT）。 这个一致性算法利用一种新颖的网络拓扑结构来有效地减少节点之间的通信时间并提供的通信复杂度。通过将PBFT 算法中未分化的节点划分为不同的层和组，并提出分层和基于组的区块链模型。然后在这个模型的基础上修改了PBFT的共识协议，采用签名机制减少节点之间的通信提高性能。

多委员会共识算法通过分片并行处理交易来提升区块链的交易吞吐量。这类共识算法相较于单一委员会共识算法引入新的问题需要解决。主要是跨分片交易处理问题、委员会成员动态管理问题以及恶意委员会检测与恢复问题。如果新加入成员分配到不同委员会的过程受到敌手的偏置,则可能造成某些委员会中恶意成员个数超过限制,敌手可能进一步控制委员会。如何检测恶意委员会并建立恢复机制是多委员会共识算法需要研究的重要问题.

### 2.1.5其他共识算法

针对工作量证明能耗高对环境不友好的问题，一些人也研究了新的能耗低的身份证明算法。A.Miller等人[38]提出了Permacoin的文件存储证明共识算法。该共识算法要求参与者有能力存储大文件的一部分。通过一个挑战应答机制来确定节点是否存储了文件部分。如果应答成功，则该节点成功获得奖励。节点最保险的方式是存储整个文件，从而就会确保挑战成功的概率。但是这个共识算法需要存储资源，因此节点不可以通过伪造身份来获益，具有抗女巫攻击性。

针对文件存储证明共识算法需要消耗大量的存储空间问题，S.Park等人[39]提出Spacecoin,采用非交互式空间证明(proof of space)达成共识。该算法以参与者能够使用的硬盘空间大小作为标准,选出区块的生产者的容量证明共识算法节点通过一次性付出的硬盘空间来获的出块权限。为了防止硬盘空间造假，节点出自加入网络的时候根据选择的存储空间大小，存储一部分具有特定序列的数据。校验节点验证时向证明者发送一个挑战，证明节点根据挑战信息生成对应数据的哈希值给校验节点。校验节点再根据返回的值判定证明者是否硬盘空间造假。可以通过多次验证的方式来避免证明人的作弊行为。这个校验节点存储空间的方法在原来的文件存储证明的机制上加入了随机性，使得节点共识过程更具有公平性。但是存在无利益攻击、长程攻击和双花攻击的风险。

针对之前的身份证明共识算法都具有资源消耗大、安全性低等问题，消逝时间证明[40]共识算法应运而生。这个共识算法是基于硬件芯片执行某个命令的等待时间来实现的,其实质是利用可信硬件产生随机数来决定下一个区块生产者。参与者在发布块之前都需要向Intel SGX中的飞地获取一个随机的等待时间,等待时间最短的节点被选为领导节点。这个算法可以提高交易吞吐量，减少能耗，并且具有很好的随机性，但是需要可信硬件的辅助，一旦硬件出现问题，则共识过程将不再可信。

针对PoW共识算法能耗高，性能低以及PoS安全性低等问题，D.Schwartz等人[41]提出了Ripple共识算法，通过信任节点列表分别对交易和区块达成共识。在确保交易安全性的前提下提高性能。目前的交易确认时间只需要4s，吞吐量可高达1500TPS。但是这类共识算法的容错率比较低，并且通信复杂度比较高。

## 2.2 DAG区块链共识算法

单一主链的区块链系统的交易吞吐量的提升是非常有限的，这类区块链容易出现链分叉从而影响系统的安全性和扩展性。为了提高区块链吞吐量，新的链存储结构—有向无环图（DAG）应运而生。DAG区块链中的每个单元代表用户发起的一笔交易, 每一条从子单元指向父单元的有向边代表一种验证关系,即用户创建交易的时候需要对先前的交易单元进行验证,将验证有效的交易单元的哈希值包含到自己的交易数据结构中,则当前的交易单元称为子交易,被验证的交易称为当前交易的父交易,每笔交易可以有多个父交易,每个交易也可以有多个子交易, 这样交易单元构成有向无环图的结构。历史交易数据不可篡改,一旦更改,则会引起整个DAG区块图的变更. DAG区块链能够实现交易处理的可扩展性,随着网络中节点的增多,交易处理速度提高,交易吞吐率增加。

基于DAG区块链的项目有IOTA[43]，Byteball(Obyte)[44]，Hashgraph[45]，Nano(RaiBlocks)[42]等。常见的DAG区块链共识算法有Tangle共识算法，见证人共识算法，基于DPoS的共识算法。这些共识算法在为系统提高了交易处理性能的情况下，也面临着一些列的安全性问题。

1. 节点自启机制 区块链系统如何确保新加入节点的可信性。如何确保节点能够快速获取可信区块链账本。由于基于DAG区块链的系统具有交易并发性，因此每个节点维护的账本是不同的，因此如何确保新加入的节点获取的是最新的账本是需要解决的问题。
2. 双花攻击问题 由于DAG区块链中允许分叉，因此容易被敌手利用发起双花攻击。共识算法需要对双花交易中的有效交易达成一致。
3. 激励机制 这类区块链通常没有挖矿费用，因此需要解决节点由于没有收益而不愿意生成交易参与共识的问题。
4. 长程攻击问题 由于节点生成交易并不需要耗费非常多的资源，因此节点可以通过伪造大量交易来确保自己的交易最终被确认。这些问题都会为DAG区块链带来演变中的安全问题。

### 2.2.1 Tangle共识算法

Tangle[43]是IOTA系统中采用的共识算法。在Tangle中引入权重的概念，每个交易都有一个自权重。交易的权重会随着之后直接或间接支持它的交易数量的增加而变大。当一个交易累积权重足够大，满足系统预设的确认阈值时，该交易将显示确认状态。理论上,如果整个网络中有足够多的交易流,则这个确认过程可以快到在几秒钟内完成。该共识机制的创新之处在于不再采用网络中的一个子集(如矿工)来专门负责维护共识,而是全网所有的参与者都来进行网络交易的验证工作。Tangle算法实现共识的过程与交易产生过程是一体的,共识算法在运行的同时处理交易可以极大的提高交易处理效率。

Tangle共识算法可以在没有任何交易费用的情况下极大的提高交易处理效率，但是会面临着一些新的安全问题。由于Tangle共识算法具有弱一致性，因此面临着双花攻击的风险。在Tangle中，交易的生成并不需要耗费资源，如果用户能够产生大量交易就可以随意控制区块链系统的共识。此外，没有交易费用就没有激励机制，这类区块链系统的活性很难保证，并且面临拒绝服务攻击和垃圾信息攻击。这个共识算法适用于具有大交易量的系统，一旦交易流变小就可能无法确认旧交易，最终影响安全性和性能。此外，新节点的安全自启机制也没有相应的解决方案。

### 2.2.2 基于见证人的共识算法

针对区块链中最容易出现的双花问题，Byteball[44]采用基于见证人的共识算法，通过建立与主链相关交易的全序来处理交易双花的问题。见证人通常是系统中长期实名且具有较高声誉的节点，他们维护系统并自愿频繁的发起交易单元，对于消极的见证人，可以被替换掉。最优父单元的选择是根据单元级别、见证级别以及交易的哈希值来确定的。通过选择最优父单元可以找到一条从任意顶端单元出发到创世单元的最优路径作为一条候选主链。多条候选主链会在某个交易单元处相交，这个相交的交易单元是一个稳定点。在每个节点中，从稳定点到创世单元的路径是相同的，这部分主链称为稳定主链。所有与稳定主链相关的交易单元可以进行一个全局排序，为每个单元分配一个主链号。最终建立在DAG区块链上的主链总序，使得全网达成共识。DAGCoin[17]是建立在Byteball上的区块链系统，因此采用相同的共识算法。但是Byteball中主链算法和见证人发布交易的频率有关系，交易确认的时间是不确定的。

为了解决IOTA和Byteball中低频次交易使得旧交易最终无法及时被确认甚至可能永远不会被确认的问题，TrustNote[35]采用双层共识的机制来解决。通过PoW的机制选择出公证共识的节点，公证节点将发布公正单元到DAG区块链上。公证共识根据公证单元严格确定其他单元的顺序。当公证单元成为主链上的稳定单元之后，公证节点才能获得公证奖励。通过找到一条以创世单元为起点的主链，并为相关交易根据主链分配主链号，建立一个主链全序来解决双花问题。为了避免出现垃圾交易，该共识算法要求所有节点提交新单元时支付交易费用。交易费用分别被支付给其最小哈希子单元和公证单元的节点。通过这个激励机制还可以吸引更多节点维护账本，确保系统活性。此外也激励节点尽可能引用最新的在靠近主链的父单元，有助于减少分叉。但是基于PoW的TrustME共识机制需要耗费大量的算力资源，并且生成单元比较慢效率不够高。并且这个机制还面临自私挖矿攻击的危险，需要方法来防止计算资源集中化，使得算力较小的见证节点缺乏竞争力的问题。同时，还需要设置一个合理安全的节点自启机制来确保新节点获取信息的安全性。

### 2.2.3 基于DPoS的共识算法

Nano[42]是一种基于区块点阵(Block Lattice)结构的新型加密货币，其共识算法采用了一个用户一条链的方式，只记录自己的交易，也只有自己可以修改自己的记录，不与其它帐户共享数据。从而可以使交易都可以并行执行，能提供秒级的交易速度和无限可扩展性，并且允许他们异步地更新到网络的其余部分，从而以极小的资源开销获得快速的交易确认。

Nano中货币流通时，需要两个交易：货币发送交易和货币接收交易。交易数据的收发可以是异步的，因此允许多笔金额同时汇入一个账户。若出现节点离线，未到账的金额会被标记，直到节点在线时完成即可。针对交易分叉问题，Nano使用的DPoS共识算法允许账户指定代表为其投票，最终得票最多的代表会广播分叉并根据高权账户在固定时间内的投票结果确定哪个分支是有效的。同时Nano中也使用了PoW共识机制，但是主要是用于交易确认。此外，网络分离后重新加入时可能会影响投票过程和结果。每个节点维护自己的账本可以很好的避免自己账本被篡改的风险，如果敌手篡改自己的账本会立即被其他节点检查到，使得篡改无效。这个就不太需要设置节点账本自启机制，可以确保节点加入系统的安全性。但是需要设置安全获取系统其他节点身份信息的自启机制，确保能够获得可信的身份信息。基于DPoS共识算法存在中心化程度高的缺点,而其中的PoW共识算法则会耗费计算资源，会面临自私挖矿攻击的危险。

### 2.2.4基于虚拟投票的共识算法

Hashgraph[45]共识算法使用Gossip协议[36]使得节点可以随机选择另一个节点传播自己的事件，通过对事件顺序和事件的时间戳达成共识达成最终共识。该共识算法采用虚拟投票机制和证人事件机制对事件的顺序达成共识。随后利用见证事件的顺序，根据事件的时间戳去中位数对非见证事件进行排序，最终实现所有事件的全局排序。这个全局排序很好的解决双花攻击的问题。Hashgraph是一种快速、公平和安全的共识算法，开创性的做异步BFT共识不会引发突发大规模消息传递风暴。

Hashgraph利用Gossip协议和虚拟投票将共识所需的通信要求降到了最低，本地计算也保证了共识的高效性。但是这个算法的共识过程比较复杂，至少需要两轮才能确认事件，并且还未在大规模公链环境下运行，安全性还没有得到具体验证。同时，还需要设置节点的身份证明机制，确保节点身份的可信性。这个共识算法还需要相应的激励机制来确保节点的活性以提高区块链系统的安全性。

# 3课题主要研究内容、预期目标

本课题的研究目标是在节点资源有限和动态变化较大的区块链网络中，以能耗小，提高交易处理效率为目标，设计安全高效的区块链共识算法，主要内容包括以下几个方面：

1. 基于节点稳定度的单共识节点的区块链共识算法

考虑节点资源有限、并且有可能快速离开共识过程，以设备能耗低和快速达成共识为设计目标，设计一个高效的单链式区块链共识算法。

1. 基于稳定委员会的区块链共识算法

针对单共识节点算法中交易确认延时高，并且单个共识节点不稳定造成某一时期内节点无法达成共识的问题，以低能耗、快速处理交易为设计目标，设计高效的稳定具有单一委员会的单链式区块链确定共识算法。

1. 基于节点稳定度的分片区块链共识算法

针对节点数量规模大，分布区域广泛，节点进出系统频繁的情况，以降低能耗、高稳定性，快速处理交易为设计目标，设计低耗能、高吞吐量的多委员会区块链共识算法。

1. 基于委员会见证的DAG区块链共识算法

DAG区块链非常适用于大规模的具有资源低，动态性高的节点的场景。这种区块链具有非常好的去中心化和高扩展性的优势，但是其安全性则备受考验。针对静态见证人容易被敌手腐蚀，见证人列表重置的易偏置性问题，以低耗能、抗偏置性为设计目标，设计可信、稳定的DAG区块链见证人委员会共识算法。

# 4拟采用的研究方法、技术路线、实验方案及其可行性分析

## 4.1技术路线方案

### 4.1.1基于节点稳定度的单共识节点区块链网络共识算法

区块链共识算法需要满足平衡好去中心化、安全性和可扩展性三者之间的关系。单共识节点的区块链共识算法具有非常耗的去中心化性和安全性。单共识节点的共识过程是节点获取信息、选出获取出块权限、生成区块，最终广播验证达成共识。考虑资源有限、节点动态高的环境下，达成共识需要以稳定性和低能耗的目标，快速稳定的使得所有节点状态达成一致。然而，由于节点的高动态性，节点在共识的过程中可能出现提前退出正在共识的进程，可能导致在这个节点时间内无法达成共识，造成资源和时间的浪费。此外，直接采用通过消耗物理资源获得出块权限的共识算法对于资源有限的节点也是不现实的。为此，在设计单共识节点的区块链共识算法时，需要考虑以下问题：第一，如何使共识算法能够抵抗女巫攻击的同时没有非常大的物理资源消耗。第二，如何使得敌手在诚实节点出块之前无法知道出块节点的信息，防止出块节点被腐蚀。第三，无能耗的出块通常会面临无利害关系攻击。当链出现分叉时，敌手可能为了增加自身获利的可能性，从而在每个分叉上都添加区块。第四，新节点或者离线节点加入网络时，敌手可以伪造一条从创世块到最新块的区块链。敌手可以贿赂节点获得节点之前使用过的重要私钥，不需要太大的成本就可以伪造假链，从而欺骗新节点，达到其目的。当节点具有很高的动态性，可能随时进入网络或者离开网络。但是有些节点在一定时间内是不会离开网络并且会比较频繁的参与到共识中。保守来看，选择这类稳定的节点作为共识节点更能确保区块链共识过程的正常进行。因此，拟结合节点的稳定度和历史上参与共识频率，设计一个更加稳定、低能耗、高效的区块链共识算法。

研究方案：

1. 网络建模：考虑在一个区域内网络节点可以随意进出，这些节点的计算资源和网络资源是有限的，每个节点具有不同的的移动速率和移动方向。节点之间可以通过发送无线信号通信，每个节点的通信半径是相同的。每个节点既可以成为共识节点，也可以只作为一般节点，并且局部地维护自己的区块链。当节点需要维护两个区块链，一个是身份链一个是交易链。身份链主要是每个节点新加入网络时为获取活跃时间支付押金的交易区块构成（或许身份链的维护可以采用其他的身份验证节点），交易链主要是由包含交易的区块构成。节点加入系统时，通过随机游走，比对好几个节点的身份链和交易链，节点最终复制了具有相同前缀且最长的交易链。
2. 共识节点选取：区块链系统中，新节点进入网络之后，需要质押金钱来获得在这个区块链系统中活动的时间，根据活动时间的长短来选择质押金额的多少。随后根据节点在这个系统中的剩余活跃时间和节点在最近个确认区块中参与共识比值来选出具有较好稳定性和可信性的节点作为共识节点。计算公式为
3. 基于稳定度的共识算法：区块链的共识过程分为两个步骤：第一步是根据当前的网络情况选择共识节点，生成区块，第二步网络中其他节点验证区块的合法性，最终达成全网共识。节点产生交易之后，将会广播交易到网络中。接收到交易的节点会验证交易的有效性，并将交易放入自己的未处理交易池中。共识节点的选择规则是根据节点的稳定度，具有最高稳定的节点会被其他节点接受作为共识节点。共识过程如下：
4. 系统中节点根据自己剩余时间和最近个确认区块中参与共识比值计算所有节点的稳定度。具有更高稳定度的节点将会有更高的概率成为共识节点。为了避免共识节点的选择过程被敌手偏置，会在选择过程中加入一个随机数进行扰动，并且以最后被确认的区块作为种子通过随机生成函数来生成一个随机数来确定自己能否获得区块生成权限，成为这一轮的共识节点。这样确保共识节点不会在生成区块之前被敌手腐蚀。
5. 当节点发现自己成为共识节点，则打包交易成为区块广播到网络中。
6. 接收到最新区块的节点会验证区块生成者身份的有效性、区块的有效性和交易的合法性性，检查是否有双花交易。如果验证通过，更新本地区块，并开始新一轮的共识。
7. 安全问题：
8. 长程攻击：由于系统生成区块并不需要消耗资源，因此为了防止系统中容易出现的长程攻击，系统可以设计检查点机制，在一定时间内对当前的最新区块链投票达成一致。
9. 无利益关系攻击：对于无利益关系攻击时，对于分叉的两条链，系统会接受节点稳定度更高并且区块哈希较小的那个分支。
10. 女巫攻击：节点的选择机制与节点投入的金钱息息相关。即使节点伪造了多个身份，也需要投入大量的金钱，但是最终成为共识节点的概率并不会因为身份的数量而有优势。
11. 激励机制：生成区块成功的节点将获得大部分区块奖励，剩余小部分区块奖励将分发给后面检查点机制中参与投票的节点。这将激励节点参与共识以及维护区块链的活性。节点的押金将在节点活动时间结束之后全额返还给节点。
12. 惩罚机制：如果节点中途离开，将只会返还部分押金。如果节点还想继续活动，则需要重新提交押金。诚实节点更愿意在区块链之后添加新的区块。一旦发现有节点作恶，则会将该节点的押金没收，并且驱逐出境，从而提高其作恶成本来降低作恶的动机，最终就使得系统更加的安全。

### 4.1.2基于稳定委员会的区块链共识算法

针对节点资源有限并且具有高动态性的情况，单共识节点共识算法的区块链无法实现强一致性只具有最终一致性，并且共识延时较长，吞吐量也受限于区块大小和区块生成时间间隔，因此这类算法并不高效。因此，拟针对资源有限且具有高动态性的规模较大的区块链系统，以稳定性、低共识时延、快速达成全局账本一致性为设计目标，采用经典区块链共识算法与经典一致性算法相结合的方式，设计更安全、快速、高效的区块链共识算法。

设计这类共识算法需要解决以下问题：

1. 初始委员会配置：在不依赖可信启动来初始化委员会配置的问题。
2. 分片委员会选举：委员会在选举时要确保委员会中诚实节点数量达到一定比例，需要防止敌手干扰委员会选举过程，造成偏置性问题。
3. 委员会一致性算法：需要解决客户端提交交易快速响应问题，提高委员会内一致性算法运行效率问题，降低委员会通信复杂度和计算复杂度问题，以及并行提议实现问题。
4. 委员会重置：在委员会重置时需要解决委员会中诚实成员达到一定比例问题；委员会成员交替时交易的处理问题；敌手影响委员会成员加入委员会的问题。
5. 节点自启动问题：考虑新节点加入时时需要解决节点快速安全的自启问题；节点获取交易信息真实有效以及识别虚假信息问题。

研究方案：

1. 网络建模：考虑在区块链系统中，网络节点可以随意进出，这些节点的计算资源和网络资源是有限的，每个节点具有不同的的移动速率和移动方向。节点之间可以通过发送无线信号通信，每个节点的通信半径是相同的。每个节点既可以成为共识节点，也可以只作为一般节点，并且局部地维护自己的区块链。每个节点需要维护两个区块链，一个是身份链一个是交易链。
2. 节点存储区块链：分片中每个节点需要维护两个区块链，一个是身份链一个是交易链。身份链主要是每个节点新加入网络时为获取活跃时间支付押金的交易信息，交易链则是系统中维护的分布式账本。
3. 节点自启机制：节点加入系统时，通过随机游走，比对好几个节点的身份链和交易链，节点最终复制了相同度最高的区块链和身份链。随后开始发起身份认证（身份链的维护可以采用其他的身份验证节点），通过支付押金获取活动时间形成一笔身份认证交易，存放在身份链上获得系统的认可。
4. 委员会候选人选取机制：区块链系统中，新节点进入网络之后，需要质押金钱来获得在这个区块链系统中活动的时间，根据活动时间的长短来选择质押金额的多少。随后根据节点在这个系统中的剩余活跃时间和节点在最近个确认区块中参与共识比值来选出具有较好稳定性和可信性的节点作为共识节点。计算公式为

初始委员会或对于区块确认数量不足K个时重置委员会，记录节点的共识比设置为零。按照时期根据节点的稳定度，再添加一个随机数来选择一定数量的委员会成员。

1. 委员会内部一致性算法：委员会内部执行BFT一致性算法算法。一致性算法过程分为以下步骤：第一步是选出首领节点（可以随机选择，也可以轮循），第二步是首领生成区块，并且作为提案发给其他成员。第三步委员会成员对于提案投票，并将投票结果广播给其他成员。第四步其他成员根据投票结果达成一致结论，将区块存进自己的本地链上，最终广播给系统其他节点。BFT一致性过程如下：PrePrepare阶段：具有最高稳定度的节点打包交易成为区块，将区块构造PrePrepare消息广播给所有其他成员。Prepare阶段：接收到PrePrepare消息后，成员会对区块进行验证。验证成功后，成员将会广播Prepare消息给所有其他成员。Commit阶段：当成员接收到个成员的Prepare消息后，则验证消息的有效性，成功验证后将广播Commit消息给所有成员。当成员接收到条Commit消息时，委员会成员对于议案区块达成一致性结果。但是这个一致性过程的通信复杂度比较高，或许可以采用别的一致性算法来提高一致性过程的效率。
2. 委员会重置：生成区块的奖励将平均分发给委员会中的成员。在完成一个任期之后，更换委员会成员。更换数量尽可能设置为一致性算法的容错数量。新的节点将取代原来委员会中活动剩余时间短的节点。当委员会重置时，将不处理交易，对于正在进行的区块提案，也将在新的委员会选定之后重新提案。
3. 惩罚机制：如果节点中途离开，将只会返还部分押金。如果节点还想继续活动，则需要重新提交押金。诚实节点更愿意在区块链之后添加新的区块。一旦发现有节点作恶，则会将该节点的押金没收，并且驱逐出境，从而提高其作恶成本来降低作恶的动机，最终就使得系统更加的安全。
4. 检查点机制；检查点机制用于对执行结果进行校验。检查点的大小默认设置为个区块，节点在写入到的整数倍个区块后达到一个检查点。此时广播该检查点的信息对区块链的一致性进行校验，校验成功则系统达到了一个稳定检查点。这个机制可以防止敌手发起的长程攻击，使得区块链系统更加安全。在确认之后，区块的收益将平均发给其委员会成员。

### 4.1.3基于稳定度的分片区块链共识算法

在大规模组网中构建的区块链系统，需要考虑节点分布区域广、并且大多数边缘设备资源有限具有高动态性。单共识节点的区块链共识算法无法满足交易处理的高效性和时效性。由于区块传输时延长，区块链更容易出现分叉问题。而基于单一委员会的共识算法在这样的大规模组网中通信效率极低，并且通信资源需求比较大。当参与共识的节点数量增加时，会使得每个节点需要需要与更多节点进行通信。此外长距离通信会由于链路不稳定造成严重的消息丢失问题。拟针对建立在组网规模大，节点分布范围广的网络中建立区块链系统，考虑边缘节点资源有限且高移动以及节点共识的通信资源高等问题，以提升区块链系统的交易处理性能，降低共识的能耗为设计目标，设计在确保安全性的前提下，提升区块链系统扩展性的分片共识算法共识算法。

分片共识算法通过将网络进行分片并行处理交易，分片存储计算的方式可以提升区块链系统的交易吞吐量。通常分片共识算法需要构建多个委员会的方式，并且每个委员会可以并行处理交易，也可以协同处理交易。对于这种多委员会共识算法需要解决以下问题：

1. 初始委员会选举：在不依赖可信启动来初始化委员会配置的问题。
2. 分片委员会选举：对于分片委员会中的成员如何进行分配。可以每个分片自己选择自己的委员会，也可以通过目录委员会分配。
3. 委员会一致性算法：执行需要解决客户端提交交易快速响应问题，提高委员会内一致性算法运行效率问题，降低委员会通信复杂度和计算复杂度问题，以及并行提议实现问题。
4. 委员会重置问题：在委员会重置时需要解决委员会中诚实成员达到一定比例问题；委员会成员交替时交易的处理问题；敌手影响委员会成员加入委员会的问题。
5. 节点自启动问题：还需要考考虑新节点加入时时需要解决节点快速安全的自启问题；节点获取交易信息真实有效以及识别虚假信息问题。

研究方案：

* 1. 网络建模：考虑在大规模组网的区块链系统中，将网络进行分片，每个分片中网络节点可以随意进出，这些节点的计算资源和网络资源是有限的，每个节点具有不同的的移动速率和移动方向。每个分片内节点可以通过发送无线信号通信，每个节点的通信半径是相同的。每个节点既可以成为共识节点，也可以只作为一般节点，并且局部地维护自己所在分片存储的区块链。

1. 节点存储区块链：分片中每个节点需要维护两个区块链，一个是身份链一个是交易链。身份链主要是每个节点新加入网络时为获取活跃时间支付押金的交易信息，交易链则是系统中维护的分布式账本。

2. 节点自启机制：节点加入系统时，通过随机游走，比对好几个节点的身份链和交易链，节点最终复制了相同度最高的区块链和身份链。随后开始发起身份认证（身份链的维护可以采用其他的身份验证节点），通过支付押金获取活动时间形成一笔身份认证交易，存放在身份链上获得系统的认可。

* 1. 分片委员会选举：区块链系统中，新节点进入分片网络之后，需要质押金钱来获得在这个分片区块链系统中活动的时间，根据活动时间的长短来选择质押金额的多少。随后根据节点在这个系统中的剩余活跃时间和节点在最近个确认区块中参与共识比值来选出具有较好稳定性和可信性的节点作为共识节点。计算公式为

初始分片委员会或对于区块确认数量不足K个时重置委员会，记录节点的共识比设置为零。按照时期根据节点的稳定度，再添加一个随机数来选择一定数量的委员会成员。

* 1. 最终共识委员会选举：从每个分片委员会中选择出一个节点作为最终共识委员会成员。选举过程最好是引入一个随机数，提高选举过程的抗偏置性。
  2. 委员会内一致性协议：委员会内部执行BFT一致性算法算法。一致性算法过程分为以下步骤：第一步是选出首领节点（可以随机选择，也可以轮循），第二步是首领生成区块，并将其作为提案发给其他成员。第三步是委员会成员对于提案投票，并将投票结果广播给其他成员。第四步其他成员根据投票结果达成一致结论，将区块存进自己的本地链上，最终广播给系统其他节点。BFT一致性过程如下：PrePrepare阶段：具有最高稳定度的节点打包交易成为区块，将区块构造PrePrepare消息广播给所有其他成员。Prepare阶段：接收到PrePrepare消息后，成员会对区块进行验证。验证成功后，成员将会广播Prepare消息给所有其他成员。Commit阶段：当成员接收到个成员的Prepare消息后，则验证消息的有效性，成功验证后将广播Commit消息给所有成员。当成员接收到条Commit消息时，委员会成员对于议案区块达成一致性结果。但是这个一致性过程的通信复杂度比较高，或许可以采用别的一致性算法来提高一致性过程的效率。
  3. 委员会重置：将时间分为多个时期，每个时期作为一个委员会的任期。每个任期内又又分为多个轮，每轮产生一个区块。在任期中最后一个轮，委员会将选举新的委员会成员。对于每个分片中，选举新的节点替换出剩余活动时间低的委员会成员。当分片委员会成员发生变化之后，最终共识委员将会从该分片委员会中选择新的节点替换掉最终共识委员会中剩余活动时间少的委员会成员。在委员会发生变化时，被打断的交易将会在新的委员会中重新被处理。
  4. 跨片交易防锁死机制：由于交易是分片存储的，对于跨片交易，需要一个机制防止交易被锁死。根据交易的数据结构，来设计一个原子性的交易机制，从而确保跨片交易即使存在问题，也不会造成锁死问题。
  5. 惩罚机制：如果节点中途离开，将只会返还部分押金。如果节点还想继续活动，则需要重新提交押金。诚实节点更愿意在区块链之后添加新的区块。一旦发现有节点作恶，则会将该节点的押金没收，并且驱逐出境，从而提高其作恶成本来降低作恶的动机，最终就使得系统更加的安全。
  6. 检查点机制；检查点机制用于对执行结果进行校验。检查点的大小默认设置为个区块，节点在写入到的整数倍个区块后达到一个检查点。此时广播该检查点的信息对区块链的一致性进行校验，校验成功则系统达到了一个稳定检查点。这个机制可以防止敌手发起的长程攻击，使得区块链系统更加安全。

### 4.1.4基于稳定度的DAG区块链共识算法

针对大规模组网中节点数量庞大、节点资源有限且具有高动态性时，单共识节点共识算法的区块链系统的扩展性非常的有限，并且面对高频交易时处理效率低。同时，这类共识算法还面临着区块链分叉问题。此外，委员会共识算法的共识过程需要通信机制。当系统规模巨大时，节点有限的网络信道资源将无法满足系统的需求，区块链系统的可伸缩性和交易吞吐量的提升非常有限。拟针对大规模高动态网络，考虑节点资源有限性和移动性，以提升区块链系统的扩展性、降低共识的能耗、提升区块链系统的交易吞吐量为设计目标，设计在确保高扩展性的前提下，提升区块链系统安全性的DAG链共识算法。

DAG区块链共识算法面临着严重的安全性问题。

第一 节点自启机制。区块链系统如何确保新加入节点的可信性。如何确保节点能够快速获取可信区块链账本。由于基于DAG区块链的系统具有交易并发性，因此每个节点维护的账本是不同的，因此如何确保新加入的节点获取的是最新的账本是需要解决的问题。

第二 双花攻击问题。由于DAG区块链中允许分叉，因此容易被敌手利用发起双花攻击。共识算法需要对双花交易中的有效交易达成一致。

第三 激励机制。这类区块链通常没有挖矿费用，因此需要解决节点由于没有收益而不愿意生成交易参与共识的问题。

第四 长程攻击问题。由于节点生成交易并不需要耗费非常多的资源，因此节点可以通过伪造大量交易来确保自己的交易最终被确认。

第五 见证委员会选举 需要设计一个具有良好抗敌手偏置性的见证人选举机制，确保委员会内诚实成员达到一定比例。并且安全的选择委员会首领，防止首领不作为或者恶意行为。

第六 委员会重置 一个委员会工作时间对应一个时期，共识算法需要合理设置每个时期的长度，防止敌手腐蚀委员会成员。在一个时期结束后，委员会重配置需要解决委员会成员交替时交易的处理问题和敌手干扰委员会新成员选择问题。重置委员会通常是只更换部分成员，随后进入新时期。

研究方案：

1. 网络建模：区块链系统是由许多无线网络节点构成。通过网络协议可以在节点建立连接，进行消息传输等功能。节点具有可移动性，每个节点既可以成为共识节点，也可以只作为一般节点。
2. DAG区块链 整个系统中，每个节点都维护一个局部的DAG区块链。由于账本只需要达到最终一致性，因此每个节点维护的账本将不会完全一致。但是最终确认的交易账本将会保持一致性。
3. 节点自启机制 节点加入系统时，需要获取可信的区块信息和节点信息。通过比较多个节点的账本，选择具有公共前缀且账本中最重的账本作为可信账本，最后添加到最忌的本地存储中作为最新的区块链账本。
4. 交易生成机制 节点通过解决一个小哈希难题来获得产生交易权限。从而避免节点大量生成空交易影响区块链中交易的确认。节点可直接将产生的交易根据系统规则选择父交易，最终添加到局部地维护的DAG区块链并广播。接收到交易的节点验证交易的有效性后将交易也添加到自己的本地链上。
5. DAG共识算法
6. 父交易选择机制 新交易生成节点会根据父交易的选择规则选择引用的交易，可以是多个。随后，根据交易累积权重来确定交易达成共识。由于DAG区块链存储的特殊性，对于交易的确认通常是根据其累积支持交易的数量来决定的。
7. 主链机制：由于在DAG区块链中链分叉是合法的，双花问题成为这类区块链系统亟需解决的一个难题。通过选定一条主链来构建一个交易的全序。即使出现双花交易，也可以根据交易的全序安全的拒绝后出现的交易。通过见证者交易成为稳定交易来确定主链进而确定各个主链相关交易的主链号。根据主链指数来确定交易的先后顺序。
8. 见证节点选择 区块链系统中见证节点需要是可信的，并且生成区块也是可信的。因此需要设计一个安全的见证交易的共识算法来确保系统的安全性。节点可能会生成的交易不能被修改，并且确保节点在系统中的稳定性。因此见证委员的选取需要具有足够的稳定性和可信性。
9. 见证节点自启动 新节点进入网络之后如果想要成为见证人，则需要质押金钱来获得在这个区块链系统中可信的时间，根据时间的长短来选择质押金额的多少。每个节点会维护一个身份链，节点进入系统后获取交易链和身份链。
10. 委员会选举机制 根据节点在这个系统中的剩余活跃时间和节点在最近个确认区块中参与共识比值来选出具有较好稳定性和可信性的节点作为共识节点。计算公式为

初始委员会或对于区块确认数量不足K个时重置委员会，记录节点的共识比设置为零。按照时期根据节点的稳定度，再添加一个随机数来选择一定数量的委员会成员。

1. 委员会一致性算法 可以采用BFT算法来生成见证交易，并通过协商达成一致。见证交易与一般交易采用相同的父交易引用机制，至少承担了确定主链的功能。
2. 委员会重置：生成区块的奖励将平均分发给委员会中的成员。在完成一个任期之后，更换委员会成员。更换数量尽可能设置为一致性算法的容错数量。新的节点将取代原来委员会中活动剩余时间短的节点。当委员会重置时，将不处理交易，对于正在进行的区块提案，也将在新的委员会选定之后重新提案。
3. 惩罚机制：如果节点中途离开，将只会返还部分押金。如果节点还想继续活动，则需要重新提交押金。诚实节点更愿意在区块链之后添加新的区块。一旦发现有节点作恶，则会将该节点的押金没收，并且驱逐出境，从而提高其作恶成本来降低作恶的动机，最终就使得系统更加的安全。
4. 激励机制 一个交易的交易费用需要支付给其子交易中哈希值最小的节点以及生成（直接或者间接）引用该交易的见证交易所属的委员会成员。通过激励机制可以激励节点的活性以及成为见证委员会成员的积极性，进而提高系统的安全性。

## 4.2可行性分析

本课题拟定的研究内容和研究方案是作者在已有工作成果，在对实际应用需求充分调研，并对当前研究现状详细分析基础上提出的。同时，国内外学者在区块链共识算法的研究中取得了大量的研究成果，这些都为本课题的研究奠定了坚实的基础，提供了可借鉴的内容。

除了上述分析之外，本人已经查阅了大量该领域的研究论文，并且熟悉多种编程语言和多种仿真手段。因此，本课题的研究方案是切实可行的。

# 5研究工作计划与进度安排

2021年12月-2022年6月：

调研区块链的应用和区块链共识算法最新研究状况。

研究基于稳定度的单链区块链共识算法。

研究基于稳定度的单一委员会区块链共识算法。

2022年7月-2022年12月：

研究基于稳定度的分片区块链共识算法。

研究基于稳定度的DAG区块链见证人委员会共识算法。

2023年1月-2022年5月：

撰写和提交毕业论文。

# 参考文献：

[1] S. Nakamoto Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf, 2008.

[2] G. Wood. Ethereum: A Secure Decentralized Generalized Transaction Ledger. Ethereum Project Yellow Paper, vol. 151, pp. 1-32, 2014.

[3] C. Dwork and M. Naor,Pricing via processing or combatting junk mail[C]. In: Advances in Cryptology— CRYPTO’92. Springer Berlin Heidelberg, 1993: 139–147. [DOI: 10.1007/3-540-48071-4\_10

[4] A. Back, . Hashcash—A denial of service counter-measure[EB/OL]. 2002. http://www.hashcash.org/papers/hashcash.pdf.

[5] M.Jackbsson and A.Juels.Proofs of Work and Bread Pudding Protocols[C]// Ifip Tc6/tc11 Joint Working Conference on Secure Information Networks: Communications & Multimedia Security. Kluwer, B.V. 1999.

[6] J. Garay, A. Kiayias, and N. Leonardos. The Bitcoin Backbone Protocol: Analysis and Applications. In Proceedings of the 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, 2015.

[7] R. Pass, L. Seeman, and A. Shelat. Analysis of the Blockchain Protocol in Asynchronous Networks. Annual International Conference on the Theory and Applications of Cryptographic Techniques Springer, Cham, 2017.

[8] I. Eyal, The Miner’s Dilemma. 2015 IEEE Symposium on Security and Privacy, San Jose, CA, May 2015, pp. 89-103, doi: 10.1109/SP.2015.13.

[9] R. Pass, E. Shi. Fruitchains: A fair blockchain. In Proceedings of the ACM Symposium on Principles of Distributed Computing, ser. PODC ’17. ACM, 2017, pp. 315–324. [Online]. Available: http://doi.acm.org/10.1145/3087801.3087809

[10] E.Heilman, A.Kendler, A.Zohar, et al. Eclipse attacks on Bitcoin's peer-to-peer network[c]. In: Proceedings of 24th USENIX Security Symposium (USENIX Security 15). USENIX, 2015: 129–144.

[11] I. Eyal, The Miner’s Dilemma. 2015 IEEE Symposium on Security and Privacy, San Jose, CA, May 2015, pp. 89-103, doi: 10.1109/SP.2015.13.

[12] I. Eyal, E. G. Sirer. Majority is not Enough: Bitcoin Mining is Vulnerable. In Financial Cryptography and Data Security. Springer, November 2014, pp. 436õ454.

[13] S. King, S. Nadal. PPCoin: Peer-to-peer crypto-currency with proof-of-stake. https://archive.org/details/PPCoinPaper (2012).

[14] Buterin V , Griffith V . Casper the Friendly Finality Gadget[J].2017. https://arxiv.org/pdf/1710.09437.pdf

[15] A. Kiayias, A. Russell,B. David, et al. Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol[C]. In: Advances in Cryptology—CRYPTO 2017, Part I. Springer Berlin Heidelberg, 2017: 357–388. [DOI: 10.1007/978-3-319-63688-7\_12]

|  |
| --- |
|  |

[16] B. David, P.Gazi, A. Kiayias, et al. Ouroboros Praos: An adaptively-secure, semi-synchronous proof-of-stake Blockchain[C]. In: Advances in Cryptology—EUROCRYPT 2018, Part II. Springer Cham, 2018: 66–98. [DOI:10.1007/978-3-319-78375-8\_3]

[17] S. D. Lerner．DagCoin:a cryptocurrency without blocks[EB/OL]. https://bitslog.files.wordpress.com/2015/09/dagcoin-v41.pdf, 2015.

[18] C. Badertscher, P. Gazi, A. Kiayias, et al. Ouroboros Genesis: Composable proof-of-stake Blockchainswith dynamic availability[C]. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS 2018). ACM, 2018: 913–930. [DOI: 10.1145/3243734.3243848]

[19] P. Debian, R. Pass, E. Shi. Snow White: Robustly reconfgurable consensus and applications to provably secure proofs of stake[J]. IACR Cryptology ePrint Archive, 2017: 2016/919. https://eprint.iacr.org/2016/919.pdf

[20] I. Grigg. EOS—An introduction[EB/OL]. 2017. https://eos.io/documents/EOS\_An\_Introduction.pdf.

[21] N. Houy. It will cost you nothing to ‘kill’ a proof-of-stake crypto-currency[EB/OL]. 2014. ftp://ftp.gate.cnrs.fr/RePEc/2014/1404.pdf.

[22] A. Chepurnoy. Interactive proof-of-stake[EB/OL]. 2016. http://arxiv.org/abs/1601.00275.

[23] J. Kwon. Tendermint: Consensus without mining. Draft v. 0.6, fall, 1(11), 2014. https://tendermint.com/static/docs/tendermint.pdf.

[24] P. Gazi, A. Kiayias, A. Russell. Stake-bleeding attacks on proof-of-stake Blockchains[C]. In: Proceedings of 2018 Crypto Valley Conference on Blockchain Technology (CVCBT 2018). IEEE, 2018: 85–92. [DOI: 10.1109/ CVCBT.2018.00015]

[25] C. Decker, J. Seidel, R. Wattenhofer. Bitcoin Meets Strong Consistency. In Proceedings of the 17th International Conference on Distributed Computing and Networking. ACM, 2016: 13:1–13:10. [DOI:  
10.1145/2833312.2833321]

[26] E. Kokoris- Kogias, P. Jovanovic, N. Gailly, I. Khoffi, L. Gasser, B. Ford. Enhancing Bitcoin security and performance with strong consistency via collective signing. In Proc. 25th conference on USENIX Security Symposium, 2016 : 279–296.

[27] I. Abraham, D. Malkhi, K. Nayak, et al. Solida: A Blockchain protocol based on reconfgurable Byzantine consensus[C]. In: Proceedings of 21st International Conference on Principles of Distributed Systems (OPODIS 2017). Lisbon, Portugal, December 18–20, 2017: 25:1–25:19. [DOI: 10.4230/LIPIcs.OPODIS.2017.25]

[28] R. Pass, E. Shi. Hybrid consensus: Efcient consensus in the permissionless model[C]. In: Proceedings of 31st International Symposium on Distributed Computing (DISC 2017). Vienna, Austria, October 16–20, 2017: 39:1–39:16. [DOI: 10.4230/LIPIcs.DISC.2017.39]

[29] Y.Gilad, R. Hemo, S. Micali, et al. Algorand: Scaling Byzantine agreements for cryptocurrencies[C]. In: Proceedings of the 26th Symposium on Operating Systems Principles, Shanghai, China, October 28–31, 2017: 51–68. [DOI: 10.1145/3132747.3132757]

[30] L.Luu, V. Narayanan, C. Zheng, et al. A secure sharding protocol for open Blockchains[C]. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016: 17–30. [DOI: 10.1145/2976749.2978389]

[31] E. Kokoris- Kogias, P. Jovanovic, L. Gasser, et al. OmniLedger: A secure, scale-out, decentralized ledger via sharding[C]. In: Proceedings of 2018 IEEE Symposium on Security and Privacy (SP 2018). IEEE, 2018: 583–598. [DOI: 10.1109/SP.2018.000-5]

[32] P.Schindler, A. Judmayer, N. Stifter, et al. HydRand: Practical continuous distributed randomness[J]. IACR Cryptology ePrint Archive, 2018: 2018/319. https://eprint.iacr.org/2018/319.pdf.

[33] M. AL-Bassam,A. Sonnino, S. Bano, et al. Chainspace: A sharded smart contracts platform[C]. In: Proceed ings of 25th Annual Network and Distributed System Security Symposium (NDSS 2018). San Diego, CA, USA, February 18–21, 2018. http://wp.internetsociety.org/ndss/wp-content/uploads/sites/25/2018/02/ndss2018\_09-2\_Al-Bassam\_paper.pdf.

[34] M. Zamani, M. Movahedi, M. Raykova. RapidChain: Scaling Blockchain via full sharding[C]. In: Proceed ings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS 2018). Toronto, ON, Canada, October 15–19, 2018: 931–948. [DOI: 10.1145/3243734.3243853]

[35] Technical White Paper. TrustNote: Fast, Scalable, Developer Friendly, https://github.com/ringringringring/document/blob/master/TrustNote-TR-2018-03.pdf, 2018.

[36] Y. Sompolinsky, A. Zohar. Secure high-rate transaction processing in Bitcoin[C]. In: Financial Cryptography and Data Security—FC 2015. Springer Berlin Heidelberg, 2015: 507–527. [DOI:10.1007/978-3-662-47854-7\_32]

[37] A.Sonnino, S. Bano, M. AL-Bassam, et al. Replay attacks and defenses against cross-shard consensus in  
sharded distributed ledgers[EB/OL]. 2019. http://arxiv.org/abs/1901.11218

[38] A. Miller, A. Juels, E. SHI E, et al. Permacoin: Repurposing Bitcoin work for data preservation[C]. In: proceedings of 2014 IEEE Symposium on Security and Privacy (SP 2014). IEEE, 2014: 475–490. [DOI: 10.1109/SP.2014.37]

[39] A. Sonnino, S. Bano, M. AL-Bassam, et al. Replay attacks and defenses against cross-shard consensus in sharded distributed ledgers[EB/OL]. 2019. http://arxiv.org/abs/1901.11218

[40] HyperLedger, Intel: Sawtooth Lake (2017). [PoET 1.0 Specification — Sawtooth v1.2.6 documentation (hyperledger.org)](https://sawtooth.hyperledger.org/docs/core/releases/latest/architecture/poet.html#introduction)

[41] D. Schwartz, N. Youngs, A. r Britto. The Ripple Protocol Consensus Algorithm[EB/OL]. https://ripple.com/files/ripple\_consensus\_whitepaper.pdf, 2018.

[42] C. Lemhieu．Nano:A feeless distributed cryptocurrency network[EB/OL]． https://content.nano.org/whitepaper/Nano\_Whitepaper\_en.pdf

[43] S.Popov. The Tangle [EB/OL]. 2016. WWW.descryptions.com/Iota.pdf, 2018.

[44] A. Churyumov. Byteball: A decentralized system for storage and transfer of value[EB/OL]. 2016.  
https://byteball.org/Byteball.pdf.

[45] B. Leemon, H. Mance, M. Paul. Hedera: A governing council and public hashgraph network[EB/OL]. 2018.  
https://www.hedera.com/hh-whitepaper-v1.4-181017.pdf.

[46] Y. Liu, J. Liu, Q. Wu, H. Yu, H. Yiming and Z. Zhou, "SSHC: A Secure and Scalable Hybrid Consensus Protocol for Sharding Blockchains with a Formal Security Framework," in IEEE Transactions on Dependable and Secure Computing, doi: 10.1109/TDSC.2020.3047487.

[47] C. Huang et al., "RepChain: A Reputation-Based Secure, Fast, and High Incentive Blockchain System via Sharding," in IEEE Internet of Things Journal, vol. 8, no. 6, pp. 4291-4304, 15 March15, 2021, doi: 10.1109/JIOT.2020.3028449.

[48] M. M. Jalalzai, C. Busch and G. G. Richard, "Proteus: A Scalable BFT Consensus Protocol for Blockchains," 2019 IEEE International Conference on Blockchain (Blockchain), 2019, pp. 308-313, doi: 10.1109/Blockchain.2019.00048.

[49] P. Berrang, P. von Styp-Rekowsky, M. Wissfeld, B. França and R. Trinkler, "Albatross – An optimistic consensus algorithm," 2019 Crypto Valley Conference on Blockchain Technology (CVCBT), 2019, pp. 39-42, doi: 10.1109/CVCBT.2019.000-1.

[50] W. Li and M. He, "EBFT: A Hierarchical and Group-Based Byzantine Fault Tolerant Consensus Algorithm," 2021 IEEE 12th International Conference on Software Engineering and Service Science (ICSESS), 2021, pp. 32-37, doi: 10.1109/ICSESS52187.2021.9522233.

[51] M. H. Ashik, M. M. Shahriar Maswood, A. G. Alharbi and D. Medhi, "FPoW: An ASIC-resistant Proof-of-Work for Blockchain Applications," 2020 IEEE Region 10 Symposium (TENSYMP), 2020, pp. 1608-1611, doi: 10.1109/TENSYMP50017.2020.9230915.