雲南大學

本科生学年论文

题　目　比特币区块链中的椭圆曲线加密方法与矿工间博弈平衡问题解析

学　　院： 数学与统计学院

姓　　名： 刘　　鹏

学　　号： 20151910042

专　　业： 信息与计算科学

指导教师： 陆正福 教授

2017 年 7 月 1 日

比特币区块链中的椭圆曲线加密方法与矿工间博弈平衡问题解析

刘鹏

（云南大学 数学与统计学院信息与计算科学专业，昆明市 呈贡区 650500）

摘 要：区块链技术从数字加密货币中产生，之后迅速往通用性技术领域发展。区块链被认为是分布式数据库，与传统数据库相比，区块链具有难以数据篡改、信息安全性高等优势。区块链中的交易打包、交易验证、区块验证等，均使用了现代密码学中的非对称加密与哈希函数等技术。作为经济学与计算机科学的交叉产物，区块链技术在设计上引用了经济学中的博弈思想，保证了在最初的竞速寻找哈希值的过程中，竞争参与者会尽可能采取合法手段而不会主动攻击其他竞争者；同时以计算机科学中的数据加密技术作为网络通信与数据存储、验证的基石。比特币区块链采用的加密技术为近年来相对较新的椭圆曲线非对称加密技术。本文从博弈论与椭圆曲线公钥体制出发，分析了两者之间在设计上的关系，剖析了两者分别在区块链“宏观”与“微观”上所发挥的作用，并反思了区块链技术是否有可能在除数字加密货币之外的其他领域中发挥作用，以及可利用区块链技术的这些领域具有哪些共同特征等。

关键词：椭圆曲线加密；区块链；比特币；矿工博弈

Analysis of Elliptic Curve Cryptography in Bitcoin Blockchain and the Nash Equilibrium Between Miners

LIU Peng

(School of Mathematics and Statistics, Yunnan University, Chenggong District, Kunming 650500, China)

Abstract: Blockchain was born from the digital cryptocurrencies, it then developed into a versatile technology field. Blockchain is considered as a distributed database. Compared with traditional database, blockchain has the advantages of hard data modification and high information security. The transaction packaging, transaction verification, and block verification in the blockchain all use techniques such as asymmetric encryption and hash functions. As a cross product of economics and computer science, blockchain ensures that competing participants will take legal measures rather than attack others. And blockchain uses data encryption technology in computer science as the basis of network communication and data storage and verification. The encryption technology used in the Bitcoin blockchain is a relatively new elliptic curve asymmetric cryptography in recent years. This paper begin with the game theory and elliptic curve cryptography, analyzes the relationship between the two, and analyzes the roles played by the two in the blockchain. Whether block chain technology may play a role in other fields besides digital cryptocurrency, and what are the common characteristics of these areas that can utilize blockchain technology? These two issues were further discussed as the extension of this paper.

0 引言

区块链技术最初只是为比特币设计的一种特殊数据库技术，它基于密码学中的椭圆曲线数字签名算法来实现去中心化的 P2P 系统设计(唐长兵 et al., 2017)。如今在使用区块链这个词时，有时是指数据结构，有时是指数据库，有时则是指数据库技术。从数据的角度来看，区块链是一种分布式数据库（或称为分布式共享总账，Distributed shared ledger），这里的“分布式”不仅体现为数据的分布式存储，也体现为数据的分布式记录（即由系统参与者集体维护）；从记录效果的角度来看，区块链可以生成一套记录时间先后、不可篡改、可信任的数据库，这套数据库是去中心化存储且数据安全能够得到有效保证；从开发架构设计看，区块链并没有采取传统的网状数据库模型、树状数据库模型以及基于表的关系数据库模型，区块链技术采用了一种类似 Git 这种版本控制软件的模式，即链式存储，这种新的数据库架构，为不可更改地写入数据量身打造。在多人共同参与写入数据的情况下，区块链的长度——即区块链的所包含的区块数目——在不停增长，妄图修改前期数据的参加者需要把被修改的数据所在数据块之后的所有数据块都修改一遍，这在区块数目稳定增长与修改者的计算力不超过整体算力的一半的情况下，可以从数学上证明是不可能的。区块链技术在没有中央控制点的分布式对等网络下，使用分布式集体运作的方法，构建了一个 P2P 的自组织网络。通过复杂的校验机制，区块链数据库能够保持完整性、连续性和一致性，即使部分参与人作假也无法改变区块链的完整性，更无法篡改区块链中的数据。区块链技术涉及的关键点包括：去中心化（Decentralized）、去信任（Trustless）、集体维护（Collective maintain）、可靠数据库（Reliable data base）、时间戳（Time stamp）、非对称加密（Asymmetric cryptography）等。

区块链技术原理的来源可归纳为数学上的拜占庭将军问题。将拜占庭将军问题延伸到互联网生活中来，其内涵可概括为：在互联网大背景下，当需要与不熟悉的对手进行价值交换活动时，人们如何才能防止不会被其中的恶意破坏者欺骗和迷惑，从而做出错误的决策。而如果进一步将拜占庭将军问题延伸到技术领域中来，其内涵可概括为：在缺少可信任的中央节点和可信任通道的情况下，分布在网络中的各个节点应如何达成共识。从这些角度来看，区块链技术解决了闻名已久的拜占庭将军问题，它提供了一种无需信任单个节点，还能创建共识网络的方法。

作为区块链技术最成功的应用，比特币系统应用工作量证明（Proof of work, PoW）的共识机制实现交易的不可篡改性和不可伪造性。PoW 共识机制的核心思想是通过引入分布式节点的算力竞争来保证数据的一致性和共识的安全性。比特币系统中，各节点基于各自的算力相互竞争，共同解决一个求解复杂但验证容易的 SHA256 数学难题，最快解决该难题的节点将获得区块记账权和系统自动生成的比特币奖励。具体过程如下：如果想产生一个区块并写入到区块链中，需要找到一个小于系统规定难度值的随机数，这样才可能被其他节点认可，并写入到区块链中。而找到随机数需要输出密码散列函数家族SHA256的哈希算法。其中，一个符合要求的输出值由N个前导零构成。零的个数取决于网络的难度值，挖矿难度越高，零的个数会越多。当输出值不满足要求时，这个随机数就会增加一个单位，直到找到为止。找到合适随机数后，节点获得记账权和相应比特币奖励，并将该过程中产生的所有交易记录在区块上，所有区块按时间顺序连接则构成区块链。一般地，比特币系统通过灵活调整随机数搜索的难度值来控制区块的平均生成时间。

在比特币系统中，产生区块的过程称为挖矿，进行挖矿的参与者称为矿工。由于比特币系统大约每 10 分钟产生一个区块，这意味着大部分矿工在一定时间内很难产生区块。为了增加获得稳定收益的可能性，矿工会选择加入开放矿池进行合作挖矿。具体地，矿池中的矿工需要耗费资源尝试产生区块，即发送完整工作量证明给管理者。但完整工作量很难产生，矿工也可以选择发送部分工作量证明获得相应收益。无论哪个矿工产生区块，获得的收益将按贡献比例分配给每个矿工。

参与者注册为矿工很简单，只需要提供一个公共的网络接口就可以加入开放矿池，因此开放矿池很容易受到攻击。有些注册矿工只发送部分工作量证明，当产生完整工作量证明时就会将其抛弃，这种攻击方式被称为区块截留攻击。在这种情形下，攻击者发送部分工作量证明，但不会对矿池产生有效收益，这也导致攻击者与其他矿工共同分享矿池收益，从而减少其矿池的收益。

研究表明，在一个开放的矿池中，矿工可以通过攻击其他矿工增加自己的收益。如果所有矿工都选择攻击对方，那么他们获得的收益将少于他们互不攻击时获得的收益。这就是 PoW 共识算法中的挖矿困境，而这种困境也对应到博弈论中经典的囚徒困境(Prisoner's dilemma)，即攻击对个体而言是最优策略，但却不是系统最优的。如何理解和分析挖矿过程中的博弈困境无疑给比特币的发展和技术开发乃至投入使用提供了理论基础。例如 Eyal 基于博弈理论，定性地分析了挖矿过程中的困境，但并没有给出纯策略存在条件以及相应证明。本文在的基础上进一步分析矿工博弈困境的纯策略和混合策略均衡，并给出两种均衡存在的条件。　　更为重要的是，PoW 共识机制存在着显著的缺陷，其强大算力造成的资源浪费（例如算力）历来为研究者所垢病，而且长达 10 分钟的交易确认时间使其相对不适合小额交易的商业应用。与此同时，随着区块链技术的发展和各种数字币的相继涌现，研究者提出多种不依赖算力而能够达成共识的机制，例如权益证明（Proof-of-Stake, PoS）、共识授权股份证明机制（Delegated Proof of Stake, DPoS）共识，缠结（Tabgle）以及 Tendermint 机制。而最理想的共识算法是系统中的节点达成的共识是一个纳什均衡，即单方面改变自己的策略都不会提高自身的收益。这为基于博弈论构建共识机制提供了新的思路。另一方面，PoW 共识过程中的挖矿困境对应经典的囚徒困境模型，其纳什均衡为互相攻击，此时的系统收益并不能达到最优。为提高系统的整体效益，有必要建立相关机制，使矿工趋向于合作，以获得较高的系统收益，从而为实现高效的共识算法提供依据。零行列式（Zero determinant, ZD）策略是近几年在博弈论中兴起的一种新方法，它能够打破传统的纳什均衡理论。

1 文献综述

文献(唐长兵 et al., 2017)从工作量证明（Proof of work, PoW）共识算法的挖矿困境入手,分析 PoW 共识过程中矿工策略选择的纳什均衡存在条件；

2 最终一致性以及比特币

定义 2-1（网络分区, Network Partition）是一类错误，指一个网络分裂为至少两个部分，且这些分裂之后的子网之间不能通信。

直觉上，任何非平凡的分布式系统不能再一个分区期间继续工作，且保持一致性。接下来介绍一致性（Consistency），可用性（Availability）和分区容忍性（Partition Tolerance）这三个指标间进行取舍的问题。

定义 2-2（一致性, Consistency）一个系统的所有节点就系统的当前状态达成一致。

定义 2-3（可用性, Availability）. 系统是可用的且正处理请求。

定义 2-4（分区容忍性, Partition Tolerance）. 分区容忍性是指分布式系统具备的一种能力：在存在网络分区的时候仍可以正确地工作。

定理 2-1（C.A.P 定理）一个分布式系统不可能同时实现一致性、可用性以及分区容忍性。它可满足其中任意两个要求，但不能同时满足三个。

证明 假定两个节点，共享某个状态。两个节点处于不同的分区中，即：他们不能通信。假定一个请求希望更新这个共享状态并联络一个节点来执行更新操作，这个节点可以采取以下两个策略之一：(1) 更新它保存的本地状态，这就导致一个不一致的状态；(2) 不更新它保存的本地状态，那么系统此时就不能响应这个请求，即不可用。

一般而言，对于一个不需要时常相应的网络而言，可以满足一种被称为最终一致性的条件，比如在众多 ATM 机与银行组成的网络中，就算 ATM 机无法与银行通信，只要 ATM 机保留有本地数据，而且银行不提供现今提取工作，就可以满足可用性，而且当 ATM 机与银行恢复通信之后就可以同步数据，从而保证银行与 ATM 机的最终状态是一致的。当然，这属于一种弱一致性。如果提款者在另一个鼓励子网络中提款就会发生双花问题。

定义 2-5（最终一致性, Eventual Consistency）如果不再对共享状态有新的更新，则最终系统进入安静（Quiescent）状态，即节点之间不需要发送额外的消息，且共享状态是一致的。

需要注意的一点是，在网络处于分区期间，不同的节点可能执行不同的更新，而这些更新可能在语义上是矛盾的。因此需要一个冲突解决机制来解决这些冲突，并使得去除分区之后的网络的所有节点在一个相同的状态上达成一致。比特币就是最终一致性的一个典型例子。

定义 2-6（比特币网络, Bitcoin Network）比特币网络是一个随机连接的覆盖网络（Overlay Network），它包含成千上万个节点，被各种各样的拥有着控制。所有节点运行相同的操作，即：这是一个去中心化的同质网络（Homogenous Network）。

定义 2-7（地址, Address）用户可以产生任意数量的私钥，并基于这些私钥构建一个公钥。地址是基于一个公钥得到的，并且被用来标识比特币系统中一笔金额的接收者。一个共/私钥对被用来唯一地标识某个地址（以及相应的一笔金额）的拥有者。

公钥与地址都是公开的信息，因此经常可以互换使用，使用地址的好处是它比公钥简短。因为很难将地址和拥有地址的用户关联起来，因此比特币经常被认为是匿名系统。地址包含一个网络标识字节，以及公钥的哈希码和校验和。通常以 base58 的编码存储。哈希算法将得到长度为 20 字节的地址，这意味着总共可以有$2^{20\times 8}$个不同的地址。如果用穷举算法来破解一个地址，就算每秒尝试 10 亿次，大约需要2^{45}年才能找到一个匹配的公/私钥密码对。根据生日悖论（Birthday Paradox），如果我们不是去穷举某一个特定地址而是采取随机算法，那么猜中的几率会上升。

定义 2-8（输出, Output）一个输出是一个元组，包含一定数额的比特币以及一个使用条件。绝大多数情况下，使用条件需要一个和某个地址对应的私钥相关联的有效签名。

使用条件是一段脚本，包含多个选项。除了一个签名外，脚本还可以要求一个简单计算的输出结果，或者是一个密码学难题（Cryptographic Puzzle）的答案。输出存在两个状态：未使用（Unspent）或者已使用（Spent）。任何输出只能被使用一次。一个地址的账户余额是所有与该地址关联的未使用输出的比特币数额总和。所有未使用的交易输出（Unspent Transaction Outputs, UTXO）以及一些附加的全局参数就构成了比特币网络的共享状态。每个在比特币网络中的节点都拥有一个改状态的完整副本。这些本地副本之间可能暂时地不一致，但是最终将重新达成一致性。

定义 2-9（输入, Input）一个输入是一个元组，包含对前面一个已经创建的输出的引用，以及用于该输出使用条件的一组参数（签名）。这些参数将证明交易创建者有权使用所引用的输出。

定义 2-10（交易, Transaction）交易是一个数据结构，描述了一次比特币的转移（使用者到接收者）情况。一个交易包含很多输入和新创建的输出。这些输入将导致所引用的输出变为已使用（即从 UTXO 中删除），此外新创建的输出将被增加到 UTXO 中。

输入用一个$(h,\ i)$元组来引用一个即将被使用的

输出，此处h是创建该输出的交易（Transaction）的哈希值，而i描述了在交易中该输出的索引值。交易在比特币网络中广播，网络中每个接收到该交易的节点都需要处理它。

算法 2-1（节点处理接收到的交易）

```
接收到交易t
for each t 中的输入(h, i) do
    if 输出(h, i)不在本地的 UTXO or 签名无效 then
        将交易t删除掉，并停止处理这个交易
    end if
end for
if 所有输入包含的金额之和 < 所有新创建的输出的金额之和 then
    将交易t删除掉，并停止处理这个交易
end if
for each t 中的输入(h, i) do
    从本地的 UTXO 中删除(h, i)
end for
将t添加到本地历史
将t发送给比特币网络中的邻居
```

这个算法可以朴素地处理交易不实的问题，但是网络中还存在着不同节点接受不同的交易，而这些交易使用同一个输出。这就是比特币网络中的双花问题，即同一份比特币可能被消费了两次。

在比特币交易网络中，交易必然处于两个状态之一：未确认（Unconfirmed）或已确认（Confirmed）。从广播中接收到所有交易是未确认的，并且会被加入到一个名为记忆池（Memory pool）的交易池中去。

定义 2-11（重复使用, Double spend）重复使用指一个特殊状况，此时多个交易都尝试使用同一个输出。只有一个交易可以是有效的，因为输出只能被使用一次。当节点在重复使用的情形下接受不同的交易时，共享状态将变得不一致。

重复使用经常是故意行为，这被称为重复使用攻击。重复使用可以导致不一致的状态，因为一组交易的有效性取决于他们到达的顺序。如果两个相互冲突的交易被同一个节点看到，该节点将认为第一个是有效的，并认为第二个是无效的。如果重复使用的问题无法解决，共享状态将会出现分叉。于是就需要一个冲突解决机制来判定冲突的交易中哪一个交易应该被确认（即被所有节点接受），由此实现最终一致性。

定义 2-12（工作量证明, Proof-of-Work）工作量证明机制使得一个参与者可以向其他参与者证明：他已在一段时间内持续使用了一定数量的计算资源。特别地，定义工作量证明函数$\mathcal{F}_d(c,\ x) \to \{\text{ture},\ \text{false}\}$，此处的d是一个正实数，表明困难程度；挑战问题$c$和随机数$x$通常是比特字符串。工作量证明函数需要具备下面的性质。

1) 当$d,\ c,\ x$都给定时，很快就能计算出$\mathcal{F}_d(c,\ x)$；
2) 对于一组给定的参数d和c，找到x使得$\mathcal{F}_d(c,\ x) = \text{true}$是可计算的，但是非常困难。困难系数$d$被用来调节找到$x$的平均期望时间。

定义 2-13（比特币 PoW 函数, PoW Function in Bitcoin）比特币 PoW 函数定义如下：

$$\mathcal{F}_d(c,\ x) \to \text{SHA}_{256}\big(\text{SHA}_{256}(c|x)\big) < \frac{2^{224}}{d}.$$

此函数将挑战c和随机数x连接起来，并且使用 SHA256 做两次哈希。将这个输出值和目标值进行比较。增大难度系数d，目标值将减小，从而提高了找到x的难度。

SHA256 是一个用于加密的哈希函数，输出具有伪随机性，目前还没有比穷举法更好的算法来寻找答案。如果所有节点都参加这个计算并使用相同的挑战，那么计算力最强的节点从概率角度讲将总是获胜。然而，在比特币的实际网络中，每个节点都是用一个节点特定的挑战来尝试找到合理的随机数x。

定义 2-14（区块, Block）一个区块是一个数据结构，在一个节点局部状态之上累积的改变将打包在区块中并传递给整个网络。一个区块包含了一组交易，一个指向

上一个节点的引用，和一个随机数（即在工作量证明阶段找到的随机数）。一个区块包含了本区块创建者（矿工）所接受并存放在自己记忆池中的所有交易，这些交易都是建立在上一个区块之后产生的。一个节点在找到一个有效的随机数来满足它的工作量证明函数之后，将广播一个区块。

算法 2-2（节点寻找区块）

1 $x = 0,\ c,\ d$，上一个区块b_{t-1}
2 **repeat**
3 $x = x + 1$
4 **until** $\mathcal{F}_d(c,\ x) = \text{true}$
5 广播区块$\mathrm{b_t} = (\text{memory} - \text{pool},\ b_{t-1},\ x)$

由于每个区块都引用它前面的那个区块，所有的区块就构成了一棵树，以创世区块（Original Block）为树根。使用工作量证明机制的首要目的是调节整个网络找到区块的速度，使得网络有时间来实现在最新一个区块上的同步。比特币通过设置难度系数来确保整个网络找到一个新的区块的平均时间为 10 分钟。

率先在上一个区块被网络接收后找到一个新的区块的人，有权把自己记忆池中的所有交易强加给所有其他节点。在接受到一个区块的时候，所有的节点都将回滚自己在上一个区块之后对本地状态所做的任何改动，并执行新区块包含的交易。我们称一个区块内包含的交易被该区块确认。

定义 2-15（奖励交易, Reward Transaction）在一个区块内的第一个交易被称为奖励交易。发现该区块的矿工将获得一定数量的新比特币以奖励它确认了一组交易。奖励交易有一个名义上的输入，其输入的总和包括一个定额的补贴加上被该区块所确认的所有交易的交易费之和。

比特币系统的交易有一个规则：所有输入之和必须大于或等于所有输出之和。但奖励交易是唯一的意外。每个奖励交易所创造的比特币数量由一个补贴机制来决定，该机制是比特币网络协议的一部分。最初，每个区块包含的补贴为 50 个比特币，这个数量在每找到 210,000 个区块，也就是大约花费四年时间之后，奖励数量就会减半。根据这个机制，能流通的比特币总量不会超过 2100 万。

设计者们期望，找到一个比特币的代价，比如能源开销与购买设备等，应该和矿工得到的奖励价值接近。

定义 2-16（区块链, Blockchain）从创世区块开始，到某个叶节点位置的最长路径，被称为区块链。区块链的作用是维护一个一致的交易历史，而且所有的节点最终都在唯一的区块链（交易历史）上达成一致。

从创世区块到某个区块的路径长度是该区块的高度。只有从创世区块到某个节点所构成的最长路径才是一个有效的交易历史。注意到由于重复使用的存在，分支之间可能会彼此冲突。由于只有在最长路径上的交易才会被最终承认，矿工们都会自发地将它们加到最长的链上，这样就会在当前状态上达成一致。

如果多个区块同时或者几乎同时被找到，那么系统将会出现分叉。分叉是很自然的，因为挖矿是一个分布式的随机过程，而两个区块是几乎同一时刻被找到。

算法 2-3（节点接收到区块）

1 接收到区块b
2 当前节点所保存的头区块为$\mathrm{b_{max}}$，其高度为h_{max}
3 将区块b加到树上去，作为其父区块p的子区块。新增区块的高度为$\mathrm{h_b} = \mathrm{h_p} + 1$
4 **if** $h_b > h_{max}$ **then**
5 $h_{max} = h_b$
6 $b_{max} = b$
7 根据从创世区块到新增区块的路径来计算 UTXO
8 清空记忆池
9 **end if**

算法 2-3 描述了一个节点在接收到一个区块时是如何更新它们本地状态的。在这个算法的作用下，不同节点之间仍然有可能出现状态分歧，也就是说接受不同的区块作为相同高度的头区块。

与扩展路径不同，切换路径有可能导致已经被确

认的交易重新变成未确认。因为在新的路径上的区块没有包含这些交易。切换路径被称为重组（reorg）。

清空记忆池包括：（1）删除在当前路径上已经被确认的交易；（2）删除与已经被确认的交易存在冲突的交易；（3）增加在前面路径中已经被确认，但是在当前路径不再被确认的交易。

为了避免在每个新节点加入时重新计算整个UTXO（这将是十分巨大的计算），所有当前的实现都使用数据结构来保存了一个区块所执行的撤销（undo）信息。这使得我们可以方便地在路径上移动，从而快速地切换路径，并更改头区块。

定理 2-2（分叉解决, Branch solve）分叉将最终被解决，并且所有节点最终都将接受同一条最长路径。于是系统确保了最终一致性。

证明 如果分叉持续存在，在两个分支上都需要不断产生新的区块，且他们的高度还得保持一致，否则较短分支上的节点将切换到较长的分支上去。随着分叉长度的增加，令各分支同时找到新节点的概率将呈指数形式下降。于是最终，将出现一个时刻，只有一个分支会被扩展并成为最长的分支。

文献(Nakamoto, 2008)指出，攻击者从给定的前期区块追赶上现有区块的概率与赌徒破产问题（Gambler's Ruin Problem）类似。赌徒破产问题是指，假设赌徒有h枚金币，每次有概率p获得一枚金币或者有概率$q=(1-p)\neq p$丢掉一枚金币，直到其所有的金币总数达到N或0游戏结束，求这种情况下赌徒最终赢得N枚硬币的概率$P(N|h)$。有两种两个状态可以确定，即$P(N|N)=1$和$P(N|0)=0$。根据全概率公式（Law of Total Probability）来分析这个马尔可夫过程（Markov Process），假设现在赌徒手中的金币数目h大于等于1但是还不到N，那么现在他能赢走所有钱的概率与下一局的胜负相关，根据可能的结果，下一局 A 要么输掉要么获胜，所以由全概率公式立即就有$P(N|h)=P(N|h+1)\times p+P(N|h-1)\times q$，上式中的$h+1$与$h-1$分别代表赌徒下一局赢了钱与输了钱。

现有概率转移公式：

$$P(N|h)=P(N|h+1)\times p+P(N|h-1)\times q$$

如果令$A_i=P(N|i)$，从而可以把上式写成数列的形式：

$$A_{h+1}=\frac{1}{p}A_h-\frac{q}{p}A_{h-1}$$

同时知道$A_0=0$，$A_N=1$（如果是破产概率，就有$A_0=1$，$A_N=0$）。

这是一个与斐波那契数列类似的二阶线性递推关系，其特征多项式(高焕江, 2010)为$x^2-\frac{1}{p}x+\frac{1-p}{p}$，特征根为1与$r=\frac{1-p}{p}$，于是得到通项公式

$$A_N=\frac{r^h-1}{r^N-1}$$

$$=\frac{\left(\frac{1-p}{p}\right)^h-1}{\left(\frac{1-p}{p}\right)^N-1}$$

William Feller 中指出，在极限情况下，即一个拥有初始赌资h的赌徒，对手是一个拥有无限资产的富翁。这时候令$N\to\infty$，可得如下结论(Feller, 1950)：$(1-p)/p\overset{\text{def}}{=}s$

$$\lim_{N\to\infty}\frac{s^h-1}{s^N-1}=\lim_{N\to\infty}\left(1-\frac{s^N-s^h}{s^N-1}\right)=\begin{cases}0 & s>1\\ 1-s^h & s<1\end{cases}$$

由此可以知，若把赌徒视为攻击者：

1) p为诚实节点获得下一个区块的概率
2) q为攻击者获得下一个区块的概率
3) q_z为攻击者在落后z个区块之后可以追上的概率

$$q_z=\begin{cases}1 & \text{if } p<q\\ (q/p)^z & \text{if } p>q\end{cases}$$

3 椭圆曲线代数系统

大多数使用公钥密码学进行加密和数字签名的产品和标准都是用 RSA 算法。近年来，为了保证 RSA 使用安全性,密钥的位数一直在增加，这对于使用 RSA 体制的应用而言是一项巨大的负担，对进行大量安全交易的电子商务与银行系统而言更是如此。近你来出现的椭圆曲线密码学（ECC）对 RSA 提出了挑战。ECC 的主要优势在于，它可以使用比 RSA 短得多的密钥得

到相同安全性，减少处理荷载。

3.1 椭圆曲线代数方程

椭圆曲线并不是椭圆，之所以称之椭圆曲线为这一类方程的样式，与计算椭圆周长的方程类似，也使用三次方程来表示的。一般，椭圆曲线的三次方程形式为

$$y^2 + axy + by = x^3 + cx^2 + dx + e$$

其中，a，b，c，d和e是实数，x和y是取值在实数集上的变量。在椭圆曲线加密种，并不需要这种普通形式，下述形式已经足够：

$$y^2 = x^3 + ax + b$$

这是一个三次方程。椭圆曲线的定义中，还需要一个称作无穷远点或者零点的元素，记作O。

当方程满足$4a^3 + 27b^2 \neq 0$时，以椭圆曲线上的所有点作为集合，可以定义一种加法，进而作出一个阿贝尔群，即一个符合封闭性、加法结合律、加法单位元、逆元存在、加法交换律这 5 条性质的代数群。

定义 3-1（阿贝尔群, Abelian group）给定一个集合G，给定G上的一个二元运算$\circ$，记为$\{G, \circ\}$，如果对于集合G中元素组成的任意一个序偶(a, b)，使得下面的五个定理成立：

1) 封闭性　若$\forall a, b \in G$，则$a \circ b \in G$；
2) 结合律 若$\forall a, b, c \in G$，则$(a \circ b) \circ c = a \circ (b \circ c)$；
3) 单位元　G中存在元素e，使得$\forall a \in G$，都有$e \circ a = a \circ e = a$；
4) 逆元　$\forall a \in G$，$\exists a'$，使得$a \circ a' = a' \circ a = e$；
5) 交换性　$\forall a, b \in G$，有$a \circ b = b \circ a$。

定理 3-1（椭圆曲线阿贝尔群）对于

4 椭圆曲线加密

5 博弈论概要

6 矿工不主动作弊的博弈基础

6.1 拜占庭协定

为了提高飞行的安全系数，学者们曾经研究过安装在飞机上的数量众多的传感器和仪器可能发生的错误。在对这些错误进行建模的过程中，科学家们发现：失灵的仪器不但会停止工作，有时还呈现出任意的行为。基于这样的认知，学者们认为仪器错误可以是任意类型的，不局限于任何模式。

定义 6-1（拜占庭）. 一个可能呈现任意行为的节点被称为拜占庭。任意行为意味着“所有能想象到的事情”，比如，根本不发送任何消息，向不同的邻居发送不同且错误的消息，以及谎报自己的输入值。

如果用博弈论的观点来看，拜占庭行为也有可能包含串谋，即所有的拜占庭节点被同一个攻击者控制。这里假定任何两个节点之间直接通信，并且没有一个节点可以伪造其他节点的发送地址——这个要求确保了单个拜占庭节点不可以扮演所有的节点。我们把所有非拜占庭节点称为**好的节点**。

定义 6-2（拜占庭协定，Byzantine Agreement）. 在一个存在拜占庭节点的系统中达成的共识被称为拜占庭协定。如果一个算法可以在存在f个拜占庭节点的情况下正确工作，则称该算法为f – 可适用（f-resilient）。

7 参考文献

FELLER, W. 1950. *An Introduction to Probability Theory and Its Applications,* the U.S., John Wiley & Sons, Inc.

NAKAMOTO, S. 2008. Bitcoin: A Peer-to-Peer Electronic Cash System.

高焕江 2010. 二阶线性递推数列的通项公式. *保定学院学报*, 34-37.

唐长兵, 杨珍, 郑忠龙, 陈中育 & 李翔 2017. PoW 共识算法中的博弈困境分析与优化. *自动化学报*, 1520-1531.

8 索引

致谢

时维七月，大三学年收官在即，行笔至此，感慨颇多。早在大三学年的第二学期开始的第一周，陆正福教授和我们一行几人仔细商讨了选题问题，在综合考虑了基础性、研究型、新颖性与可拓展性之后，陆导最终敲定：以区块链技术作为最终方向，题目可以涉及该技术中的数字加密技术，也可从经济学角度分析整个架构设计。

眼下正值比特币、以太坊之类的数字货币大行其道之时，一个比特币的价格甚至炒到了一万美元以上，虽然价格多有摆动，但是总体来看，比特币价格的居高不下印证了区块链这一技术的可靠与高可拓展性。同行五人，开始了针对比特币技术的研究。国内主流期刊上发表的文章大多都在探讨区块链的应用，而在进行入手研究之前，我一度认为除了用于发币，区块链并没有什么独到的其他用途。基于这种不解，我从区块链的核心机制开始分析，主要从系统设计角度考虑，并在文章末尾分析了区块链的底层支柱性技术：哈希函数与椭圆曲线非对称加密。

感谢陆老师给的支持，感谢学校老师给的良好资源。区块链技术内涵绝不是一篇文章能透彻分析的，不过能做一点分析，对于以后的拓展总有好处。

2018-06-30

于云南大学