第 38 卷第 4 期　　重庆工商大学学报(自然科学版)　　2021 年 8 月
Vol. 38　No. 4　　J Chongqing Technol & Business Univ(Nat Sci Ed)　　Aug. 2021
doi:10. 16055/j. issn. 1672-058X. 2021. 0004. 006

一种基于区块链 PoS 共识算法的改进研究*

钟增胜[1,2]
(1. 中南大学 计算机学院,长沙 410083; 2. 重庆工商大学 招生就业处,重庆 400067)

摘　要:区块链共识算法保证了区块链中的区块按时间戳有序生成,算法优劣直接影响区块链系统性能。PoS 共识算法是一种区块链公有链采用的主流共识算法,但生成区块的时间具有随机性,有些情况下区块间隔时间过长,不能满足商业应用场景需要;针对 PoS 共识算法生成区块的速度存在的性能局限,改进设计的 Silkworm 算法,通过智能合约对最快生成区块时间和最慢生成区块时间进行定义,结合 Raft 算法进行主节点选举;在有交易的情况下,当 PoS 共识算法未在定义的最快时间内生成区块时,Silkworm 算法确保由主节点自动快速生成区块;在无交易情况下,当 PoS 共识算法未在定义的最慢时间内生成区块时,Silkworm 算法也由主节点生成区块。而当主节点关闭或出故障时,PoS 共识算法仍然生效正常生成区块。通过实验验证:Silkworm 算法能较大提升基于 PoS 共识算法的区块链的性能,保证了区块链的安全性和健壮性,更能满足商业应用场景的需要。

关键词:区块链;共识算法;PoS;PoW;Raft
中图分类号:TP311. 1　　**文献标志码**:A　　**文章编号**:1672-058X(2021)04-0036-06

0　引　言

化名“中本聪”的学者在 2008 年发表了《比特币:一种点对点的电子现金系统》[1]论文后,在 2009 年 1 月开发出比特币系统。比特币系统通过算法实现了价值转移,不需要通过中介机构。2010 年 5 月有人用 1 万比特币(BTC)购买价值为 25 美元的商品,确定了比特币最初单价为 0. 002 5 美元[2],在数字货币交易市场上,比特币价格在 2017 年 12 月达到 2. 0 万美元,2021 年 2 月达到 5. 8 万美元[3]。

比特币的底层支撑技术是区块链,“中本聪”巧妙地将相关技术结合在一起构成区块链的核心技术,包括了 P2P 网络技术、哈希运算技术、非对称加密技术和 PoW 共识算法等,比特币是区块链系统目前最成功的应用,越来越多的商业应用场景引入区块链技术。

共识机制的核心是在共识算法保障下,在一定的时间内,解决分布式系统中状态的一致性问题,使得各节点承认且不可篡改。在区块链系统中,主要解决分布式场景下各节点交易数据和交易状态的一致性问题,实现去中心化的多方互信,共识算法是区块链的核心技术,是实现互联网从信息互联到价值互联的关键技术。

区块链按节点权限可分为公有链和许可链。公有链是指区块链节点没有准入机制,参与节点能随时加入或退出,代表性的共识算法主要有两种:一是工作量证明(Proof of Work,PoW),通过消耗算力去获取记账权的概率,算力越大概率越高,优点是随机性强、公平性好,缺点是耗能、共识效率低。二是权益证明(Proof of Stake,PoS),通过持有资产去获取记账权的概率,资产越多概率越大,优点是节能环

收稿日期:2021-02-01;**修回日期**:2021-04-02.
* **基金项目**:重庆工商大学 2019 年校级科研项目资助(960419055).
作者简介:钟增胜(1974—),男,江西萍乡人,博士研究生,从事区块链技术及项目应用、数据可视化及可视分析研究.

保,缺点是权力集中。许可链指参与的节点需要授权,未经授权的节点无法接入区块链,代表性的共识算法有 RAFT(Replicated and Fault Tolerant),优点是效率高、容易理解,缺点是不能容纳作恶节点。

文献[4]研究了共识算法的演化历程,分析了 PoW 的优缺点,比较了不同共识算法的特点,但未提出如何解决性能问题的建议。文献[5]将不同共识协议分步骤进行解耦比较,进行了一定的性能分析,缺乏对 PoS 深入分析。文献[6]对 Raft 算法进行了改进,但未研究与 PoS 融合。已研究的成果对 PoS 共识算法的安全和性能进行过分析,但改进研究较少。

采用 PoS 共识算法的区块链生成区块的时间具有随机性,有些情况下区块间隔时间过长,不能满足商业应用场景需要。Silkworm 算法能弥补 PoS 共识算法的性能局限,同时提升区块链的安全性和健壮性。

1　共识算法原理

1.1　PoW 算法原理

使用 PoW 作为共识机制典型的公有链有比特币(Bitcoin)[1]、以太坊(Ethreum)[9]等。

比特币系统中通过不断重试 n Nonce 值,n Nonce 的范围为 $0 \sim 2^{32}$,若满足不等式(1),即满足生成区块条件,寻找到 n Nonce 符合条件的该节点可以打包交易记录并组装到区块中,再通过 P2P 网络将区块发往其他节点验证。

$$\text{SHA256}(\text{SHA256}(\text{version}+\text{prev\_hash}+\text{merkle\_root}+n\ \text{time}+n\ \text{bits}+n\ \text{Nonce}+x))<\text{TARGET} \quad (1)$$

其中,SHA256 为生成 256 位消息摘要的哈希算法[10],version 为版本号,prev_hash 为前一区块哈希值,merkle_root 为当前区块交易树根哈希值,n time 为时间戳,n bits 为当前难度值,x 为区块填充信息,TARGET 为目标值。

比特币系统通过前 2 016 个区块的生成区块时间和单个区块的难度计算新的 TARGET,以保证系统生成区块时间动态维持在 10 min。

1.2　PoS 算法原理

为避免 PoW 算法造成大量的算力资源浪费,PoS 算法以节点持有代币的数量和时间表示权益,权益越大越容易获得生成区块条件。

$$\text{Hash}(n\ \text{StakeModifier}+tx\ \text{Prev. block.}\ n\ \text{Time}+tx\ \text{Prev. offset}+tx\ \text{Prev. nTime}+tx\ \text{Prev. vout.}\ n+n\ \text{Time})<bn\ \text{Target}*bn\ \text{CoinDayWeight} \quad (2)$$

其中,Hash 为哈希算法,n StakeModifier 为权重修正因子,tx Prev. block. n Time、tx Prev. offset、tx Prev. n Time、tx Prev. vout. n 为未花费的交易支出(Unspent Transaction Outputs,UTXO)属性,n Time 为时间戳,bn Target 为目标值,bn CoinDayWeight 为币龄。

节点通过不断重试所持有的 UTXO,若满足不等式(2),即可以打包交易记录并组装到区块中,再通过 P2P 网络将区块发往其他节点验证。

1.3　Raft 算法原理

Raft 共识算法的整体原理框架是一个基于 Log 复制机制的状态机[6]。所有节点有 3 种状态:领导者(Leader)、跟随者(Follower)和参与者(Candidate),具体流程如图 1 所示。

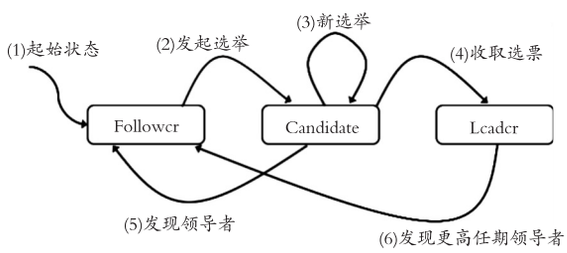

图 1　节点状态转换图

Fig. 1　Node status conversion

(1) 起始状态:节点刚启动时自动进入 Follower 状态。

(2) 发起选举:选举定时器到期后,节点切换为 Candidate 状态发起选举。

(3) 新选举:在一次选举超时到来前若还没有 Leader,则保持在 Candidate 状态开始新选举。

(4) 收取选票:收到超过半数节点选票,切换状态为新的 Leader。

(5) 发现领导者:若收到 Leader 或更高任期消息,切换回 Follower 状态。

(6) 发现更高任期领导者:若收到更高任期消息,切换回 Follower 状态。

每一个节点状态中都保存当前任期号(Current Term),节点在进行通信时都会带上本节点的当前任期号。如果一个 Candidate 或者 Leader 状态的节点发现自己的当前任期号已经小于其他节点了,那

么将切换到 Follower 状态。

2 PoS 算法区块链性能问题

衡量区块链性能的指标一般有平均每秒处理的交易数(Transaction Per Second,TPS)、平均生成区块时间等,采用 PoS 共识算法的数字货币,最早有点点币(Peercoin)[7],TPS 为 7 左右,平均生成区块时间为 600 s,量子链(QTUM)[8,11],TPS 为 70 左右,平均生成区块时间为 128 s。

以量子链为例,它结合了比特币和以太坊的优势,打通了比特币的 UTXO 模型和以太坊的智能合约生态,在 5 000 个区块前使用 PoW 算法,其后一直使用 PoS 算法。

根据量子链官方微信公众号发布的数据[8],2019 年 9 月 1 日—10 月 15 日区块 438 439 ~ 465 623 时间间距分布如图 2 所示。

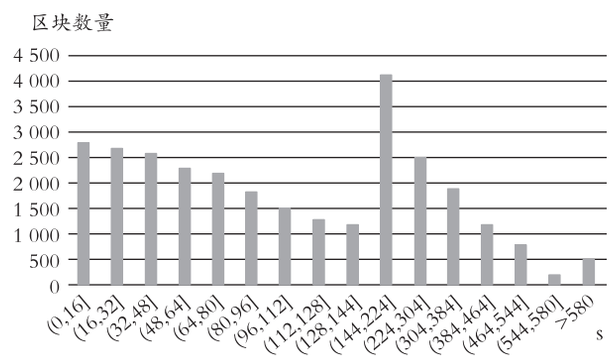

图 2 量子链 438 439 ~ 465 623 区块间隔时间图

Fig. 2 Time intervals between blocks QTUM 438 439 ~ 465 623

通过量子链官方浏览器[12]拉取了 2020 年 12 月 1 日区块 743 647 ~ 743 683 实时数据,如图 3 所示。

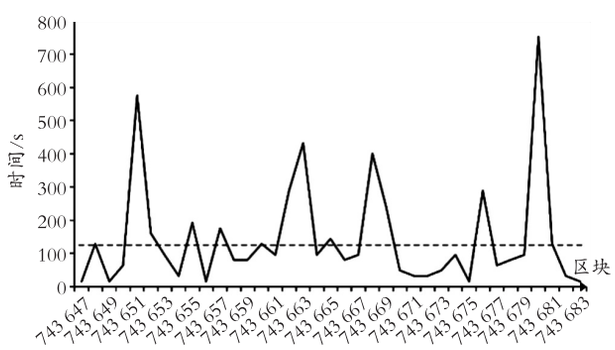

图 3 量子链 743 582 ~ 744 257 区块间隔时间图

Fig. 3 Time intervals between blocks QTUM 743 582 ~ 744 257

从图 2 和图 3 可以看出:量子链的最快生成区块时间 16 s,最慢生成区块时间 752 s,平均生成区块时间为 128 s,16 s 内生成的区块数量占比 9%,128 s 内生成的区块数量占比 58%。

PoS 共识算法生成区块时间间隔有较大幅度的变动,而商业应用场景与区块链结合,对单笔交易而言,往往要求生成区块时间间隔小且稳定。因此,改进 PoS 共识算法的目标是减小区块间隔,更快速地确认链上交易和实现价值转移。

3 PoS 共识算法的改进

利用智能合约定义参数,通过投票选举出主节点(Leader),主节点判断 PoS 是否在约定时间范围内生效,若生效则会清空内存交易池由 PoS 节点打包生成区块,若未生效则主节点进行打包生成区块。改进后的算法定义为 Silkworm 共识算法。

3.1 部署时间参数智能合约

定义区块产生的最小间隔时间 SwMinInterval,默认为 12 s,定义区块产生的最大间隔时间 SwMaxInterval,默认为 360 s。将智能合约部署到区块链生效后得到参数合约 ID。

若要改变最小或最大间隔时间,重新部署智能合约可得到新的参数合约 ID。

3.2 部署主节点选举智能合约

定义选举任期间隔时间 SwElectionInterval,如 24 h。

定义节点参与条件,如持有代币达到 5 万或线下评审(考虑节点综合实力等),将合约部署到区块链生效后得到选举合约 ID。

符合条件的节点调用智能合约进入 Follower 列表。Follower 节点定时调用智能合约竞选 Leader。

3.3 部署主控智能合约

定义是否启用 Silkworm 算法标志 SwEnabled,默认为 True。定义选举合约 ID。定义区块高度 SwHeight 1,参数合约 ID 1,区块高度 SwHeight 2,参数合约 ID 2。

区块高度 SwHeight 1 和 SwHeight 2 用于切换新老参数合约,以便实现平稳过渡。

3.4 Silkworm 运行流程

Silkworm 共识算法流程如图 4 所示:

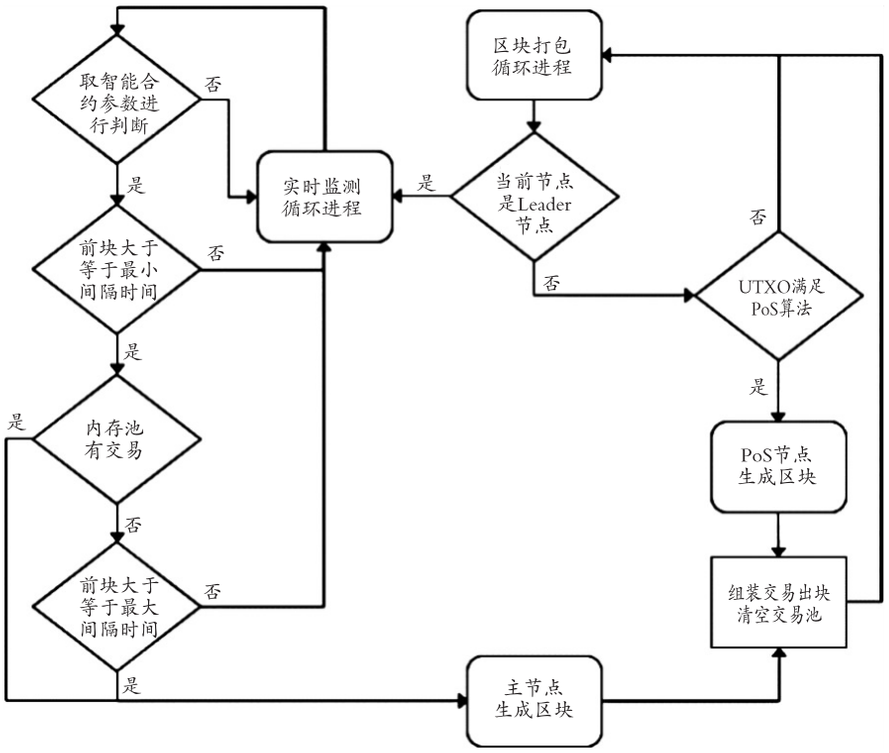

图 4　Silkworm 共识算法流程图

Fig. 4　Silkworm consensus algorithm flowchart

步骤 1　任意节点均有单独的进程执行共识算法进行区块打包循环,首先查找主控智能合约,判断启用标志 SwEnabled 是否为 True,查找主节点选举智能合约,判断本节点是否 Leader 状态。

步骤 2　若步骤 1 不符合,则按 PoS 共识算法规则,满足条件就由 PoS 节点生成区块,组装交易出块并清空区块链节点的内存交易池。

步骤 3　若步骤 1 符合,则取智能合约参数,判断当前时间戳与前一区块的时间差,是否大于等于最小间隔时间 SwMinInterval。

步骤 4　若步骤 3 不符合,则忽略,若步骤 3 符合,判断当前内存交易池是否存在交易,若存在交易,主节点直接生成区块,组装交易出块并清空区块链节点的内存交易池。

步骤 5　若步骤 4 中没有交易,而当前时间戳与前一区块的时间差,大于等于最大间隔时间 SwMaxInterval,主节点也直接生成区块,保证区块的后续确认不受影响。

4　实验验证

4.1　实验准备

4.1.1　源代码改造

从 https://github. com/qtumproject/qtum 下载量子链源码,开发环境采用 vmware+utuntu18. 04+Qt5. 12. 3,按下列步骤改造为一条新的区块链。

步骤 1　修改 src/chainparams. cpp 中的 CreateGenesisBlock 函数,计算出的创世纪块符合预设值。

步骤 2　修改 src/chainparams. cpp 中 pchMessageStart 网络字节值和 src/chainparamsbase. cpp 中 rpc 端口,使其不用于 QTUM 链。

步骤 3　适当修改其他配置文件,将源码编译通过,生成 Linux 版本的客户端。

4.1.2　服务器

在阿里云开通 10 台 2vCPU8GB 内存的服务器,操作系统为 Ubuntu18. 04 64 位,网络互通。所有服务器部署改造后的新区块链客户端并启动成功。

4.1.3　初始化区块

使用 rpc 命令 generate 初始化 5 000 个区块,这些区块是通过 PoW 共识算法生成的,检查 10 个服务器的区块是否同步正常。再等待 24 h,新区块链自动使用 PoS 共识算法生成 675 个区块。

4.1.4　部署智能合约

(1) 根据第 3.1 节、第 3.2 节、第 3.3 节部署相关智能合约。

(2) 根据第 3.4 节修改 src/miner. cpp 源码,重新编译发布。

4.2 实验步骤

为方便观察实验结果,人工指定 Leader 节点。

在不启用主节点的情况下,2 h 内随机发送交易,观察产生区块结果,共进行3组。

在启用主节点情况下,2 h 内随机发送交易,观察产生区块结果,共进行3组。

4.3 实验结果分析

实验数据对比情况如表1所示。在不启用主节点的情况下,PoS 共识算法生效,区块产生的间隔时间最大、最小和平均与 QTUM 相近,说明新区块链工作状态正常,交易越频繁,对生成区块间隔平均时间略增。

表1 在不同条件下发送交易的比对情况

Table 1 Comparison of transactions sent under different conditions

主节点	组别	交易交数	区块		出块间隔时间/s		
			区间	数量	最大	最小	平均
不启用	1组	100	5 680~5 735	56	600	16	128
	2组	150	5 750~5 803	54	600	16	132
	3组	200	5 810~5 859	50	720	16	144
启用	1组	100	5 900~6 064	147	360	12	48
	2组	150	6 050~6 240	191	360	12	40
	3组	200	6 250~6 481	232	360	12	32

在启用主节点的情况下,Silkworm 共识算法生效,因为时间参数智能合约中规定了区块间隔时间的最大值为360 s,最小值为12 s,在有交易时,区块生成区块间隔时间恒定在12 s,在没有交易时,生成区块间隔时间有变化,最大不会超过360 s,平均时间在128 s左右。

从实验结果可以看出:Silkworm 共识算法使区块链的性能达到了预期的目标,也能够满足商业应用场景的使用。若有必要,也可以通过智能合约调小生成区块间隔时间,但不能低于4 s。

在区块链没有交易的情况下,为避免更快生成更多的空块,节约存储资源,Silkworm 共识算法自动不生效,由 PoS 节点起主导作用。在主节点因网络等原因失效的情况下,PoS 节点仍然发挥作用,保障了区块链的稳定性和安全性。

5 结束语

给出了一个 PoS 共识算法的改进方法,该方法通过智能合约治理区块链的网络参数,对最快生成区块时间和最慢生成区块时间进行定义,结合 Raft 算法选举主节点,实现了在有交易的情况下,当 PoS 共识算法未在定义的最快时间内生成区块时,Silkworm 算法确保由主节点自动快速生成区块;在无交易情况下,当 PoS 共识算法未在定义的最慢时间内生成区块时,Silkworm 算法也由主节点生成区块。而当主节点关闭或出故障时,PoS 共识算法仍然正常生成区块。

通过实验验证,启用主节点使用 Silkworm 共识算法,能保证交易在规定的最快时间内生成区块,没有交易时自动降低区块生成速度,节约存储资源,这种区块链更能满足商业应用场景的要求。

Silkworm 共识算法还存在一些不足,当主节点频繁生效时,PoS 节点生成区块的概率有可能会降低,效率上比纯许可链更低。

区块链共识算法还在不断发展中,Silkworm 共识算法在基于公有链 PoS 共识算法上进行改进,还需要在安全性、孤立块和分叉方面进行更深入的验证。

参考文献(References):

[1] SATOSHI N. Bitcoin: A Peer-to-Peer Electronic Cash System [EB/OL]. (2008-10-31)[2021-03-02]. https://bitcoin.org/bitcoin.pdf

[2] 李翀. 比特币会成为货币吗? [J]. 当代经济研究, 2015(4):60-65+2+97
LI C. Will Bitcoin Become A Currency? [J]. Contemporary Economic Research, 2015(4):60-65+2+97(in Chinese)

[3] WWW.BTC126.COM. Bitcoin Market Trend Chart [EB/OL] (2021-03-02)[2021-03-02]. https://price.btc126.com

[4] 武岳,李军祥. 区块链共识算法演进过程[J]. 计算机应用研究,2020,37(7):2097—2103
WU Y, LI J X. Evolution Process of Blockchain Consensus Algorithm [J]. Application Research of Computers, 2020,37(7):2097—2103(in Chinese)

[5] 夏清,窦文生,郭凯文,等. 区块链共识协议综述[J]. 软件学报,2021,32(2):277—299
XIA Q, DOU W S, GUO K W, et al. Survey on Blockchain Consensus Protocol[J]. Journal of Software, 2021,32(2):277—299(in Chinese)

[6] 王日宏,张立锋,周航,等. 一种结合 BLS 签名的可拜占庭容错 Raft 算法[J]. 应用科学学报,2020,38(1):93—104
WANG R H, ZHANG L F, ZHOU H, et al. A Byzantine Fault Tolerance Raft Algorithm Combines with BLS Signature. Journal of Applied Sciences, 2020,38(1):

93—104(in Chinese)
[7] KING S, NADAL S. PPCoin: Peer-to-peer Crypto-currency with Proof-ofstake[EB/OL]. (2012-08-19)[2021-03-02]. https://www.peercoin.net/whitepapers/peercoin-paper.pdf
[8] JACKSON B. QIP-26:Qtum Speed up[EB/OL]. (2020-10-21)[2021-03-02]. https://mp.weixin.qq.com/s/-5Xsrnt4av-tMzKKwFz2AA
[9] VITALIK B. Ethereum Whitepaper[EB/OL]. (2021-02-09)[2021-03-02]. https://ethereum.org/en/whitepaper
[10] BACK A. Hashcash: A Denial of Service Counter-measure[EB/OL]. (2002-08-01)[2021-03-02]. http://www.hashcash.org/papers/hashcash.pdf
[11] QTUM. ORG. Qtum Blockchain New Whitepaper[EB/OL]. (2020-02-24)[2021-03-02]. https://qtum.org/user/pages/01.home/Qtum%20new%20whitepaper_cn.pdf
[12] QTUM. QTUM Blockchain Browser [EB/OL]. (2021-03-02)[2021-03-02]. https://qtum.info/block

An Improvement on Blockchain-Based PoS Consensus Algorithm

ZHONG Zeng-sheng

(1. School of Computer Science, Central South University, Changsha 410083, China;
2. Admission Office, Chongqing Technology and Business University, Chongqing 400067, China)

Abstract: While ensuring that blocks in the blockchain are generated in an orderly manner according to the timestamp, Blockchain consensus algorithm has a direct effect on the performance of the blockchain system. PoS consensus algorithm is a mainstream one adopted in the public chain of blockchain, blocks are generated randomly and in some cases the time interval between blocks is too long, making it difficult to meet the needs of commercial scenarios. To address performance limitations of PoS consensus algorithm, we design a Silkworm algorithm, which defines the fastest and slowest time for the generation of blocks through smart contract, and which selects the master node by combining Raft algorithm. When there are transactions, if PoS consensus algorithm does not generate blocks within the defined fastest time, the Silkworm algorithm ensures that blocks are automatically generated by the master node. In the case of no transactions, when the PoS consensus algorithm does not generate blocks within the defined slowest time, the Silkworm algorithm also generates blocks through the master node. It still generates blocks normally when the master node shuts down or fails. As verified in experiment, Silkworm algorithm can greatly improve the performance of blockchain based on PoS consensus algorithm, ensure the security and robustness of blockchain, and better meet the needs of commercial application scenarios.

Key words: blockchain; consensus algorithm; PoS; PoW; Raft

责任编辑:罗姗姗

引用本文/Cite this paper:

钟增胜. 一种基于区块链 PoS 共识算法的改进研究[J]. 重庆工商大学学报(自然科学版),2021,38(4):36—41

ZHONG Z S. An Improvement on Blockchain-Based PoS Consensus Algorithm[J]. Journal of Chongqing Technology and Business University (Natural Science Edition),2021,38(4):36—41