迷你区块链方案

J. D. Bruce
2017 年 3 月第三版
www.cryptonite.info

摘要

几乎所有的 P2P 加密货币都使用笨重的“区块链”方案以防止双重支付和类似的攻击，并且通常不使用某种形式的伪去中心化方案去管理交易。这里我们提出一个纯粹的 P2P 加密货币方案，其中旧的交易可以被网络遗忘。由于节点只需要区块链的最新部分即可与网络同步，我们称这部分链为*“迷你区块链”*。我们认为这个修剪过程中产生的安全性损失可以通过一个小的*“证明链”*来解决，并且货币所有权数据的安全性损失可以通过一个数据库来解决，这个数据库记录了所有非空地址的余额，成为*“账户树”*。证明链确保迷你区块链安全，迷你区块链确保账户树安全。本文将描述这三个机制如何协同工作以形成一个系统提供高级别的完整性和安全性，但是比其他所有纯粹的 P2P 货币要精简地多。它还提供了其他潜在的好处。比如更快的交易、更低的费用、更快的网络同步、支持高流量，以及更多自定义消息的区块空间，甚至可能增加匿名性。

1. 引言

五年多以前，“Satoshi Nakamoto”首次将比特币发布到公共领域并且永远地改变了许多人对金融和经济的看法[1]。当时的难度低，区块链规模较小。头两年的事情看起来很棒，许多人认为区块链在很长一段时间内都不会出现问题，所以这个问题被搁置在一旁。我们现在是 2014 年 7 月，区块链的大小接近 20GB[2]。虽然仍然可以控制，但它正在成为一个令人担忧的严重问题。

比特币的核心开发人员现在将大部分注意力集中在处理不断增加的网络流量上。今天，bitcointalk.org 论坛几乎每天都在讨论有关减小区块链大小和减少同步时间的方法。迄今为止采取的最有效的措施之一是从 Berkeley DB 切换到 LevelDB[3]。BDB 速度慢很多，切换到 LDB 后在同步和块验证速度方面都显著提升了性能。

另一个有前景的方案是“终极区块链压缩”项目[4]，希望通过区块链修剪技术来实现“近乎最佳的区块链压缩”。该项目使用“账户树信息”，能够“通过合并挖矿在另外的区块链中进行维护和验证”。区块链修剪技术确实很有前途并且能被证明可以提供高度的压缩能力，但是它增加了额外的复杂性并且不能以完全令人满意的方式解决所有的规模性问题。

bitcointalk 论坛还有许多关于更改最大块大小限制的讨论（有很多支持和反对）。一群反对这一改变的人发布了“比特币区块大小问题的视频”[5]，他们反对增加最大的区块大小，因为“运行一个节点会变得更加昂贵”。比特币的核心开发者负责人 Gavin Andresen 回复说，“区块大小将会增加。你的视频只会让很多人什么都不担心”。这个问题似乎每天都在变得更加紧迫。

最大的块大小肯定会在某个时候增加，现在的交易容量被限制为每秒 7 个交易[6]，最终这还不够。提高比特币的最大区块大小将需要协调硬分叉，因为所有老客户都不知道如何处理更大的区块。因此，有正当理由担心增加最大的块大小，但这并不意味着不需要它。有一些必须考虑的缺点和优点。

增加最大区块大小会增加交易带宽并降低费用，但会导致区块链增长更快，给节点带来更多压力。bitcointalk 论坛的许多成员表示，最大块大小不是问题，而是他们指责 Satoshi Dice 等赌博服务通过许多交易向网络“发送垃圾邮件”。其他成员指出，Satoshi Dice“正在为网络提供压力测试，并表明除非取消块大小限制，否则将会出现问题”[5]。

2. 我们应该担心吗？

正如刚才提到的，有几种方法可以尝试解决区块链过大的问题。在 Bitcoin Stack Exchange 上，一位用户 Sean Chapman 询问“是否有任何关于区块链规模随时间变化的研究？”。最优回答的作者 Meni Rosenfeld 解释说尽管他不知道有任何研究符合 Chapman 的要求，但他可以概述我们不必担心区块链扩展性的 5 个原因[7]：

1) 不是每个用户都需要运行一个完整的网络节点。
2) 花费的输出可以从区块链中删除。
3) 链下交易有助于减轻网络压力。
4) 交易费用可以抵消区块存储成本。
5) 在可预见的未来，摩尔定律依然正确。

Satoshi 在 2008 年底谈到了第一点，当时他说随着网络的发展，挖矿“将越来越多地留给拥有专用硬件服务器的专家”[8]。这也许是一个令人满意的解决方案，但它会导致持续的中心化，最终这些专家将通过让较小的参与者更难参与来控制大部分的网络算力。第二点已经提到过；修剪技术提供了希望，但这是一个复杂的过程，结果是有限的。

最终区块链压缩方案等提议的主要问题是，Satoshi Dice 等服务产生的大量“灰尘”仍然会阻塞系统。比特币将交易联系在一起的方式使得我们无法从加密货币中实现我们真正需要的扩展性能力。最后 3 个点对解决问题有一定的有效性，但不管如何，我们仍然坚持使用永不停止增长的区块链，并且仍然没有真正轻量级的方案。

我们应该寻找创新的方法并以简洁和令人满意的方式解决这些可扩展性问题。比特币 Wiki 指出“在非常高的交易率下，每个区块的大小可能超过半 GB”[6]。

比特币达到那种网络流量水平并尝试将其全部存储在不断增长的区块链中真的可行吗？对于比特币等货币来说，高度集中化可能是最终唯一的解决方案。很明显，随着时间发展，我们需要一个更好的解决方案。

比特币很可能会在这里停留一段时间，任何新的山寨币都不太可能突然让比特币过时。迷你区块链方案在某些领域提供了更好的功能，但由于我们使用了简单交易模型，它也缺乏大多数货币所具有的一些功能。我们的目标是在 Satoshi 的工作基础上创建一个开放和自由的最佳加密货币市场，让竞争能够蓬勃发展。这里提出的方案可以为我们提供真正创新和独特的优势。

3. 寻找解决方案

接下来是一个全新的可选加密货币的提议，它在许多方面与比特币相似，但在其他方面也非常不同。对比特币代码库进行任何大的更改是极其困难的，不幸的是，以下方案与比特币不兼容。该方案通过取消交易链接消除了对完整区块链的需求，因此允许在经过足够的时间后丢弃所有交易，但这样做会从协议中删除脚本，这是比特币无法做到的。

本文中描述的建议解决方案来自于理解区块链的不同目的，然后将该功能分成单独的机制，每个机制都经过优化以服务于它们的目的。区块链有 3 个主要功能。比特币区块链将这些功能组合成一个单一的机制，因此不能很好地扩展。它需要存储大量实际上并不需要永久存储的数据。分解区块链的功能是关键。

区块链的功能：

1) 协调网络如何处理交易
2) 封装保护网络的工作量证明
3) 管理账户余额；记录货币的所有权

由于该提案的原始版本是在 2013 年初编写的，因此它已经得到了显著的改进和修改[9]。在其他 bitcointalk 论坛成员的帮助下，迷你区块链方案得到了完善，创建了一个项目 Wiki 以扩展原始白皮书的想法，并使用该方案实现了一种名为 Cryptonite 的新货币[10]。它的实施过程极大地帮助了微调迷你区块链方案的相关支撑概念。

本质上，迷你区块链方案通过将所有非空地址的余额存储在我们称为“账户树”的结构中来工作，因此我们实际上不需要任何交易来计算任何给定地址的余额。我们删除了脚本系统以及整个联锁交易的想法，并用一个更简单的概念取而代之，交易在账户树上执行基本操作，例如“从地址 A 的余额中减去货币并添加到地址 B 的余额”。

交易中的输入和输出不指向其他交易，它们只是指向帐户树中的地址，因此交易不会像在比特币中以相同的方式链接在一起。我们可以在经过一段安全的时间后丢弃所有交易（足以使 Secret Chain Attack 攻击不可行，稍后讨论）。Cryptonite 节点能够删除所有早于一周的交易，但他们可以选择存储尽可能多的历史记录，整个链不太可能丢失。

4. 帐户树

该提案从“帐户树”的概念开始。如果我们只需要知道所有非空地址的余额，我们为什么要记录每一笔交易并将其永久保存？区块链的第三个功能被账户树取代。账户树本质上可以被认为是去中心化的“资产负债表”。它将包含每个唯一的非空地址和所有这些地址的余额，以及其他一些使提款限制成为可能的字段（稍后会详细介绍）。

当地址余额发生变化时，我们需要做的就是更新帐户树中的数字，而不是向其中添加新数据。当然，这不会提供真正有限的数据量，因为新的非空地址会一直出现，但它尽可能接近有限。它在某种意义上是有限的，因为货币的可分性有限，我们不能真正期望世界人口或互联网用户的数量永远持续增长。在任何情况下，它都是可扩展和可管理的。

即使有 100 亿人口，每个人有 10 个不同的非空地址，我们也只需要跟踪 1000 亿个地址。由于我们可以从数据库中删除空地址，并且由于交易只需要对等方在该数据库中移动数字而不是向其中添加新数据，因此帐户树的大小应该始终保持相当小。当我们达到任何接近 1000 亿个独特的非空地址时，我们的计算机将快得多。

帐户树中非空地址的所有者用他们的私钥证明他们的所有权。与比特币一样，交易被创建为一组签名的数据并通过网络广播。与比特币一样，接受交易的矿工随后将其放入他们的区块中，并努力解决一个难题以将其放入迷你区块链（下一节将详细介绍）。接受区块的节点将通过转移硬币或做任何必要的事情来更新他们自己的帐户树副本。

提议的数据库被命名为“帐户树”，因为它应该具有哈希树结构。树中的每个“帐户”都有一个相应的哈希值，并充当树底部的叶节点。作为一棵哈希树，我们可以将每个账户的哈希组合起来，构建一个哈希金字塔，并计算顶部的“主哈希”。请注意，“帐户”不是像“比特币帐户”那样的地址集合。在这种情况下，每个帐户仅引用一个地址或叶节点（显然，正常的“帐户”也会存在）。

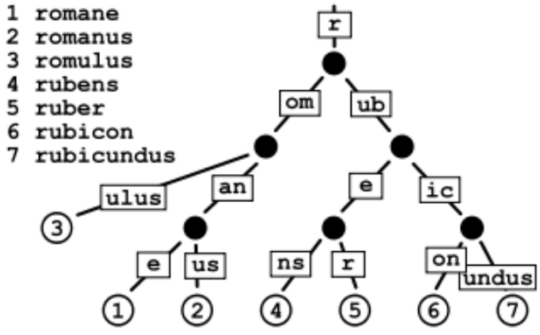

Figure 0-a

Figure 0-a 显示了一个通用的基数树/trie 结构（来源：维基百科）。在现实中 Cryptonite 使用了一种结合 merkle 哈希的二进制基数 trie 树。图中未显示的是节

点是如何哈希的。所有哈希一起在顶部产生主哈希/根哈希。即使树中只有一个帐户以任何方式更改，主哈希也会更改。这个哈希树系统为我们的数据提供了完整性，因为主哈希也存储在块头中，因此树受到区块链的保护。

使用二进制基数 trie 结构的优点包括：

- 它非常适合查找地址（公钥哈希）
- 它比许多其他树结构更快且内存效率更高
- 树的小部分可以在没有整体的情况下进行验证，并且可以证明完整性
- 将相同的数据插入任何顺序的 trie 总是会生成相同的结构

5. 迷你区块链

迷你区块链提供了我们区块链的第一个功能。迷你区块链本质上只是一个普通的区块链，只是我们不需要保留历史区块的副本。同样，它并不是真正有限的，因为改变最大块大小可能会增加迷你区块链的平均大小。在本文的后面，我们描述了一种动态确定的最大块大小的机制，但它不是该方案的必要部分。

如果我们要使用一组节点哈希和一个主哈希来跟踪我们的数据库，我们不能允许每个单独的事务按需更改数据库。我们必须将它们分解成以周期性时间间隔插入数据库的事务组。如果没有将交易组中的交易作为块来解决，我们就没有可行的方法来维护账户树。这产生了对区块链的内在需求，但由于我们可以丢弃旧块，所以我们将其称为“迷你区块链”。

比特币需要完整的区块链，因为这是确定所有地址全部余额的唯一真正方法。然而，我们有账户树来完成管理账户余额和记录货币所有权的任务。我们不需要完整的区块链，我们可以扔掉旧块并节省大量的磁盘空间。然而，我们确实保留了几百或几千个最新区块，这就是我们的迷你区块链。迷你区块链还为我们提供了一定程度的安全性。

每个区块的头部都嵌入了主哈希，我们可以从头开始验证迷你区块链中的每个区块，确保每个区块中的交易始终与前一个区块中的主哈希相对应。由于每个区块在被接受到迷你区块链之前都需要一个工作量证明过程，因此攻击者很难生成假的迷你区块链。虽然很困难，但如果我们完全删除旧块，这远非不可能。

使用比特币，我们可以从一开始就一直验证到最新的点，因为我们拥有完整的区块链。如果攻击者从可用的最旧区块创建一个新的迷你区块链，那么新节点将很难将其与真正的迷你区块链区分开来，因为在最旧的区块之前，他们没有记录历史发生过的事情。攻击者可以根据需要花费一定时间来建立他们的迷你区块链的累积难度，因为这不是一个他们必须超越的不断增长的链。

然后攻击者可以开始广播假链，它可能传播到足以施加成为主链的风险。证明链通过提供一种机制来解决这个问题，该机制可以充当存储长期工作量证明历史的容器，以便我们可以计算任何链的总累积难度。我们必须维护块头，而不是完全删除旧块，以便我们始终可以跟踪任何给定链的历史并比较每条链的总累积难度。

6. 证明链

证明链提供了我们区块链的第二个功能，它本质上只是一个区块头链。当节点丢弃旧块时，它们不会丢弃块头，只会丢弃交易。所以基本上迷你区块链是一个除最近交易外所有区块链都被修剪掉的区块链。这意味着所有节点仍然可以使用区块头链来验证具有最高累积难度的最佳迷你区块链，并且由于帐户树，他们不需要旧交易来计算地址余额。

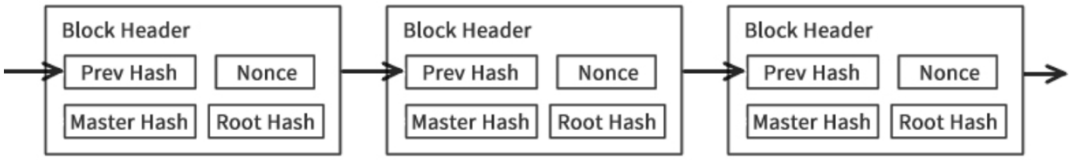

Figure 1-a

我们拥有本质上是普通区块链的东西，因此挖矿可以像在比特币中一样工作。节点必须对块头进行哈希处理，并在当前目标下方搜索生成的哈希值。可以丢弃旧交易并仅存储区块头链，因为证明解决方案不依赖于知道区块中的交易。与比特币安全的原因相同，它是安全的，证明链中的每个证明都会输入下一个证明，因此几乎不可能生成假证明链。

主哈希需要在区块头中，如 Figure 1-a 所示，因为它允许节点验证区块中的交易并确保其帐户树已被区块正确更改。存储在块头中的主哈希是在块中的交易被应用到账户树之后计算的。我们可以从证明链的开始一直计算到它进入迷你区块链的地方，我们可以验证我们拥有的最新区块是否有效。

证明链证明了哪个迷你区块链具有最长期的计算支持。攻击者不再需要永远坐在那里生成假证明链，因为证明链必须输入迷你区块链。因此，现在如果攻击者试图创建一个完全无效的迷你区块链，他们还需要一个强大的证明链来配合它。这基本上将我们带回到典型的完整区块链方案提供的安全级别，但在所有情况下它仍然不是完全安全的。

主要问题是攻击者可以使用无效的帐户树秘密建立合法的证明链，然后在他们认为它太长以至于没有人可以追溯到那么远的历史时将秘密链发布到网络。在那种情况下，新节点将无法检测到哪个链是真实的。 我们称之为“The Secret Chain Attack”。 如果一周后可以丢弃所有交易（如在 Cryptonite 中），则攻击者必须将大部分哈希能力维持一周以上（秘密）。

我们相信，即使这种攻击确实发生，也不会是灾难性的，因为：

1) 攻击只会影响超过一周未与网络同步的节点，其他节点都可以检测到攻击并拒绝假链。
2) 新节点可以检测到可能正在进行的秘密链攻击，尽管他们不知道哪条链是真实的。
3) “社区检查点”的发布可以将节点指向正确的链，以防这种攻击确实发生。

7. 网络□为总结

网络同步

网络同步通过 4 个步骤实现：

1) 获取累积难度最高的证明链。
2) 获取与证明链相关的迷你区块链。
3) 通过请求切片和验证哈希来构建帐户树。
4) 使用最近的交易来完成账户树的同步。

首先，节点将使用“headers first”的方法来定位具有最高累积难度的链。然后它将获取连接到这些块头的最新块的集合。然后它将尝试获取帐户树的切片，直到它拥有完整的树。帐户树结构允许节点证明它已收到完整的切片，以便节点可以确定它拥有所有帐户。最后，节点可以使用最近的事务将树的所有切片更新为最新的主哈希。

不需要任何人存储旧的账户树数据，但节点必须构建一棵与其拥有的主哈希完全一致的账户树；他们不能混合和匹配与不同主哈希关联的切片。这就是为什么必须确定每个切片的高度，以便我们知道在第 4 步中需要应用哪些事务。由于所有切片都包含主哈希，因此当节点请求特定切片时，它可以将切片与特定块进行匹配通过比较哈希。

节点通常会尝试获取在可以安全丢弃旧事务的点附近同步的切片。其他节点能够提供这些旧切片，因为它们可以撤消对帐户树的最近更改并根据请求生成旧切片。新节点将尝试在这样一个旧点构建帐户树，因为它们需要能够创建一个大型“反转数据库”，其中包含撤消对帐户树所做更改的指令，这对于生成旧切片和处理分叉很有用。

简单来说，新节点会尝试围绕可以修剪旧区块的点构建完整的账户树，然后使用来自最近区块的交易“快进”，同时构建一个反转数据库，这样它的反转可以追溯到不再需要反转的地步。当然，这个过程的描述非常简单，您必须阅读 Cryptonite 的源代码才能更详细地了解同步的工作原理。

请注意，此过程不依赖于对块的大量检查，人们信任他们拥有的块，因为证明链支持它们。验证证明链非常容易，一旦完成，节点只需要确保它获得的块与证明链匹配。由于正在构建帐户树，唯一重要的是它最终拥有最新块的主哈希。一旦完成并且节点同步，它可以通过接受有效块开始正常更新帐户树。

交易

比特币通过阅读区块链来跟踪地址余额以查看发生了什么，它是一个持续的分类账本，而不是一个独立的资产负债表。比特币交易使用包含“输入”和“输出”的系统，大多数新交易的输入通常参考之前交易的输出。迷你区块链方案使用基本的输入和输出概念，但输入指向账户树中的账户，输出也指账户树中的账户。

输入账户将为发送到输出账户的硬币提供资金。此操作将导致输入账户余额减少，输出账户余额增加。费用仍然照常使用，以优先交易并为矿工提供激励。

显然，如果某笔交易将任何账户的余额减少到 0 以下，或者如果它请求任何与任何余额的价值相冲突的东西，或者它尝试任何它没有权限做的事情，则它不应该被接受为有效。

为了确保网络不会多次处理相同的签名交易，交易还必须包含“lockheight”字段。一旦锁定高度超出了节点需要保留的块范围（我们称其为“可见”块），事务就会无效，并且相同的 txid 不能在任何可见的块中包含两次。这使得不可能两次使用相同的 txid。但是，此解决方案要求 txid 不可延展。

在尝试解决交易延展性时需要考虑几件事，但最重要的是我们不能在对交易进行哈希处理时包含签名，因为使用相同的密钥签署相同的数据每次都会产生不同的签名。发件人将对 txid 签名，但多次签名不会改变 txid，只会改变签名。因此，尝试更改交易内容将始终更改 txid，从而导致签名无效。

帐户树

账户树有多种实现方式，但数据结构必须满足一定的要求：

1) 所有数据都应该能够通过确定性哈希（主哈希/账本指纹）有效地汇总。
2) 高效支持 4 种操作：添加账号、修改账号、删除账号、查找账号。
3) 每次修改后，应该能够有效地更新帐户树主哈希。
4) 应该允许有效验证账户子集的正确性，而无需下载整个结构。

当硬币被发送到一个尚不存在的地址时，新帐户将作为叶节点插入到 trie 结构中，并在地址清空时从 trie 中删除。当地址确实存在时，交易将简单地改变 trie 中的现有帐户。迷你区块链在树更新时进行协调。当一个节点收到一个新区块时，他们将通过相应地更改其帐户树来执行该区块中列出的交易。

提议的帐户树结构允许以“资产负债表”格式汇总所有地址余额，并允许所有节点安全地丢弃旧交易。但是，如果没有办法协调帐户树何时以及如何更改，我们仍然无法确保节点之间的一致性。这就是迷你区块链的用武之地。每次新区块被接受到迷你区块链中时，节点将使用该区块以一致且协调的方式更新其帐户树的副本。

在分布式网络中，每个节点都不可能在看到交易的那一刻就应用交易。交易必须连接在一起并批量应用。这样的交易列表组合在一起形成一个块，并与标头一起形成区块链。节点收集交易并将其应用于帐户树以实现新的帐户树状态。新树状态的主哈希包含在块头中。接收此类块的其他节点可以自己重放交易并检查哈希匹配。

8. 讨论新的网络协议

动态最大块大小

正如本文的介绍部分所述，关于最大块大小存在很多激烈的争论。一个值得讨论的潜在新网络协议是动态最大块大小的想法。它可以是一个浮动值，可能由几个因素决定。我立即想到的两种方法是 1）基于采矿的投票系统和 2）分析一些先前块并计算平均块大小以得出新的块大小限制（例如 2*average）的系统。

投票系统听起来是可行的，它可以让我们通过群体共识来管理最大区块大小，但它赋予了矿池等群体很大的权力来决定最大区块大小。更好的解决方案是简单地计算一些最近块的平均大小，然后将其乘以某个值以得出新的最大块大小，并具有一些任意的下限。这样我们就不需要将所有投票数据存储在区块中。这是我们决定采用 Cryptonite 的方法。

即使使用我们的轻量级方案，也确实需要最大块大小，因为我们的网络只能在几乎停止工作之前处理这么多的流量。即使使用有限的区块链，它也可以通过足够大的块快速增长。然而，在未来我们可能能够处理更大的块，因此我们需要一种随着时间的推移逐渐改变它的方法。如前所述，硬分叉不是很方便。自动调整系统将是自我调节的并且更加无缝。

修剪帐户树

迷你区块链方案中最庞大的部分实际上是账户树。如果有足够的时间，账户树可能会充满许多低余额账户，如果我们可以在账户足够老时从树上修剪掉这种类型的“灰尘”，那将是有利的。有几种方法可以实现这一点，但它们都不是远程简单或易于实现的。最好的方法似乎是收取“账户维护”费用以维护账户树中的账户。

从账户提款时将收取费用，并与交易费用一起包括在内。费用将根据发送帐户的年龄计算。通过这种方式，低余额账户最终会达到余额 0 并从树中删除。即使没有从账户中提款，我们也可以有一个系统来允许修剪账户，如果他们支付了账户维护费用，这些账户的余额将是非正数。

这种类型的系统的好处是它在帐户树中存储数据时会产生成本，这在经济上是有利的，并且有助于保持帐户树的紧凑。该系统的另一个有用功能是我们可以将维护费用反馈到“coinbase 账户”（支付区块奖励）并确保区块奖励永远不会达到 0，而不会实际增加货币供应量，只需将硬币回收回来通过采矿系统，在不中断铸造过程的同时维持有限的货币供应。

提款限额

由于我们的资产负债表方法，在此方案中可以很容易地为个人账户设置提款限额。提款限额规定了每个区块可以从账户中提款的最大硬币数量，这可能出于多种不同的原因而有用。需要在账户结构中添加三个额外字段才能实现这一点：1）账户上次修改的时间（这对于修剪账户树也很有用）2）当前提款限制 3）可能排队的提款限制。

该限额由账户所有者使用特殊交易自行设置，新账户的默认提款限额是无限制的。提款限额的主要目的是帮助防止双重支出，并使商家对确认次数较少的交易更有信心。如果他们知道每个区块只能从账户中提取一定数量的硬币，那么他们知道即使是 0 确认交易也可能通过，因为攻击者无法一次提取所有硬币。

提款限额如何运作的简要概述：

1) 向网络发送特殊交易以修改您账户的提款限额。 限制指定为每个块的硬币数量，并保存到队列字段中。此类更改将在例如生效。100 个区块，

然后排队的值会覆盖实际的提款限制值。

2) 网络接受特殊交易，在 100 个区块后，它将拒绝任何会导致超出新指定限制的交易。

商家可以通过以下方式确保他将收到资金：

1) 检查发送账户是否有排队提款限额更改。
2) 检查发送账户余额是否足够高，不能过快清空。
3) 确保事务的优先级不低，并且在网络中传播得足够多。

9. 确定技术规格

迷你区块链

如果证明链提供了我们大部分的安全性，乍一看似乎几乎没有必要存储超过 1 或 2 个块的任何内容。当然，至少有几百个区块是必要的，因为如果太短，Secret Chain Attack 攻击就会变得更加可行。出于几个原因，我们至少需要一些合理数量的最小区块历史记录来保存所有节点。我们认为 1 周对于 Cryptonite 来说是一个不错的数字，但在这个特定领域有很大的试验空间来找到最佳权衡。

我们需要考虑的另一个重要因素是块之间的时间。设置太短可能会出现问题，例如由于同时解决块而导致的孤立块数量增加，但是设置太长会使仅等待 1 个确认完全不切实际。根据对山寨币的简单调查，最佳出块时间似乎在 1 或 2 分钟左右。Cryptonite 恰好有 1 分钟的出块时间，因为它非常快，但不是那么快，它会导致太多的孤立块。

货币供应和分配

比特币使用 2.1 万亿个单位，每个硬币由 1 亿个单位组成，总共有 2100 万个硬币可以被小数点后 8 位整除。然而，比特币内部使用 64 位整数，它可以处理更多的硬币，所以 2100 万听起来像是一个相当随意的数字，但实际上并非如此。许多应用广泛使用双精度浮点数，但双精度浮点数最多只能存储 2^53 的整数，这就是比特币最终只有 2^50.9 个单位的原因。

但是，可以使用扩展精度浮点数来利用 64 位整数的全部范围。这使硬币供应的粒度更高，并且基于自然上限。一个 64 位粒度的硬币供应由大约 1844 亿枚硬币组成，每个硬币都可以被小数点后 8 位整除。我们在 Cryptonite 中使用了扩展精度浮点数，因此我们不再受双精度浮点数的限制，从而实现了极大的硬币供应。

代币分配有很多种方式，使用 Cryptonite 需要 10 年才能开采出一半的代币供应，重复半衰期为 10 年，但每个区块都会调整区块奖励，使其逐渐变化时间。区块奖励的突然大变化对网络来说是不健康的，并且不清楚为什么比特币被设计为在区块奖励的变化之间有如此长的时间间隔。 Cryptonite 也出于类似的原因更新每个区块的难度。

10. 结论

在本文中，我们描述了比特币网络协议的一种变体，旨在消除对完整区块链的需求并显着减少对长期数据存储的需求。这是通过将区块链的功能分离为优化以执行某些任务的单独机制来实现的。结果提供了一种纯粹的 P2P 加密货币，具有许多好处，例如增加了块空间。也有人建议使用同态加密技术来实现高水平的隐私[11]。

帐户树和迷你区块链的性质也可能提供更高级别的用户隐私，因为旧交易可能无法定位，但我们不太可能没有专门用于存储完整链的节点。我们确实以牺牲一些安全性为代价实现了更高水平的可扩展性，但没有什么是不能处理的。最终结果确实具有高级别的安全性，但同时它是超级紧凑和可扩展的。在许多方面它都优于比特币，但并非在所有领域（例如脚本功能）。

加密货币的未来会是什么样子？随着可扩展加密货币技术的出现，它看起来更好了。加密货币可以改变世界的运作方式，但前提是它的扩展能力足以满足世界的需求。光有想法是不够的，我们必须实施这些想法。从这个意义上说，中本聪是非常受人尊敬的，因为他主动创建了一个极其复杂的系统，该系统建立在一系列以前从未测试过的新奇的概念之上，他永远改变了世界。

参考文献

[1] Nakamoto, S. 2008. Bitcoin: A peer-to-peer electronic cash system. http://bitcoin.org/bitcoin.pdf

[2] Blockchain.info. 2013. Blockchain Size Data. https://blockchain.info/charts/blocks-size

[3] Andresen, G. 2013. Bitcoin-Qt / bitcoind version 0.8.0 released. https://bitcointalk.org/index.php?topic=145184

[4] Reiner, A. 2012. Ultimate blockchain compression. https://bitcointalk.org/index.php?topic=88208

[5] Todd, P. 2013. Bitcoin Blocksize Problem Video. https://bitcointalk.org/index.php?topic=189792

[6] Bitcoin Wiki. 2013. Scalability. https://en.bitcoin.it/wiki/Scalability

[7] Rosenfeld, M. 2012. Are there any studies into the size of the blockchain scaling over time? http://bitcoin.stackexchange.com/questions/2798/

[8] Nakamoto, S. 2008. Re: Bitcoin P2P e-cash paper. http://www.mail-archive.com/cryptography@metzdowd.com/msg09964.html

[9] Bruce, J. 2013. Cryptocurrency with Finite "Mini-Blockchain" https://bitcointalk.org/index.php?topic=169311

[10] Mini-blockchain Project, 2014. Cryptonite. http://cryptonite.info/ and wiki: http://cryptonite.info/wiki/

[11] Franca, B.F. 2015. Homomorphic Mini-blockchain Scheme. http://cryptonite.info/files/HMBC.pdf