doi:10.3969/j.issn.1002-0802.2018.06.026

区块链的税收智能合约设计 *

陈宇翔，张兆雷，刘地军，彭 笛，李 枫
（中国电子科技集团公司第三十研究所，四川 成都 610041）

摘 要：新兴的区块链和智能合约技术提供了解决资产交易问题的新途径。作为分布式账本数据库的区块链，记录了不可变更的数据资产。区块链上的智能合约直接控制资产交易，构造了基于智能合约的可信税收业务计算环境，为完全数字化资产的记录和转移奠定了基础。因此，基于 Hyperledger Fabric 区块链平台研究智能合约，给出了智能合约的税收场景设计，包括智能合约的执行模型、税收区块链的总体架构、Go 语言的合约实现及说明，分析了税收场景设计距离工程应用的距离、优缺点、可能的法律、监管问题及展望，为企业、政府开发基于区块链技术的智能合约提供参考。

关键词：智能合约；区块链；税收；法律；联盟链
中图分类号：TP316 **文献标志码**：A **文章编号**：1002-0802(2018)-06-1384-07

Smart Contract Design for Tax Scenario based on Block-Chain

CHEN Yu-xiang, ZHANG Zhao-lei, LIU Di-jun, PENG Di, LI Feng
(No.30 Institute of CETC, Chengdu Sichuan 610041, China)

Abstract: As a distributed account database, the block-chain records the non-changeable data assets. Smart contracts on the block-chain directly control asset transactions, thus to construct a trusted tax business computing environment based on smart contracts and lay a foundation for the recording and transfer of fully digital assets. Based on the Hyperledger fabric block-chain platform, the smart contract is explored, and the taxation-scenario design of the smart contract also given, involving the implementation model of the smart contract, the overall structure of the tax block-chain, and the implementation and description of the Go language contract. Meanwhile, the distances of tax-scenario design to engineering application, advantages and disadvantages, possible legal and regulatory issues, and prospects of tax-scenario design and engineering applications are analyzed, thus to provide certain reference for companies and governments in their developing smart contracts based on block-chain technology.

Key words: smart contract; block-chain; tax; law; consortium

0 引 言

区块链技术[1]是国内外企业、金融和政府关注的重点。智能合约作为区块链的重要组成，是区块链被称为颠覆性技术的主要原因。智能合约（Smart Contract）在区块链中可以概括为一段代码，运行在去中心化的、复制的账本上，能保持自己数据的状态，控制数字资产，对收到的外界信息和资产回应。

从用户看，智能合约是一个自动执行账户，满足预设条件就会释放转移资金。从开发人员看智能合约则是网络服务器，这些服务器不是被架在互联网上使用 IP 地址，而是架在区块链上。因此，区块链及其智能合约被认为是实现价值互联网的重要手段，最终实现人们在网络上像传递信息一样方便、快捷、低成本的传递价值。

* 收稿日期：2018-02-22；修回日期：2018-05-14 Received date:2018-02-22;Revised date:2018-05-14

1　智能合约的发展

早在 20 世纪 90 年代，密码学家 Nick Szabo[2] 就提出了智能合约的概念，用程序代码实现人们约定的合同条款。程序从外界获取信息识别并判断，满足条件则触发程序完成相应的操作，如资产转移、交易等。但是，由于缺乏支持可编程合约的数字系统和技术，Szabo 的理论迟迟没有实现，只有一些原始粗糙的应用，如刷卡机（PoS）、自动售货机等。

在生产产品和提供服务的过程中，要把人和物、人和人整合，才能创造更大的价值。比如，汽车发动机的几千个零件涉及多方面专业，需要不同工厂协作完成。所以，建立合约将某个部件生产外包给别人，别人也许会继续外包，形成一个合约网络。在双方约定的过程中，大家希望交易利于自己，所以要创建共赢的交易。如果用程序代码实现合约网络，则能更好地保障参与方的权益。

区块链的出现解决了平台问题，如去中心化、不可篡改、过程透明和可追踪等特点，天然适应于智能合约，也使智能合约被企业、金融机构所关注。

区块链 1.0 以比特币为代表，在去中心化的网络中支付加密货币。区块链 2.0 则要实现更宏观的目标，对市场去中心化，使网络中不仅交易货币，还包括其他数字资产，如房产、汽车等。区块链 3.0 则转向社会治理，如身份管理、公证、仲裁审计何投票等领域 [3]。

当前，区块链和智能合约技术的结合成为解决方案的研究热点，如 Hyperledger[4]、以太坊 [5–6] 和 Codius[7] 等，都通过建立在可执行基础设施上的可编程合约语言实现智能合约。智能合约作为区块链上的脚本，在节点间以分布式形式执行，类比法律法规在交易、合同中的应用。智能合约按照双方事先约定的条件，保证合约安全、可信、按需以及可监管的执行。表 1 展示了典型的智能合约项目平台。

表 1　三种知名区块链平台的智能合约应用对比

区块链平台	沙盒环境	编程语言	应用范围	准入机制
比特币	—	Script	比特币交易	公有链
以太坊	EVM	Solidity/Serpent	Dapp/ 以太币交易	公有链
Hyperledger Fabric	Docker	Go/Java	企业级应用	联盟链

不同的区块链平台针对不同类型的应用场景。公有链平台以比特币为代表，任何人都可参与记账；私有链平台通常用于金融机构或公司、政府的内部审计；设计的税收智能合约场景跨联盟和组织，通常多个机构组织具有记账权，属于典型的联盟链应用。因此，本文选择最具代表性的 Hyperledger Fabric 区块链平台进行设计。

2　智能合约的工作原理

智能合约[8–9]一般包括以下内容：交易（Transaction）处理，保存机制，一个完备的状态机用于接收和处理各种智能合约，事务保存和状态处理则在区块链上完成。

调用智能合约 [10] 需要满足触发条件，一旦符合条件，就从合约代码传出预设的数据资源。它最重要的特点是，输入智能合约的是一组事务，处理后的输出也是一组事务。类似于生活中的合同，使一组复杂、有触发条件的数字承诺按照参与方的意志自动执行。通用的智能合约模型如图 1 所示。

智能合约系统 Hyperledger Fabric 是一个开源区块链底层系统，像安卓一样提供了丰富的 API 接口，可以使人在上面开发各种区块链应用场景。主要较成熟的平台中，Hyperledger Fabric 是面向企业联盟的区块链应用，也是此次应用设计选择的主要原因。它的智能合约又被称为 Chaincode，选用 docker 容器作为沙盒环境，容器中有被签名的磁盘映像和 Go 语言运行的 SDK。上层应用通过 gRPC 或 REST，与运行在 Hyperledger Fabric 节点上的智能合约进行通信。

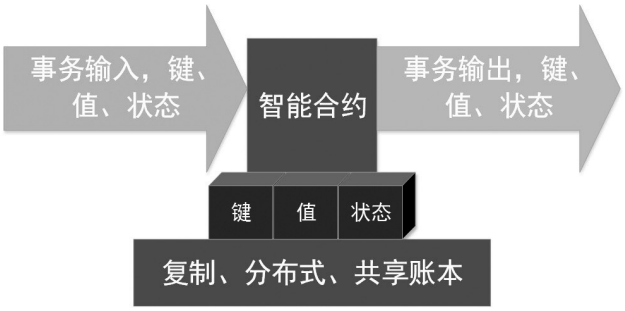

图 1　智能合约模型

3　税收交易区块链的总体架构

区块链从比特币发展到引入智能合约的以太坊，再到联盟链 Hyperledger Fabric，尽管各应用实现上有所差别，但体系架构上有一定的相似度。图 2 给出了税收区块链的体系架构及相关软件模块。

图 2 税收区块链体系架构及模块

最底层的网络层包括了 P2P 组网机制、数据传播机制和数据验证机制，使得区块链具有自动组网功能，可以适应网络变化而查询到完整的结果集，也方便数字资产交易类应用。Hyperledger Fabric 的 P2P 协议基于 HTTP/2 协议实现，节点时刻监听网络中广播的数据。当收到邻居节点发来的新区块交易时，根据预先设定的背书策略，验证这些交易和区块是否有效，验证内容包括数字签名、工作量证明等。只有通过验证的交易，才会被加入正在构建的区块。通过验证的区块，才会接入区块链并转发。其中，加入正在构建的区块，还涉及排序节点（Orderer）。

共识层使用先进的 PBFT（Practical Byzantine Fault Tolerance）算法解决分布式一致性问题，在 $n \geqslant 3f+1$ 的条件下解决拜占庭将军问题（n 是网络中总节点数，f 是最多可容忍的恶意节点个数），并在异步通信中使拜占庭协议的复杂度从指数级别降低到多项式级别。在税收智能合约设计中，共识流程如下：在网络中设定一个主节点 orderer（排序节点，如图 3 所示），负责把网络中收集的交易排序后生成列表，并向其他节点扩散；其他各节点收到交易列表后，根据列表中交易排序模拟交易；模拟完排序的交易后，生成交易结果的哈希摘要，并向其他节点广播。每个节点如果收到 $2f$ 个节点的摘要与自己相同，就向全网广播 commit 消息；如果某节点收到 $2f+1$ 条 commit 消息，就可以生成新区块，并把交易提交到区块链和状态数据库。该共识流程设计的拓扑结构，如图 3 所示。值得一提的是，org1 和 org2 的锚节点设置（本案例设置图中 0 和 2 号节点分别代表 org1 和 org2 的锚节点）体现了联盟链特征，可防止单点失败。Org1 的 peer 节点与 Org2 的 peer 节点加入通道，通过对方组织的锚节点来发现对方所有加入通道的节点。

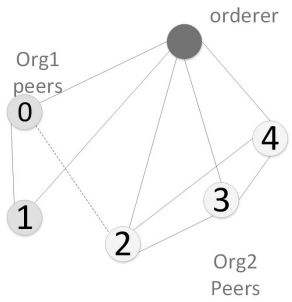

图 3 智能合约网络的节点拓扑结构

数据层在数据结构方面如图 4 所示，设计上基于文档时间戳证明各类文档的创建时间，并提供数字公证服务。时间戳服务器对文档、时间和指向之前文档的 hash 指针签名，后续文档重复该操作，形成基于时间的证书链，无法篡改。每个区块都包含类似图 4 中所示的区块头和区块体。块头存放 Merkle 根、时间戳和前块哈希等。块体存放批量交易数据。一个块内所有交易不断两两哈希生成最终的 Merkle 根，实现了交易的不可篡改性和支付的可验证性。每个区块通过前一区块的哈希把所有块连接在一起。块头包含难度值、nonce 等数据。在数据模型方面，Hyperledger Fabric 采用了基于账户的模型，可根据账户快速查询当前状态，支持功能丰富的通用应用。在存储方面，按照日志文件格式存储，通过哈希键值检索数据（通过交易哈希检索交易数据，通过区块哈希检索区块数据），索引和状态数据一般存储在 Key-Value 数据库。Hyperledger Fabric 目前支持 LevelDB 和 CouchDB 数据库存储和索引。

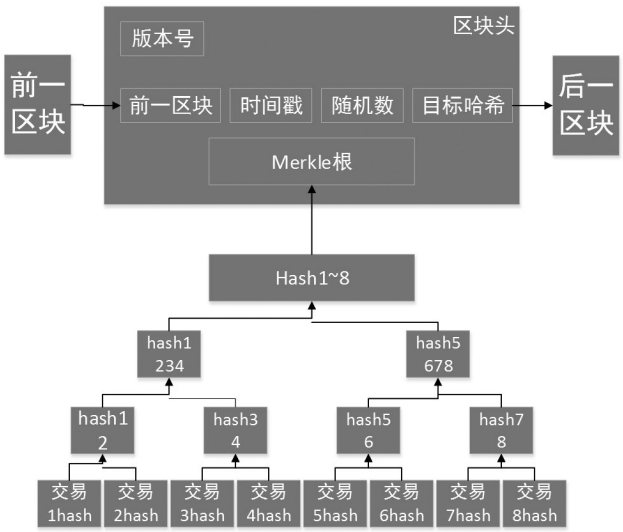

图 4 区块链数据结构

智能合约层扩展了区块链功能，用算法程序实现

预设场景，编制合同条款。Fabric 平台中的智能合约称为 Chaincode（链码），选用 Docker 作为沙盒，运行机制如图 5 所示。按预设场景编写完合约代码后，将其安装并实例化到区块链的网络节点上。部署后的合约被打包为 Docker 镜像，各节点基于该镜像启动一个新 Docker 容器并执行合约中的实例化方法，然后等待调用。应用层通过调用智能合约实现各种交易场景。如果是转账等数据修改类调用，要在网络中的所有节点达成共识，然后修改会被记在区块链，结果会存储在状态数据库。在本用例中，转账的金额和税收的金额被记入区块链，而账户余额包括税收账户存储在状态数据库中。如果是查询，则不需节点共识。

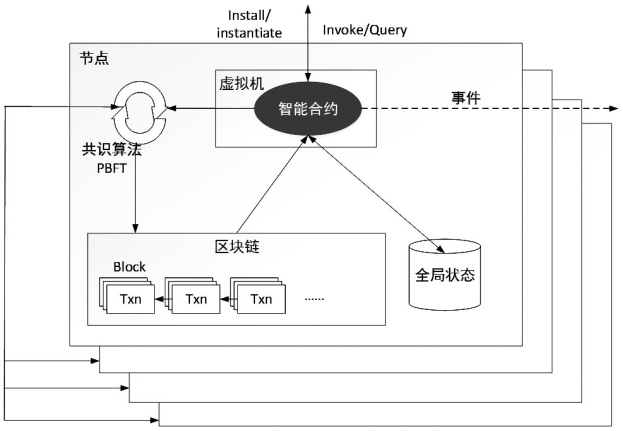

图 5　智能合约的运行机制

4　税收智能合约的设计

市场交易中，常常出现偷税漏税的情况。比如，饭店老板常常以赠送小礼品的方式拒绝开发票，大型企业在交易时使用银行转账存在虚假报账偷税漏税的情况。小数额交易太过琐碎，税务局追查不便；大数额交易即使追查，也会耗费巨大人力物力。在支付宝、微信等电子支付手段越来越普遍的情况下，提出以智能合约的方式实现对偷税漏税的零容忍。

在税收区块链中，针对交易数额进行设计。图 6 是交易中的税收过程。当区块链收到代码的触发交易时，从区块链读取代码，基于交易数额、合约状态与外界信息执行代码，并通过 Fabric 区块链平台提供的智能合约接口查询区块链信息，把结果返回链上。应用中涉及的接口如表 2 所示。

在交易场景中，对每笔转账设置几种交易金额的税收比例，以合约方式自动完成交易场景中的税收过程。Hyperledger Fabric 平台的智能合约也称 chaincode（链码），是应用层与区块链交互的渠道和交易的来源。编写合约的实质是实现链码接口的 Init、Invoke 和 $Quer_y$ 函数，分别对应部署合约、执行交易和查询状态。

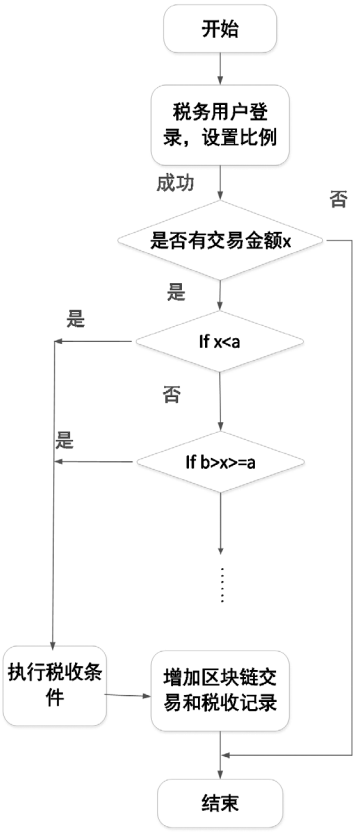

图 6　交易的税收过程

表 2　税收合约设计中使用的接口

接口名	输入	输出
GetFunctionAnd Parameters	Null	函数和数组变量
PutState	key	布尔值
GetState	key	state，布尔值
DelState	key	布尔值
Start	结构体函数	布尔值
GetTxID	Null	交易 id

Go 程序在运行前会对所有对象加载，生成相应的节点和组织的证书和每个节点的公私钥。该智能合约主要实现以下场景：初始化 3 个角色 A、B、C，A 客户向 B 商家付款，总金额的税率比例自动转入税务局的 C 账户；可以查询 A、B、C 三个账户的金额，确认交易成功，也可以删除账户。可以根据法律、税率，不同交易金额设定分段函数执行每笔交易的税收功能，如：

$$\begin{cases} y = x \times x_1, & x < a \\ y = ax_1 + (x-a)x_2, & a \le x < b \\ y = ax_1 + (b-a)x_2 + (x-b)x_3, & b \le x < c \\ \cdots\cdots \end{cases} \quad (1)$$

其中 x_1、x_2……代表税率设置。对每次的交易判断其交易金额所处的区间，然后自动执行税收功能。表 3 为税收合约设计中的主要函数。

表 3 税收合约设计中主要函数

主要函数	功能
init	初始化账户
invoke	实现交易转账
query	查询 A、B、C…账户余额
delete	删除账户

合约代码的可信是通过把代码部署在区块链上实现的。交易触发时，网络中的节点会读取链上的代码并执行，触发机制如图 7 所示，通过 Go 实现说明。在 Go 语言运行前，会将代码安装在区块链节点上并实例化。该过程也可理解为对象的加载，这样在运行中可以知道所有属性和方法。通过 shell 可执行文件或手动输入，可以使合约动态获取对象信息和调用对象的方法。智能合约被实例化后会被编译为字节码，等待触发执行。当触发时，通过 Fabric 区块链平台提供的接口与合约信息交互。

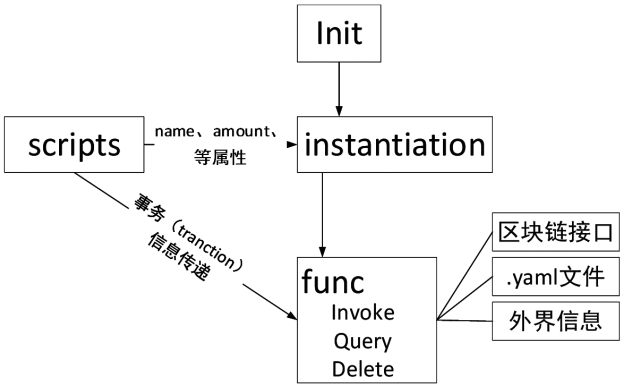

图 7 基于 Go 的智能合约执行模型及软件模块

税收智能合约的全生命周期如图 8 所示，主要包括：创建点对点交易的沙盒环境、在节点对智能合约的信息进行注册登记和节点传递参数进行场景初始化。此三者为合约部署流程。链码部署成功后，会创建连接到部署它的 VP 节点的 gRPC 通道，以接受后续 Invoke 或 $Quer_y$ 指令。节点传递 transaction（交易）或 $quer_y$ 信息给接口，以调用相关函数执行税收场景，其中图 8 中的步骤 4 ~ 步骤 7 都属于代码执行阶段。智能合约类比社会中的法律法规，维护网络环境中事先约定的秩序，可以根据不同场景和约定调整。

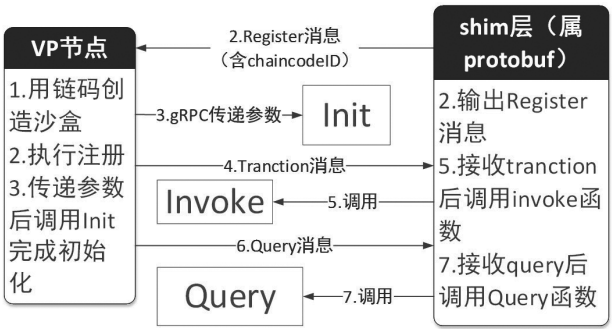

图 8 税收智能合约的全生命周期活动

5 优势及劣势分析

5.1 优势分析

设计基于区块链的税收应用场景，借助了区块链作为分布式数据库的优势。

5.1.1 可信信息共享

数据库通过去中心化的架构，由多方共同管理维护，任何一方不能完全操作数据。区块链依靠每个区块头的哈希指针和块体的 Merkle 树实现不可篡改，而共识机制则加固了数据的安全性，促进了公平公正。

5.1.2 可追溯还原

区块链上存储了运行以来的所有数据，对交易纠纷、税收公正都有益处，方便追溯还原所有历史，提高监管、审计和执法的效率。

5.1.3 高可用可信

税收合约自动执行所有电子交易的税收功能。为相关的报账、审计工作提供方便，网络中无需参与人相互信任，审计监察也无需担心偷税漏税事件。每个事务都要发送者签名，且全网共识后才记录在区块中。多节点备份降低了原有数据库的建设和维护成本。

5.2 劣势分析及发展方向

本次借助 Hyperledger Fabric 平台设计税收应用场景，本身也存在很多不足需要克服。

5.2.1 交易量低

当前，主流区块链架构吞吐量不大，主要原因是它们采用的共识算法系统开销巨大。不管是比特币区块链采用的 PoW，还是 Hyperledger Fabric 采

用的 PBFT 算法，都是以牺牲电力或带宽性能确保安全的。例如，比特币吞吐量为 7 TPS，以太坊为 20 ~ 30 TPS，Hyperledger Fabric 平台虽有进步，但也不过 2 000 TPS。基于智能合约的税务场景虽然设计可行，但工程角度远不能满足当前巨大的电子交易量。

从以上数据可以看到，各区块链平台系统性能的关键瓶颈受制于共识算法。典型代表 Fabric 平台采用的 PBFT 共识算法虽然理论证明完备，但大规模应用面临广播带来的网络开销过大问题。在以后的发展中，包括选取部分主节点共识的算法、基于特定场景减少网络广播的共识算法、融合 PoW 与 PBFT 共识算法的优势等都会受到关注。

5.2.2　事务处理效率

税收合约设计中，Hyperledger Fabric 依靠底层的 LevelDB 提供事务处理。LevelDB 属不支持严格事务的 Key-Value 数据库，一旦单点合约执行失败，需要依靠共识算法同步恢复一致。

除了上面分析的共识算法发展可能部分提高效率外，在区块链平台性能改善和数据积累过程中，平台的数据将趋向结构化数据，如 SQL on Blockchain，使现有分析查询工具无缝接入。

5.2.3　扩展性差

中心化数据库通过增加节点数提高吞吐量。目前，以 Fabric 为代表的联盟链平台随节点增加整体性在下降。因此，扩展性方案还需改进。

除了提到的选取部分主节点共识提高效率外，它的吞吐性能改善也可能借助比特币社区提出的隔离见证、增大区块、闪电网络和雷电网络等扩大区块容量，提高吞吐量。

从智能合约税收应用场景的优劣势分析可见，当前平台本身的限制仍然是限制区块链进入社会、转化为工程应用的瓶颈。从 20 世纪 90 年代 Nick Szabo 提出智能合约概念但缺乏可编程技术手段和数字系统的环境相比，当前智能合约的设计环境已然大为改观，距离进入工程应用并不久远。

6　结论与展望

本文提出了基于 Hyperledger Fabric 平台的带有税收交易场景的智能合约设计方案，包括智能合约的执行模型、状态模型与合约模板，并给出了 Go 语言的具体实例解释。可以看到，Fabric 为智能合约提供了一个实现平台，为合约提供了存储代码、状态的地方，把执行合约的环境与一致性哈希算法融合，构成了基于 Fabric 平台的智能合约系统。智能合约随着区块链的发展正处于初级阶段，并且依然存在安全、法律和智能等方面的问题。后续将会对智能合约进一步研究，引入零知识证明、秘密共享等密码技术，研究智能合约隐私保护功能，包括智能合约的一致性安全性研究和可用于合约建立的工程方法与技术等 [11-12]。

智能合约存储在区块链节点以分布式形式执行，类似社会中的合同、监管等方面的法律法规。从税收智能合约的设计可以看出，在应用场景上，技术与法律有了交叉是一个必然趋势。分析对比智能合约与法律的关系，如表 4 所示。在以后越来越多的应用中，智能合约要与法律一致，必须经过专业法律知识的人士的指定审核，遵守法律并具有法律效应，学科交叉将会越来越多 [13-15]。

表 4　传统合约与智能合约的比较

合约种类	建立基础	安全性	预测性	成熟度	特点
传统合约	合同法	法律束缚	灵活，涉及人的主观判断	历经考验	人工、本地、不确定
智能合约	比特位上的软件	密码学电脑科学	固定、僵硬	经验少	自动化、全球、无情

除法律外，智能合约在公共服务 [16-18] 方面还有以下要求：具有合约利益的人才可接触相应合约信息，合约相关的知识、控制和执行都需要保护，如合约执行方的过程可验证、对违反合约的制裁、合约方可以从用户界面观察合约执行的所有状态和过程等。

本文是对区块链智能合约在交易领域及公共服务方面的一次有益探索，虽然智能合约还有很多问题要面对，但它一定会成为信息互联网向价值互联网转变的关键技术之一，会有广泛的应用前景。

参考文献：

[1] Nakamoto S.Bitcoin:A Peer-to-Peer Electronic Cash System[EB/OL].[2018-05-23].https://bitcoin.org/bitcoin.pdf.

[2] Szabo N.A Formal Language for Analyzing Contracts [EB/OL].(2017-01-12)[2018-01-05].http://nakamotoinstitute.org/contract-language/.

[3] Antonopoulos A M.Mastering Bitcoin:Unlocking Digital Cryptocurrencies[M].USA:O´Reilly Media Inc.,2014.

[4] Hyperledger.Project Charter[EB/OL].[2018-05-23]. https: //www. hyperledger.org/about/charter.

[5] Ethereum White Paper.A Next-generation Smart Contract and Decentralized Application Platform[EB/OL].(2015-11-12)[2018-01-09].https://github.com/ethereum/wiki/wiki/White Paper.

[6] Ian A.Ethereum's Vitalik Buterin Explains How State Channels Solve Privacy and Scalability[EB/OL].(2017-11-15)[2018-01-05].http://www.ibtimes.co.uk/ethereums-vitalik-buterin-explains-how-state-channels-address-privacy-scalability-1566068.

[7] 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(04):481-494.
YUAN Yong,WANG Fei-yue.Blockchain:the State of the Art and Future Trends[J].Acta Automatica Sinica,2016,42(04):481-494.

[8] Vitalik B.A Next-generation Smart Contract and Decentralized Application Platform[EB/OL].(2017-01-17)[2018-01-05].https://github.com/ethereum/wiki/wiki/%5BEnglish%5D-White-Paper.

[9] 中国区块链技术和产业发展论坛 . 中国区块链技术和应用发展白皮书 (2016)[EB/OL].(2016-10-18)[2018-01-05].http://www.cbdforum.cn/index/article/rsr-6.html.
China Blockchain Technology and Industry Development BBS.China Blockchain Technology and Application Development White Paper[EB/OL].(2016-10-18)[2018-01-05].http://www.cbdforum.cn/index/article/rsr-6.html.

[10] Szabo N.Smart Contracts[R].1994.

[11] David B.Blockchain Revolution[EB/OL].(2016-10-21)[2018-01-05].http://www.slideshare.net/15Mb/blockchain-revolution.

[12] Swan M.Blockchain Thinking:the Brain as a Decentralized Autonomous Corporation[J].IEEE Technology and Society Magazine,2015,34(04):41-52.

[13] Christopher A.The Path to Self-Sovereign Identity[EB/OL].(2017-11-15)[2018-01-04].http://www.coindesk.com/path-self-sovereign-identity.

[14] Scalability B W.Scalability[EB/OL].(2017-11-15)[2018-01-05].https://en.bitcoin.it/wiki/Scalability.

[15] uPort.The Wallet is the New Browser-Medium[EB/OL].(2017-11-15)[2018-01-05].https://medium.Com/@ConsenSys/uport-the-wallet-is-the-new-browser-b133a83fe73%7B%5C#%7D.1l0vsfq2p.

[16] Cheema G S,Popovski V.Building Trust in Government[EB/OL].(2017-11-15)[2018-01-05].https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?Htmlfid=GBE03801USEN&.

[17] Djuri B.Towards Self-Sovereign Identity using Blockchain Technology[EB/OL].(2017-11-15)[2018-01-05].http://essay.utwente.nl/71274/1/Baars_MA_BMS.pdf.

[18] 何蒲 , 于戈 , 张岩峰等 . 区块链技术与应用前瞻综述 [J]. 计算机科学 ,2017,44(04):1-7.
HE Pu,YU Ge,ZHANG Yanfeng,et al.Block Chain Technology and Application Forward Review[J]. Computer science,2017,44(04):1-7.

作者简介：

陈宇翔（1993—），男，硕士，主要研究方向为身份管理、区块链；

张兆雷（1985—），男，硕士，高级工程师，主要研究方向为信息安全；

刘地军（1984—），男，硕士，主要研究方向为通信与信息系统；

彭　笛（1981—），男，硕士，主要研究方向为通信与信息系统；

李　枫（1993—），男，硕士，主要研究方向为密码学。