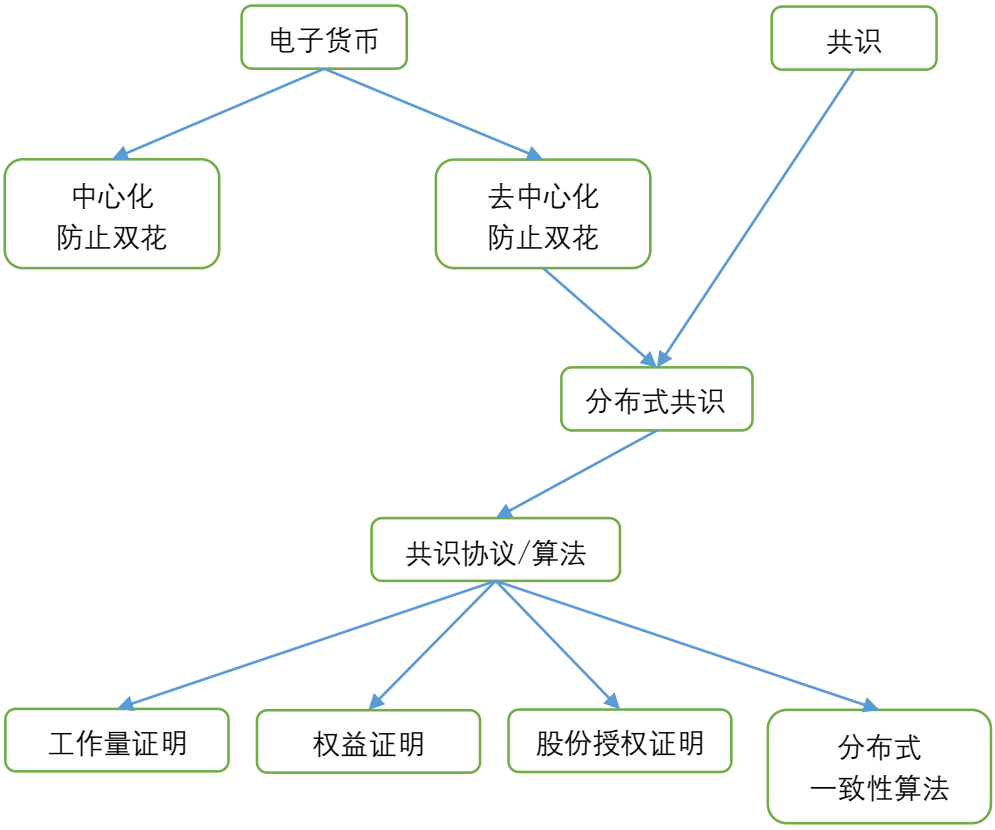
区块链共识机制

思维导图



摘要

传统的中心化货币系统通过一个可信的权威机构来确保每次交易的可靠性，以及解决交易中可能的争议。在以比特币为代表的去中心化电子货币系统中，不存在这样的机构，因此，需要使用一种分布式的共识机制，维护全局账本（交易记录）的一致性和有效性。本文将介绍分布式共识机制的由来，和目前主要的 4 种分布式共识协议的具体内容。

关键词：区块链，分布式共识，共识协议，PoW，PoS，DPoS，分布式一致性算法。

一、电子货币与共识机制

1.1 电子货币及其交易方式

近年来，以比特币（Bitcoin）为代表的电子货币和交易系统迅速发展。广义上的电子货币大体可划分为两类：一类仍然是传统金融体系的一部分，账户与现实实体绑定，作为对传统交易方式的扩展；另一类电子货币自身有一个相对独立的货币和交易体系（但是要参与现实世界的金融行为，这一类电子货币仍需在现实的货币体系中有确定的“价值”）,而账户一般不与现实实体绑定，保证一定程度的匿名性。

在第二类电子货币中，假如 A 要与 B 进行一次交易（即 A 向 B 转账），A 需要生成一个转账记录（transaction），在其中填上 B 的公钥的 hash 值（即 B 的账户地址），以及自己要转给 B 的货币的相关信息，然后用自己的私钥对这个记录进行签名，最后把记录写入账本（ledger）。[1] 由于电子信息容易复制，如何避免 A 将相同的货币多次转账（double-spending）自然成为这种电子货币面对的主要问题。

1.2 电子货币防止双花（double-spending）的方法

上述电子货币按照是否有中心化的可信权威机构（trusted central authority，如银行）又可以分为两种：中心化的和去中心化的。

在中心化的无身份电子货币系统中，一种有效的防止双花的方法是，接收方每次收到的转账都提交给银行，银行负责销毁旧币并给接收方发送等额的新币。这样每次交易使用的货币都是全新的，一旦存在双花，接收方就会向银行提交已经被提交过一次的货币，从而双花被检测出来。

在去中心化的系统中，不存在这样的银行，需要用密码学的方法解决双花问题。其中的关键问题是：在不存在全局时钟的分布式系统中，需要维护一个全局一致的账本，账本中按被所有节点承认的时间顺序记录了所有的交易历史（实际上出于效率考虑，会将若干交易打包成区块（block），再将区块写入账本），其中每条交易的合法性都能被每个节点检查。对这个账本的维护，就需要所有节点形成一个分布式共识。[2]

二、分布式共识

2.1 共识和分布式共识

共识（consensus）这一概念源于计算机科学，是最弱的一种同步概念。在并发计算中，共识对象提供了一种 decide 方法，每个线程进行同步时需要调用一次这个方法，并提

供一个输入参数。在一次同步过程中，每个参与并发的线程都调用同一个共识对象的 decide 方法，每个线程有不同的输入参数。decide 方法向所有线程返回相同的输出，并且这个输出值与其中某个线程的输入值相同，即认为该线程先于其它线程，所有线程达成了共识。[3][4]

分布式共识（distributed consensus）是一个类似的概念，起源于对分布式系统一致性的维护。在区块链中，假定有若干个节点各自将一些交易打包成一个区块（在不同节点上这些交易通常不会完全相同），每个节点都会试图把自己的区块记入账本中。所谓记入账本，就是在区块链网络中广播自己的区块，区块中的 hash 指针指向当前区块链的最新区块（即尝试将自己的区块上链）。但由于需要维护账本的全局一致性，每次只能允许一个节点的区块上链，因此需要这些节点运行一个分布式的共识协议，确定实际被写入账本的区块。

2.2 分布式共识协议/算法

分布式共识协议的作用类似于共识对象的 decide 方法，但有一些细微的不同之处。n 个节点（其中某些节点可能是恶意的，比如试图将一个含有双花交易的区块写入账本）运行一个分布式共识协议时，各自输入一个不同的值，最终协议需要从中确定一个值作为返回输出，并且这个值必须来自诚实的节点。[2]

分布式共识机制解决了区块链如何在分布式场景下达成一致性的问题。常用的分布式共识机制有工作量证明机制、权益证明机制、股份授权证明机制以及分布式一致性算法等几种。[5]

三、分布式共识协议

3.1 工作量证明（Proof of Work）

工作量证明机制要求节点在产生区块时解决一个求解复杂但容易验证结果正确性的数学难题。解决了这样一个难题的节点的区块可以上链，并被其它节点验证和承认。

工作量证明是比特币中使用的共识协议。在比特币中，有一个事先选定的 hash 函数和一个难度值。节点将一些交易打包以后，设置好指向区块链中最新区块的 hash 指针，然后在自身区块的 Nonce 域尝试填入不同的值，使得区块的 hash 值小于难度值。这是一个难以求解的数学问题，目前只能依靠算力暴力枚举所有可能的 Nonce，需要枚举的次数随着难度值高位 0 的数量的增加呈指数增长。

当一个节点成功找到这样一个 Nonce 后，它就将区块广播到网络中。其它节点都能验证这个区块的合法性（包括它的 hash 值以及其中交易的合法性），验证通过，表明其它节点承认了这一区块在链上，达成分布式共识。[1]

对于每一个普通节点来说，区块链中最长的链上整个系统花费的算力最多，也就意味着新的区块继续链接的可能性最大，因此可以保证诚实的节点都会尝试链接在它上，形成主链。

如果一个恶意节点打包了不合法的交易（如双花），其它节点不会承认，因此这个区块不会出现在主链上。如果一个恶意节点试图篡改账本历史，它需要从篡改的区块开始，重新逐块构造一个新的分支，并使得这个分支的长度最终超过主链。在诚实节点拥有大部分算力的情况下，历史被篡改的可能性极小。

工作量证明的缺点是需要消耗大量的算力，而这些算力唯一的作用是形成分布式共识；另外，区块链本身形成共识的时间较长，因此交易系统的吞吐量不会太高。

3.2 权益证明（Proof of Stake）

由于工作量证明本身消耗很多的计算资源，权益证明在 2012 年[6]被提出，一个实例是 PPCoin。

权益证明中的基本概念是币龄（coin age）。币龄的计算方法是币值×持有时间，例如持有 10 个币 90 天将得到 900 coin-days 的币龄。当这 10 个币被花掉时，它们的币龄也会同时被消耗掉。

在权益证明的协议中，节点对消耗币龄最多的链达成共识（在工作量证明中，节点对消耗算力最多的链达成共识）。一个区块所消耗的币龄分成两个部分：一部分是区块内打包的普通交易消耗币龄之和，另一部分是铸币交易消耗的币龄。铸币交易中，矿工向自己支付一定数额的币，这些币将转化为自己拥有的新币（加上一定比例的奖励），同时它们的币龄将被消耗掉。在铸币交易中还有一个 kernel input 域，类似于工作量证明中的 Nonce，用于尝试计算不同的 hash 值。与工作量证明不同的是，矿工在铸币交易中消耗的币龄越多，要达到 hash 目标的难度就越小。

与工作量证明相比，权益证明达成共识更快，也更加节能和安全。但同时恶意节点的攻击成本也降低了很多。

3.3 股份授权证明（Delegated Proof of Stake）

股份授权证明与权益证明的区别在于，交易消耗的币龄用来进行选举，由被选出的“代表”节点来完成产生和验证区块的工作。[7] 因此，在股份授权证明的共识机制中，只需要有限的“代表”节点之间达成共识即可，这极大地减少了产生和验证新区块的时间。即使“代表”节点中存在恶意节点，在新的投票中也将不会有诚实节点给它投票。

有趣的是，从权益证明机制到股份授权证明机制的发展，与人类社会发展史上从直接民主到间接民主的发展历程有一定的相似之处。

3.4 分布式一致性算法

与上述机制不同，分布式一致性算法是传统的分布式系统一直研究的内容。分布式一致性算法有许多种，如 PBFT, Paxos, Raft, Ripple, Tendermint 等算法。例如起源于拜占庭将军问题的实用拜占庭容错（Practical Byzantine Fault Tolerance）算法，在所有节点全局可见，而且有不超过 1/3 的节点恶意或宕机的条件下，剩下的节点通过 3 个阶段（pre-prepared, prepared, commit）的通信过程仍然能够达成共识，每一轮共识可以产生一个新的区块。Ripple 算法中的节点分为两类，受集体信任的 server 参加共识过程，client 产生普通交易。每个 server 维护一个唯一节点列表（unique node list），server 通过与列表中的节点通信来确定是否记录一次交易。在宕机/恶意节点不超过 20%时，节点能够维护全局一致的账本。[7]

四、总结

除了上述算法之外，还有权威证明（proof of authority，节点拥有一个身份，可以争取作为公认可信的 validator）、空间证明（proof of space/capacity，用内存/硬盘大小代替工作量证明中的算力）等分布式共识协议以及其它的分布式一致性算法。

总之，上述种种分布式共识协议/算法所关注的核心问题在于，如何在分布式的结点网络中维护一个全局一致的区块链。不难看出，在这个问题中，去中心化程度和对系统安全性的要求越高，共识算法的效率就会越低，这可能是去中心化导致的无法避免的代价。

参考文献：

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[J]. 2008.

[2] Narayanan A, Bonneau J, Felten E, et al. Bitcoin and cryptocurrency technologies: a comprehensive introduction[M]. Princeton University Press, 2016.

[3] 林惠民，吕毅，吴鹏，杨潇潇. 并发数据结构与多核编程 讲义[H]. 2018.

[4] Herlihy M, Shavit N. The art of multiprocessor programming[M]. Morgan Kaufmann, 2011.

[5] 徐海霞. 区块链技术及其应用 课件[S]. 2018.

[6] King S, Nadal S. Ppcoin: Peer-to-peer crypto-currency with proof-of-stake[J]. self-published paper, August, 2012, 19.

[7] Zheng Z, Xie S, Dai H, et al. An overview of blockchain technology: Architecture, consensus, and future trends[C]//Big Data (BigData Congress), 2017 IEEE International Congress on. IEEE, 2017: 557-564.