比特币白皮书详解

付尧: fuyao@stu.pku.edu.cn

北京大学-区块链课程

版本：1.2

日期：2021 年 11 月 27 日

摘　　要

本文主要是对比特币白皮书做一个较为深入的剖析，参考了几乎所有互联网能查到的对于白皮书的解读，也加入了自己的一些想法作为第一次课的作业。因为感觉解读的资料不够连贯和全面，于是写了这篇自己解读的文章。本文主要从四个方面解读比特币白皮书，第一部分是引论，主要是对原文整体介绍和原文的摘要以及第一章的解读。第二部分是比特币核心设计，涉及原文第二章到第七章，也是本文的重点。第三部分是补充说明，涉及原文余下的章节 (9-12 章)。最后一部分是自己的看法和参考文献[1]。本文力求把论文的初衷和想解决的问题讲明白，更多的是为了能够让读者常看常新。如果有理解有误的地方，欢迎各位批评指正。

目录

[1]参考文献只放了重点参考的三篇文献

1 引论

1.1 比特币白皮书概述

本节课程的名称是《区块链》，说到区块链的最开始的地方，绕不开的就是比特币的发明者中本聪，以及他在 2008 年发表的一篇白皮书论文《比特币：一种点对点的电子现金系统》这是学习区块链 (当然也是学习比特币) 最早最权威，也是第一手的资料。本文主要是想剖析这篇论文到底讲了些什么，以及中本聪设计比特币的初衷是什么。

中本聪，也叫 Satoshi Nakamoto，关于他的真实身份，一直是区块链领域最大的未解之谜，外界对他身份的猜想不断，也出现了种种的阴谋论，但是最终都被证伪。但是可以基本确认的是，2008 年 10 月，中本聪在一个密码学网站的邮件列表中，发表了这篇题为《比特币：一种点对点的电子现金系统》的论文，详细描述了如何创建一套去中心化的电子交易体系。而且这种体系不需要创建在交易双方相互信任的基础之上。

很快，2009 年 1 月，中本聪开发出了首个实现了比特币算法的客户端程序，并首次进行了挖矿，获得了第一批的 50 个比特币。在 2010 年 12 月 5 日，在维基解密泄露美国外交电报这一事件期间，比特币社区呼吁维基解密能够接受比特币捐款，以打破金融封锁。但是中本聪表示坚决反对，他认为比特币还处在摇篮中，经不起这种涉及政治的冲突和争议。七天后的 12 月 12 日，中本聪在比特币论坛中发表了最后一篇文章，提及了最新版本的一些小问题。随后不再露面，email 通讯也逐渐终止了。目前比特币社区主要由 Bitcoin Core 团队和比特币基金会管理。

我们回头来看看这篇可以被载入史册的论文，这篇论文一共 9 页，分 12 个章节，大约 3500 字，原版在bitcoin.org网站上可以下载。比特币是一套复杂精密的系统，它涉及了非常多领域的技术和知识，如果想要彻底搞懂比特币的原理，光看这一篇论文是不够的，还需要大量的延伸阅读。对于刚开始直接看中本聪的这篇论文的同学，门槛着实有一点高，因为内容比较抽象和跳跃，还需要一定的密码学，计算机网络，计算机数据结构等领域的知识储备，但它毕竟是作者为数不多的细致讲解比特币和区块链的文章，所以本文争取用相对简单的语言来解读这篇论文的核心思想。能让看到这篇文章的同学对比特币的设计原理有大体的了解，更重要的是，一起回顾中本聪发明比特币时的初衷。

1.2 比特币白皮书第 1 章: Introduction

比特币白皮书是中本聪在 2008 年 10 月发表的一篇论文，论文的标题是《比特币：一种点对点的电子现金系统》，在摘要中，中本聪总结了设计比特币的目的和要解决的问题。首先中本聪说，比特币的目的就是通过构建完全点对点技术实现一种电子现金系统，达到无需金融中介，也无需双方信任的情况下实现在线电

子化支付，中本聪通过数字签名 (Digital Signature)，哈希散列 (hash)，工作量证明 (Proof-of-work)，动态链式存储 (Ongoing Chain) 等技术，设计了一系列的机制，使得比特币能够解决电子支付中的双花，记录篡改等问题，并最终证明该系统安全有效，稳定可行。

图 1: 比特币的核心技术

论文的第一章是概述 (Introduction)，中本聪说，目前网上交易主要是基于信用模式 (trust based model) 交易往往由第三方金融机构来承担交易双方的信用评估、担保和资金清算工作。阿里巴巴的支付宝就是典型的这种模式。他说这种模式有两点弊端，第一点就是金融中介的存在增加了交易成本，也增加了对交易规模，交易金额的限制。第二点就是线上支付存在可逆性，因此它不适用于一些不可逆的商品或服务。

首先来说，支付的可逆性是指交易后如果发生争端，顾客可以通过第三方支付机构申请退款，第三方机构有权在违背商家意愿的情况下强制划扣其收入。当然这种情况下，买方也要退货给卖方。但是有一些商品和服务是不可退的。比如餐饮业，服务业。吃完饭总不能再吐出来，按摩完后悔了，不是想要的服务，想要退钱。所以说双方没有建立足够的信任。商家就要时刻提防客户，并且索要客户大量个人信息来防止客户欺诈。现实中这种欺诈并不少见，商家只能自认倒霉。

中本聪讲这会被记入销售成本。如果有一种不可逆的支付手段，这种收款不确定性而产生的问题是可以避免的。其实现金就是一种不可逆的支付方式。所以在某些交易中，卖家就更倾向收现金，这也就是为什么中本聪称比特币是一种电子现金系统，因为它跟现金一样，是一种不可逆的支付手段。

总结起来，比特币支付系统有三点特性。第一点 (Based on cryptographic proof instead of trust) 是用密码学代替了信用，因为人的信用不一定可靠，但是数学是可靠的。第二点 (Without third party financial institutions) 是去金融中介化，提升了效率，实现了纯粹的点对点支付。第三点 (No reverse transactions) 是杜绝可逆支付，保护卖家不被欺诈，系统也可以兼容第三方担保机制，来保护买方利益。

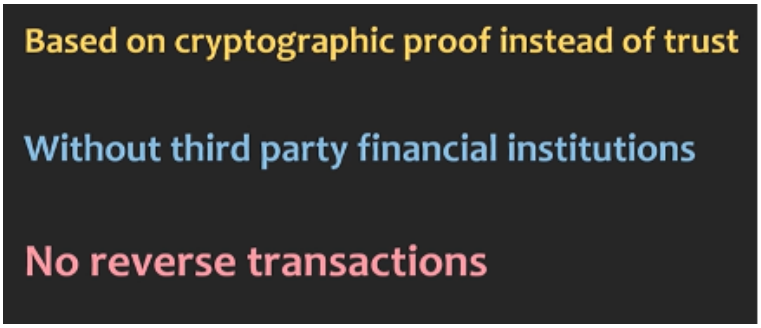

图 2: 三大特性

最后中本聪还强调，我们的系统能够通过点对点的分布式技术，基于密码学的工作量证明等机制解决双

重支付问题。只要这些节点和算力足够分散，诚实节点算力大于攻击者的算力，整个系统就能够保证安全。中本聪反复强调的双重支付问题 (double-spending problem)，简称双花问题，就是指同一笔钱，比如比特币，被重复支付两次，却不被发现，或发现时为时已晚。比特币中的各种复杂机制，主要就是为了解决双花问题。

这里关于双花问题只是简单引入，后续有详细介绍。我们可以这样理解，因为在现实中“钱”或者说“纸币”，是一种物品，无法被花多次，但是数字的世界可以很容易复制，那么确认作为数字的“钱”只可能被花一次，就变成了一个比较棘手的任务，如图3。一般的情况都是有一个权威来做仲裁 (比如阿里的支付宝)，但这又和区块链的核心思想“分布式”背道而驰。如何解决双花问题，在我看来就是比特币甚至是区块链最初诞生的核心，后续我们会重点讨论这个问题。

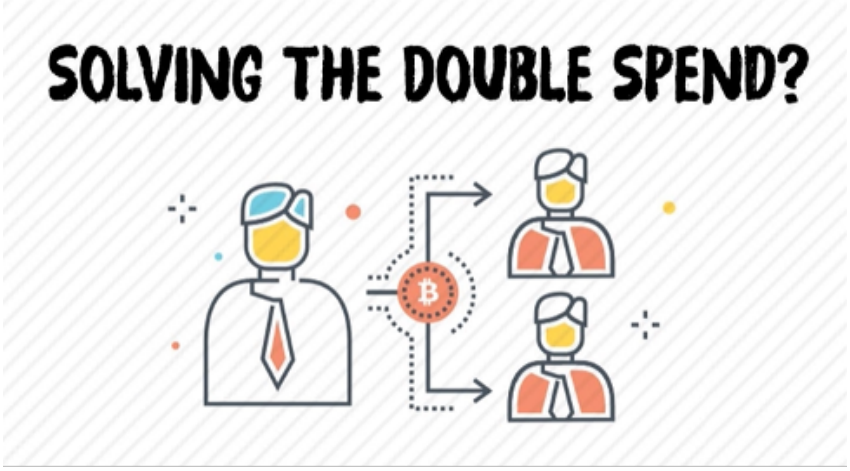

图 3: 双花问题

至此我们就讲清楚了比特币和区块链诞生的初衷，接下来就是白皮书较为核心的部分了，主要介绍了中本聪对于各种机制的设计。

2 核心设计

2.1 比特币白皮书第 2 章: Transactions

论文的第二章是交易 (transactions) 这一章讲了比特币中非常重要的概念，就是交易和记账体系，因为对一个货币系统，如何完成收付款，验证交易的有效性，以及存储交易记录，是必须要解决的问题。中本聪说，我们定义的电子货币是一条数字签名链，每个拥有者都通过将上一次交易和下一个拥有者的公钥的哈希值的数字签名添加到此货币末尾的方式将这枚货币转移给下一个拥有者，收款人可以通过验证数字签名来证实其为该链的所有者。

这里也就是区块链的核心了，但是原文中讲的比较抽象，我们具体来看一下。首先，常规的记账体系是基于账户和余额。比如银行系统中，我们每个人有一个账户，账户里面记录了余额，同时记录了我们每一笔交易。这些交易对应了账户余额的增减，核心是账户余额，附属的可以得到所有的交易明细。但是比特币是基于中本聪称为 UTXO 的记账模型 (Unspent Transaction Output) 未花费交易输出，这是一种基于交易的记账体系。在这个体系中，数据库记录了一笔笔的交易，包括收付款双方的信息、金额、上一笔交易的哈希值等信息。如果要计算某个账户拥有多少电子货币，需要在帐本中找到其参与的交易，来计算该用户当前的余额。在这个体系中，核心是每一笔交易，附属的可以得到账户余额。

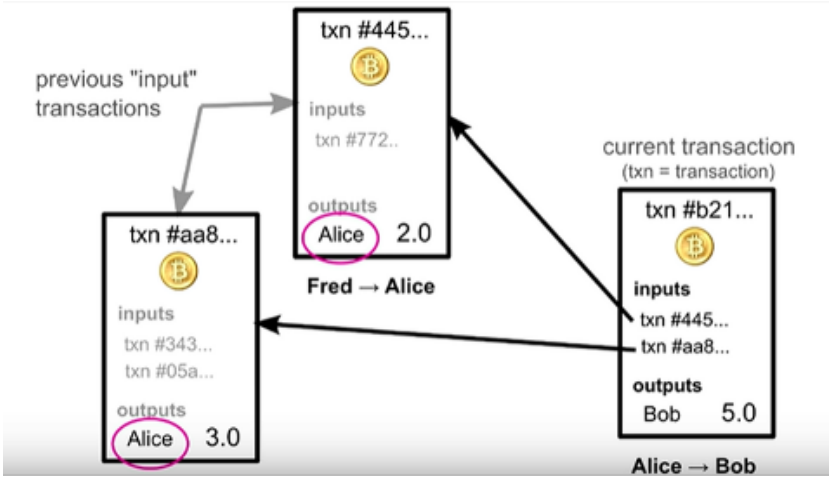

图 4: 简化的交易信息

这里的配图来自论文原文 (图4)，表示了一笔笔交易构成的链式结构 (其实比特币每个区块中，存储的就是大量的这种交易数据)。

我们来看一次具体的交易过程，图5中记录了三笔交易完成了从 0 号拥有者到 1 号拥有者，最终到 3 号拥有者的转账数据。

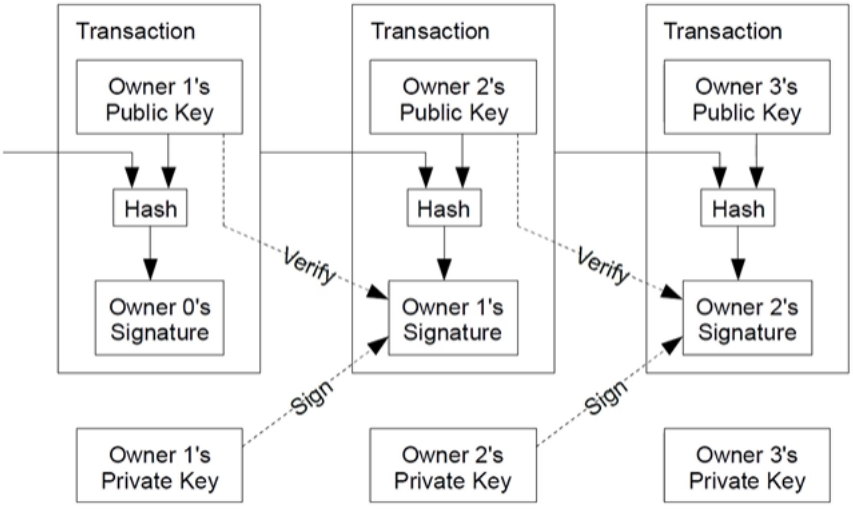

图 5: 交易过程

现在我们来看图中的中间这笔交易，付款方是 1 号拥有者，收款方是 2 号拥有者。这笔交易首先记录了 2 号拥有者的公钥，我们将第一笔交易的全部数据拼接 2 号交易者 (也即收款方) 的公钥，生成一个哈希值，然后用 1 号拥有者 (也即付款方) 的私钥进行签名 (sign)，也打包在记录中，这就记录了一次交易的主要数据。那么这笔交易是否真实有效呢，也就是付款方的钱是真的吗？其实，收款方可以用付款方的公钥来验证他的签名是否是其本人，因为所有人的公钥都是公开的。后面的交易也是一样的原理，以此类推。以上就是区块链核心中的核心了，如果读者朋友们不太理解，那就是对于密码学的一些基础知识不太理解了，这里做一下解释：

这个过程涉及密码学的三个概念：公钥、私钥和数字签名。其中，基于非对称加密算法会得到一对公钥

和私钥，公钥通常是公开的信息，负责数据加密，私钥是非公开的，只有解密权的人拥有，用于解密和签名。

非对称加密算法是一种算法，比较常见的有“RSA”和“迪菲-赫尔曼椭圆曲线”(ECDH)[2]。和对称加密算法中只有一个秘钥不同，这类算法的核心就是生成一对互相关联的公钥和私钥，信息的加密和解密是两个过程，分别用到了这两个秘钥，一个完整的信息传递过程需要同时用到这两个信息。

关于公钥和私钥最核心的需要满足以下两个特点：

```
1. 公钥加密的信息，只能由对应的私钥解开
2. 私钥加密的信息，只能由对应的公钥解开
```

其中第一条特点，常被用来「传递信息」，第二条特点常被用来「数字签名」

1. 传递消息：当 Alice 想给 Bob 传递消息时，只需要用 Bob 的公钥加密这条消息，那根据以上第一个特点，理论上来说就只有 Bob 的私钥可以解开这个消息。
2. 数字签名：当想证明一个东西的归属时，Alice 将这个东西用自己的私钥加密，那么任何人想要验证这个东西属于 Alice，就可以用 Alice 的公钥解 (根据以上第二个特点，理论上只有 Alice 的公钥才能解开)，同时也验证了 Alice 当初用私钥加密的东西没有遭到破坏。

再次回看图5，图中包含了两个过程，第一个过程是付款方用收款方的公钥进行加密，进行信息传递，第二个过程是收款方用付款方的公钥来验证这笔交易是付款方发起的，也就是数字签名的验证过程。

比特币使用椭圆曲线算法，生成公钥和私钥。公钥是公开信息，用于标识用户，验证比特币的所有权，私钥只有持币者拥有，保证了比特币的归属权，用于转账交易。

如果仍然理解起来有困难，我们可以粗暴的把公钥想象成银行卡号，私钥理解成银行卡密码[3]。实际上公钥的确对应了每个人比特币钱包的地址，也就是收款账号，而私钥用于签名和授权付款。比特币的数字签名就是持有比特币的付款方，生成的一段防伪字符串，通过验证这个字符串，一方面证明该交易是付款方发起的，同时证明交易信息在传输中没有被篡改。

比特币系统中涉及许多密码学的概念。正如中本聪所说，他希望用密码学代替信用。

通过数字签名的技术，收款人已经能够验证付款方身份的真实性，但是不能证实付款方没有进行双重支付。也就是说付款方是否同时给多人支付了这枚电子货币。中本聪说通常情况下会引入一个可靠的第三方机构，比如央行或者铸币厂，由他们来负责发行、结算和交易审核，这样就可以解决上述问题。但是比特币的使命就是不再依赖第三方金融中介，所以这种方案肯定是不行的，另一种方案就是不如让收款人知道付款人最近的其他交易，这样就可以判断在这笔交易前，付款人有没有造成双重支付的其他交易，如果系统不依赖任何第三方来审核账本，那么就需要公开所有的交易数据，让所有的参与者来监督和维护一个共同的账本，并达成一种历史共识，最终只会有一个有效的，带有时间顺序的记账数据公之于众。这就是比特币使用的分布式记账系统。

2.2 比特币白皮书第 3 章: Timestamp Server

论文的第三章是时间戳服务器 (Timestamp Server) 这一章很短，只有一段话和一张图，主要是强调将时间戳信息纳入区块链信息中，所谓时间见证了一切。图中就是一个个区块构成的链，也就是区块链[4]。比特币大概每 10 分钟产生一个新的区块，每个区块中记录了这段时间的所有的交易信息，也就是图6中的这些 item，

[2]这里只做简单解释，深入讨论不在本次的研究范围内，不过还是建议能够看一下 RSA 的相关实现，在其他很多领域都有应用

[3]这里之所以说“粗暴”，是因为这样只能解释“数据传递”，无法解释“数字签名”

[4]这里只是一个简化版本，方便理解，真正的存储结构将在第 7 章介绍

当前区块的数据加上前一个区块的哈希值，再打上时间戳，就构成了当前区块的哈希，这个哈希还要参与下一个区块哈希的运算，这就让所有区块像编年史一样，按照时序紧密连接，你中有我，我中有你，并同步到所有节点。因此历史区块信息几乎无法篡改，因为当要修改某个区块数据，就要修改其后所有已生成区块的哈希数据，还要让大多数节点形成共识，我们后面将证明这几乎不可能 (论文 11 章)。

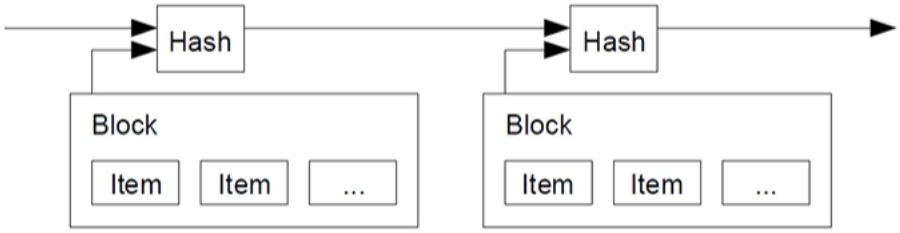

图 6: 区块链

2.3 比特币白皮书第 4 章: Proof-of-work

论文的第四章是工作量证明 (proof-of-work) 简称 POW。这里也是非常重要的一章，因为会解释我们现在非常疯狂的现象“挖矿”到底在挖什么？为什么要挖矿？挖出来的矿有什么价值?

前面提到，整个账本由全体分布式节点共同维护，那么接下来就面临谁拥有记账权的问题，中本聪通过引入工作量证明机制，公平有效的解决了这个问题。POW 理念早在亚当贝克的 Adam Back 的哈希现金 (Hashcash) 中曾经使用过。中本聪涉及的工作量证明，就是让所有节点去计算一个哈希数值，并且尽可能地小，满足图7中的不等式，这个计算的复杂程度，随着 target 目标值的变化而动态调整，目标值越小，难度越大。

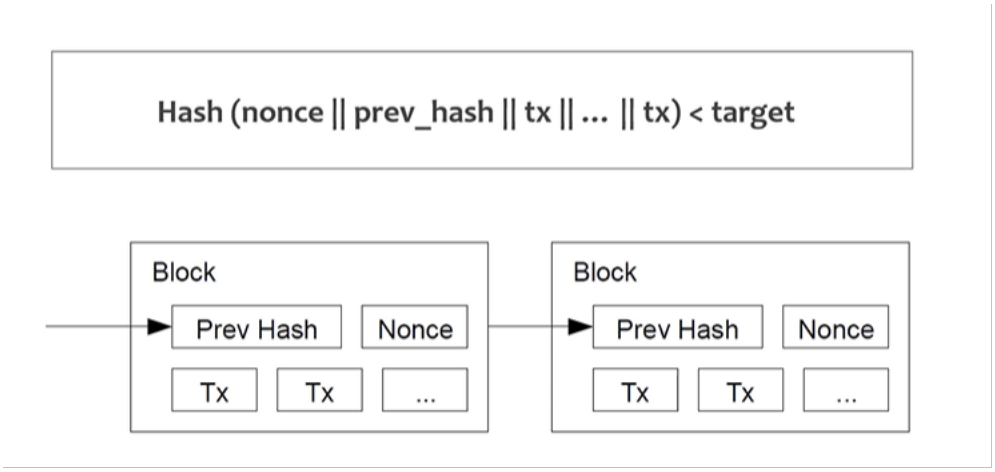

图 7: 工作量

实践中会让这种复杂度维持在合理的计算成本下，大概 10 分钟左右计算出来一次的水平，这其实就是矿工挖矿的过程，大量遍历 SHA-256 哈系数，寻找一个解，这种算法的好处是，遍历起来十分麻烦，但是验证起来却非常简单，只需执行一次哈希即可。为了保证每次工作量证明都是重新计算，每个区块还会加入一个随机数 Nonce，正如图中所示，每个区块中就包含了前一个区块的哈希，一个随机数，以及所有交易数据，

这样我们的区块结构也逐步在完善。

中本聪在这一章节的最后又进一步的解释到，工作量证明还解决了在多数决定中确定投票方式的问题。工作量证明本质是按照 CPU 算力进行投票，最长的聊代表了最多数的投票结果，因为它有最大工作量证明的算力投票到这条链上，如果多数 CPU 算力被诚实节点控制，那么诚实的链就会增长的最快，并超过其他的竞争链。

前面提到过，要修改过去的一个区块的信息，那么其后已产生的区块信息都要修改，因为他们包含了前面区块的哈希值。从这一章我们知道，修改区块并没有那么简单，因为需要记账权，也就是工作量证明，这个成本及其高，也就是说要修改过去的某个区块，攻击者必须要重做这个区块，以及其后所有区块的工作量证明，从而赶上并超过诚实节点已达成共识的最长链，这种概率会随着后续区块的增长而大大减小，成本也会远远大于收益，我们会在 11 章证明这个观点。

为了抵消硬件运算速度的增加，以及平衡不同时期运行节点的利益，工作量证明的难度将由移动平均数法来确定每小时生成区块的平均个数。这里中本聪考虑到了摩尔定律，因为几年后或者几十年后，运算能力会大幅提升，但是没有关系，如果区块生成的过快，系统会让区块生成的难度增加，最终还是保持在 10 分钟的水平。

在这一小节的最后，我们来回答最初的那几个问题，如果这一小节都没有看懂也没有太大的关系，只需要知道以下的几个结论：

1. *矿工挖矿到底在挖什么*：要找到一个值，使得基于这个值计算出的哈希值满足给定最大值的条件 (比给定值小)。
2. *为什么要挖矿*：挖到矿的节点 (第一个计算出这个值的人) 可以获得记账权 (记最近这十分钟发生的交易)。这里很明显有两个问题，为啥要抢记账权，是有什么好处吗 (第 6 章)，第二个问题是如果真有什么好处，那如果两个人几乎同时找到了值 (挖到矿)，算谁的？而且第二个问题会由于网络传输的不稳定性增大这种可能 (第 5 章)。
3. *挖出来的矿真的有价值吗*：挖出来的矿没有价值，只是一个数字，但是中本聪人为的给他赋予了一个价值，同刚才说的第一个问题一样，将在第 6 章进行解释。

2.4 比特币白皮书第 5 章: Network

论文的第五章是网络 (Network)，如何设计一个网络，来部署整个分布式系统呢。熟悉计算机的同学可能知道，分布式系统最难的部分就是设计一个“共识算法”，Raft 算法和 Paxos 算法都是为了解决此类问题[5] 这也是在和人打交道时候最难的问题，如何运营与管理组织 (在分布式网络中就是都听 leader 的意见最终能够达成共识)。

那中本聪是如何设计有效的机制，来部署整个比特币的分布式节点网络呢？在第五章中，中本聪将运行整个比特币网络分为以下 6 个步骤：

1. *广播交易*：每一笔新的交易都要向所有节点广播。
2. *收集交易*：每个节点要将新交易收集到一个区块中。
3. *工作量证明*：每个节点要为它的区块寻找工作量证明。
4. *广播区块*：当一个节点找到了工作量证明，就要向所有节点广播这个区块。
5. *验证区块*：节点必须在验证区块内所有交易都是有效的，而且没有双重支付的情况下，接收这个区块。

[5]Raft 算法是对 Paxos 算法的改良版，业界用的较多，感兴趣的读者可以自行探索

6. 接受区块：最后一步，节点使用这个区块的哈希值，作为上一个区块的哈希值，记录在新区快中，这就表示接受了这个区块。

一个诚实的节点必须要遵循以上六点要求 (事实上诚实的节点就在周而复始的运行这六个步骤)，不太严谨的总结就是，1. 系统下发工作量，2. 大家力争当领导，3. 领导带领大家记账的循环。

由于节点总认为最长的链是正确的，并持续维护和延长它。所以如果两个节点同时广播了不同的新区快，有些节点可能先收到其中一个，其他节点先收到另一个，这种情况下节点基于先收到的区块工作，但也保存另一个分支，以防其变为更长的链，就是要脚踏多条船。当下一个工作工作量证明被找到后，僵局就会被打破，从而其中一个分支变得更长。在另一个分支上工作的节点，将会切换到更长的链上，就是说节点要学会见风使舵。新交易的广播不必到达所有的节点才生效，只要达到部分节点，不久就会打包进区块中，区块广播也是能容忍一定程度信息丢失。如果一个节点没有收到某个区块，那他将在收到下一个区块时，发现它丢失了一个区块，然后再去请求这个区块。所以比特币系统有很强的容错性和鲁棒性的。

和计算机要求的比较严格不同，比特币的网络中会经常存在的大量不同的节点，它并不强调实时的一致性，更像“最终一致性”。[6]

现在我们来回答在上一个小节中提出的问题：如果两个人几乎同时找到了值 (挖到矿)，算谁的？答案是都算，收到了就保留，之后总会有一条链开始变得更长。

2.5 比特币白皮书第 6 章: Incentive

论文的第六章是激励 (Incentive)，是对于挖矿的激励，为了推动矿工挖矿，中本聪人为的赋予“矿”(之前提到的那个哈希计算前的值) 一个价值。根据我们上一个小节的总结，矿工 (诚实的网络节点) 的成本主要是 CPU 时间和电力，存储成本，网络传输成本[7]

具体的过程是这样：约定每个新区快中的第一笔交易是特殊交易，专门用于奖励给区块创建者的新货币。这样不仅增加了对网络节点的激励，也提供了一种分发新货币到流通领域的方法。因为我们的系统中没有中央机构来发行货币，新货币按固定的量，稳定的产生和分发。就像金矿矿工消耗资源，并增加黄金到流通域一样，在这里消耗的是 CPU 时间和电力，这一部分换来的就是挖矿奖励。还有一部分激励是交易手续费，比特币交易没有强制规定手续费的金额，只要交易的输出值小于输入值，差价就作为交易费，被加到包含此交易的区块奖励中。这里需要解释一下为什么会出现输出值小于输入值的情况，是因为在比特币中，我们认为的金额自动合并不会发生，如果 Bob 给 Alice 转了 3 个比特币，Rose 给 Alice 转了 2 个比特币，那么 Rose 就有了 3 个比特币和 2 个比特币，而不是 5 个比特币，如果 Alice 想花 1 个比特币的话，用 2 个比特币就会导致输入是 2 个比特币，输出是 1 个比特币，差价的 1 个比特币就变成了手续费。

一旦预定量的货币进入了流通领域，激励将变为只含有交易费，这样可以避免通货膨胀。总结起来，比特币的激励 = 挖矿奖励 + 交易手续费，发生时间就是每个区块的第一笔交易奖励给矿工。这里解释一下，在实践中，比特币发行总量大约 2100 万枚，每个区块的产量，也就是矿工的收益，每 4 年会减半一次，所以当产出的比特币越来越少，或者所有币发行完了，挖矿奖励就只剩下交易手续费的部分了。

同时，这种激励也会有助于鼓励节点保持诚实，比如一个攻击者，如果它有能力聚集比所有诚实节点更多的 CPU 算力，而他的目的仅仅是完成一次双花攻击，或者是完成一次欺骗交易，这是非常不值的，因为如果他有这个能力，还不如做矿工，可以获得稳定的挖矿收益，所以说这个机制让遵守规则比破坏系统有更高

[6]最终一致性是计算机中的概念，这里并不完全一致

[7]通过后续章节的总结，我们会发现存储成本和网络传输成本很低，主要是 CPU 算力的成本

Time	Mining reward / block	Block height
2009.1	50	0
2012.11	25	210000
2016.7	12.5	420000
2020.x	6.25	630000
2024.x	3.125	840000
2028.x	1.5625	1050000
2032.x	0.78125	1260000
2036.x	0.390625	1470000
2040.x	0.1953125	1680000

图 8: 每四年减半

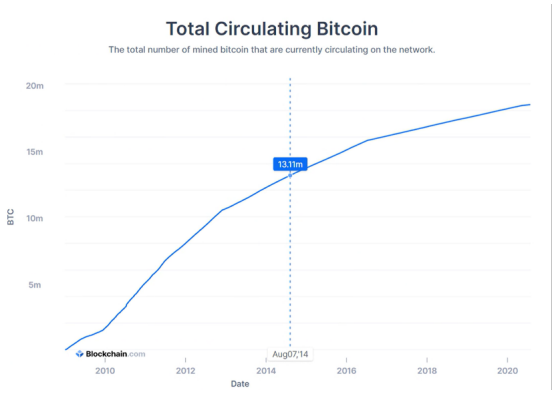

图 9: 发行总量

的收益。

以下两张图，图8表示了比特币每次奖励的变化，图9表示了发行量随着时间的变化情况。

总结一下，中本聪人为的给“矿”定了一个价值，这个价值驱动了整个比特币帝国的运转，除了最重要的激励矿工进行挖矿，运转整个比特币帝国，另一方面也通过这种方式让比特币在市场流通，最后也让有更多算力的节点，有更大的动机做诚实的节点，而不是搞破坏。

2.6 比特币白皮书第 7 章: Reclaiming Disk Space

论文的第七章是回收磁盘空间 (Reclaiming Disk Space)，这一部分和下一章节的内容解释了两个事情，第一为什么之前计算矿工成本的时候，存储成本不重要，既然所有的交易都要存，存储的成本和同步的成本应该都很高。第二个事情就是所谓的“区块链”，区块中究竟存储的是什么。

理解这一章需要一些前置的计算机知识，哈希和二叉树[8]。这一章中本聪主要讲如何利用一种叫默克尔树的二叉树数据结构，来压缩数据，节约交易数据的存储成本，因为我们知道，每个完整的节点要保存所有的历史交易数据，这个数据量可能会很大。如图10中所示 (论文原图)，就是区块链中一个区块的数据结构形式，这是一个默克尔树的树状结构。

关于默克尔树，我们用另一个图来详细说明，图11由一个根节点，一组中间节点和一组叶节点组成。叶节点包含存储数据和它的哈希值，中间节点是它的两个子节点的哈希值，根节点也是由它的两个子节点内容的哈希值构成，所以默克尔树也被称作哈希树。当一笔货币的最新交易被足够多区块覆盖，也就是形成共识的时候，那么这个区块中，它过去拥有者的交易数据就可以丢弃，只保留最新的数据，以及之前数据根节点的哈希值，同时哈希值没有被破坏，依然可以验证数据的有效性，也就是零知识证明。所谓零知识证明就是说，验证一个事件正确与否，并不需要验证者重现整个事件。举个例子来说，我们下载资源经常看到资源附有一个 MD5 的字符串，通过短短的 MD5 字符串就能验证，当前下载的资源，与源资源相比，是否完整一致，这就是零知识证明。默克尔树的哈希值，既不占据空间，又能提升索引效率，还能够用于零知识证明。

中本聪还通过举例来做了一笔估算，他说每个不包含交易的区块头大约是 80bytes，如果每十分钟生成一个区块，每年会生成 4.2MB 数据，在当时 (2008) 年在售的典型计算机有 2GB 内存，而且根据摩尔定律的预测，每年内存增加 1.2GB，所以就算区块头一定要存在内存里，存储也不是问题。本章的核心就是将大量数据进行哈希运算后，来增加其分布式索引的性能，并且通过维持一个较小的高效索引，也就是默克尔路径，进而来管理复杂的大量数据。

[8]本文不再额外解释，感兴趣的读者自行搜索

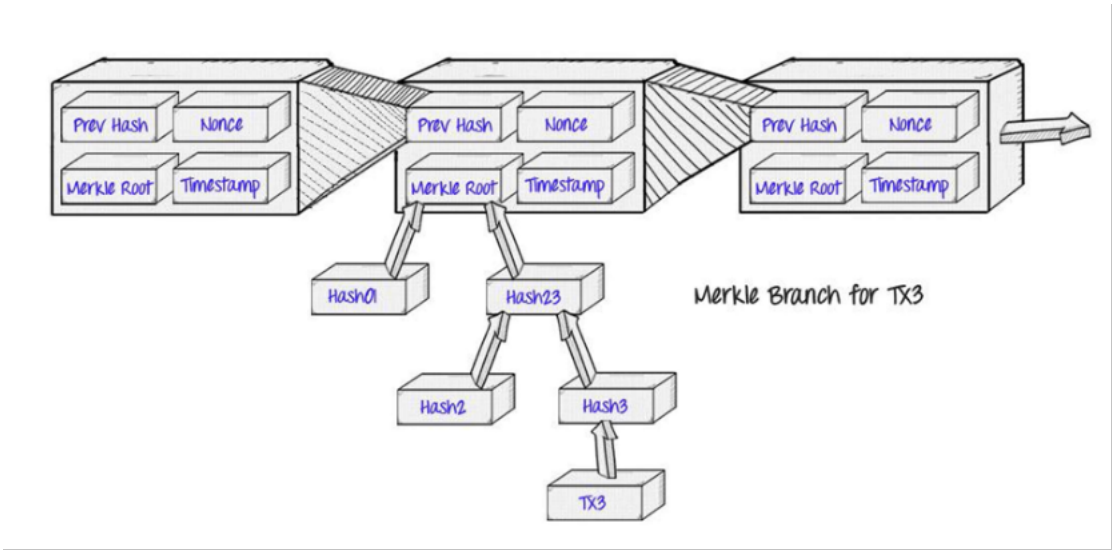

图 10: 带有默克尔树的区块链

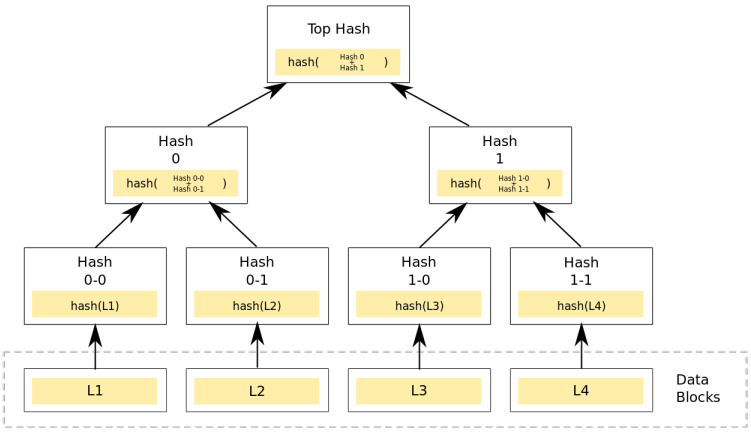

图 11: 默克尔树

至此，我们就完成了最重要的第二部分: 比特币的核心设计的解读，余下的是一些额外说明补充，涉及原文的 8 到 12 章。

3 补充说明

3.1 比特币白皮书第 8 章: Simplified Payment Verification

论文的第 8 章讲的是简化的支付验证，simplified payment verification，简称 SPV，是指不用运行一个完整的网络节点，也可以进行支付验证的方式，我们经常看到一些比特币钱包，介绍自己是 SPV Wallet，或者叫做轻钱包，就是这个意思。一个完整的比特币节点，或者叫做全节点，至少需要几百 G 的磁盘空间，数据还需要实时的同步和更新，这对普通用户来说门槛太高了，因此中本聪介绍了简化支付验证的可能性，实际上用户只要有最长共识链的区块头数据就行了，它可以通过向其他网络节点查询来确保他有了最长链的数据，然后将交易链接到对应的默克尔分支，直到有节点接受了这笔交易，以及后面的区块确认此交易，并最终被

整个网络接受。如果面临攻击，这种简化验证就会有较高风险，实际上这种简化认证的钱包客户端也要设计一定的机制来保证安全性，对于高频的首付款公司或者企业级服务商，中本聪还是建议维护一个自己的全节点。

到目前我们就可以给出整个区块的存储了，如下图12，和之前的不同点主要是在区块头中增加了默克尔书的根节点 (代表了整棵树的哈希)，如果是个全节点，就会存储整棵默克尔树 (主要是叶子结点的数据)

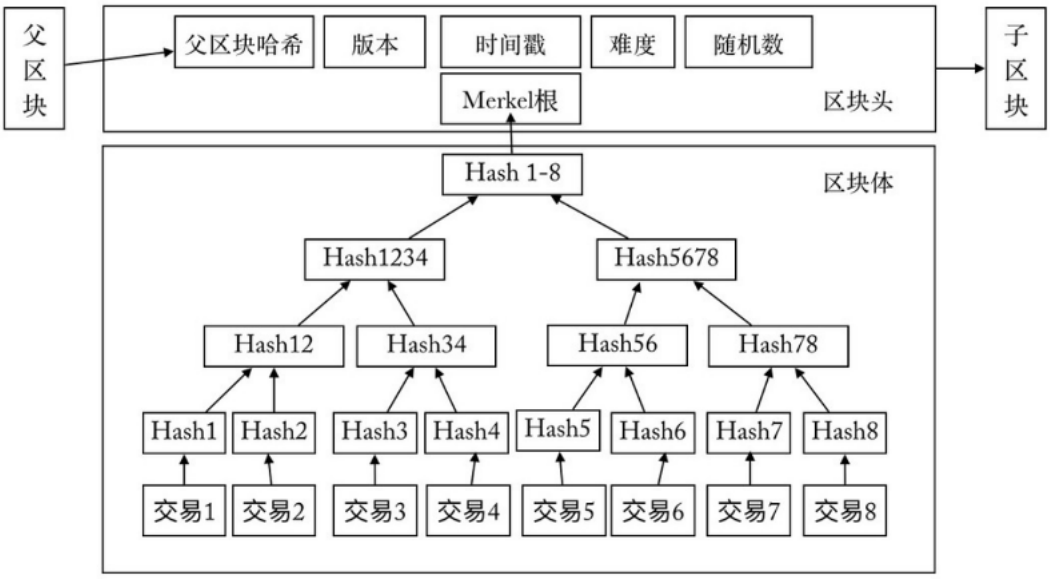

图 12: 区块存储

至此我们也可以回答上一小节提出的问题，为什么存储空间并没有太耗费资源，就是因为网络中大多数节点并不用存储全部信息，只需要存储根节点 (整棵树的哈希)

最后我们讨论一下简化的支付验证具体的过程。以上图为例，当节点需要进行支付验证的时候，比如需要验证交易 4，只需要提供 hash3，hash12，hash5678，就可以得到根节点的哈希[9]，再和其他节点中保存的区块头中的根节点哈希进行比较就可以知道交易 4 是否在整个链路中了。

3.2 比特币白皮书第 9 章: Combining and Splitting Value

论文的第 9 章讲的是合并和分割交易额 (Combining and Splitting Value) 这一章比较简单，主要补充说明了如下事实：实际上每笔比特币交易可以有多个输入方和多个输出方，也就是多个付款方和多个收款方，收款方还可以包括付款方自己，也就是找零的过程。对于用户来说，合并交易的好处不仅可以提升效率，也可以节约手续费。

3.3 比特币白皮书第 10 章: Privacy

论文的第 10 章是关于隐私的问题 (Privacy) 前面提到比特币不同于传统金融机构，必须要公开完整的账本，也因此所有的交易都是公开的，这就涉及了隐私的问题。对于传统机构也就是银行来说，作为交易的中介机构，用户交易信息只有用户本人和银行可以获取和提供，这一定级别上保护了用户隐私。比特币是通过

[9]可以得到 hash4，配合 hash3，可以得到 hash34，配合 hash12，可以得到 hash1234，以此类推

保持公钥的匿名性，来保护用户隐私。也就是说用户无需注册和实名审核，在匿名的状态下，就可以获得比特币账户。这样就打破了真实身份信息和交易之间的关联，公众能看到有人正在发送一定量货币给别人，但是不能将交易关联到某个人，举一个金融市场的例子: 股票交易，股票市场中每笔交易的时间和交易量 (行情) 是公开的，但市场上的每个参与者都看不到交易双方究竟是谁。

3.4 比特币白皮书第 11 章: Calculations

论文的第 11 章是计算 (Calculations)，本章主要是将通过公式和代码来论证，在实践中比特币网络被攻击成功的概率非常低。我们假设一个攻击者试图生成一条比诚实链更快的替代链，假设这个目标达到了又能怎么样呢？能不能凭空创造比特币，或者拿走别人的比特币呢？这个答案是否定的，因为这需要破解别人的账户私钥，这是一个难度极高的数学问题，如果没有私钥，这就是一笔无效交易 (前面提过交易的最后一步是付款方的私钥签名)，诚实的节点不会接受无效的交易作为支付，诚实节点永远也不会接受一个包含无效交易的区块，这在第 5 章网络中提到过。所以，攻击者只可能通过改变他自己的某笔交易，来拿回他不久前已经支出的钱，也就是通过双重支付来完成攻击。

本章剩余部分就是通过建立概率模型来估算这种概率，后面的证明略微复杂，如果没有心情的往下看证明推导的同学，可以记住结论，结论就是就是，如果攻击者找到区块的概率无法超过诚实者，那么这个概率很小，而且随着落后的区块越多，概率呈指数级减小。

以下是具体证明[10]：中本聪说，诚实链与攻击链之间的竞争可以被描述为二项随机游走模型 Binomial Random Walk，约定成功事件是指，诚实节点被延长一个区块，也就是两条链的差距 +1，失败事件是指攻击链延长一个区块，也就是两条链的差距-1。这就好比两边在赛跑，那么攻击者从某一落后位置赶上诚实链的概率 (因为攻击者重做了某个区块，需要重做该区块后面的所有区块，相当于从落后追赶) 类似于赌徒破产理论 Gambler's Ruin Problem。就像一个拥有无限信用的赌徒，从一定亏损开始，或者说从落后开始，进行无限次的赌博，试图达到盈亏平衡，我们可以计算他达到盈亏平衡的概率，也就是一个攻击者赶上诚实链的概率。

这里我们假设，p 是诚实节点找到下一个区块的概率，q 是攻击者找到下一个区块的概率，q(z) 表示攻击者从落后 z 个区块赶上诚实链的概率，那么当 p<q 的时候，q(z)=1，因为攻击者总是能够更快找到下一个区块，那么在落后有限个区块的情况下，就一定能追上来。当 p>q 的时候，$q(z) = (q/p)^z$，因为我们可以把每一步的追赶当作一次独立事件。假设 p>q，概率 q(z) 将随着攻击者需要赶上的区块数的增多成指数级下降。如果他没有在早期快速赶上，那么他落后的越远，赶上的机会就越渺茫。

中本聪举了一个例子，我们考虑一个新交易的收款人要等多久才能确保付款人不能够再改变这个交易，这个问题很现实，就是收款人"落袋为安"的时间期望值。我们假设付款人是攻击者，他想让收款人相信他已经完成付款，然后再一段时间后改编成支付回他自己，或者支付给其他人，最终收款人会受到警告，但付款人希望对方收到警告时为时已晚。其实在每次首款前，收款人都可以临时生成一对新密钥，然后将公钥，也就是收款地址，给付款人，这样付款人就不会提前直到收款人的常用收款地址，也就无法提前对交易进行签名，这样能防止付款人 (攻击者) 预先准备好一条区块链，然后执行交易。总之无论怎样，交易一旦发出，不诚实的付款人就开始秘密的在一条包含他替换版交易的链上搞事情了。

假设收款人等到交易被加到区块中，而且后面追加了 z 个是诚实区块，他虽然不知道攻击者确切的进度，但我们假设城市区块按平均时间生成，大概 10 分钟一个，攻击者可能的进度将是一个泊松分布，期望值表示攻击者追上多少个区块，为了计算攻击者当前仍然能赶上的概率，我们给每个他可能达到的进度的泊松密度乘以他在那个进度时能赶上诚实链的概率，论文原文证明如图13。

[10]这里的证明不复杂，是一个简单的泊松分布的推导，读者不必再次纠结时间，可以直接跳过

The recipient waits until the transaction has been added to a block and z blocks have been linked after it. He doesn't know the exact amount of progress the attacker has made, but assuming the honest blocks took the average expected time per block, the attacker's potential progress will be a Poisson distribution with expected value:

$$\lambda = z\frac{q}{p}$$

To get the probability the attacker could still catch up now, we multiply the Poisson density for each amount of progress he could have made by the probability he could catch up from that point:

$$\sum_{k=0}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \cdot \begin{cases} (q/p)^{(z-k)} & \text{if } k \le z \\ 1 & \text{if } k > z \end{cases}$$

Rearranging to avoid summing the infinite tail of the distribution...

$$1 - \sum_{k=0}^{z} \frac{\lambda^k e^{-\lambda}}{k!} \left(1 - (q/p)^{(z-k)}\right)$$

图 13: 泊松分布的相关证明

左边是泊松分布的概率密度函数 (攻击者能达到 k 个区块进度的概率)，右边是之前计算过的当达到 k 个区块时，还能最终赶上 z 个诚实区块的概率。这两部分相乘得到所有的事件概率，然后对所有的事件概率积分得到总概率值，再通过变换来避免无限的尾部求和，得到最终的结果。

最后中本聪还给了一段 C 语言代码和运行的数据结果，如图14。结论就是，当收款人被攻击的交易后面增加了 z 个区块后，攻击者的虚假链仍然能赶上诚实链的概率，随 z 的增长呈指数下降。

```
#include <math.h>
double AttackerSuccessProbability(double q, int z)
{
    double p = 1.0 - q;
    double lambda = z * (q / p);
    double sum = 1.0;
    int i, k;
    for (k = 0; k <= z; k++)
    {
        double poisson = exp(-lambda);
        for (i = 1; i <= k; i++)
            poisson *= lambda / i;
        sum -= poisson * (1 - pow(q / p, z - k));
    }
    return sum;
}
```

```
P < 0.001
q=0.10   z=5
q=0.15   z=8
q=0.20   z=11
q=0.25   z=15
q=0.30   z=24
q=0.35   z=41
q=0.40   z=89
q=0.45   z=340
```

```
q=0.1
z=0    P=1.0000000
z=1    P=0.2045873
z=2    P=0.0509779
z=3    P=0.0131722
z=4    P=0.0034552
z=5    P=0.0009137
z=6    P=0.0002428
z=7    P=0.0000647
z=8    P=0.0000173
z=9    P=0.0000046
z=10   P=0.0000012

q=0.3
z=0    P=1.0000000
z=5    P=0.1773523
z=10   P=0.0416605
z=15   P=0.0101008
z=20   P=0.0024804
z=25   P=0.0006132
z=30   P=0.0001522
z=35   P=0.0000379
z=40   P=0.0000095
z=45   P=0.0000024
z=50   P=0.0000006
```

The probability drop off exponentially with z!

攻击者的虚假链能赶上z个诚实区块的概率，随z的增长呈指数下降!

图 14: C 语言代码

3.5 比特币白皮书第 12 章: Conclusion

论文的最后一章是总结部分 Conclusion，中本聪总结说，我们已经提出了一种不依赖信任的电子交易系统，参考了通用的数字签名货币体系，这种体系虽然提供了强有力的所有权控制，但是由于缺乏防止双重支

付的方法而不够完善。为了解决这个问题，我们提出了一种工作量证明机制，并在点对点的分布式网络中记录公开的交易数据，只要诚实节点控制了多数的 CPU 算力，对于攻击者来说，交易历史将很快在计算上变得几乎不可再被更改。他还总结说，这是一个结构简洁而健壮的网络，节点只需很少的协同就能同时工作，他们不需要被认证，因为信息不会被发送到某个中心位置，只需要被广泛的传播。所以节点可以随时的离开和重新的加入网络，节点使用 CPU 算力来投票，通过努力延长有效区块来表达接受，通过拒绝在无效区块 (包含无效交易的区块) 上工作来表达抵制，任何必要的规则和激励，都可以通过这个共识机制来加强。这也就是区块链的核心思想，一种去中心化的经济学。

4 思考与感悟

4.1 思考与感悟: 比特币与区块链的发展

至此，关于比特币白皮书的解读就告一段落了。其实，我们提到的中本聪，很多技术都不是他首创，他最核心的贡献就是把前人的经验加以总结，并以一个真实的货币将所有技术都串了起来。

早在早在 20 世纪 30 年代，加密货币的最初设想就已经出现了。

1. 1982年：大卫乔姆提出了不可追踪的密码学网络支付系统，事后诸葛亮来看，这其实才是今天比特币的老祖宗。
2. 1991年：Stuart haber 和 Scott stornetta 发表论文: 《How to TimeStamp a Digital Document》他们提出用时向戳确保数字文件安全的协议，这也时是今天所谓区块链链条的雏形。
3. 1991年：菲尔齐默尔基于 RSA 公钥加密体系开发了一个邮件加密系统 PGP, 它能够保证邮件内容不被篡攻。
4. 1997牛：亚当拜克 (Adam back) 发明了一种哈希现金 (Hashcash) 算法机制，而哈希算法在比特币的白皮书中, 就被中本聪用来解决了零信任基础的共识问题。
5. 1998年：戴伟（Wei Dai）提出了匿名的、分布式的电子加密货币系统 B-money。在比特币的官网上，B-money 被认为是比特币的精神先导，中本聪与他的交流很多。

从这里的历史来看，似乎看到了牛顿总结了伽利略、惠更斯、笛卡尔、开普勒等人的贡献，写出《原理》一样的那种传承，带来的震撼。

4.2 思考与感悟: 工作量机制的讨论

回到这篇论文，通过本文的大致解读，我们似乎跟着中本聪走了一遍他的思路，想提出的第一个探讨是关于工作量机制，也就是挖矿算是比特币中最为人所熟知的部分。如果你不挖矿，那么你可能是深度学习从业者，如果你也不从事计算机行业，那你可能玩游戏，如果你也不玩游戏，那也一定听说过比特币挖矿，和显卡涨价

关于中本聪设计的核心之一：工作量证明机制，第一次看到这里的我很疑惑，为什么要设计一个没有意义的谜题，让机器去解，计算机又不需要训练脑力，如果真要让计算机去算什么东西，不能让它算点有意义的东西吗，显卡那么值钱，一方面有人用它做 GPU 加速的深度学习计算，另一方面有人用它做“无意义”的挖矿计算。不能让第二部分的人去算第一部分人的需求吗？首先，关于第一个问题，中本聪设计工作量机制就是为了在所有节点中找到一个带头的来记账，因为总要有节点作为 leader 来统一大家记账的方法 (典型的分布式结构)，在比特币的世界里有没有政府或者银行，就需要想办法设计一个机制让大家轮流当 leader。

那既然这样，是否能够让系统在整个体系里随机选择一个节点作为 leader 呢。答案也是否定的，因为我们讨论过，系统会为这个“leader”发放奖励也就是发比特币 (这也是比特币世界比特币总量增加的唯一方法，相当于央行发行货币)。中本聪把这个作为挖矿激励，也作为比特币世界中的唯一发行货币的渠道。如果真的可以什么都不做 (或者说做了什么都一样) 只是简单等着系统随机选取一个 leader 来发放奖励，那么比特币就没有“价值”，作为货币最根本的属性就无法体现出来了。换句话说，现在比特币值的钱，从某种角度上看很大程度上就是值的挖矿时耗费的成本 (还有一部分是市场对于比特币的信心，这就是市场的部分了)。这里其实和真实的挖矿很像，比如钻石矿，它的价值就是人类赋予的，然后钻石的价值就是挖钻石矿的成本，和市场对于钻石的信心，这里面还涉及营销，文化等其他原因导致的市场需求 (比特币的世界就很单纯，只有一个由代码驱动的类似上帝的系统给挖的矿「定价」)

说到这里我们来解决上面提到的问题：能不能让挖矿的人去算深度学习那帮家伙的计算问题，或者其他有意义的数学问题？答案也是否定的，中本聪之所以这么设计，最根本的需要是保证比特币体系的有序，也即大约每 10 分钟需要能够计算出来，否则产生的交易无法被记录 (只能等着有人成功挖到矿才能记账)，时间越久，比特币作为货币的价值就会越低，试想一下，如果我今天用比特币发生一笔交易，这笔交易需要等一天甚至两天才能被确认，这样的货币可能也没有什么生命力了。

事实上，目前挖矿的时间已经远超过之前中本聪定义的 10 分钟了，主要原因是整个世界都陷入了这场“毫无意义”的军备竞赛中了。这应该是中本聪预料不到的，论文中中本聪考虑了摩尔定律 (简单来说就是算力的增长大体上符合某种规律) 对挖矿速度的影响才设计了这样一个比赛。但是中本聪没有考虑到市场狂热和人性的贪婪，使得挖矿的军备竞赛愈演愈烈。现实生活中美苏的军备竞赛是因为人类对权力的欲望，比特币中的竞赛是市场的选择。现实中的竞赛被各种其他原因 (经济政治等) 缓和，最终迎来了目前的和平。那么比特币的最终是什么呢？当货币发完之后 (每次挖矿激励越来越少)，是政治、经济、社会各种力量借势发挥，最终一地鸡毛，还是我们的社会，金融结构发生了更具弹性的变化以迎来新的转机。那个时候的世界又有什么危机和挑战呢？

4.3 思考与感悟: 停滞性通货膨胀和通货紧缩

第二点想要讨论的是通货膨胀和通货紧缩，在宏观经济学中有这样一个概念，作为通货膨胀长期发展的结果，会出现经济停滞 (stagnation)，失业及通货膨胀 (inflation) 同时持续高涨的经济现象。对于政府来说这是一个两难的问题，可能需要组合政策权衡利弊减轻影响。在现实世界中，货币存在通货膨胀，但是随着科技的发展，物质的积累，我们可以通过各种政策平衡这种影响，甚至是金融之外的因素（战争，政治等）但是在比特币的世界中，中本聪为了消除通货膨胀的影响，设计的机制是每四年减半发行货币，最终会导致比特币总量一定，当发展到后来肯定会出现通货紧缩。从这个角度看，无论是方向，还是具体的政策 (每四年减半)，无论这是否会真的延长比特币的生命力，这都是一次很有益的社会尝试。

4.4 思考与感悟: 去中心化的讨论

第三点想要讨论的是比特币提出的去中心化，我们真的不需要中心化吗，如果每个人做好自己的事，甚至都不需要组织，也没有什么复杂的共识算法，只需要一个根据代码执行的“上帝”就能让社会正常运转吗，从技术角度来看论文没有给出明确的解释，这里可能需要进一步查询资料，假设这个中本聪构造的这个世界真的可以无限发展下去，不同的节点比例 (不同比例的诚实节点和攻击节点，对应不同比例的社会组成) 的最终的影响 (全局影响，不是中本聪论证的是否在一次交易中能够追上) 是什么，概率又是多少，这部分感觉可以深挖很多内容。

从另一个角度来看，这种去中心化，分布式的体系似乎刻画了人类社会的发展变革过程，从边缘到中心再到边缘。或者说在一种失控中获得控制的过程。正如《失控》的作者凯文凯利 Devin Kelly 所说，如果货币，这个现代生活中最中心化的事物，都可以在一定程度上去中心化，那么也许任何事物其实都可以去中心化了。

4.5 思考与感悟: 社会机制的讨论

第四点想讨论的是社会机制: 我们可以会看到区块链设计的核心就是引导节点诚实 (当有能力的时候，可以做的破坏还不如诚实带来的收益大)，和降低被攻击的概率 (即使不顾一切攻击，整个网络被影响的概率也很低)。这是一个很好的比特币网络节点设计，也是一个很好地分布式系统的设计，这种设计和计算机领域中由于数据量增长而采取的分布式设计不同，这里的每个节点的背后都是人，都有自己的思想，都可以根据环境做出选择，可以很好地模拟社会组织和分工，所以说这可能是很好的一种社会机制和社会实验，相信这种机制在脱离了“货币”这个概念之后依然能体现出它的价值。

4.6 思考与感悟: 总结与展望

至此，白皮书全文就讲完了，中本聪用他这篇 12 章，3500 字的论文让世界震惊，无论是从论文中总结出来的技术“区块链”，还是论文的核心“比特币”，都已经让这个世界陷入癫狂。这些年外界对比特币的研究和评价非常多，从信息学，经济学，货币银行学，哲学，社会学，政治学，生物进化论，甚至宗教等多种视角，给予了诠释，我认为比特币除了像中本聪定义的它是一种点对点的电子现金系统，更重要的是他构建的这种区块链模型，给科技界带来了一种思想变革，甚至说形成了一种社会思潮，一种人文精神。

当再有十几年甚至几十年，比特币这种火热的现象过去之后，我们的生活，我们的社会一定会因为这篇论文而发生改变。就像星星之火，可以燎原。比特币从上线至今已经有十多年，它所衍生的区块链行业也在不断发展进步，出现了像智能合约 (smart contract)，去中心化金融 (DeFi)，去中心化自治组织 (DAO)，星际文件系统 (IPFS) 等等的技术创新和社会实验。

目前区块链正在颠覆许多行业，但是我相信这只是个开始，它的威力还远远没有发挥出来，我们拭目以待吧。

5 参考文献

参考文献

[1] 读懂比特币白皮书，区块链开山之作，中本聪到底说了什么？| Bitcoin whitepaper | 李查说 Richard Talks

[2] 比特币白皮书个人翻译 + 注解

[3] 金色财经: 比特币白皮书解读