计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2018.20170658
55(11):2452-2466,2018

基于区块链的智能合约技术与应用综述

贺海武[1] 延 安[2] 陈泽华[3]
[1](中国科学院计算机网络信息中心 北京 100190)
[2](太原理工大学信息工程学院 太原 030024)
[3](太原理工大学大数据学院 太原 030024)
(hehaiwu@gmail.com)

Survey of Smart Contract Technology and Application Based on Blockchain

He Haiwu[1], Yan An[2], and Chen Zehua[3]
[1](*Computer Network Information Center, Chinese Academy of Sciences, Beijing* 100190)
[2](*College of Information Engineering, Taiyuan University of Technology, Taiyuan* 030024)
[3](*College of Data Science, Taiyuan University of Technology, Taiyuan* 030024)

Abstract With the flourishing development of blockchain technology represented by bitcoin, the blockchain technology has moved from the era of programmable currency into the era of smart contract. The smart contract is an event-driven, state-based code contract and algorithm contract, which has been widely concerned and studied with the deep development of blockchain technology. The protocol and user interface are applied to complete all steps of the smart contract process. Smart contract enables users to implement personalized logic on the blockchain. The blockchain-based smart contract technology has the characteristics of de-centralization, autonomy, observability, verifiability and information sharing. It can also be effectively applied to build programmable finance and programmable society, which has been widely used in digital payment, financial asset disposal, multi-signature contract, cloud computing, Internet of things, sharing economy and other fields. The survey describes the basic concepts of smart contract technology, its whole life cycle, basic classification and structure, key technology, the art of the state, as well as its application scenarios and the main technology platforms. Its problems encountered at present are also discussed. Finally, based on the theoretical knowledge of the smart contract, we set up the Ethereum experimental environment and develop a system of crowdsale contract. The survey is aimed at providing helpful guidance and reference for future research of smart contract based on blockchain technology.

Key words smart contract; blockchain; ethereum; distributed application; formal method; crowdsale contract

摘 要 随着以比特币为代表的区块链技术的蓬勃发展,区块链技术已经开始逐步超越可编程货币时代而进入智能合约时代.智能合约(smart contract)是一种由事件驱动的、具有状态的代码合约和算法合同,随着区块链技术的深入发展而受到广泛关注和研究.智能合约利用协议和用户接口完成合约过程的所有步骤,允许用户在区块链上实现个性化的代码逻辑.基于区块链的智能合约技术具有去中心化、

收稿日期:2017-09-12;**修回日期**:2018-03-05
基金项目:中国科学院百人计划项目(1101002001);国家自然科学基金项目(61402319)
This work was supported by the One Hundred Person Project of the Chinese Academy of Sciences (1101002001) and the National Natural Science Foundation of China (61402319).
通信作者:陈泽华(zehuachen@163.com)

自治化、可观察、可验证、可信息共享等特点,可以有效构建可编程金融和可编程社会,广泛应用于数字支付、金融资产处置、多重签名合约、云计算、物联网、共享经济等多个领域.首先阐述了智能合约技术的基本概念、全生命周期、基本分类、基本架构、关键技术、发展现状以及智能合约的主要技术平台;然后探讨了智能合约技术的应用场景以及发展中所存在的问题;最后,基于智能合约的理论知识,搭建了以太坊实验环境并开发了一个众筹智能合约系统,旨在为基于区块链的智能合约技术的研究与发展提供参考与借鉴.

关键词 智能合约;区块链;以太坊;分布式应用;形式化方法;众筹合约

中图法分类号 TP391

区块链是比特币的基础支撑技术,随着近年来比特币的快速发展与普及,引起了多方的广泛关注.2016 年 1 月,英国政府发布区块链专题研究报告[1];同年 12 月,中国政府将区块链技术列入《"十三五"国家信息化规划》[2],旨在加强新技术的基础研发和前沿布局.区块链被认为是继大型机、个人电脑、互联网、移动社交网络之后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、央行纸币信用后的第 4 个里程碑[3].

区块链技术的应用发展有 3 个阶段:1)区块链 1.0,即可编程货币,如比特币;2)区块链 2.0,即可编程金融[3],其中智能合约是其代表性应用;3)区块链 3.0,即可编程社会,如去中心化应用(decentralized application)、去中心化自组织(decentralized autonomous organization)[4-5].目前,区块链已经开始超越区块链 1.0 时代,进入到区块链 1.5 时代,并向可编程金融,即智能合约时代过渡.

2016 年以来,以以太坊(Ethereum)[6]为代表的智能合约技术成为各界关注的热点,引起了政府部门、金融机构、科技企业的广泛关注.2016 年 12 月,首届智能合约专题研讨会在微软纽约市总部举行,分析与探讨了智能合约的应用场景.2017 年 2 月,欧洲议会在《区块链如何改变我们的生活》[7]报告中指出,智能合约技术是最具潜力的区块链应用;同月,企业以太坊联盟(Enterprise Ethereum Alliance)成立,致力于将以太坊开发成企业级区块链,其成员既有摩根大通、荷兰银行等大型金融机构,也有微软、Intel 等科技企业.

"智能合约"(smart contract)的概念产生于 1995 年,由密码学家 Szabo[8]首次提出,他指出"智能合约通过使用协议和用户接口来促进合约的执行".从本质上讲,智能合约是由事件驱动的、具备状态的、部署于可共享的分布式数据库上的计算机程序,现存智能合约的工作原理类似于其他计算机程序的 If-Then 语句[9].智能合约只是以这种方式与真实世界的资产进行交互.当一个预先设定的条件被触发时,智能合约执行相应的合同条款.

Szabo 指出,计算机在某一天可以代替人力、机械设备等进行更加复杂的数字资产交易.未来的某一天,这些自动执行的程序可能取代某些处理特定金融交易的专家或机构.智能合约的发展虽然处于初级阶段,但其潜能显而易见,它将合约参与者、合约协议以及参与者与协议之间的复杂关系程序化了.目前基于区块链的智能合约技术的发展呈现出技术产业创新驱动的态势,但在学术方面的研究相对滞后,截至 2017 年 7 月,以万方知识服务平台为中文数据源,以 EI Village 为英文数据源的检索显示,目前标题包含"智能合约/smart contract"且与区块链技术相关的学术论文仅有中文 9 篇和英文 27 篇.

本文的内容有 6 方面:1)简要介绍了智能合约的底层技术基础——区块链技术,概述了智能合约的定义、全生命周期、优点及分类;2)凝练总结了智能合约的基本架构、关键技术;3)简要介绍了智能合约的主要技术平台;4)阐述了智能合约的发展现状与应用场景;5)概要总结了智能合约现存的问题并搭建了以太坊实验环境,开发了一个众筹智能合约系统;6)总结与展望.

1 智能合约背景知识

1.1 区块链技术简介

区块链技术起源于 2008 年,由一位化名为"中本聪"(Satoshi Nakamoto)的学者提出,其在文献中所描述的区块链是一种按照时间顺序将数据区块以链条的方式组合成特定数据结构,并以密码学方式保证的不篡改和不可伪造的去中心化共享总账(decentralized shared ledger)[10].比特币是最早的区块链应用场景,其本质是由基于区块链技术的分布式

网络利用密码学算法生成的数字加密货币. 数字加密货币领域一直面临着两大难题:双重支付问题和拜占庭将军问题(Byzantine generals problem)[11]. 而区块链技术的出现,为解决这两大难题提供了有效的途径. 双重支付问题是指用"同一笔钱"在两次或多次交易中完成支付. 拜占庭将军问题是指在缺少可信任中心节点的情况下,分布式系统如何达成共识和建立互信的问题[12]. 区块链技术,在不需要第三方信用机构的前提下,通过分布式数据库、数字加密技术和独特的共识算法解决了去中心化系统的双重支付问题,实现了一个无需信任单个节点的去中心化的可信任系统. 区块链的共识算法的理论基础是拜占庭容错(Byzantine fault tolerant, BFT). 常见的共识算法有工作量证明(proof of work, PoW)[13]、权益证明(proof of stake, PoS)[14]、授权权益证明(delegated proof of stake, DPoS)[15]、实用拜占庭容错(practical Byzantine fault tolerance, PBFT)[16]、授权拜占庭容错(delegated Byzantine fault tolerance, DBFT)[17]等.

狭义的区块链是去中心化系统中各节点共享的数据账本,区块结构如图 1 所示,每个区块分为区块头和区块体两部分,涉及链式结构、Hash 算法、Merkle 树和时间戳等技术要素[18].

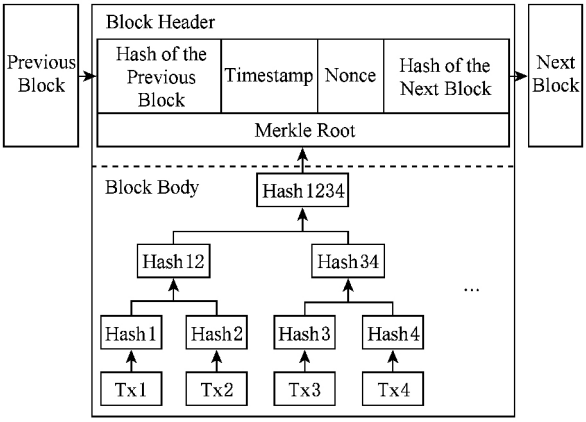

Fig. 1 Structure of block

图 1 区块结构

区块链技术的本质是它在网络空间建立了一种分布式的一致性标准,对所有的数字事件在分布式数据库上创建确切的无法篡改的记录并且使得区块链中的所有参与方都能确切、可信地了解所发生的数字事件.

区块链技术出现后,其去中心化、去信任、规则透明、集体维护、不可篡改等特性,恰好为智能合约提供了安全可靠的记录载体和执行环境. 首先,区块链技术采用纯数学的方法,在不牺牲隐私性,也无需第三方信用机构参与的条件下,可以对所有过去、当前的数字事件(如行为、资产等),建立分布式的一致性表达[19]. 其次,区块链提供了可供用户进行编程的脚本系统,进一步增强了区块链应用的灵活性,如在以太坊中,具备图灵完备、功能强大的脚本系统,使得基于智能合约的更为高级的分布式应用得以实现.

1.2 智能合约概述

智能合约有许多非形式化的定义,Szabo[8]创造性地提出,"智能合约就是执行合约条款的可计算交易协议";以太坊的智能合约是基于区块链的数字资产控制程序[6]. 狭义来讲,智能合约是涉及相关商业逻辑和算法的程序代码,把人、法律协议和网络之间的复杂关系程序化了. 广义来讲,智能合约是一种计算机协议,一旦部署就能实现自我执行和自我验证,已经不仅仅局限于金融领域,并且在分布式计算、物联网等领域都有广阔的应用前景.

类似于传统合约,智能合约全生命周期包括:合约生成、合约发布、合约执行 3 个部分,如图 2 所示:

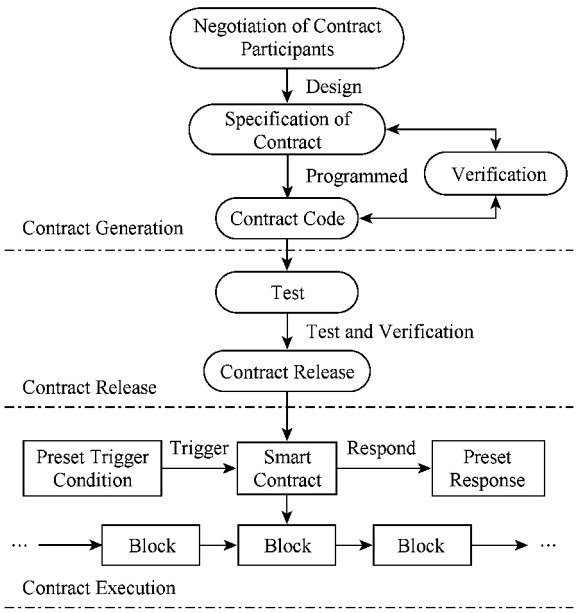

Fig. 2 The whole life cycle of smart contract

图 2 智能合约全生命周期图

合约生成,主要包含合约多方协商、制定合约规范、进行合约验证、获得合约代码 4 个环节. 具体实现过程为:由合约参与方进行协商,明确各方的权利与义务,确定标准合约文本并将文本程序化,经验证后获得标准合约代码. 其中涉及 2 个重要环节:合约规范和合约验证. 合约规范需要由具备相关领域专业知识的专家和合约方进行协商制定. 合约验证在基于系统抽象模型的虚拟机上进行,它是关乎到

合约执行过程安全性的重要环节,必须保证合约代码和合约文本的一致性.

合约发布与交易发布类似,经签名后的合约通过 P2P 的方式分发至每一个节点,每个节点会将收到的合约暂存在内存中并等待进行共识. 共识过程的实现:每个节点会将最近一段时间内暂存的合约打包成一个合约集合,并计算出该集合的 Hash 值,最后将这个合约集合的 Hash 值组装成一个区块并扩散至全网的其他节点;收到该区块的节点会将其中保存的 Hash 值与自己保存的合约集合的 Hash 值进行比较验证;通过多轮的发送与比较,所有节点最终会对新发布的合约达成共识,并且达成共识的合约集合以区块的形式扩散至全网各节点,如图 3 所示. 其中每个区块包含以下信息:当前区块的 Hash 值、前一区块的 Hash 值、时间戳、合约数据以及其他描述信息.

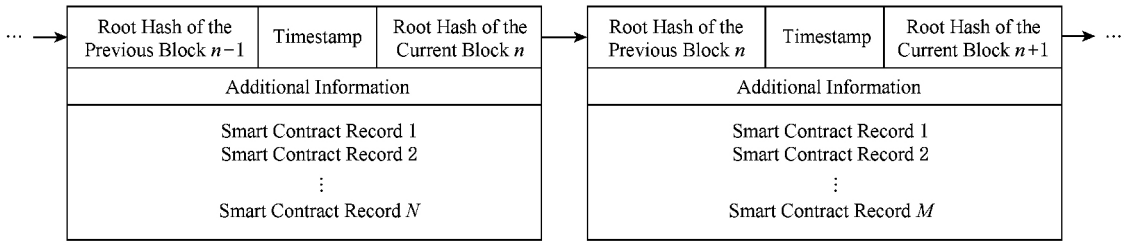

Fig. 3 Blockchain diagram of smart contract

图 3 智能合约的区块链示意图

智能合约的执行是基于“事件触发”机制的. 基于区块链的智能合约都包含事务处理和保存机制以及一个完备的状态机,用于接受和处理各种智能合约. 智能合约会定期遍历每个合约的状态机和触发条件,将满足触发条件的合约推送至待验证队列. 待验证的合约会扩散至每一个节点,与普通区块链交易一样,节点会首先进行签名验证,以确保合约的有效性,验证通过的合约经过共识后便会成功执行. 整个合约的处理过程都由区块链底层内置的智能合约系统自动完成,公开透明,不可篡改.

智能合约的实现,本质上是通过赋予对象(如资产、市场、系统、行为等)数字特性,即将对象程序化并部署在区块链上,成为全网共享的资源,再通过外部事件触发合约的自动生成与执行,进而改变区块链网络中数字对象的状态(如分配、转移)和数值. 智能合约可以实现主动或被动的接受、存储、执行和发送数据,以及调用智能合约,以此实现控制和管理链上数字对象. 目前已经出现的智能合约技术平台,如以太坊、Hyperledger 等,具备图灵完备的开发脚本语言,使得区块链能够支持更多的金融和社会系统的智能合约应用.

现今,虽然智能合约尚未得到广泛应用,但其技术优点已经得到研究人员的广泛认可. 总体来说,智能合约具有 7 个优点:

1) 确定性. 智能合约在不同的计算机或者在同一台计算机上的不同时刻多次运行,对于相同的输入能够保证产生相同的输出. 对于区块链上的智能合约,确定性是必然要求,因为非确定性的合约可能会破坏系统的一致性.

2) 一致性. 智能合约应与现行合约文本一致,必须经过具备专业知识的人士制定审核,不与现行法律冲突,具有法律效应[20].

3) 可终止性. 智能合约能在有限的时间内运行结束. 区块链上的智能合约保证可终止性的途径有非图灵完备(如比特币)、计价器(如以太坊)、计时器(如 Hyperledger Fabric)等.

4) 可观察和可验证性. 智能合约通过区块链技术的数字签名和时间戳,保证合约的不可篡改性和可溯源性. 合约方都能通过一定的交互方式来观察合约本身及其所有状态、执行记录等,并且执行过程是可验证的.

5) 去中心化. 智能合约的所有条款和执行过程都是预先制定好的,一旦部署运行,合约中的任何一方都不能单方面修改合约内容以及干预合约的执行. 同时,合约的监督和仲裁都由计算机根据预先制定的规则来完成,大大降低了人为干预风险.

6) 高效性和实时性. 智能合约无需第三方中心机构的参与,能自动地实时响应客户需求,大大提升了服务效率.

7) 低成本. 智能合约自我执行和自我验证的特征,使其能够大大降低合约执行、裁决和强制执行所产生的人力、物力成本.

就当前发展而言,以区块链技术为基础的智能合约大致分为 3 类:1)“Chaincode”,即常说的链上

代码,如金融活动由交换数据变为交换代码;2)“智能法律合约”,包括不同方面所产生的权利和义务,并且在法律上可执行,通常以复杂的法律文本来表达,不仅涵盖个人行为,还可能涉及时间依赖和次序依赖等一系列依赖关系,例如 PrimaveradeFilippi 加密账本交易法律框架[21],用链上智能合约来补充或代替现有法律合同,成为智能合约代码和传统法律语言的结合;3)“智能应用合约”,即在区块链上部署基于智能合约的分布式链上应用,创建有商业价值的全新合约形式,如 M2M(机器对机器)商业模式.

2 智能合约基本架构与关键技术

智能合约基本架构如图 4 所示. 总体来说,区块链智能合约包含数据层、传输层、智能合约主体、验证层、执行层以及合约之上的应用层这 6 个要素. 数据层包括链上数据和链下数据,它们是智能合约运行的必要数据源. 传输层则封装了用于支持“链上-链上”和“链上-链下”进行通信、数据传输的协议. 智能合约主体包括协议和参数. 验证层主要包含一些验证算法,用于保证合约代码和合约文本的一致性. 执行层主要封装了智能合约运行环境的相关软件. 应用层则是基于前 5 个要素的基础产生的相对高级的各种应用,它主要是为智能合约与其他计算机、应用程序通信服务的. 本节将从智能合约主体、数据加载方式、执行环境、验证方法和扩展性的实现 5 个方面探讨智能合约的关键技术.

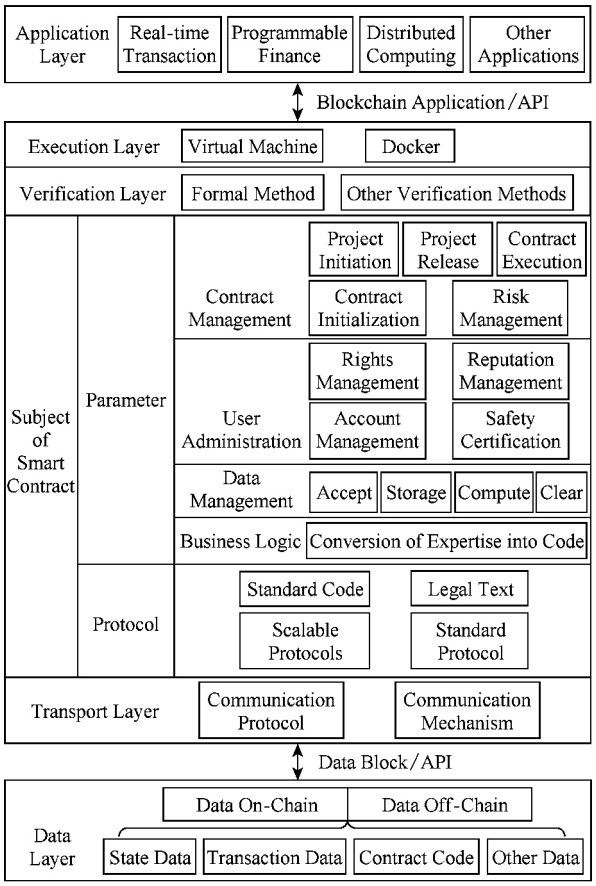

Fig. 4 Basic framework of the smart contract

图 4 智能合约基本架构

2.1 合约主体

智能合约主体为基于标准化的合约应用提供了复杂的协议框架,可以通过识别智能合约的关键参数来识别合约的行为和状态. 智能合约主体主要包括协议和参数 2 个部分:

1) 协议是由标准机构发布的合法文本的程序化描述[22]. 协议包括合法的标准文本和标准参数,其中每个参数都有一个标识,分别代表一种类型. 可以说,协议是一个完全实例化的模板.

2) 参数包括业务逻辑模块(主要参数)和各种附件模块,如数据管理模块、用户管理模块、合约管理模块等. 业务逻辑模块包括定制的合法文本和参数,是对应用领域专业知识的程序化描述,由合约参与方协商产生,涉及多方的权利与义务. 业务逻辑模块的合法文本和参数来自协议部分的标准文本和参数,但根据应用场景而有所不同. 附件模块在业务逻辑的基础上,结合具体应用场景的需要,实现对智能合约的补充和完善. 数据管理模块,封装了实现数据接收、暂存、计算、清除等功能的代码程序;用户管理模块,主要实现了合约用户的权限管理、安全认证、信誉管理等功能;合约管理模块,主要功能是当合约被调用时,结合用户需求,实现合约的生成、验证发布、部署执行、状态查询以及风险处理等功能. 各模块根据应用需求,可以定制子协议和子标准,如计算安全标准、风险预警标准、模块交互协议等. 所有参数都是合约的关键部分,因为它们不仅直接反映了各方之间的业务关系而且影响合约的自动执行.

2.2 数据加载方式

数据层包括状态数据、交易数据、合约代码、应用数据等,出于可观察和可验证的目的,状态数据和交易数据一般都采用链上存储方式. 应用数据和合约代码的加载方式则分为链上和链下 2 种. 目前绝大多数区块链系统均采用链上方式,将代码和应用数据发布到链上,然后再从链上加载数据和代码并执行,其缺点是代码和应用数据将永久地存在于区块链中,不利于更新维护,占用节点存储资源,随着时间的积累将带来巨大的存储负担. 链下方式是指

将智能合约的散列值存储于链上,并通过以散列值为索引的存储网络或可信赖的数据源来保存完整的合约代码,如 IPFS(inter planetart file system)系统、Tower Crier 平台[23]. 散列值是由合约代码内容计算而得,这样既可以保证合约的不可篡改性,又可以节约节点大量的存储空间和加强合约的隐私性.

2.3 执行环境

目前主流的智能合约执行环境的设计主要分为 2 种:虚拟机和容器(docker). 无论是虚拟机还是容器,它们的作用都是在一个沙箱中执行合约代码,并对合约所使用的资源进行隔离和限制. 虚拟机通常是指通过软件模拟的具备完整硬件功能的、能像真实机器一样执行程序的计算机的软件实现,如 VMware. 出于降低资源开销、提升性能和兼容性的目的,绝大多数区块链会采用轻量级的虚拟机结构,如以太坊虚拟机(Ethereum virtual machine, EVM).

容器通常是指借助容器引擎,让开发者可以打包其应用以及依赖包到一个可移植的容器中,也可以实现虚拟化. 容器使用沙箱机制,相互之间不会有任何接口,如 Hyperledger Fabric 使用 Docker 作为智能合约的执行环境. Docker 本身没有采用虚拟化技术,程序是直接运行在底层操作系统上,代码执行的效率很高. 但与轻量级虚拟机相比,其过于庞大的架构,使得部署和启动 Docker 本身需要消耗大量的时间和计算资源.

我们在 1.2 节已经提到,智能合约本质上是区块链上可执行的代码,那么在智能合约的执行过程中,我们需要关注 2 个问题,即指令的执行速度和运行环境的启动速度. 对于智能合约而言,运行环境的启动速度比指令的执行速度更加重要[24]. 这是因为,针对轻量化的虚拟机或容器,智能合约的代码中很少会涉及到 IO 相关的指令,所以这些指令代码易于优化. 而智能合约的每次调用,都必须在一个新的虚拟机或容器中进行,因此运行环境启动时间对整个智能合约系统影响较大.

2.4 验证方法

智能合约是对某领域专业知识的程序语言描述,对合约所涉及的核心利益(如资产)的安全性、合约代码的逻辑正确性有了更高的要求,必须保证合约文本与合约代码的一致性,合约验证是保证这些要求的重要途径. 目前,形式化验证是智能合约领域的主流验证方式. 形式化方法是基于数学的描述和推理计算机系统性质的技术,常用于软件的规范、开发和验证[25]. 形式化方法主要包括形式归约和形式验证. 形式验证是建立在形式归约的基础上,验证已有程序是否满足其归约要求[26]. 目前常见的形式验证方法主要有 2 种:演绎验证和模型检测. 演绎验证是基于定理证明的思想,采用逻辑公式描述系统,优点是可以处理无限状态的问题但做不到完全自动化,如 STeP. 模型检测是基于状态搜索的思想,主要针对有穷状态系统,如 SPIN. 模型检测可以实现完全自动化,并且在验证性质得不到满足时,搜索终止可以给出反例,这种信息往往反映了系统设计中的错误.

智能合约的形式化验证主要包括 4 个部分:代码生成、形式化描述、形式化验证和一致性测试. 代码生成,是指用编程语言对合约文本进行程序化描述. 形式化描述,是指通过建模语言和建模工具对形式化合约文档进行建模[27]. 一致性测试强调被测系统与给定标准的一致性,通过测试的合约代码实现的外部特性与标准合约文本一致[28]. 形式化验证法可以检查智能合约的很多属性,如可达性、公平性、死锁等. 将形式化验证法应用于智能合约,使得合约的生成和执行有了规范性约束,保证了合约的可信性.

2.5 扩展性的实现

可扩展性通常是指如何处理更大规模的业务. 对于一个系统的扩展性,我们通常有 2 种方法,即垂直扩展和水平扩展. 与水平扩展相比,垂直扩展是基于单台设备最大处理能力的串行系统的扩展性,容易较快触及成本、技术的极限,因此水平扩展是当下的主流措施,即将串行系统改造成并行系统,对指令进行并行处理.

区块链本质上是一个分布式数据库,存储着各种数据以及数据间进行交换和计算的规则,而智能合约就是这些规则的代码实现. 因此,实现智能合约的并发执行,将成为提高区块链系统扩展性的重要途径,如以太坊提出的分片(sharding)方案,即架构中的全球验证程序集合中的节点被随机分配到特定的“碎片”,其中每个碎片并行处理全局状态的不同部分,从而确保工作是跨节点分布处理.

3 智能合约主要技术平台

以太坊和 Hyperledger Fabric[29]是目前较为成熟且极具代表性的智能合约技术平台. 本节将以以太坊和 Hyperledger Fabric 为例介绍智能合约技术平台.

3.1 以太坊

以太坊是一个基于区块链数据结构的、可实现

智能合约的、开源的底层系统，在 2013 年由 Buterin[6] 在他的文章“以太坊：下一代加密货币和分散应用平台”中提出. 以太坊的目标是基于脚本语言、数字加密货币和链上元协议(on-chain meta-protocol)概念进行整合和提高，使得开发者能够创建任意的基于共识的、可扩展的、标准化的、特性完备的、易于开发和协调的分布式应用.

以太坊虚拟机(EVM)是在以太坊智能合约及其应用的运行环境，提供了一种图灵完备的脚本语言——Ethereum virtual machine code，这使得任何人都能够创建智能合约及其去中心化应用，并在其中自由定义所有权规则、交易方式和状态转换函数. 以太坊智能合约的核心要素如图 5 所示，主要包括账户、交易、Gas、日志、指令集、消息调用、存储和代码库 8 个部分.

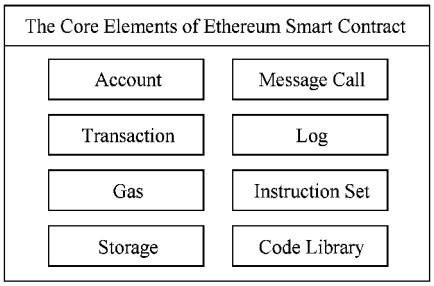

Fig. 5 The core elements of Ethereum smart contract

图 5 以太坊智能合约核心要素

账户是以太坊的核心操作对象，主要分为 2 类：外部账户和合约账户. 外部账户类似于一般区块链电子货币账户，并且外部账户有能力创建并部署智能合约. 合约账户由外部账户创建，其地址由合约创建者的地址和该地址发出过的交易数量计算得到. 合约账户既含有货币余额状态还有合约存储状态. EVM 的指令集被刻意保持在最小规模，以尽可能避免可能导致共识问题的错误出现. 指令集具备常用的算术、位、逻辑和比较操作，以及条件和无条件跳转.

以太坊智能合约旨在实现 4 个目的：1)存储对其他合约或外部实体有意义的值或状态；2)作为具有特殊访问策略的外部账户；3)映射和管理多个用户之间的关系；4)为其他合约提供支持. 基于这 4 个目的，以太坊智能合约有着广泛应用，如储蓄钱包、云计算、版权管理系统[30]、身份和信誉系统、去中心化存储以及去中心化自治社会(decentralized autonomous society)[31]等.

截至 2017 年 5 月，全球已有 200 多个以太坊应用诞生，如分布式众筹平台 Betfunding[32]. 以太坊公有链和开源架构的特性，使得以太坊成为了最流行的智能合约及其分布式应用开发平台之一.

3.2 Hyperledger Fabric

Hyperledger[33]是 Linux 基金会于 2015 年 12 月发起的旨在推动各方协作，共同打造基于区块链的企业级分布式账本底层技术，用于构建支撑业务的行业应用平台. Fabric[29]是 Hyperledger 的一个子项目，目标是实现一个通用的许可链(permissioned chain)的底层基础框架，其采用模块化架构提供可切换和可扩展的组件，包括共识算法、加密算法、数字资产、智能合约等服务. 超级账本的设计原则是“用例驱动”，目前，Fabric 项目主要支持 5 种用例：数字支付[34]、金融资产管存、供应链、主数据管理和共享经济[35].

Fabric 的逻辑架构如图 6 所示，由成员资格服务、策略服务、区块链服务和智能合约服务 4 个部分构成. Fabric 智能合约实质上是在验证节点(verification node)上运行的分布式交易程序，用以自动执行特定的业务规则，最终更新账本的状态. Fabric 智能合约分为公开、保密和访问控制 3 种类型，分别由拥有不同权限的成员发起.

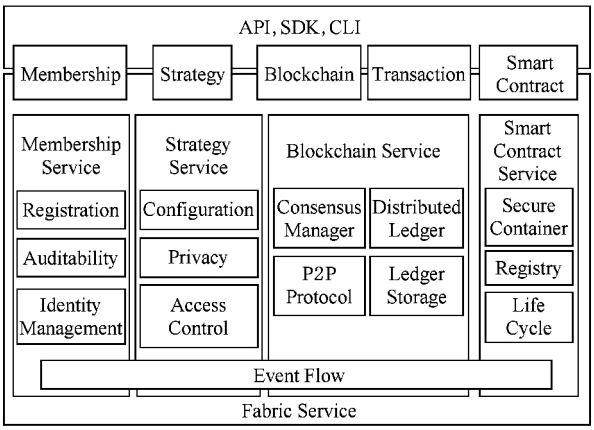

Fig. 6 Fabric project architecture diagram

图 6 Fabric 项目架构图

Fabric 智能合约代码的执行过程图如图 7 所示，分为 6 个步骤：①客户端发送执行请求给任意

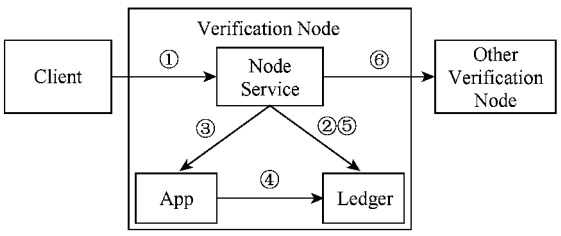

Fig. 7 Diagram of smart contract execution of Fabric

图 7 Fabric 智能合约代码执行示意图

一个验证节点;②验证节点收到请求后,向本地账本发送启动智能合约的指令;③验证节点创建隔离的运行环境,启动合约代码;④合约执行过程中,更新本地账本的状态;⑤合约调用完成后,验证节点向本地账本确认交易;⑥验证节点向其他验证节点广播交易.

与以太坊不同,Fabric 虽然也是开源的,但是 Fabric 主要是为联盟链服务的,其更加强调商业需求和实际应用需要.

4 智能合约研究现状与应用场景

4.1 国外研究现状

从国外发展来看,在科研领域,截至 2017 年 7 月 31 日,以 EI Village 为数据源,标题中包含"blockchain"的外文文献有 152 篇,标题中包含"smart contract"且与区块链技术相关的外文文献有 27 篇,如图 8 所示. 从图 8 可以看出,国外在区块链和智能合约的研究与探索方面起步较早. 以康奈尔大学为代表,Emin Gun Sirer 教授所在的 IC3 研发团队推出了 Solidus 协议、区块链保密查询工具 Town Crier 等理论科研成果,与其合作的企业包括微软、花旗银行和区块链技术公司 Chain 等.

2015 年被称为区块链元年,自这一年以来,区块链掀起了前所未有的热潮,全球金融机构和各大银行争相展开对区块链技术的研究. 据《2015 年度全球数字货币(区块链)创业投资报告》显示,2015 年度共发生全球数字货币及区块链相关投资 65 起,总金额达 4.9 亿美元. 其中,涉及智能合约和商业应用等方面的投资有 21 起,总金额达 5 628 万美元. 其中,智能合约平台 Symboiont 和 Mirror 获得 700 万美元和 1 280 万美元投资. 2016 年,RSK 公司 Rooststock 智能合约平台相继获得 110 万美元投资,同年 5 月 DAO(分布式自治组织)作为主攻智能合约的区块链项目,自众筹金额高达 2.45 亿美元. 更详细的国外主要智能合约项目如表 1 所示:

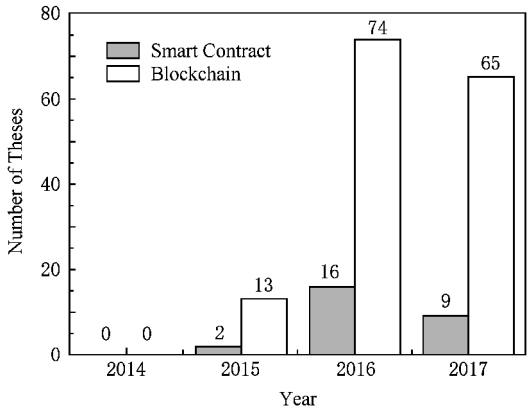

Fig. 8 Search result of EI Village

图 8 EI Village 检索结果

Table 1 Main Foreign Smart Contract Projects

表 1 国外主要智能合约项目

Research Institution	Smart Contract Project	Project Introduction
Ethereum	Ethereum Smart Contract Platform	It is a new generation of smart contracts and a decentralized application platform, and it can be used to program and develop smart contract agreements.
RSK	Rootstock Smart Contract Platform	It is a smart contract distributed platform, and it can be implemented as a side chain, adding value and functionality to the core bitcoin network.
IBM	Hyperledger	It is blockchain digital technology and transaction verification of open source projects, and it can mainly be used in the financial industry.
Slock. it	Slock. it	It is a smart contract system based on the blockchain technology, and it focuses on the research and development of the Ethereum blockchain data and the Internet of things (IoT) solution.
Ripple Laboratory	Coduis	It is a smart protocol released by Ripple Laboratory, and it is applied to the Ripple platform to achieve the guidance of currency circulation.
ConsenSys	BTC Relay	It is the Ethereum where users can be allowed to verify bitcoin transactions by using Ethereum's smart contract feature.

4.2 国内研究现状

从国内发展来看,区块链技术起步较晚. 在科研领域,截至 2017 年 7 月 31 日,以万方数据库为数据源,标题中包含"区块链"的中文文献有 352 篇,标题中包含"智能合约"且与区块链技术相关的中文文献有 9 篇,如图 9 所示. 从图 9 可以看出,自 2016 年以来,区块链技术在国内呈现出爆发式的发展态势,但智能合约技术尚处在发展初期. 蔡维德教授所在的北航区块链实验室致力于区块链、智能合约、数字社会等技术的研究与探索,理论成果涉及共识算法、应用问题、架构问题等多个方面,如双链并发模型[36]、并行拜占庭协议、"金丝猴"分布式链网模型等;马小峰

教授所在的同济金融科技研究院推出了“中国银联-同济区块链测评标准体系”,“基于区块链的信用校园”等多项区块链技术与应用;中国科学院软件所发布了区块链基础组件 RepChain(reactive permission chain),是一种采用响应式编程实现的自主可控的许可链,以推动区块链与行业应用结合为目标,具有标准化、模块化、可视化的特点. 除此之外,2015 年国内各地开始纷纷成立研究联盟,共同推动区块链技术的发展. 2015 年 10 月首届全球区块链峰会“区块链——新经济蓝图”在上海举办. 2015 年 12 月,中国区块链应用研究中心(北京)成立. 2016 年 1 月,中国区块链研究联盟在北京成立,致力于推动区块链的相关学术研讨和实践工作.

国内基于区块链的智能合约应用探索还处于早期研究阶段. 据《全球区块链+创投报告》,截至 2017 年 4 月底,全球总共 455 家区块链公司累计获得融资额为 19.47 亿美元,其中中国共有 61 家位列全球第 2. 2016 年 5 月腾讯联合发起金融区块链合作联盟,推出腾讯云的联盟链云服务;同年6月百度

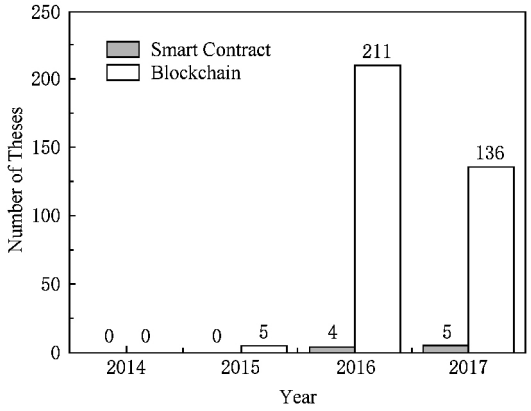

Fig. 9 Search result of Wanfang Data

图 9 万方数据库检索结果

战略投资 Circle;同年 7 月,阿里系的蚂蚁金服在全球 XIN 公益大会上表示其区块链技术即将上线,旨在打造基于区块链的公益平台;同年 9 月,中国银联与 IBM 合作,预演“使用区块链技术的跨行积分兑换系统”;2017 年 5 月,小蚁区块链发布智能合约 2.0,从数字资产平台全面升级为智能经济平台. 更详细的国内主要智能合约项目如表 2 所示:

Table 2 Main Domestic Smart Contract Projects

表 2 国内主要智能合约项目

Research Institution	Project Introduction	Research Institution	Project Introduction
Ping'an Insurance	Participation in the Ethereum smart contract platform	Alibaba Group	The public financial clouds based on the blockchain
Letv Finance	Working with Stellar to pay cross-border	WeBank	Cloud services based on tencent union chain
Ant Financial	The public service platform based on blockchain	Wanda Group	Hercules project launched in cooperation with others
BOC Hong Kong	Rootstock smart contract platform	JD Finance	Digital bill and ABS cloud
CZBank	Participation in the Hyperledger	WanXiang Group	Participation in the Coduis

4.3 智能合约的应用场景

智能合约具有确定性、实时性、自治性、可观察、可验证、去中心化等特点,在数字支付、金融资产处置、云计算、物联网、共享经济等方面有着广阔的应用前景.

1) 数字身份. 智能合约可以让用户拥有和控制自己的数字身份,构建以用户为中心的个人网络,例如个人数字信誉和数字资产[37]等;同时还可以指定哪些个人数据可以或不可以与他人进行共享.

2) 数字记录. 智能合约可以实现合规性的自动化,提高数据的透明度,降低服务费用,实现记录的自动处理. 例如利用智能合约技术进行临床试验数据的管理,可以提高数据的透明度[38].

3) 证券. 基于智能合约实现数字化终端到终端的证券工作流程,用于资本化股权结构表管理能够极大地简化其工作流程,如帮助私人公司自动股息支付、股票分割和负债管理等流程,区块链证券公司 Symbiont 已经开始推动股票证书向使用加密区块链签名转变.

4) 金融贸易. 智能合约可以推动简化全球商品转移,带来更高资产流动性. 实现信誉证明和贸易支付流程的自动化发起,可以在客户、供应商和金融机构之间创建一种更高效、风险更小的流程.

5) 物联网. 利用智能合约在设备之间创建服务市场,创建分散的、共享的经济应用程序,兼顾隐私性与数字资产的价值性,促进服务和资源的共享,如 Blockchain-IoT[39].

6) 供应链. 智能合约能够为供应链的每一个环节提供更高的可见性,简化多重机构系统,与物联网设备进行协调,跟踪被管理的资产和产品,降低欺诈

和盗窃风险.例如 Everledger 与 IBM 已经将区块链用于供应链以提升其可见性.

7)保险.智能合约体系下的保险合同都是数字化的,被保存在区块链账本之中,无法篡改;同时可以自动化保险索赔流程,提供接近瞬时的处理、验证和付款服务.基于智能合约的保险业务,可以提供智能定制服务,可根据投保种类、时间、期限、理赔记录等自动匹配最佳投保方案,大大降低传统保险业的服务成本.

8)分布式计算.基于区块链技术,利用智能合约实现的分布式计算有着广阔的应用前景和现实意义.在全面进入区块链 1.0 时代后,全球有数以万计的具备算力的节点接入数字货币网络(如比特币网络)从事挖矿工作.利用智能合约实现的分布式计算,是实现将闲置节点作为计算资源供应商与客户(计算资源需求方)进行智能匹配,充分开发与利用现行网络计算资源,简化计算服务流程,降低计算服务成本,例如 iEx.ec[40] 旨在提供一个可扩展、安全、易于交互的数据集和计算资源的分布式应用程序.

5 智能合约的现存问题

虽然基于区块链的智能合约技术以其独特的优势吸引了众多研究者,但区块链智能合约技术还处在发展初期,存在诸多问题.除此之外,如何协调去中心化、低能耗、安全三者之间的关系,还有待进一步的研究.本节将从效率、隐私、安全、标准不统一 4 个方面,探讨和分析区块链智能合约技术发展中有待解决的问题.

5.1 效率问题

效率是影响智能合约可用性的重要因素.

1)数据存储问题.智能合约区块链记录了整个区块链网络从诞生至当前时间点的一切状态改变记录,并要求每个节点保存一份数据备份,这对日益增长的海量数据的存储和同步来说是极为困难的.例如以太坊区块链,完全同步自创世区块以来的全部区块数据需要约 180 GB 的存储空间,新添加进网络的节点完全同步区块链数据所花费的时间就长达 1 周.虽然以太坊数据区块中既包含智能合约代码,也包含交易数据等,但其从诞生到现在不过 3 年多的时间,即使将智能合约单独成链,按照以太坊愈加活跃的走势和时间累积效应,其区块链数据库过于庞大是一个急需解决的问题.尽管轻量化区块链可以部分解决此问题,但大部分区块链的轻量化都是以牺牲可靠性和安全性为代价的,因此如何协调轻量化与可靠性、安全性之间的关系还有待进一步的研究.同时,适用于工业级的解决方案仍有待开发[41].

2)状态确认的效率问题,这主要涉及 2 个问题:双重确认和闭锁问题.当具备访问权限的不同节点修改同一智能合约的同一个状态时,由于确认过程时间差的存在,将面临"双重确认"问题,即同一个状态被写入 2 次或多次,这有可能导致智能合约中的某个状态被错误地修改或覆盖.如节点 1 先于节点 2 提交修改申请,但由于确认过程时间差的存在,节点 2 的申请可能先于节点 1 被确认,而当节点 1 被确认时又会覆盖之前节点 2 的修改."闭锁问题",即优先获得确认的状态会产生闭锁合约的效果,使合约拒绝其他节点的访问.以太坊目前每秒能处理 10~20 笔交易,而确定交易则要等待下一个区块的生成,平均时间为 12 s.因此如何提高区块链处理事务的能力,是区块链智能合约技术亟待解决的重要问题.目前,研究者已经开始尝试解决此类问题,例如数据分片技术和索引技术.区块链技术平台 Zilliqa[42] 提出了一种基于分享协议的区块链,其采用分片技术,在测试网上每秒能处理近 1 400 笔交易.虽然相比以太坊区块链,其交易效率有了极大的改善,但对于部署一个面向世界所有人共同参与的大规模智能合约项目,处理大规模交易的抗压能力问题,仍然欠缺全面解决效率问题的方法.

5.2 隐私问题

智能合约风险管理和危机应对场景尚不完善.目前智能合约的隐私保护是基于非对称密码学的原理,具有很高的安全性,但随着数学研究和量子计算机技术的进一步发展,未来非对称加密算法存在被破解的可能,智能合约在隐私和安全方面仍然存在薄弱环节.

首先,区块链智能合约中的各用户并非完全匿名,准确地说,应该是假名性.智能合约、个人账户等都是通过一种地址标识(例如以太坊公私钥地址)来实现在区块链网络上的数据传输[43].但是,一旦当用户和现实世界的事务发生关联,如数字货币钱包代理商等,用户的地址标识就会变为网络代号(例如论坛的网名),虽不知道用户具体身份,但任何与用户相关的数据和行为都可以关联到这个代号上.同时,随着反匿名身份甄别技术的发展,智能合约用户的匿名性将难以保证.其次,区块链上的数据是公开透明的,通过各种数据挖掘技术,可以发现很多地址的互相关系,一旦真实身份泄露,用户的所有信息

都将公开.因此,智能合约风险管理方面的预案及相应技术手段的不完善,将成为影响智能合约应用发展的关键因素.

根据区块链技术的特点,区块链的隐私保护机制可分为3类:网络层的隐私保护、交易层的隐私保护和应用层的隐私保护[44].目前,隐私保护主要是通过增加恶意节点检索、获取和解读区块链数据的难度来实现的,如斯雪明教授团队[45]推出的拟态防御技术,以及网络层数据混淆技术、数据失真技术.随着区块链技术发展的日新月异,隐私保护问题将愈加突出,但目前的隐私保护方案都存在一定的不足,还有待进一步的研究.

5.3 安全问题

传统合约是基于自然语言描述的;而智能合约是用计算机代码来阐述、验证和执行合约,对保证数字资产和资源的安全性提出了更高的要求.智能合约最终会取代合约实体,但正如2.4节所述,智能合约涉及复杂的时间依赖和次序依赖关系,合约代码的不确定性和不一致性将导致智能合约本身存在漏洞,进而导致合约执行结果的不确定性,最终会导致法律责任的不确定性[46].2016年5月,史上最大的一个众筹项目The DAO由于智能合约在设计之初就存在漏洞,攻击者利用DAO.sol代码中"splitDAO"函数在递归发送模式上存在的漏洞盗取了大量以太币[47].2017年7月19日,Parity Wallet的多重签名钱包"multi-sig"代码中发现一个漏洞,即多重签名钱包的创建过程是无保护的,使得攻击者可以任意重置现有钱包的所有权和使用参数,这致使Parity Wallet中的3个大额以太币账户被盗取.因此,智能合约必须保证其逻辑属性的正确性以及合约代码和合约文本的一致性,并且能够自动生成可信任的执行代码.目前已有学者或团体提出了关于智能合约的验证方法,如OYENTE语义符号工具[48],但尚不完善,仍有众多关于智能合约的逻辑完整性、可验证性、安全性上的问题有待深入研究与探索.

5.4 标准不统一问题

智能合约和其相关平台的构建以及监管的标准是智能合约面临的一个重大挑战.当前,关于智能合约的标准有多个版本,主要由分散的智能合约应用联盟创建,如2015年5月Chain和花旗银行等金融机构发布了区块链方面的开放标准.2016年末以来,智能合约已经不仅局限于金融领域,基于智能合约的高级应用开始蓬勃发展,如iEx.ec.虽然各大商业联盟的标准正在逐步建立和完善,但在全球层面或国家层面仍然缺乏一个统一的技术开发标准,制约了智能合约及其应用的可扩展性和兼容性.

6 基于智能合约的众筹系统

本节基于智能合约相关理论,搭建了以太坊私有链实验环境,实现了一个众筹项目.本次实验环境为:CPU为Inter® Core™ i5-4210 M,主频为2.60 GHz,内存为8 GB,操作系统为Windows 8,编程语言为Solidity.

众筹智能合约,是在股权众筹发起的初始阶段,由发起人、平台、领投人等多方共同签署的一份合约.这份合约用于约定各成员的责任和义务,并且存在于区块链中,在无第三方中心机构的条件下,由区块链实现合约的自动运行、监管以及保证合约的不可篡改性.众筹智能合约一般存在3个特征:众筹目标(如4 000 ether)、众筹时间(如20 d)和不同众筹结果所对应的操作(如目标失败退回全款、目标成功时受益人获得加密代币).

众筹智能合约系统的实现主要包括3部分:实验环境的搭建、私有链的创建和智能合约系统的开发.

上述智能合约系统的实验环境是基于以太坊客户端Go-Ethereum[49]搭建的,该实验环境所需的支持软件如表3所示:

Table 3 Support Software of Experiment Environment

表3 支持软件

Software	Description
Solc	Solidity language compiler
Testrpc	The local test environment of Ethereum
Truffle	The development framework of the Ethereum smart contract which can be quickly compiled and deployed locally
Node.js	JavaScript runtime environment

私有链的创建是通过配置创世区块和启动节点来实现的.

创世区块的配置实际上是区块参数的配置.创世区块本质是一个json文件,其关键参数如表4所示.同一个区块链网络中,各个节点的创世区块必须是相同的,否则各个节点无法联通.为了便于本系统区块的快速生成,其区块参数的配置较为简单,这样

有利于合约的快速部署与验证.创世区块的具体参数配置为

```
{
  "config":{
    "chainId":10;
    "homesteadBlock":0;
    "eip155Block":0;
    "eip158Block":0;
     }
  "coinbase":"0x0000000000000000000000000000
              0000000000000";
  "difficulty":"0x40000";
  "extraData":"";
  "gasLimit":"0x2fefd8";
  "nonce":"0x0000000000000042";
  "mixhash":"0x0000000000000000000000000000
              0000000000000000000000000000
              0000000";
  "parentHash":"0x0000000000000000000000000
               0000000000000000000000000000
               0000000000000";
  "timestamp":"0x00";
  "alloc":{};
}
```

Table 4 The Parameter of the Block

表 4 区块参数

Parameters	Description
nonce	A 64-digit random number used to mine.
mixhash	Similar to *nonce*, *mixhash* is also used to mine, which value is related to the previous block.
difficulty	The difficulty of generating new blocks.
alloc	Used to preset accounts and the amount of Ether.
coinbase	Miner's Account.
timestamp	Set the timestamp for the genesis block.
parentHash	The Hash value of the previous block.
gasLimit	The upper limit of gas, used to limit the sum of the transaction information contained in the block.
extraData	Additional information.

在此系统中,启动节点的配置是通过编写启动文件来实现的.启动文件因实验设备、区块链网络结构的不同而略有差异.启动文件的关键参数如表 5 所示.

本文开发的众筹智能合约系统主要由支付模块、安全计算模块、紧急事件处理模块、权限管理模块、众筹模块、数字代币模块 TTC(test test coin)构成.其中,支付模块、安全计算模块、紧急事件处理模块、众筹模块的详细介绍如表 6~9 所示.

Table 5 Parameter of the Boot File

表 5 启动文件的参数

Parameters	Description
identity	Identification of the blockchain used to indicate the name of the current network.
init	Set the path to the genesis block file and create the initial block.
datadir	Set the storage path of the current blockchain data.
port	Network Listener Port.
rpc	Start the rpc communication to deploy and debug smart contract.
rpcapi	Set rpc client, usually db, eth, net, web3.
networkid	Set the network ID for the current blockchain to distinguish between different networks.
console	Start command-line mode to execute code in geth.

Table 6 Function of the Payment Module

表 6 支付模块功能函数表

Function	Description
asyncSend	Implement the function of asynchronous sending of data.
withdrawPayments	Implement the function of withdrawa.

Table 7 Function of the Security Calculation Module

表 7 安全计算模块功能函数表

Function	Description
safeMul	Reliable multiplication calculation.
safeDiv	Reliable division calculation.
safeSub	Reliable subtraction calculation.
safeAdd	Reliable addition calculation.
*min*64	Reliable 64 bit minimum calculation.
*max*64	Reliable 64 bit maximum calculation.
*min*256	Reliable 256 bit minimum calculation.
*max*256	Reliable 256 bit maximum calculation.

Table 8 Function of the Emergency Management Module

表 8 紧急事件处理模块功能函数

Function	Description
release	Be called at the end of an emergency to return to normal state.
emergencyStop	Be called in an emergency to trigger a stop state.

Table 9 Function of the Crowdsale Module

表 9 众筹模块功能函数

Function	Description
payable	Allow to accept ether. It corresponds to investment in the ether.
start	Start the crowdsale.
receiveETH	Access to investment funds (ether), increase transaction records, and investment records, and distribute TTC by exchanging rate.
emitTTC	Based on the crowdsale funds, compute the variable part of research and development funds (ttc_team), contingency reserve funds (ttc_reserve) and founding team bonuses (ttc_bounty).
bonus	calculation of the amount of TTC bonus according to the investment date.
receiveApproval	The function is called to refund when the minimum crowdsale fund is not reached.
finalize	The function is called to lock the crowdsale contract when crowdsale is completed.
drain	Troubleshooting.

智能合约的众筹算法如算法 1 所示:

算法 1. 众筹算法.

输入:最大众筹金额 *maxGap*、**最小众筹金额** *minGap*、**时间** *Time_crowdsale*、**最小投资金额** *minInvest*;

输出:若众筹成功,则调用函数 *finalize*,**按规则发放数字代币;若众筹失败,则调用函数** *receiveApproval*,**根据投资记录退回所有资金.**

部署智能合约并调用函数 *start*();

```
crowdsaleClosed=False;
While crowdsaleClosed=False do
  If receiveInvestRequest=False then
    continue;
  End If
  If Time_crowdsale<deadline then
    If TTC_total<maxGap then
      bonus();
      receiveETH();
    Else
      crowdsaleClosed=True;
      emitRLC();
      finalize();
    End If
  Else If TTC_total<minGap then
      receiveApproval();
    Else
      crowdsaleClosed=True;
      emitRLC();
      finalize();
    End If
  End If
End While
```

智能合约的投资算法如算法 2 所示:

算法 2. 投资算法.

输入:投资金额 *gapSender*;

输出:若投资成功,则调用函数 *sendInvestRequest*,**发送投资请求;若投资失败,则结束本次访问.**

访问智能合约并开始投资;

```
Input gapSender;
If crowdsaleClosed=True then
  break;
Else
  If Time_crowdsale<deadline then
    If gapSender>minInvest then
      If TTC_total+gapSender≤maxGap
        then
        If SenderBalance>gapSender then
          输入支付密码;
          sendInvestRequest();
        End If
      End If
    End If
  End If
End If
```

7 总结与展望

自 2009 年比特币诞生以来,以其为代表的区块链技术迅速崛起,已经成为学术界和产业界的热点研究课题. 经历了以数字货币为代表的区块链 1.0 时代,未来几年的区块链的研究方向将以"区块链 2.0 应用为主,区块链 3.0 应用为辅". 区块链 2.0 是智能合约时代,可以适应更为复杂的应用场景和更为高级的功能需求,使其在金融和社会系统中具有广泛的应用前景;同时,基于智能合约的其他高级应用也具备蓬勃的发展潜力,智能合约技术有望成为实现物联网、大交易量区块链、去中心化云存储和去中心化域名服务器的一个有效途径.

目前,智能合约技术的基础理论和技术研究尚处于起步阶段,仍缺乏对基础理论、关键技术以及对

行业发展至关重要的科学问题的研究与探索.该领域存在的挑战性问题较多,详见第5节介绍的4个方面.需要指出的是,对于智能合约所面临的挑战性问题,本文仅仅只是选择其中具有代表性的一部分给予介绍或归纳,并不涵盖所有的研究方向与问题.

本文系统地介绍了智能合约技术的全生命周期、基本架构、关键技术、研究现状、主要技术平台和应用场景以及探讨了可能存在的问题,是对目前智能合约技术研究成果的一个总结和归纳.同时,开发了一个众筹合约系统,探索和实践了智能合约的相关理论.本文介绍这个研究领域的初衷是希望展现智能合约技术的研究现状与前沿性问题,以便能为相关领域的学者提供参考和借鉴.

参 考 文 献

[1] Hancock M, Vaizey E. Technical report by the UK government chief scientific adviser [EB/OL]. [2017-07-19]. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/492972/gs-16-1-distributed-ledger-technology.pdf

[2] National Development and Reform Commission. The Thirteenth Five-Year National Informatization Plan [EB/OL]. [2017-07-19]. http://ghs.ndrc.gov.cn/ghwb/gjjgh/201705/t20170502_846421.html
(国家发展和改革委员会. "十三五"国家信息化规划 [EB/OL]. [2017-07-19]. http://ghs.ndrc.gov.cn/ghwb/gjjgh/201705/t20170502_846421.html)

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

[4] Vasek M. The age of cryptocurrency [J]. Science, 2015, 348(6241): 1308-1309

[5] Hodson H. Bitcoin moves beyond money [J]. New Scientist, 2013, 220(2945): 24-24

[6] Buterin V. A next-generation smart contract and decentralized application platform [EB/OL]. [2017-07-19]. https://github.com/ethereum/wiki/wiki/White-Paper

[7] Philip B. How blockchain technology could change our lives [EB/OL]. [2017-07-19]. http://8btc.com/doc-view.html?id=1319

[8] Szabo N. Formalizing and securing relationships on public networks [EB/OL]. [2017-07-19]. http://www.firstmonday.org/ojs/index.php/fm/artic le/view/548/469

[9] He Pu, Yu Ge, Zhang Yanfeng, et al. Survey on blockchain technology and its application prospect [J]. Computer Science, 2017, 44(4): 1-7,15 (in Chinese)
(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(4): 1-7,15)

[10] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system [EB/OL]. [2017-07-19]. http://www.bitcoin.org/bitcoin.pdf

[11] Antonopoulos A M. Mastering Bitcoin: Unlocking Digital Crypto-Currencies [M]. Sebastopol, CA: O'Reilly Media, Inc, 2014

[12] Swan M. Blockchain thinking: The brain as a decentralized autonomous corporation [J]. IEEE Technology & Society Magazine, 2015, 34(4): 41-52

[13] Dwork C, Naor M. Pricing via processing or combatting junk mail [C] //Proc of the 12th Annual Int Cryptology Conf. Piscataway, NJ: IEEE, 1992: 139-147

[14] Larimer D. Transactions as proof-of-stake [EB/OL]. [2017-07-19]. http://www.8btc.com/pos-white-book

[15] Larimer D. Delegated proof-of-stake white paper [EB/OL]. [2017-07-19]. http://8btc.com/doc-view-151.html

[16] Castro M, Liskov B. Practical Byzantine Fault Tolerance and Proactive Recovery [M]. New York: ACM, 2002

[17] NEO Smart Economy. NEO white paper [EB/OL]. [2018-01-23]. http://docs.neo.org/en-us/index.html

[18] Bitcoin. Bitcoin core integration [EB/OL]. [2017-07-19]. https://github.com/bitcoin/bitcoin

[19] Yuan Yong, Wang Feiyue. Blockchain: The state of the art and future trends [J]. Acta Automatica Sinica, 2016, 42(4): 481-494 (in Chinese)
(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)

[20] Hu Kai, Bai Xiaomin, Gao Lingchao, et al. Formal verification method of smart contract [J]. Journal of Information Security Research, 2016, 2(12): 1080-1089 (in Chinese)
(胡凯, 白晓敏, 高灵超, 等. 智能合约的形式化验证方法 [J]. 信息安全研究, 2016, 2(12): 1080-1089)

[21] Wright A, De Filippi P. Decentralized blockchain technology and the rise of Lex cryptographia [J]. Social Science Electronic Publishing, 2015, 34(4): 41-52

[22] Clack C D, Bakshi V A, Braine L. Smart contract templates: Foundations, design landscape and research directions [EB/OL]. [2017-07-19]. https://arxiv.org/abs/1608.00771

[23] Zhang Fan, Cecchetti E, Croman K, et al. Town crier: An authenticated data feed for smart contracts [C] //Proc of the 23rd ACM Conf on Computer and Communications Security. New York: ACM, 2016: 270-282

[24] Zhang Zhengwen, Da Hongfei. Reconstruction of smart contracts: Parallel universe and wireless extension [EB/OL]. [2017-07-19]. http://8btc.com/thread-49429-1-1.html
(张铮文, 达鸿飞. 重构智能合约:平行宇宙与无线扩展 [EB/OL]. [2017-07-19]. http://8btc.com/thread-49429-1-1.html)

[25] Kenneth M. Symmetry and model checking [J]. Formal Methods in System Design, 1996, 9(1/2): 105-131

[26] D'Silva V, Kroening D, Weissenbacher G. A survey of automated techniques for formal software verification [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2008, 27(7): 1165-1178

[27] Frantz C K, Nowostawski M. From institutions to code: Towards automated generation of smart contracts [C] //Proc of the 1st IEEE Int Workshop on Foundations and Applications of Self-* Systems. Piscataway, NJ: IEEE, 2016: 210-215

[28] Zhang Yingbei, Gong Zhenghu, Wang Lechun. Research and implementation of LDP conformance testing [J]. Computer Engineering and Science, 2004, 26(11): 14-16 (in Chinese)
(张颖蓓, 龚正虎, 王乐春. LDP 一致性测试的研究与实现[J]. 计算机工程与科学, 2004, 26(11): 14-16)

[29] The Linux Foundation. Smart contract engine [EB/OL]. [2017-07-19]. https://www.hyperledger.org/projects/fabric

[30] Watanabe H, Fujimura S, Nakadaira A, et al. Blockchain contract: Securing a blockchain applied to smart contracts [C] //Proc of the 2016 IEEE Int Conf on Consumer Electronics. Piscataway, NJ: IEEE, 2016: 467-468

[31] The Economist. The DAO of accrue: A new automated investment fund has attracted stacks of digital money [EB/OL]. [2017-07-19]. https://www.economist.com/news/finance-and-economics/21699159-new-automated-investment-fund-has-attracted-stacks-digital-money-dao

[32] Jacynycz V, Calvo A, Hassan S, et al. Betfunding: A distributed bounty-based crowdfunding platform over ethereum [J]. Advances in Intelligent Systems and Computing, 2016, 474: 403-411

[33] The Linux Foundation. About the hyperledger project [EB/OL]. [2017-07-22]. https://www.hyperledger.org/about

[34] Carrillo P N, Peña C I, Rosa J L D L. Eurakos next: A cryptocurrency based on smart contracts [C] //Proc of the 19th Int Conf of the Catalan-Association-for-Artificial-Intelligence. Ohmsha: IOS, 2016: 221-226

[35] Huckle S, Bhattacharya R, White M, et al. Internet of things, blockchain and shared economy applications [J]. Procedia Computer Science, 2016, 98: 461-466

[36] Tsai Wei-Tek, Yu Lian, Wang Rong, et al. Blockchain application development techniques [J]. Journal of Software, 2017, 28(6): 1474-1487 (in Chinese)
(蔡维德, 郁莲, 王荣, 等. 基于区块链的应用系统开发方法研究[J]. 软件学报, 2017, 28(6): 1474-1487)

[37] Yasin A, Liu Lin. An online identity and smart contract management system [C] //Proc of the 40th Annual IEEE Computer Software and Applications Conf Symp. Piscataway, NJ: IEEE, 2016: 192-198

[38] Nugent T, Upton D, Cimpoesu M. Improving data transparency in clinical trials using blockchain smart contracts [EB/OL]. [2017-07-19]. https://f1000research.com/articles/5-2541/v1#referee-response-17773

[39] Christidis K, Devetsikiotis M. Blockchains and smart contracts for the Internet of things [J]. IEEE Access, 2016, 4: 2292-2303

[40] iEx.ec. The iEx.ec project [EB/OL]. [2017-07-19]. http://iex.ec/wp-content/uploads/2017/04/iExec-WPv2.0-English

[41] Eyal I, Gencer A E, Renesse R V. Bitcoin-NG: A scalable blockchain protocol [C] //Proc of the 13th USENIX Conf on Networked Systems Design and Implementation. Berkeley: USENIX Association, 2016: 45-59

[42] Zilliqa. Next-Gen high throughput blockchain platform [EB/OL]. [2018-01-23]. https://www.zilliqa.com

[43] Kosba A, Miller A, Shi E, et al. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts [C]//Proc of the 2016 IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2016: 839-858

[44] Zhu Liehuang, Gao Feng, Shen Meng, et al. Survey on privacy preserving techniques for blockchain technology [J]. Journal of Computer Research and Development, 2017, 54(10): 2170-2186 (in Chinese)
(祝烈煌, 高峰, 沈蒙, 等. 区块链隐私保护研究综述[J]. 计算机研究与发展, 2017, 54(10): 2170-2186)

[45] Si Xueming, Wang Wei, Zeng Junjie, et al. A review of the basic theory of mimic defense [J]. Engineering Sciences, 2016, 18(6): 62-68 (in Chinese)
(斯雪明, 王伟, 曾俊杰, 等. 拟态防御基础理论研究综述[J]. 中国工程科学, 2016, 18(6): 62-68)

[46] Savelyev A. Contract law 2.0: 'Smart' contracts as the beginning of the end of classic contract law [J]. Information & Communications Technology Law, 2017, 26(2): 116-134

[47] Andy D. The DAO [EB/OL]. [2017-07-22]. http://ethfans.org/posts/127

[48] Luu L, Chu D H, Olickel H, et al. Making smart contracts smarter [C] //Proc of the 23rd ACM Conf on Computer and Communications Security. New York: ACM, 2016: 254-269

[49] Ethereum. Go-Ethereum [EB/OL]. [2017-07-19]. https://github.com/ethereum/go-ethereum

He Haiwu, born in 1977. PhD, professor. Senior member of CCF. His main research interests include blockchain technology, big data, cloud computing, parallel and distributed computing.

Yan An, born in 1991. MSc candidate. His main research interests include blockchain technology, intelligent control and machine learning (anyan_tyut@163.com).

Chen Zehua, born in 1974. PhD, professor. Senior member of CCF. Her main research interests include blockchain technology, industrial big data and intelligent control.