# 《区块链技术与应用》课程设计——共识算法

#### 前言

在前半个学期中，我们学习了《区块链技术与应用》这门课程，在这门课程中我们了解了区块链技术大概是个什么样子，能够用来做些什么，然后在具体的生活中又有哪些应用，面临着哪些问题。在完成了对课程的学习后，我们小组按照老师的要求安排决定完成一个课程设计，主要是设计区块链中的共识算法，一共有五个算法，分别为pos，pow，dpos，pbft，raft。其中余志伟同学（学号：M202173793）完成pbft和raft部分，徐子清（学号：M202173797）同学完成pow部分，杨劲帆（学号：M202173794）完成pos部分，孙锐（学号：M202173786）完成dpos部分。

#### 算法介绍

##### 2.1pow共识

Pow（Proof-of-Work），俗称工作量证明机制。简单来说，就是一份用来确认你做过一定量工作的证明，它就像你的大学毕业证，证明了你确实有四年大学的学习经历。工作量证明机制，就是用工作量结果来证明贡献大小，再根据贡献大小确定记账权和奖励。

这个证明过程，是依赖计算机进行数学运算进行的。可以理解为：大家都去解答同一个题目，谁先算出来，谁就负责记账，并得到相应报酬，这个报酬就是网络产生的数字货币。比如，在比特币网络系统里，谁先解题解出来，就能先得到比特币作为奖励。

Pow的优点是，完全去中心化，公平公正，不需要中心化的管理机构，用户（即节点）之间实现了公平竞争，谁先解出题目，谁就获得相应收益。

但Pow也有不可忽视的缺点：浪费能源。大家一起算题目，都要耗费算力，而最终有效的只有一个用户所做的功，其他人做的都是无用功。而计算机是靠电力来带动的，大家一起用计算机算题，其实也耗费了大量电力资源。比如，比特币就是采用POW共识机制，每年需要消耗价值几十亿美元的电力，因此一直遭人诟病。

##### 2.2pos共识

PoS的原理类似于现实世界中的股份制，拥有股份越多，话语权就越强，获得记账机会的概率就越大。其安全性来自于抵押经济的价值，理论上 Staking 的数量越高收益越高。

与 PoW 不同，PoS 共识机制的核心是网络中的币，这些币可以在交易所甚至 OTC 市场大量购买，不需要通过挖矿获取，所以参与 Staking 几乎没有门槛。在 PoS 共识机制中，超级节点可以通过运行和维护网络获得奖励，所以运行节点的动机是很强烈的。

PoW 和 PoS 是两种不同的算法，它们决定分布式网络中的共识，以及如何奖励共识维护者。换句话说，PoW 和 PoS 阐述了如何发行新币为网络提供激励，奖励那些提供资源的共识维护者。这些币的价值背后都有资源支撑（矿机或者一定数量的币），而且拿到奖励的前提是你必须付出诚实的行动。

PoS 的安全性来自于抵押经济的价值，理论上 Staking 的数量越高收益越高。虽然不是所有人都愿意或有能力去运行节点，但是 Staking 机制对于去中心化是很重要的，因为 PoS 允许所有持币人在没有运行节点的情况下投票并参与共识。

PoS 里的 Validator 更像是投资人，他们需要对所持数字货币背后的技术有很深的理解，同时对协议的发展作出贡献。大型PoS 币种一般都有独立的研究团队。

为了获得更多的投票，这些 Validator 一般需要持有大量的币。通过 Staking 或者提供 Staking 服务，这些资金会被锁定一定时间，如果想动用部分资金，那他们可能不能继续为所有的客户提供服务。这样 Validator 就成为了长期投资者，他们更有动力驱动技术和价值。激励越高，投入越多。

##### 2.3pbft共识

在PBFT算法中，至多可以容忍不超过系统全部节点数量的三分之一的拜占庭节点“背叛”，即如果超过三分之二的节点正常，整个系统就可以正常工作。 PBFT算法采用密码学相关技术（RSA签名算法、消息验证编码和摘要）确保消息过程中无法被篡改和破坏。算法的基本过程如下:

1.首先通过轮换或随机算法选出某个节点为主节点，此后只要主节点不切换，则成为一个视图（view）

2.在某个试图中，客户端将请求<Request,operation,timestamp,client>发送给主节点，主节点负责广播请求到所有其他副本节点。

3.所有节点处理完成请求，将处理结果<Reply,view，timestamp,client,id\_node,response>返回给客户端。客户端检查是否收到了至少f+1个来自不同节点的相同结果，作为最终结果。 主节点广播过程包括三个阶段的处理：预准备（pre-prepare）阶段、准备（prepare）和提交（commit）阶段预准备和准备阶段确保在同一视图内请求发送顺序正确；准备和提交阶段则确保在不同视图之间的确认请求是保序的：

预准备阶段：主节点为从客户端收到的请求分配提案编号，然后发出预准备消息<<pre-prepare,view,n,digest>,message>给各副本节点，其中message是客户端的请求消息，digest是消息的摘要。

准备阶段：副本节点收到预准备消息后，检查消息合法，比如检查通过则向其他节点发送准备消息<<prepare,view,n.digest,id>>,带上自己的id信息，同时接收来自其他节点的准备信息，收到准备消息的节点对消息同样进行消息合法性检查，验证通过后，则把这个准备消息写入消息日志中，集齐至少2f+1个验证过的消息才进入准备状态。

提交阶段：广播commit消息，告诉其他节点某个提案n在视图v中已经处于准备状态。如果集齐至少2f+1个验证通过的commit消息，则说明提案通过。

##### 2.4raft共识

我们主要分以下三部分对Raft进行讨论：

Leader election——a new leader must be chosen when an existing leader fails. （领导人选举）

Log replication——the leader must accept log entries from clients and replicate them across the cluster, forcing the other logs to agree with its own.（日志复制）Safety——the key safety property for Raft. （安全性）

正常工作过程中，Raft分为两部分，首先是leader选举过程，然后在选举出来的leader基础上进行正常操作，比如日志复制操作等。

一个Raft集群通常包含2N+12N+1个服务器，允许系统有NN个故障服务器。每个服务器处于3个状态之一：leader、follower或candidate。正常操作状态下，仅有一个leader，其他的服务器均为follower。follower是被动的，不会对自身发出的请求而是对来自leader和candidate的请求做出响应。leader处理所有的client请求（若client联系follower，则该follower将转发给leader)。candidate状态用来选举leader。

##### 2.5dpos共识

DPOS全称是委托股权证明 (Delegated proof os Stake). DPOS算法分为两个部分:区块生产者选举，生产调度．

区块生产者选举(election a group of block producers)过程确保利益相关者最终得到控制，因为当网络不顺利运作时，利益相关者损失最大。人们如何当选对于如何逐分实现共识几乎没有影响．因此，本文将重点讨论如何在选定区块生产商后达成共识．

为了帮助解释这个算法，我想假设3个块生产者A，B和C.因为共识需要2/3 + 1来解决所有情况，所以这个简化模型将假设生产者C被认为是攻击者。 在现实世界中，将有21个或更多的块生产者。就像工作证明一样，一般规则是最长的连锁胜出。任何时候，一个诚实的同行看到一个有效的严格的更长的链条，它会从当前的分叉切换到更长的分叉。

我将通过示例展示DPOS如何在最可想象的网络条件下运行。这些例子应该可以帮助你理解为什么DPOS很健壮并且很难打破。

###### 2.5.1各情况下运行分析

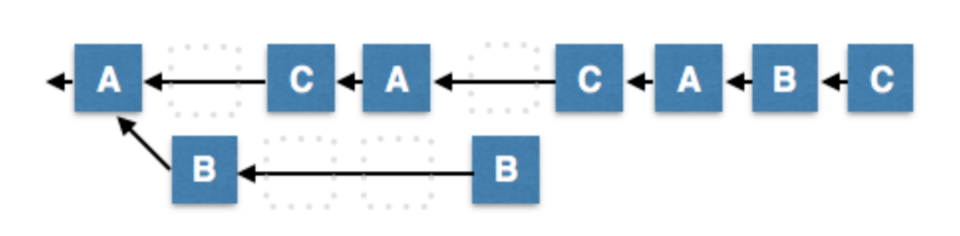
###### 正常情况

在正常操作下，块生产者轮流每3秒产生一个块。假设没有人错过他们，那么这将产生最长的链条。 块生产者在任何其他时间段生成块都是无效的。



###### 少数派未连接分支

高达1/3的节点可能是恶意或故障，并创建一个少数叉。在这种情况下，少数叉将只产生一个块每9秒，而大多数叉将产生2块每9秒。再次，诚实的2/3多数将比少数人长。



###### 少数派未连接多分支

少数人可以尝试生产无限数量的分支，但是他们的所有分支都将比主要连锁店短，因为少数人的生产环节比大多数人慢。

###### 网络破碎

在某种情况下，网络完全有可能被分割很多个碎片，而没有大部分区块生产者。 在这种情况下，最长的链将会成为最大的少数。 当网络连接恢复后，较小的少数民族将自然地切换到最长的连锁店，并且明确的共识将被恢复。有可能有3个分支其中2个最长的分支是相同的长度。在这种情况下，在第三条最短链上的生产者会放弃原有链，重新加入两条最长链之一的网络。生产商的数量是奇数的，所以不可能长期保持相同。之后，我们将覆盖生产者洗牌，这将随机化生产顺序，以确保即使两个分支具有相同数量的生产者，分支将以不同速度增长．

###### 连接少数派双生产

在这种情况下，少数B在他们的时隙产生了两个或更多个备选块。下一个计划生产者（C），可以选择从B产生的任何一个区块中建立。当这发生时，它将成为最长链，并且选择B1的所有节点将切换叉。不多的少数不良生产商试图传播的替代品并没关系，它们将永远不会成为超过一轮的最长链的一部分。

#### **算法功能对比**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | PoW | PoS | DPoS | PoET |
| 记账权分配 | 算力 | 权益 | 权益 | 随机时间 |
| 安全威胁 | 算力集中 | 活跃节点不足 | 生产者作恶 | 不适用于公有区块链 |
| 平均出块时间 | 10min（比特币） | 64s（黑币） | 3s（比特股） |  |
| 代表性项目 | 比特币、莱特币 | 点点币、黑币 | 比特股、柚子 | 随机领导者选举 |

Pow：

优点：其算法采用了目前行业内大家普遍认可的寻找随机数，比较简单且容易实现。同时就是安全性非常的高，因为犯错的成本很高，所以也就导致了比特币安全运行十年依然没有出现任何大的系统隐患。

缺点：因为矿机设备需要耗费了大量的电力和水力，处理效率比较低。现在有矿池的出现，算力集中在个别几家，那也就可能他们的算力超过全网的51%，就有做坏事的条件了，比如强行分叉一条新的链。随着大矿池的出现，也就越来越中心化。违背了区块链的初衷。

Pos：

优点：矿工不需要去拼算力，而且也不会浪费太多的算力，这样间接性的就会缩短达成共识的时间。性能上会比POW要来的高。

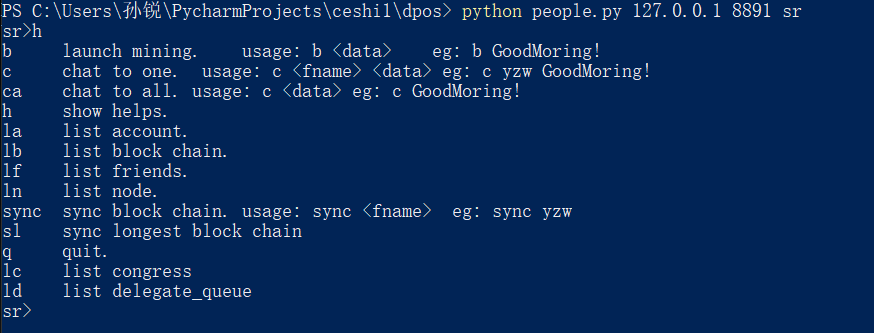
缺点：拥有代币的大户可以坐享其成，而且所有参与者可以持币拿利息。卖币的人也会少了，大家想着存着币拿利息，也不利于流动性。还有一个间接的危险，就是会吸引黑客攻击大量币的钱包。

Dpos：

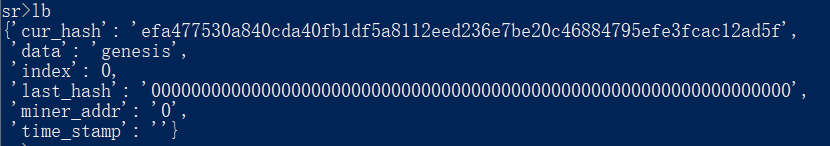
优点：继承了POS的有钱且比POS更快的效率和更高的性能。

缺点：为了性能，对去中心化做了一个妥协。比如EOS来说，就21个超级节点，那么这个区块链网络已经不是真正意义上的“去中心化”了，反而变成了“弱中心”或者说“部分去中心化”。

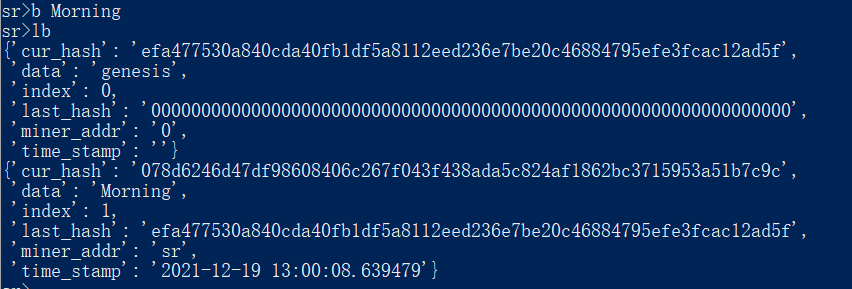
#### 4.算法执行结果



当前可用命令列表



区块链列表



开始挖掘区块