Computer Science and Application 计算机科学与应用, 2019, 9(9), 1655-1666
Published Online September 2019 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2019.99185

Hans汉斯

Analysis of the State Capacity Problem of Ethereum 2.0

Na Tian, Zhihuai Li*, Huawei Li

School of Information Science and Technology, Dalian Maritime University, Dalian Liaoning
Email: *qhlee@dlmu.edu.cn, 1946533325@qq.com

Received: Aug. 14th, 2019; accepted: Aug. 29th, 2019; published: Sep. 5th, 2019

Abstract

As one of the most well-known public chain projects, Ethereum has the world's largest blockchain open source community and the most active public chain in the world. But at the same time, performance is also a huge bottleneck. Therefore, Improving Transaction Per Second (TPS) is an important goal for Ethereum to survive and develop in the next step, but at the same time, capacity issues will be exposed. According to the improvement plan of Ethereum 2.0, the paper analyzes the possible state requirements and hidden dangers. Highlights include: related calculations and assessments; node synchronization, expected capacity requirements, and potential hazards when TPS is expected. Finally, it summarizes and looks forward to the key research directions in the future.

Keywords

Blockchain, Ethereum, Transition Throughput, Performance, Capacity

以太坊2.0的状态容量问题的分析

田　娜，李志淮*，李华威

大连海事大学，信息科学技术学院，辽宁　大连
Email: *qhlee@dlmu.edu.cn, 1946533325@qq.com

收稿日期：2019年8月14日；录用日期：2019年8月29日；发布日期：2019年9月5日

摘　要

作为最知名的公有链项目之一，以太坊拥有全球最大的区块链开源社区，也是全球最活跃的公有链。但同时，性能也是极大的瓶颈问题。因此，提高交易吞吐量(Transactions Per Second，TPS)是以太坊下

*通讯作者。

文章引用: 田娜, 李志淮, 李华威. 以太坊 2.0 的状态容量问题的分析[J]. 计算机科学与应用, 2019, 9(9): 1655-1666.
DOI: 10.12677/csa.2019.99185

一步能够生存和发展的重要目标，但与此同时，容量问题将会暴露出来。按照以太坊2.0的改进方案，分析可能存在的状态容量需求和隐患问题。主要包括：相关的计算与评估；达到预期TPS时的节点同步情况、容量需求以及可能的隐患问题。最后进行总结并展望了未来的重点研究方向。

关键词

区块链，以太坊，交易吞吐量，性能，容量

1. 引言

区块链是一种分布式、去中心化的计算与存储架构，首次出现在中本聪发表的《比特币：一种点对点式的电子现金系统》[1]中。因其去中心化、可追溯、不可篡改等特点，被认为是继大型机、个人电脑、互联网、移动/社交网络之后计算范式的第五次颠覆式创新，是人类信用进化史上继血亲信用、贵金属信用、央行纸币信用之后的第四个里程碑[2]。

2013 年，Vitalik Buterin 提出了“以太坊[3]”的概念。以太坊在比特币提出的区块链技术[4]的基础上，创新地引入了智能合约[5]，用户在以太坊区块链上不仅仅发送交易，还可以调用自定义的代码，这使得以太坊上的生态迅速繁荣起来。以太坊虚拟机(Ethereum Virtual machine，EVM)是以太坊的核心，以太坊区块链上每个节点都运行着 EVM，这支持用户在以太坊网络中创建并调用一些复杂的逻辑，除此之外，允许任何人在平台中建立和使用通过区块链技术运行的去中心化应用。截止到 2019 年 3 月 26 日，以太坊区块链上已有 2399 个 DApp [6]，称以太坊是全球最活跃的公有链实至名归。

以太坊的联合创始人 Joseph Lubin 近日在接受采访时表示，以太坊区块链将在 18~24 个月内扩展 1000 倍。

然而，为了可以实时完成对新区块和未确认交易的验证，所有用户的账户信息以及所有智能合约状态都需要驻留在内存中，每一个全节点都会需要承担这样的一个负荷，如果要出块，还需要做额外的事情。对于以太坊系统，如果以太坊系统 TPS 提升 1000 倍，或者用户规模扩大 1000 倍，互联网上的普通服务器顺利部署一个全节点将是一个难题。新节点参与到系统的门槛提高，将会影响区块链系统的去中心化程度，如果一个普通人无法独立部署一个全节点，而只能由专业矿场操作的话，整个系统将会退化成一个中心化服务器，从而变得容易被攻击，因此，在提高系统性能的同时，容量问题不容忽视。

目前区块链面临的问题主要有三类，分别是隐私问题，性能问题和容量问题。本文主要对以太坊系统的容量问题进行分析。

在本文中，通过计算分析以太坊区块的交易吞吐量、区块大小和叔块率，来探讨在以太坊 2.0 [7]系统中，容量需求和节点的同步情况，以及可能存在的隐患问题。最后总结全文并讨论未来可行的解决方案。

2. 以太坊相关介绍

以太坊是一个重要的区块链应用平台，是先进公有链技术的代表之一。以太坊是一个基于交易的状态机，区块链中的每个区块对应一个状态，每产生一个区块，以太坊中的状态就会转换到下一个状态，通过状态转换使得运行以太坊中的所有节点保持数据的一致性。如图 1 所示，以太坊从创世块开始，产

生一个区块，系统状态就发生一次转换，不断产生的交易持续刷新当前状态，如图 1 所示。

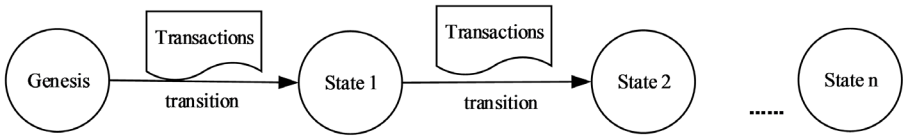

Figure 1. Ethereum state transition diagram
图 1. 以太坊状态转换图

2.1. 以太坊整体架构

以太坊基本结构主要分为三层，分别是：底层服务、核心层、顶层应用。其架构如图 2 所示。

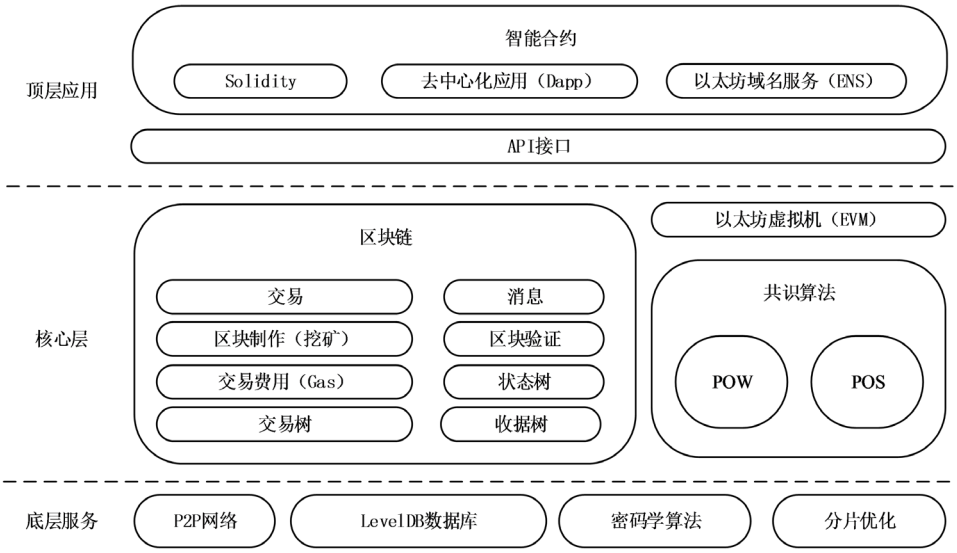

Figure 2. Ethereum overall architecture
图 2. 以太坊整体架构图

底层服务中 LevelDB 数据库中存储了交易、区块等数据，密码学算法为区块的生成、交易的传输等进行加密，分片优化加快了交易验证的速度，共识算法用于解决 P2P 网络[8]节点之间账本的一致性，顶层应用中的去中心化应用(DApp)需要在以太坊虚拟机(EVM)上执行，各层结构相互协同又各司其职，共同组成一个完整的以太坊系统[9]。

2.2. 数据结构

以太坊使用了 MPT 树(Merkle Patircia Tree，MPT)作为数据组成形式，用来组织管理用户的账户状态、交易信息等重要数据，MPT 树融合了 Merkle 树[10]和 Trie 树两种数据类型的优点。

2.2.1. Merkle 树

Merkle 树是一种树形结构，大多数是二叉树，也可以是多叉树，它由一组叶节点、一组中间节点和一个根节点组成。如图 3 表示一棵 Merkle 树，如果底层的交易被篡改了，那么其对应的叶节点散列值也会改变，最终导致 Merkle 树根值发生改变。如果一个恶意用户尝试在树的下部加入一个伪造的交易，将会导致根节点的改动以及区块散列的改动，这样协议就会将其记录为一个完全不同的区块[9]。

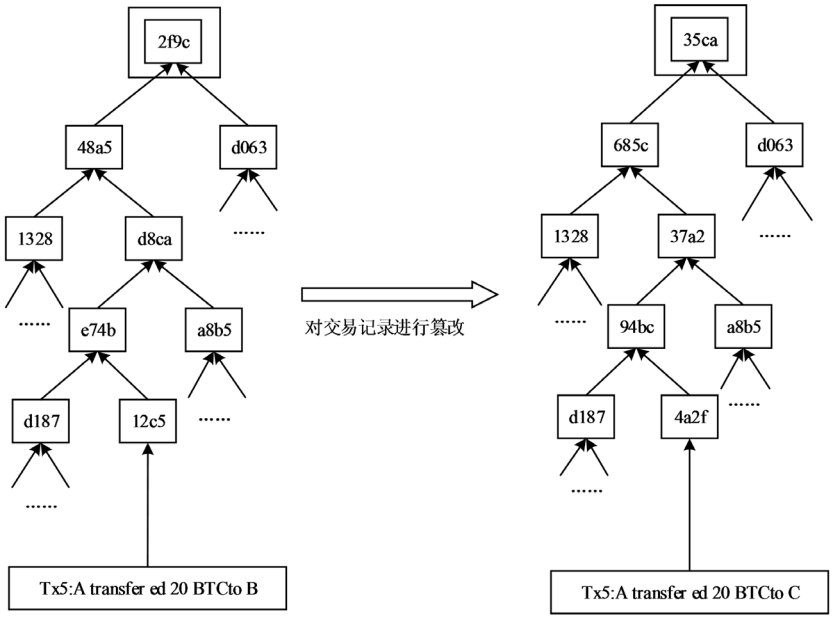

Figure 3. Merkle tree
图 3. Merkle 树

2.2.2. Trie 树

Trie 树也叫做 Radix 树，又称字典树，用于保存关联数组，其 key 的内容通常为字符串，代表着从根节点出发到对应 value 的一条真实路径。value 存储在叶节点中，是每条路径的最终节点，Trie 树的节点在树中的位置由其 key 的内容决定，即 Trie 树的 key 值由被编码在根节点到对应 value 的路径中。如图 4 所示，是一个有 6 个叶子结点的 Trie 树结构，则其 key 值分别为：to；tea；ted；ten；A；inn。

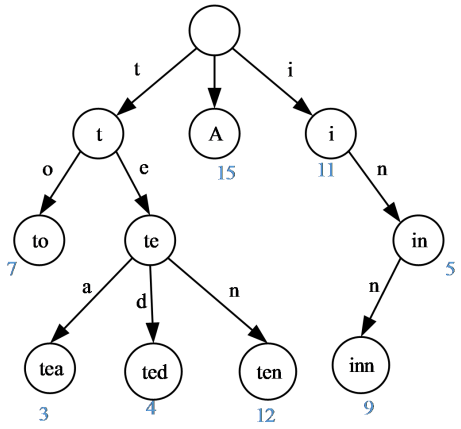

Figure 4. Trie tree
图 4. Trie 树

由图 4 可知，如果有两个 value 基于相同前缀的 key，它们相同前缀的长度占自身比例越大，则这两个 value 在 Trie 树中的位置越靠近。

2.2.3. Merkle Patricia 树

以太坊采用的 Merkle Patricia 树，综合了上述两种树的优点，并对其进行改进。在以太坊系统中，为

MPT 树新增了空节点、叶节点、扩展节点和分支节点四种不同类型的节点来提高效率。

在图 5 所示的状态树中，节点 A、E 是分支节点，节点 B、D、F、G 是叶节点，节点 C 是扩展节点。

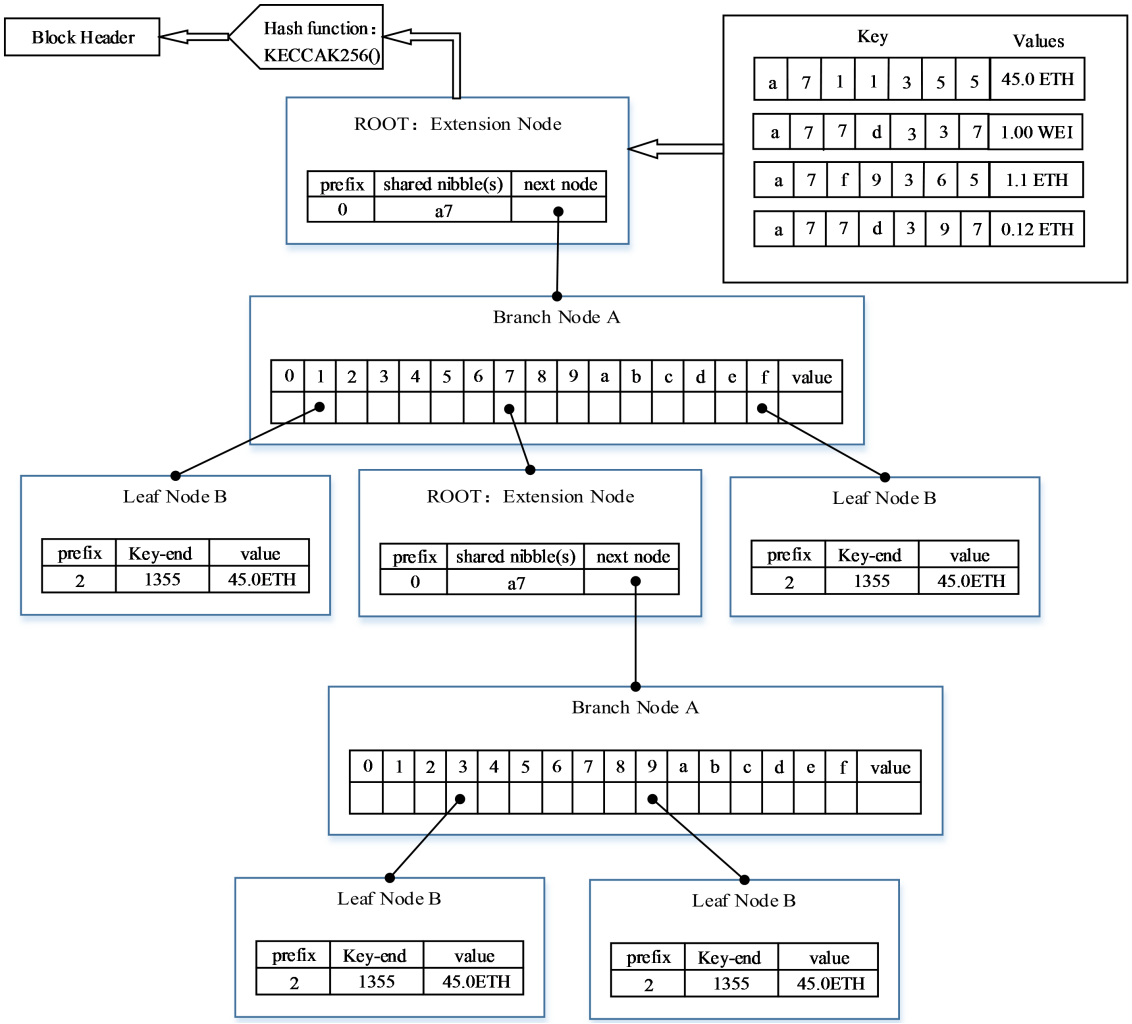

Figure 5. Merkle Patricia tree
图 5. Merkle Patricia 树

以太坊区块头中除了 MPT 树，还存储了三棵树，分别是交易树、状态树和收据树。状态树中每个叶节点表示一个账户，是用来记录各个账户的状态的树，需要经常更新；每个区块都有一颗独立的交易树，区块中交易的顺序主要由“矿工”决定；每个区块都有自己的收据树，收据树代表每笔交易相应的收据。

2.3. 存储

在以太坊中，数据的存储大致分为三个部分，分别是：状态数据、区块和底层数据。以太坊中的区块、交易等数据最终都被存储在 LevelDB 数据库中，存储形式是[k,v]键值对，存储结构图如图 6 所示。

2.4. 区块结构

区块本质上是一个数据包，以太坊的交易记录保存在区块中。区块也可以理解为记录一段时间内发生的交易和状态结果的数据结构，是对当前账本状态的一次共识。在以太坊中，区块主要由三部分组成：

区块头、叔区块头和交易列表三部分组成。

区块头是区块的核心,区块头中每个字段的意义如表 1 所示。

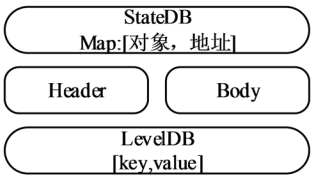

Figure 6. Storage structure diagram
图 6. 存储结构图

Table 1. Block header field and meaning
表 1. 区块头字段及意义

名称	类型	意义
ParentHash	common.Hash	父区块的哈希值
UncleHash	common.Hash	叔区块列表的哈希值
Coinbase	common.Address	打包区块的矿工的地址，用于接收旷工费
Root	common.Hash	状态树的根哈希值
TxHash	common.Hash	交易树的根哈希值
ReceiptHash	common.Hash	收据树的根哈希值
Bloom	Bloom	交易收据日志组成的 Bloom 过滤器
Difficulty	*Big.Int	区块难度
Number	*Big.Int	区块号，从 0 开始
GasLimit	uint64	区块中所有交易消耗的 Gas 上限
GasUsed	uint64	区块中所有交易使用的 Gas 的和
Time	*big.Int	区块产生的 unix 时间戳，一般指打包区块的时间
Extra	[]byte	区块的附加数据
MixDigest	common.Hash	哈希值，与 Nonce 的组合用于工作量计算
Nonce	BlockNonce	区块产生时的随机值

3. 计算与评估

以太坊 2.0 将在 18~24 个月内实现扩展 1000 倍的目标，接下来，在这一小节中，将分别对以太坊交易吞吐量、区块大小和出块时间进行计算，并分析实现扩展目标后可能存在的隐患问题。

3.1. 交易吞吐量

以太坊交易吞吐量指的是在以太坊区块链中每秒能够处理通过的交易数量，用 TPS 来表示交易吞吐量，公式如下：

$$\mathrm{TPS} = \left(\mathrm{gasLimit}/\mathrm{gas}\right)/\mathrm{time} \tag{1}$$

其中，gasLimit 是单个区块允许的最多 gas 总量，gasLimit 决定着区块容量大小，gasLimit 大小由矿工决定；gas 在这里指的是单笔交易的消耗；time 是区块出块时间。

目前以太坊上 gasLimit 平均大小为 8,000,000，在以太坊上消耗 gas 数量最小的操作是发送支付交易，支付交易的 gas 数量为 21,000。目前以太坊平均出块时间为 15 秒。则代入上述公式可得

(8,000,000/21,000)/15 = 25，即以太坊目前交易吞吐量最大可达到每秒 25 笔交易。

由公式可以看出，以太坊上交易吞吐量主要与两个参数有关：

(1) 区块大小。

(2) 出块时间。

出块时间不变的情况下，区块容量越大，包含的交易数量越多，交易吞吐量越大，但是区块大小也会影响出块时间，区块过大，会导致出块时间过长；出块时间越小，交易吞吐量越大，但出块时间大小也会影响区块大小，出块时间过小，系统来不及处理过多交易，从而导致区块变小。

若以太坊 2.0 中，TPS 达到现在的 1000 倍，将会导致区块增大以及出块时间增加，区块过大或出块时间过长，都有可能产生出块困难、网络阻塞等问题。

接下来分析区块大小和叔块率。

3.2. 区块大小

区块包括区块头、叔区块头和交易列表，区块头和叔区块头平均所占内存为 540 B，区块所占内存主要由区块内包含的交易数量决定。

经计算，目前以太坊上交易吞吐量最大为每秒 25 笔交易，出块时间为 15 秒，则目前以太坊区块内最多可装 375 笔交易，由以太坊浏览器可得每个区块大小和每个区块中交易数量，由此计算每笔交易大小约为 175.5 B，如图 7 所示 [1]。此处取每条交易 180 B，计算可得，目前以太坊平均区块大小为 68,040 B。

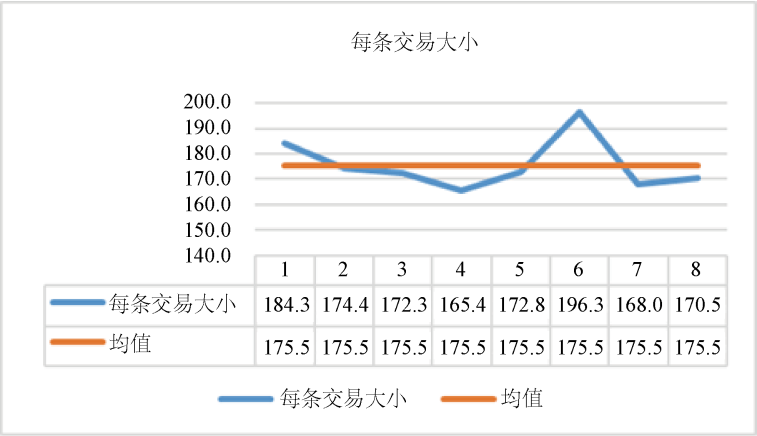

Figure 7. Transaction size
图 7. 交易大小

以太坊通过来控制区块大小，由函数可得如下公式。

$$\text{gasLimit} = \begin{cases} (1+1/1024)*G, & G' > 2/3G \\ (1-1/1024)*G, & G' \le 2/3G \end{cases} \tag{2}$$

其中，gasLimit 是新出区块的 gasLimit，G 表示父区块的 gasLimit，G' 表示父区块的 gas 使用量，公式表示，当父区块 gas 使用量 $G' > 2/3G$，会较上次增大 gasLimit，反之减小。假设每次父区块 gas 使用量都满足 $G' > 2/3G$，那么，每次新出的区块的最大容量都会线性缓慢增加。在以太坊中，矿工挖矿时将会得到新区块奖励、引用叔块的奖励和在区块中运行合约的 gas，区块越大，矿工得到的奖励越多，但是在出

[1] 计算交易大小，数据是根据以太坊浏览器中提供的数据，生产的图表。以太坊浏览器：https://etherscan.io/

块时间不变的情况下，相应的带宽要求也会增加，带宽达不到要求，将会导致区块中交易来不及处理，从而导致较高的分叉率。因此区块过大将会导致出块困难、网络阻塞等问题。

3.3. 叔块率

叔块率(uncle rate)在交易的gas消耗的改变下如何变化是我们用来鉴定以太坊区块链能安全得承受多大压力的一个重要因素。比特币无限研究[11]表明，因为交易扩散技术的提升，现在的扩散时间已经降低到约 0.008 秒每千字节。如果一个区块需要更长的时间来扩散，那么它变成陈腐块的机率也越高。公式表示如下。

$$R = (0.008 * C)/T + r \tag{3}$$

在上述公式中，R 表示总区块率；C 表示出块区块容量大小，单位为千字节；T 表示以太坊当前出块时间；r 表示以太坊当前叔块率。由公式可知，每增加 1 秒的扩散时间，会相应增加 1/T 的叔块率。由上一小节可知，现在以太坊上区块大小为 68,040 B，则区块扩散时间为 0.54 秒，以太坊出块时间为 15 秒，根据公式得出，区块率增加 3.6%，目前以太坊叔块率为 7.5%，总叔块率为 11.1%。

以太坊出块时间与矿池总算力和挖矿难度相关，根据矿池算力调整挖矿难度，将出块时间控制在 15 秒左右，那么在区块出块时间不变的情况下，交易吞吐量越高，区块容量越大，广播该区块所需时间越长，那么该区块成为叔块的概率越大。以太坊为避免大量的算力浪费，采用了 GHOST 协议，对产生或者发现并引用孤块的矿工进行奖励，即叔块奖励，以减少因区块广播不及时导致的区块链分叉问题，但是根据 GHOST 协议，如图 8 所示，每个区块最多引用两个叔块，且为避免矿工故意在链上制造分叉后等待被后边的区块引用而获得奖励，以太坊限制链中七代及以内的叔块可以得到奖励，超过七代的叔块将不会得到奖励，当叔块率提高到不能全部被引用时，将会造成大量叔块成为孤块，影响区块链的共识，从而造成系统的安全性降低。

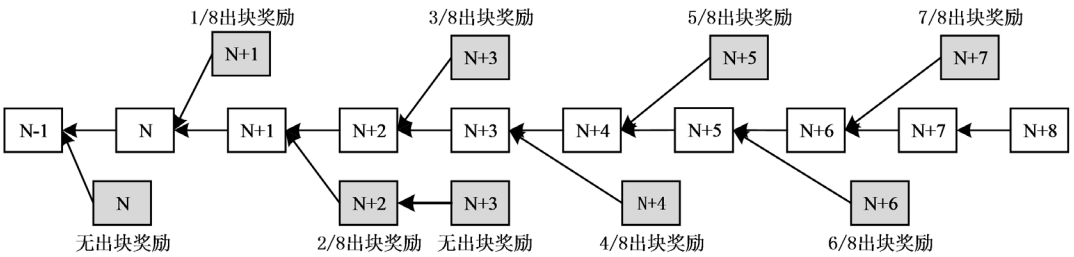

Figure 8. GHOST protocol
图 8. GHOST 协议

BTC 和 ETH 发行量的对比图如图 9 所示 [2]，以太坊 token 的产生与比特币不同，比特币发行量大约每四年减半，以太坊的发行每年产量被限制在 7200 万以太币的 25%，即每年以太币的矿产量不高于 1800 万，如图中虚线所示。

这 1800 万矿产量包括区块奖励、叔块奖励以及叔块引用奖励，叔块率增加，会在一定程度上导致叔块奖励和叔块引用奖励的增加，相应的，会造成区块奖励减少，即区块减少，交易吞吐量提高的同时，区块减少，会导致大量交易排队等候，部分交易始终没有被矿工看到，从而造成长时间不被确认，最终超时的问题，随后出现潜在的中心化问题。

[2] 比特币和以太坊 ETH 发行量对比图，是依据官方承诺的比特币发行量每四年减半，以太坊发行量每年不超过 1800 万，根据这个规律，做出 BTC 和 ETH 从出生到 2045 年这几十年的总发行量变化趋势。

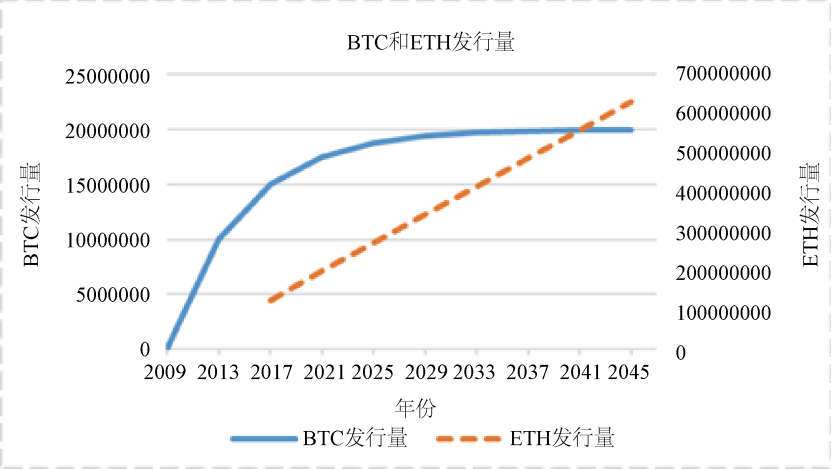

Figure 9. BTC and ETH circulation comparison chart
图 9. BTC 和 ETH 发行量对比图

4. 分析与探讨

区块链也可以看做一个全网节点共同维护的分布式账本，每个节点都是平等的，任何更改都要全网达成共识，并被记录下来。在以太坊系统中，保证去中心化的条件是满足绝大多数用户都能参与，因为只有保证了参与全网的条件大众容易满足，才能具备良好的去中心化特性，因此，对于分布于全球的以太坊而言，不能要求部署全节点的机器具备顶级的配置，否则将大幅降低节点数量，导致去中心化程度降低。

以太坊共识达成的过程要经历如下四个步骤：

(1) 交易产生：两个用户间发起一笔转账交易或用户与智能合约发生一次交易；

(2) 构造区块：以太坊网络中的节点根据账簿的最新状态，在所有未确认的交易中选择验证合法的交易进行打包，放入构造的区块中；

(3) 竞争出块：记账节点竞争获得出块权；

(4) 广播区块：记账节点获得出块权之后，向全网广播构造的区块，更新账本信息[12]。

4.1. 节点同步

上述共识达成过程要完成交易验证、区块验证、智能合约计算、区块信息存储等多个步骤。这些步骤与计算机 CPU、内存、网络带宽、硬盘容量等资源息息相关。目前一台普通计算机的配置，每秒可处理上万条交易，因此，在可控成本范围内，CPU 不会成为性能瓶颈。在当前以类金融为主流应用场景的情形下，区块链系统最首要的性能瓶颈是区块数据的广播延迟造成的，本质上受限于互联网的带宽和通讯延迟，这一点直接制约了 TPS；一旦吞吐量实现了大幅提升，容量问题马上会出现。

对于现有的单链系统，有一个难以逾越的物理限制，即全节点的平均带宽。根据以太坊官方在第 5,828,433 个区块[13] (2018 年 6 月 1 日中午左右开采)的快照，如图 10 所示，用一个测试节点同步到主网的顶端，每 24 小时放置一条水平蓝色线，造成坡度变化的原因可能是区块 GasLimit 的增加，由图可得，同步时间已超过 12 天。

同步到第 5,828,433 个区块后的数据大小为 341 GB，分布如图 11 所示。

按照图中数据计算，节点同步速度只有每秒 0.337 Mb，换算成带宽为 3 Mbps，也就是说当前全节点的平均带宽只有 3 Mbps，按此带宽速度计算，吞吐量上限为 141 TPS。当以太坊吞吐量超过 141 TPS 时，

新加入的节点将会永远同步不完整个网络的数据。全节点的平均带宽只有 3 Mbps 的原因是，以太坊节点分布于全球各地，当一个新加入节点进行同步时，需要若干节点的中继，网络延迟会比较大，从而导致无法全部利用带宽。

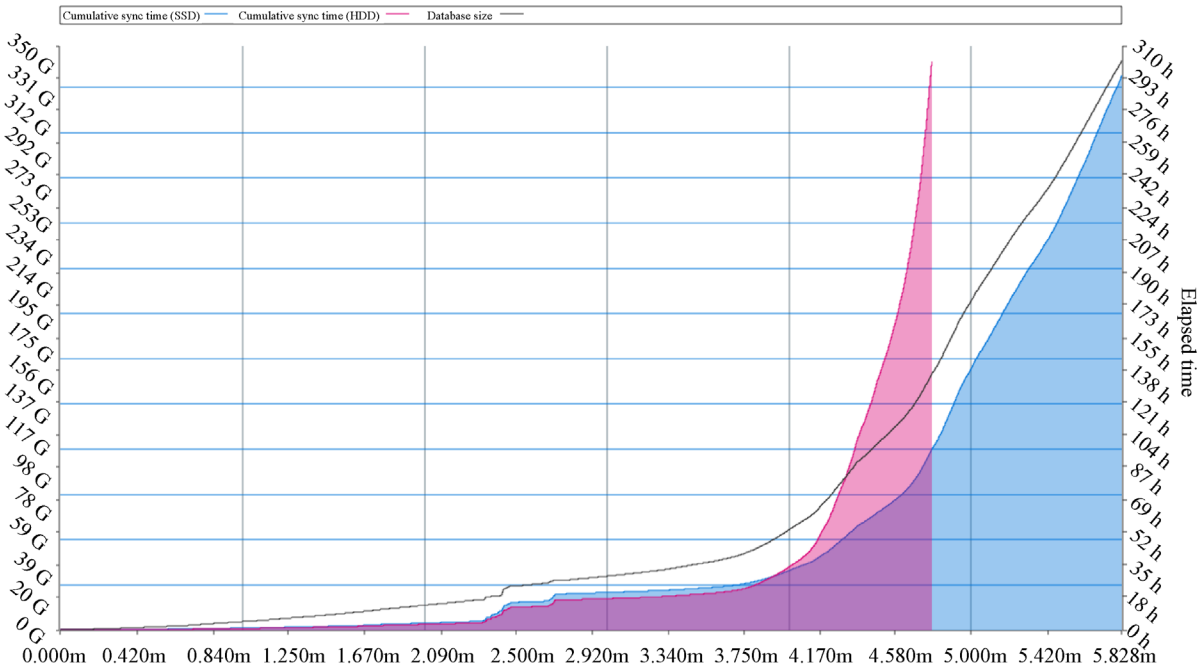

Figure 10. Synchronization time
图 10. 同步时间

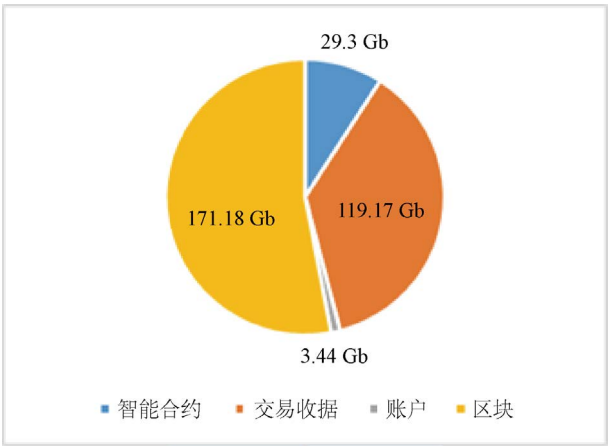

Figure 11. Data distribution map
图 11. 数据分布图

对以太坊系统来说，以太坊 2.0 阶段将会扩展 1000 倍，以互联网带宽中位数 13 Mbps 为例，无论是基于 POW 共识机制还是以太坊 2.0 使用的 POS 共识机制，即使完全丧失去中心化，忽略传送协议以及层层封包等额外引入的代价，以目前每秒处理 25 笔交易的吞吐量来算，假定出块时间不变，则计算得以太坊 2.0 的吞吐量上限为 609,375 TPS。超过该吞吐量会出现部分全节点脱网，长时间无法赶上全网的区块增长，离最新的区块越来越远，此时，一个新的全节点无法加入到网络中。这样导致实际只能有限个

节点参与到以太坊网络中，不能做到真正的去中心化，同时也容易产生分叉和双花问题。

同样以 13 Mbps 带宽为例，以太坊 TPS 理论上限为：

$$\mathrm{TPS}_{\max} = 13\ \mathrm{Mbps}/8\ \mathrm{Mb}/180\ \mathrm{B} \approx 9466 \tag{4}$$

这距离以太坊 2.0 版本扩展 1000 倍的改进目前还有一定的差距，可见，要达到此目标，必须要提高带宽标准，不考虑网络延迟等各种因素，带宽的最低标准为 35 Mbps。

当然，这只是一个理论值，当以太坊区块链扩展 1000 倍之后，区块大小理论也将会大幅提升，此时，广播该区块所需时间将会大大增加，待处理交易将会逐渐累积，可能会造成网络阻塞、分叉率增加等问题。

4.2. 容量问题

在区块链中，存储分为内存存储和硬盘存储。

4.2.1. 状态硬盘容量

硬盘主要存储所有的账本信息和确认后的区块，其容量需求与交易数量线性相关，对于目前以太坊区块链中的单个节点来说，硬盘容量还不会对系统的性能产生限制。但是，以太坊 2.0 扩展完成后，TPS 提高到现在的 1000 倍，彼时，硬盘成本也会直线上升。

取目前以太坊 TPS 为 25，区块中交易大小为 180 B，则计算可得以太坊 2.0 每年全网的交易数据如下：

$$180\ \mathrm{B} * 25000 * 3600 * 24 * 365 \approx 129\ \mathrm{TB} \tag{5}$$

可见，当达到以太坊 2.0 的扩展目标时，每年全网交易数据将达到 129 TB，这对硬盘容量的需求也将会是一笔不小的开销，目前普通计算机还未能达到此配置。

4.2.2. 内存容量

为了可以实时完成对新区块和未确认交易的验证，所有用户的账户信息以及所有智能合约状态都需要驻留在内存中，这个占据了主要的内存开销，内存容量同用户量(地址数量)以及智能合约数量线性相关。以太坊网络中，目前(截至 2019 年 3 月 31 日)全网地址总数为 60,163,582，每个地址平均消耗 68 个字节，智能合约数量达到 3430 个，若平均每个智能合约占用 300 KB，取地址总数为 61,000,000，则所占用内存至少为：

$$\mathrm{RAM} = 6.1\times10^{7} * 68\ \mathrm{B} + 300\ \mathrm{KB} * 3430 \approx 4.84\ \mathrm{GB} \tag{6}$$

假设一台普通的计算器内存为 8 GB，智能合约数量不变的情况下，当地址数量达到 110897414 时，所占内存已超过 8 GB，也就是说，当地址数量最多为 110,897,414 时，这台计算机的内存已达不到以太坊网络所需要的内存要求。在以太坊 2.0 中，将达到扩展 1000 倍的改进目标，假设用户数量扩大 10 倍，内存容量需求将达到 40 GB，这远远超过了普通计算机的配置要求，将导致只有一小部分的人有条件部署全节点，而一大部分的节点无法加入到区块链中。对全节点的要求变高，如果全节点只能由专业矿场操作，普通人无法独立部署一个全节点的话，那么整个系统将会退化成一个多地部署的中心化云服务了，进而变得容易被攻击，也容易被封禁。

5. 总结与展望

可见，交易性能瓶颈和状态容量瓶颈是现在区块链技术，尤其是公链领域，比较难以攻破的两大难题。公链是完全去中心化的，链上节点数量越多，去中心化程度越高，若现有架构没有改变的条件下，

提高性能，那么对节点的配置要求也会提高，要求越高，可加入的节点越少，中心化趋势就会越来越明显。这是两难的问题。

任何一个在线系统，如果没有一个大容量、高吞吐的基础设施，就无法承载哪怕仅仅一个互联网级别的应用。以太坊 2.0 扩展 1000 倍的改进目标，在性能大幅提高的同时，严重的状态容量需求问题将会暴露出来。因此，要获得大幅度的伸缩性，必须能够有一个合理的设计，使得单个节点仅仅负责整个网络吞吐量和容量的一部分，这样有可能在不升级节点配置条件、维持全节点的负荷比较小的前提下，使得全网的性能和容量有大幅提升。

在以太坊2.0系统中将引入分片技术[14]，将节点划分到不同的分片中，各分片内仍执行相同的任务，但分片间执行不同任务，从而可提升全网性能。分片技术分为网络分片、交易分片和状态分片。交易分片似乎能在一定程度上提升性能，但与此同时，也带来了更大的带宽和状态容量压力。因此，只有实现状态分片[15]才能在本质上解决以太坊可扩展性问题。

只有通过同时切分全网的工作量和状态容量，才可以在大幅提高 TPS 的同时，支持 10 亿以上级别的用户量，并且保持每个参与到这个网络中的全节点仅有一个合理的负荷，让大部分互联网上的普通终端都可以轻松部署一个全节点，共同参与网络的维护和治理。

致　谢

感谢李志淮老师的指导。

参考文献

[1] Nakamoto, S. (2008) Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf

[2] Swan, M. (2015) Blockchain: Blueprint for a New Economy. O'Reilly Media, Sebastopol, CA.

[3] Ethereum White Paper. A Next-Generation Smart Contract and Decentralized Application Platform. https://github.com/ethereum/wiki/wiki/White-Paper

[4] 袁勇, 王飞跃. 区块链技术发展现状与展望[J]. 自动化学报, 2016, 42(4): 481-494.

[5] Szabo, N. (1997) Formalizing and Securing Relationships on Public Networks. *First Monday*, **2**. https://www.firstmonday.org/ojs/index.php/fm/article/view/548/469

[6] State of the DApps. https://stateofthedapps.com/zh/stats/platform/ethereum#new

[7] Vitalik, B. (2018) ETH 2.0: The Road to Scaling Ethereum. Devcon4.

[8] Parameswaran, M., Susarla, A. and Andrew, B. (2001) P2P Networking: An Information-Sharing Alternative. *Computer*, **34**, 31-38. https://doi.org/10.1109/2.933501

[9] 闫莺, 郑凯, 郭众鑫. 以太坊技术详解与实战[M]. 北京: 机械工业出版社, 2018: 15-29.

[10] Berman, P., Karpinski, M. and Nekrich, Y. (2011) Optimal Trade-Off for Merkle Tree Traversal. *Theoretical Computer Science*, **372**, 26-36. https://doi.org/10.1016/j.tcs.2006.11.029

[11] Peter, R. (2015) Block Size Limit Debate Working Paper: A Transaction Fee Market Exists without a Block Size Limit. https://www.bitcoinunlimited.info/resources/feemarket.pdf

[12] 王嘉平. 区块链公链如何才能快起来(上)[EB/OL]. https://www.8btc.com/article/291181

[13] Akhunov, A. (2018) Recent Data on Turbo Geth Performance. https://medium.com/@akhounvo/recent-data-on-turbo-geth-performance-f7fb28b06a65

[14] Luu, L., Narayanan, V., Zheng, C., *et al*. (2016) A Secure Sharding Protocol for Open Blockchain. In: *CCS*'16, ACM, New York, 17-30. https://doi.org/10.1145/2976749.2978389

[15] Wang, J. and Wang, H. (2019) Monoxide: Scale out Blockchain with Asynchronous Consensus Zones. In: *NSDI*, USENIX Association, Boston, 95-112.

知网检索的两种方式：

1. 打开知网首页：http://cnki.net/，点击页面中“外文资源总库 CNKI SCHOLAR”，跳转至：http://scholar.cnki.net/new，搜索框内直接输入文章标题，即可查询；
   或点击“高级检索”，下拉列表框选择：[ISSN]，输入期刊 ISSN：2161-8801，即可查询。
2. 通过知网首页 http://cnki.net/顶部“旧版入口”进入知网旧版：http://www.cnki.net/old/，左侧选择“国际文献总库”进入，搜索框直接输入文章标题，即可查询。

投稿请点击：http://www.hanspub.org/Submission.aspx
期刊邮箱：csa@hanspub.org