



BIT TRUST SYSTEM

比特信任系统

—

无需信任，自由交易



摘要

本文提出了一种完全通过点对点技术实现的基于交易记录的信任计算网络, 它使得C2C交易(该白皮书中C2C交易特指去中心化, 点对点关于某种标的交易)能够直接由双方发起并完成交易, 中间不需要通过任何的中心化机构。比特币(bitcoin)系统解决了电子现金的问题, 但仍然无法解决去中心化在线交易的信任问题。以太坊通过智能合约可以解决基于其主网发行token的交易信任问题。但在广泛的区块链应用场景, 涉及数字货币与法币、商品、服务等P2P交易领域, 尚没有有效方法解决信任问题。

目录

1 简介	1
2 去中心化C2C交易的信任问题	2
3 系统功能	3
3.1 系统功能 ······	3
3.2 系统各元素说明 ······	4
3.3 一个去中心化C2C交易举例 ······	4
4 系统架构	5
4.1 多链结构 ······	5
4.2 模块化设计 ······	5
4.3 数据格式与存储 ······	5
4.4 加密算法 ······	6
4.5 交易链 ······	8
4.6 BIUT链 ······	8
4.7 BIU链 ······	9
4.8 自治域 ······	9
4.9 网关与智能合约设计 ······	9
5 PGPOW共识算法	10
5.1 节点群落 ······	10
5.2 节点群落的生成 ······	11
5.3 BIU链共识机制及维护 ······	11
5.4 交易链共识机制及维护 ······	12
6 网络	13
7 信任算法	14
7.1 信任机制的实现 ······	14
7.2 名词定义 ······	14
7.3 基于BIU链交易记录的Activity Rank计算 ······	15
7.4 基于BIUT链的Importance Rank计算 ······	16
7.5 基于交易链的信誉和信任计算 ······	17
7.6 信任机制算法总结 ······	20
8 激励	22
8.1 BIUT的经济模型 ······	22
8.2 BIU的经济模型 ······	24
9 公链商店	25
9.1 公链商店的代码框架 ······	25
9.2 组件 ······	26
10 总结	26
11 代码地址	26

1 简介

与比特币出现和发展的时间几乎同步，互联网诞生了众筹、众包、共享经济、P2P金融和P2P保险等新经济商业模式。乘坐陌生人的汽车，参加一个陌生人创意产品的投资，住到一个陌生人家里，把设计等任务外包给一个陌生人，把钱借给一个陌生人等等这些都呈现明显的C2C商业的特征。目前这些商业模式通过买家的评论和评分来进行信誉评价。而买家通过卖家的信息展示，以及这些评价体系来做购买决定。

这些新经济的商业组织往往一开始发展迅速，很快就裹足不前甚至向原有业态倒退，共享租车平台成了出租车公司，众包变为选择供应商，中国的P2P借贷变为中心化资金池集体性诈骗，深层的原因是新的信任手段缺失。新瓶装旧酒，这些商业模式起源于信任的重构，但在原有互联网技术下，无法提供系统级的信任解决方案。

在区块链行业中，有了去中心化的货币比特币，也有了去中心化协作的初步形式，但缺乏去中心化信任机制的工具，因此除了币币交易以及募集资金的功能，迟迟无法在有实际应用，甚至简单的法币转账依然无法通过区块链实现，只要和法币、链下资产、商品和服务等有结合，就无法去中心化和去信任化，所有的交易都受制于信任机制。而区块链项目行业的投资市场，陌生人把有流动性价值的BTC投给新的项目，也是一种新的金融众筹形式，也同样需要基于新型的合作信任。但因为信任机制的缺失，跑路项目众多，造成“劣币驱逐良币”。

人类社会的信任经历了三个阶段，农业社会是习俗型信任，工业社会是契约（合同）型信任，信息社会是合作型信任。对应的信任建立方法是基于人际关系，信任中介担保的合同和信息系统。目前传统互联网贸易中，广泛使用的是工业时代的契约信任机制。在比特币的白皮书中中本聪也提出，传统的基于互联网的电子交易，都需要中心化组织作为第三方机构，为交易双方解决信任评估。这种模式也是契约信任机制，交易的信任成本高，还需要交易双方提供完全不必要的隐私。

2 去中心化C2C交易的信任问题

在电商发展中,如eBay、亚马逊、淘宝、美团大众点评等均采用的是中心化通过评价算法来计算信任度。这种中心化评价系统的信任机制效果正在降低甚至失效,表现在卖家集体性选择刷单,卖家选择成本很高,评价的可信度不高,算法不透明等现状。

探索P2P技术在电子商务领域的应用,Lightshare很早就曾经使用P2P技术创建C2C商城,而Openbazaar也尝试使用区块链技术提供C2C交易服务,并提供了BTC, BCH, ETH等数字货币支付功能。但因信任机制建立的缺失,并未对交易效率有提升。

从逻辑上看,比特币网络解决了Alice传送电子信息给Bob,信息多重发送的问题,是Alice和Bob的单向信息发送的博弈问题。互联网C2C交易中需要解决的是Alice发送电子货币给Bob, Bob发送电子货币(实物资产、商品、服务)给Alice的双向信息传递的博弈问题。

从博弈论角度分析,多次博弈交易可以产生直接信任。单次交易可以通过抵押来解决违约成本问题,而多次博弈通过交易记录,多次交易历史提供参考,并用经济激励造成违约成本上升。当一个账户在系统中的信誉带来的利益远大于其违约成本的状态下,该账户在交易中会希望不作弊完成交易。

最早Karl等人提出并解决了P2P网络环境下信任管理的问题,越来越多的人尝试使用算法解决复杂性网络的信任问题,信任模型有很多,常见的有:Peer Trust模型使用评价和评价奖励因素,通过局部声誉值和全局声誉值计算信任。EigenTrust模型使用可信节点,基于分布式哈希表,利用直接信任计算全局信任,筛选恶节点并剔除网络。PowerTrust模型从可信节点选取,提高 EigenTrust 模型的收敛速度,和Power节点算法发现改善了EigenTrust模型。Gossip Trust 受绯闻传播特点的启发,并行计算节点的全局信誉。

基于LeaderRank和Trust Rank算法计算账户的活跃度,并将P2P的去中心化信任算法用于账户的信任值计算,通过PGPOW和类DPOS激励模型,建立一种新的分布式信任系统。交易双方双向完成交易,认为该交易成功,然后完成区块链记录。然后系统以区块链的交易记录为基础,根据账户间的关于某个交易标的的交易历史,动态地计算账户的活跃度、重要度、直接信任值,信誉值和推荐信任值。用户根据交易对手的推荐信任值和信誉值便能判断交易对手的可靠性,降低恶意账户被选为交易对手的概率。

3 系统功能

3.1 系统功能

通过Bit Trust System系统,使得交易无需第三方中介,无需和交易的人有人际或契约信任,可以自由交易。



▲ 系统功能示意图

蜂巢节点承担区块链全节点功能,并通过BIU交易记录, BIUT交易记录, 交易链交易记录, 根据相应算法, 分别运算活跃度, 重要度以及特定交易标的信誉值。轻节点向蜂巢节点提出查询请求后,如果没有该节点的活跃度, 重要度和信誉值, 或该节点的区块已经有更新,则蜂巢节点计算后存储并回答查询请求结果。

每条交易链对应某交易标的交易记录, 并计算出账户关于该交易标的直接信任值, 信誉值。每种交易标的并不一样。一个高信誉的BTC的卖家, 并不意味着会是一个合格的logo设计师卖家。

卖家所使用的账户的活跃度和重要度在整个系统中是全局性的。活跃度算法和重要度算法中, 有防备女巫攻击等因素, 因此账户的活跃度和重要度越高, 其在交易中违约的可能性越小;买家同样如此。

3.2 系统各元素说明

- **交易链:**用来存储特定交易标的的信息及对应的交易记录
- **BIUT:**信用币
- **BIUT链:**用于存储应用层使用的数字货币BIUT的交易记录
- **BIU:**信用令
- **BIU链:**用于支付、转换Gas、实现Token转账与记录、支持智能合约系统
- **蜂巢节点:**一般情况下为普通PC电脑,共同维护整个系统运作。
- **轻节点:**一般为手机端,可以自行选择是否成为蜂巢节点,如果成为蜂巢节点,就会加入节点群落与其它蜂巢节点共同完成轮循周期,下载BIU链信息。
- **节点群落内矿工:**节点群落内成功解出难题的蜂巢节点,拥有生成新的区块,打包数据的权利和义务,并可以获得生成新区块的奖励。

3.3 一个去中心化C2C交易举例

数字货币的C2C交易,通过轻节点和蜂巢节点的验证机制,使用智能合约可以确保交易成功进行。更广泛的场景,比如使用纸币购买数字货币(与数字货币购买商品或服务类似),无法获取支付信息,因此使用智能合约。我们以下面这个例子简化说明这个过程。

以使用美元纸币购买BIUT的C2C交易为例,买家Alice希望用美元买BIUT,在DAPP中蜂巢节点查询到不同卖家的BIUT交易信誉值以及报价。Alice选择卖家Bob交易发起后, BIUT卖家向数字货币交易智能合约发出请求,并将相应数量的BIUT打入合约相关的指定地址。在成功完成交易情况下, Bob线下或者银行账户等收到相应款项后,点击收到款项,则智能合约触发账户将BIUT转给Alice。

如果交易发生争议的情况,Alice完成付款,Bob并不承认收到。智能合约暂时锁定BIUT,并如果Alice为高活跃度、高重要性、高信誉值用户,则抵押费提交争议解决,并提交相应材料。根据算法在BIUT的信任守门人中任意选取3个,其中一个为第一指定法官。法官判定Bob违约后,Bob未上诉,则归还Alice争议抵押费。并将智能合约的BIUT转给Alice。如果Bob上诉提供材料后,并提交争议抵押金后,

则再增加信用守门人18个,投票决定侵权与否。投票Bob为违约后, Bob的争议抵押金将奖励给信用守门人。除了提交的资料外,系统降维法官提供Bob账户的活跃度、重要度、信誉值以及全网交易记录。

随着系统的发展,多次交易和博弈后,交易活跃的诚信账户活跃度、重要度、信誉值会增加,交易双方能快速选择决策,而违约账户基本被排除活跃账户系统。因为违约成本高,因此系统中活跃账户在特点交易标的中为诚信账户,实现去中心化,无需信任中介也可快速交易,促进市场交易速度。

4 系统架构

4.1 多链结构

区块链首先是一个去中心化的系统,任何数据都将是公开透明可追溯的且不可修改的,我们不存在用于储存交易信息的服务器。其次加密机制确保了用户私钥和地址的高安全性,同时对交易信息选择性加密,又保护了用户交易过程中的隐私。

系统当中,采用多链平行结构,分为交易链、BIUT链和BIU链,交易链可分自治域,每个自治域包含一个自己的交易链。

4.2 模块化设计

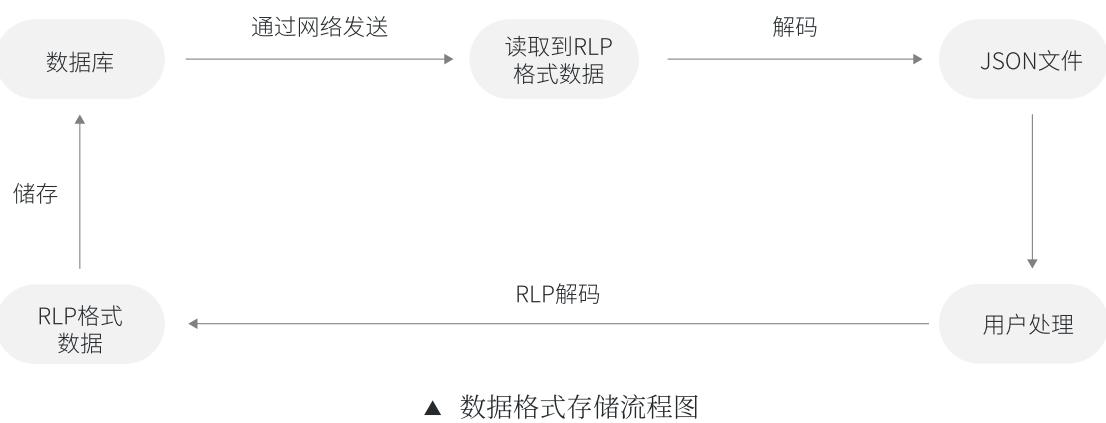
系统采用模块化设计,从多链结构的链、非对称加密算法、信任相关的算法和功能,系统级和用户级智能合约,共识算法和数据结构,都采用模块化去耦合的设计,具有优良的扩展性。

4.3 数据格式与存储

系统中使用JSON格式和RLP编码的方法来处理和存储区块的信息。JSON是一结构简洁清晰的数据交换格式,易于开发者使用,系统方便解析和生成,有利于提高传输效率。系统采用了作为ethereum项目原生的RLP编码,该编码格式符合到系统设计目标。RLP编码可以嵌套任意长度的二进制数组,专门用于列表结构的数据处理。具有以下优势:

- 结构清晰简洁, 仅仅分析短短几个字节长度的前缀, 就可以很清晰的了解数据结构。
- 易于实现保证绝对的字节表达一致性。
- 提供一个显式顺序的key/value maps。
- 在编码长度上比bencode 算法有优势。

具体流程图如图

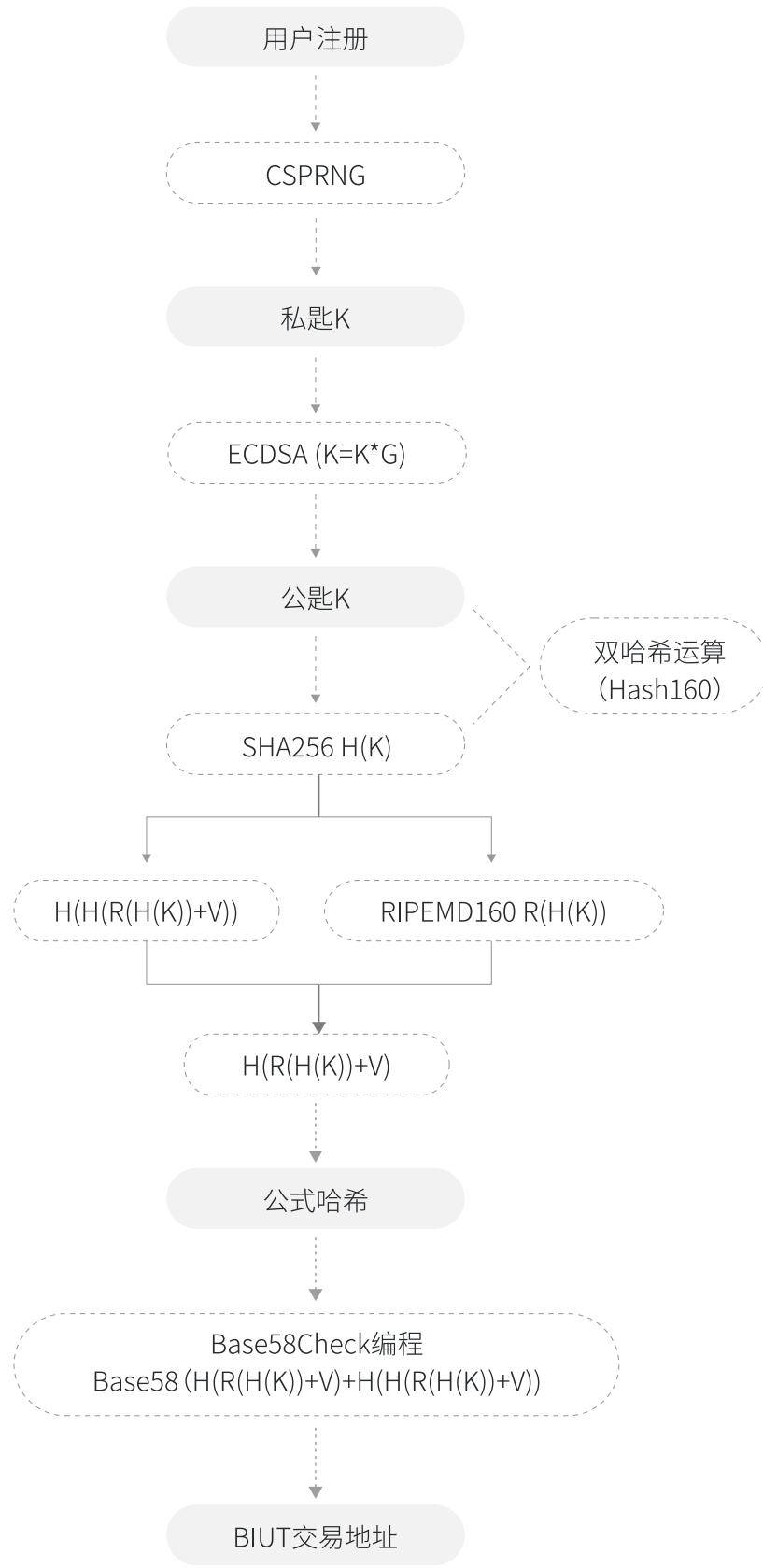


4.4 加密算法

采用非对称加密方式中的椭圆加密, 其加密算法为secp256k1, 此算法也正是比特币所用的加密法。椭圆加密的工作效率远高于非对称加密RSA算法, 达到同样级别的安全强度, RSA所用的字符长度约是椭圆加密字符长度的6倍多。因此椭圆加密大大降低了对整个系统的运算负荷, 且此算法不可逆。

当用户注册后, 产生随机数k做为用户私钥, 经过单向加密函数ECDSA(k)加密, 得到用户公钥K。此时, 使用一个单向加密哈希函数Hash(K)得到用户交易地址。生成私钥使用伪随机数生成器CSPRNG。通常情况其长度为 256 位, 是一个二进制数, 以 64 位十六进制表示, 每个十六进制占 4 位。

用户注册以及交易地址产生结束后, 发送用户的公钥K和交易地址。公钥做为私钥和交易地址的桥梁, 其重要作用是验证交易的真实性, 通过运算发送交易的地址是否和该公钥生成的地址一致; 公钥也可验证私钥的签名, 用来验证交易的私钥签名。

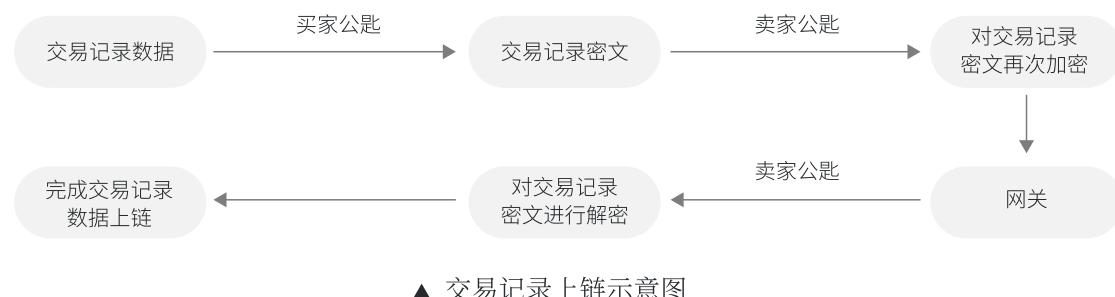


▲ 加密过程示意图

4.5 交易链

4.5.1 交易记录数据的结构及上链流程

交易记录数据的主要内容为：卖家地址、买家地址、交易记录和交易状态。其中的交易记录需要在上链之前进行加密，交易记录内容储存商品名称和商品价格。



- 交易记录数据中的交易记录首先会用买家公钥进行加密，生成交易记录密文，此密文会被存储在自治域的交易链中。
- 第一步生成的交易记录密文再次使用卖家私钥进行加密，并发送至系统网关。
系统网关中的系统级智能合约会用卖家公钥对二次加密的交易记录密文
- 进行解密，如果可以解密，则可以确定此交易记录的真实性（确实由卖家生成并发出），完成对交易记录真实性的验证。
- 在完成验证后由对应轮循内的负责节点封装上链。

通过此流程上链的交易数据本质上只能由交易相关的卖家和买家进行解密后查看，与此交易无关的其它结点无法查看这次交易的具体内容，只能知道发生了交易。实现了链上交易记录的保密性及匿名性。

4.6 BIUT链

BIUT链用于存储应用层使用的数字货币BIUT(信用币)的交易记录，BIUT链的区块记录用于重要度算法，用于计算账户重要度。信用币为交易链的支付货币，也是交易链信誉度算法的计量工具。

公链共同参与者角色众多，包括用户、投资者社群、开发者等。在传统模型之下，token的消耗是巨大的，不利于扩展生态以及做激励机制；在BIUT的设计当中，充分考虑了所有参与者的利益；因此，模型中BIUT的设定，很好地平衡了生态参与者的利益。

4.7 BIU链

BIU链负责维护交易链以及BIUT链的出块和安全。当BIU链出现分叉时,即由于网络延时等原因出现两个不同区块拥有同一个父区块的现象时,我们采取与比特币相似的方法,即承认较长链的方法来解决分叉问题。分叉的处理是在网络状态事件处理机实现,基本的逻辑如下:

当节点A接收到新的区块,如果区块的高度小于或者等于自己的区块链长度,则直接丢弃该区块。如果接收到的区块的高度刚好等于区块长度,则直接把该区块加入到自己的区块链中。如果接收到的区块的高度大于自己区块链长度超1,则直接向发送区块的节点请求所有超出部分的区块。在接收到区块的同时,会进行一致性验证,如果验证没有通过,则会比较双方的区块链长度,强制更新成较长的区块链。

4.8 自治域

自治域为跨链技术的预留逻辑结构,每个交易链都对应一个自治域,而整个交易链也是一个自治域。其他该系统以外的区块链也以自治域的形式与系统进行交易数据交换。自治域的存在极大的提高了我们系统的可扩展性,以及并行处理能力。

4.9 网关与智能合约设计

4.9.1 BIUT网关

智能合约分为用户智能合约以及系统级智能合约。用户方智能合约可由用户自行定义交易的行为方式。系统级智能合约由系统直接提供。我们将所有系统级智能合约的集合定义为BIUT网关。

4.9.2 系统级智能合约

BIUT系统级智能合约保存于交易链中。BIUT系统级智能合约主要应用于全网通用的智能合约。比如域名智能合约,通过BIUT支付Gas费的自动购买BIU的智能合约和法币支付的上链智能合约等。

4.9.3 用户智能合约

BIUT用户智能合约保存于交易链中。针对特定交易链的智能合约,例如买家的分享行为奖励。当卖家与买家之间的一笔交易完成时,卖家节点会在VM环境中运行智能合约。该智能合约会为此次的交易信息生成区块并且开始对买家的分享行为进行监听。

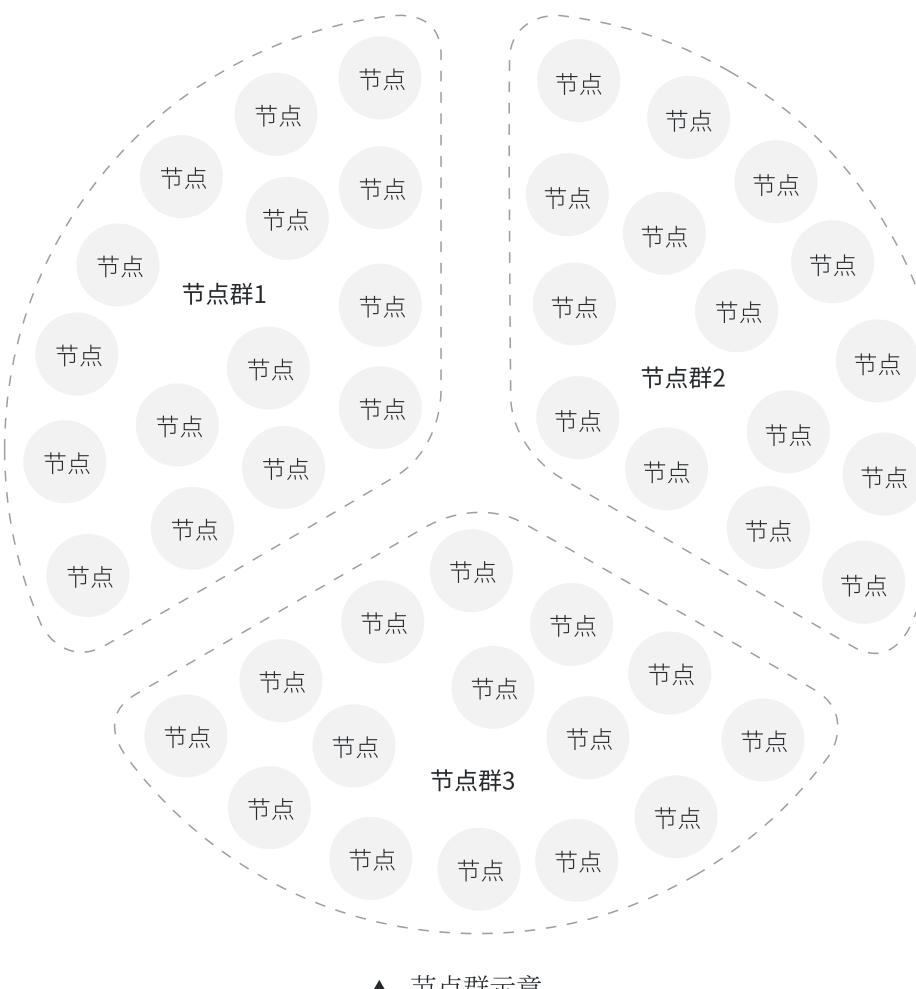
5 PGPOW共识算法

5.1 节点群落

系统在网络层面上只由轻节点与蜂巢节点组成,在逻辑层面上由数量庞大的自治域组成,每个自治域又包含唯一一条交易链。每一笔在dapp中完成的交易会被储存在对应的交易链中。自治域按照商品标的种类进行划分,也就意味着每条交易链对应一个商品种类。

为了对数量庞大的交易链进行维护(如每笔交易数据打包验证和上链工作,每一条交易链的新区块的生成工作),节点群落由大量蜂巢节点构成。从本质上说,节点群落解决了对大量交易链维护所需的算力问题以及BIU链新区块生成问题。

BIU链的新区块由节点群落轮换生成,假设系统划分成了1到10个节点群落,则这10个节点群落顺序生成新的区块,在一个轮循完成后,重新投票进行新一轮的节点群落划分。



5.2 节点群落的生成

每个节点群落由蜂巢节点投票生成, 具体流程如下:

- 假设我们的系统由10个节点群落组成, 节点群落标号1到10;
- 每个蜂巢节点给自己的每一个相邻节点生成一个随机数x, 这个过程称为投票;
- 每个蜂巢节点会得到来自相邻节点的投票, 获得的最多的点数进行运算, 运算结果就为这个节点所在的节点群落标号;
- 获得相同节点群落标号的蜂巢节点, 属于同一个节点群落。

通过上述机制, 划分每个节点群落, 同时避免了投票作弊。在每个轮循完成后, 系统会自动再次进行节点群落的重新划分。为防止攻击行为, 我们引入如下公式:

$$S = 5 * \sin x + 5$$

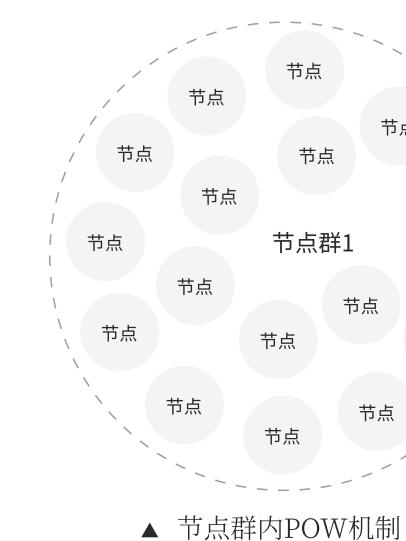
x为产生的随机数, S表示节点群落标号, 比如S的值为0-1即表示为节点群落1, 1-2表示节点群落2。划分结束后, 或其他蜂巢节点可选择性检查此蜂巢节点获得票数及所在节点群落, 若发现此节点上报结果与所在节点群落不符合, 说明此节点存在作弊行为, 将对此节点做出惩罚。

5.3 BIU链共识机制及维护

BIU链每20秒产生一个区块, 10个节点群落轮换生成BIU链上的新区块, 并轮换对交易链进行维护, 一次完整的轮循为一个周期, 一个周期结束后, 重新进行新的节点群落划分及开始新的周期。

被轮换到的节点群落内部进行POW机制, 本节点群落内的各个节点竞争一个生成Token区块的机会, 生成Token区块的节点获得Token及Gas奖励。

为防止POW能源上高消耗的问题, 我们将设置难度降低, BIU链的难度每经过2016个区块改变一次,



保障大部分节点用户可以参与计算, 计算难度公式为:

$$\text{计算难度} = \frac{\text{difficulty_target}}{\text{current_target}}$$

目标(target)为一个128位长的Hash值。目的只是为决出一个可以生成Token区块的节点, 另外可以相对保证这个蜂巢节点的机器性能较高, 可以胜任数据打包的任务, 这个节点完成对BIU链上的数据打包及上链。

在节点群落内的各个蜂巢节点, 由于共同维护了系统的正常运作, 即使没有竞争成为节点群落内矿工节点, 也可以获得少量BIU奖励。

由于节点群落包含了很多蜂巢节点, 这些蜂巢节点大概率包含了系统内全部的交易链, 这些交易链由在被轮循节点群落内的所有蜂巢节点共同维护。如果发生节点群落没有包含全部交易链的情况, 由当前节点群落最新加入的蜂巢节点向全网广播, 收集BIU链的完整信息。

5.4 交易链共识机制及维护

交易链是根据交易标的品类产生的自治域区块链, 所有的用户不一定都参与到每一个交易链中。对于交易链也是使用同样的节点群落划分, 交易链每8秒产生一个区块, 保证交易的高效性, 若活跃性不高的自治域可在2016个区块后做调整。

假设节点1, 节点2, 节点3同时拥有交易链A, 则节点1, 节点2, 节点3各自同时生成一个随机数并在节点群落内进行广播, 之后大家对比各自的点数, 由获得随机数最小的节点完成对交易链A中数据的打包上链, 以及交易链A新区块的生成。将数据打包上链后进行全网广播, 本自治域的其他节点进行验证。

对交易链的维护操作不会给节点带来任何奖励, 这个是在节点群落内蜂巢节点的义务, 由于自治域本身的交易链高度不会很高, 另外需要完成的打包数据也不会很大, 所以对节点机器性能的要求不高。

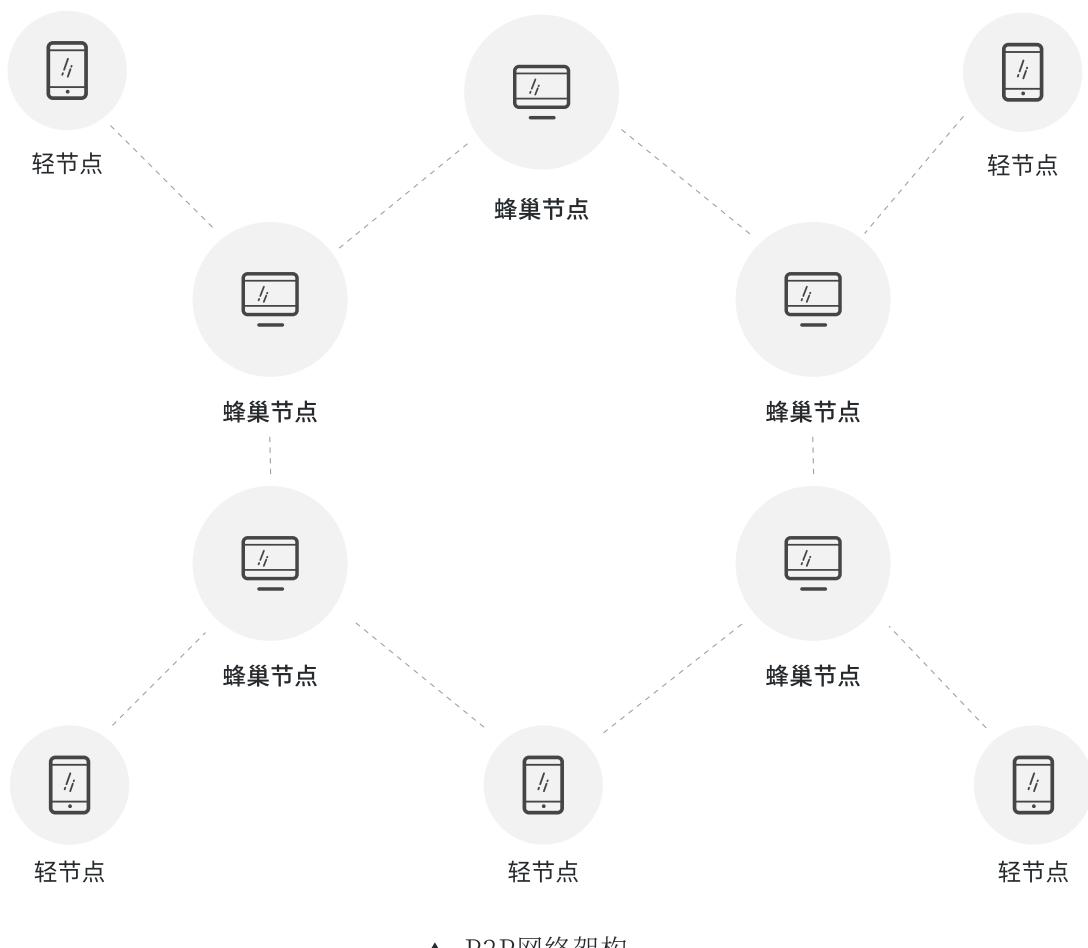
每条链等待6个区块高度, 若出现分叉行为, 所有节点群落检查链的高度, 投票查出最高链, 得票超过51%的为合法链。

6 网络

BIUT区块链网络为一个类似于KaZaA的结构化分布式网络, 是一种典型的第三代P2P网络。将网络中所有的节点信息汇集成立式哈希表, 表内信息包含关键字和所在节点的地址, 然后将这分布式哈希表分割后分别存储到网络中的节点中。通过该表能发现存储与关键词对应的哈希表内容所存放的节点。该网络具有高度灵活性, 高度结构化, 扩展性强。

BIUT网络由轻节点、蜂巢节点组成。

- **轻节点:**一般为手机客户端, 只与蜂巢节点进行连接, 数据从蜂巢节点拉取。
- **蜂巢节点:**一般为PC客户端, 可以与其它蜂巢节点进行连接, 其所在域内的数据与其它蜂巢节点进行同步。



7 信任算法

7.1 信任机制的实现

在C2C交易场景下,当违约成本大于违约收益时候,那么用户倾向于完成交易。在一个C2C交易系统内,用户最安全的选择是与可信度高的用户交易。账户关于某个标的的历史成交记录,代表了该账户关于该标的在该交易市场的重要性排名,通过算法给予更高的评价。我们通过信誉算法和推荐信任值来解决交易账户的成交能力评价,并展示给用户,从而借助交易记录和算法实现系统级的信任。

在系统中,每次交易完成后,蜂巢节点把交易评价存储在本地。当一个轻节点需要计算一个账户的信誉值时,从蜂巢节点获取相关信誉值。蜂巢节点根据区块的交易记录,查询是否有新的交易,是否需要更新信誉值。

- 如果没有新的交易记录,则返回原有信誉值即可;
- 如果需要更新,首先从区块中获得活跃度、重要度、直接信任值、全区信誉值和推荐信任值。如果两个账户之间不存在直接的交易记录,就需要依靠系统推荐信任,算法,计算交易对象的推荐信任值。

7.2 定义

- 定义1. 活跃度排名(Activity Rank)是根据BIU账户交易的关系,算出网络中不同账户的活跃度排名。
- 定义2. 重要度排名(Importance Rank)是根据BIUT的账户交易关系,交易量、币龄和余额,计算的账户重要度排名。
- 定义3. 直接信任值(Direct Trust)是根据账户之间直接交易的历史记录,计算出对交易对象账户的信任值。
- 定义4. 推荐信任值(Recommendation Trust)是账户A和账户B之间没有交易关系,根据系统计算计算而形成信任值.
- 定义5. 信誉值(Reputation)是全局信任,是整个网络群体对某账户的信任。

7.3 基于BIU链交易记录的Activity Rank计算

根据BIU链中的交易关系, 使用leader rank 算法, 计算每个账户的BIU活跃度排名Activity Rank。

算法含义:

- 如果一个账户有很多其他账户与之产生交易, 说明这个账户比较重要, 也就是Activity Rank排名会相对较高。
- 如果一个Activity Rank值很高的账户, 由于转账或交易产生了gas费用, 那么被动交易到的账号的Activity Rank排名会相应地因此而提高。
- AR只与账户交易关系有关, 与交易量无关, 定义为向量矩阵。

考虑抵抗漂白攻击和女巫攻击等因素, 设定如下

- 计算所有和系统黑洞账户0000000000000000有过转账记录的账户, 其他账户活跃度为0。系统黑洞账户作用: 所有账户的手续费均转移至黑洞账户, 并且通过黑洞账户将gas费奖励给矿工。
- PGPOW的矿工需要提供算力, 但普通PC电脑即可挖矿, 根据算法, 矿工账号的AR相对其他账户大, 因此矿工账户设定为可信账户, 将可信账户输出的矩阵信息用于评估环节, 为交易链中计算推荐信任, 能有效减少女巫攻击。

将系统中的账户依据相互转账的关系, 组织成一个Web图 $G_w = \langle V, R \rangle$, 其中, V 是账户的集合, 也就是Web图中的顶点集合, R 是账户与账户之间的转账关系的集合。在Web的 G_w 中, 为了表示账户之间的转账关系定义 $R_{i,j} \in R$ 表示账户*i*到账户*j*的转账。

然后采取活跃度分裂的方式计算活跃度, 活跃度分裂是如果一个账户的活跃度为1, 且指向n个账户, 那么它所转账的每一个账户得到的信任值为 $1/N$, 由此一个账户的信任值是它从所有转入到它的账户得到的活跃度的总和, 此方式避免矩阵当中孤立点造成的无法收敛问题、同时在应用层面也平衡了新加入者的权益。

AR_i 定义为账户 i 的活跃度。 $AR_i(k)$ 为经过 k 步计算后, 系统稳定态账户 i 的一阶活跃度。然后, 黑洞账户将其活跃度平均分配给其他账户, 从而获得账户的全局活跃度。账户 i 的最终的全局活跃度 AR_i 值计算如下:

如果该账户和黑洞账户没有交易记录, 则:

$$AR_i = 0$$

如果该账户和黑洞账户有交易记录, 则:

$$\begin{cases} AR_i(0) = 1/n \\ AR_i(k) = \sum_{j=1}^{n+1} \bar{a}_{ji} AR_j(k-1), k = 1, 2, \dots \\ AR_i = AR_i(k) + AR_g(k)/n \end{cases}$$

式中, n 为网络账户数 (不包括黑洞账户以及和黑洞账户没有交易记录账户); $AR_g(k)$ 为第 k 步背景账户 v_g 的活跃度; \bar{a}_{ij} 为 web 图的矩阵元素:

$$\bar{a}_{ij} = \begin{cases} 1/k_i^{\text{out}}, & \text{存在账户 } i \text{ 向账户 } j \text{ 转账的关系} \\ 0 & \text{否则} \end{cases}$$

式中, k_i^{out} 为节点 v_i 的出度。

7.4 基于BIUT链的Importance Rank计算

利用 BIUT 链的交易记录, 使用 trust rank 核心算法, 先计算每个账户的 TR 值, 再在根据 TR 值计算 Importance Rank (IR) 值。

Trust Rank 算法设定: 可信账户很小几率参与欺诈攻击。先用 AR 识别的可信账户(即“种子”账户), 那么由可信账户指向的账户也可能是可信账户, 即其 TrustRank 也高, 与“种子”账户的指向越远, 账户的 TrustRank 越低, 在此算法中包含了路径因素的影响。

TrustRank 算法首先设置一个好的账户种子集 SP_g , 即认为集合中的账户是可信账户, 将这些账户的初始活跃度设置为 1, 之后将攻击账户的活跃度设置为 0, 其他账户设置为 0.5。若可信账户集合中的某账户到一个账户所有的路径上都不包

含垃圾账户，则将此账户活跃度设置为1。

计算账户的TR值。

$$TR(i) = (1-d) + \sum_{j=1}^{k_i^{in}} \left(c_{ij} * TR(j) \right) d$$

其中， $TP(i)$ 是账户 Vi 的 Trust Rank 值， d 为阻尼系数； k_i^{in} 为账户 vi 的入度。

$$\bar{c}_{ij} = \begin{cases} 1/k_i^{\text{out}} & \text{账户 } i \text{ 指向账户 } j \\ 0 & \text{otherwise} \end{cases}$$

式中， k_i^{out} 为账户 vi 的出度。

计算每个账户的交易重要度Importance rank(IR)：

$$IR = TR \times AR^T \times \frac{\ln(CA_i^{\sum t_i})}{1 - e^{\delta C}}$$

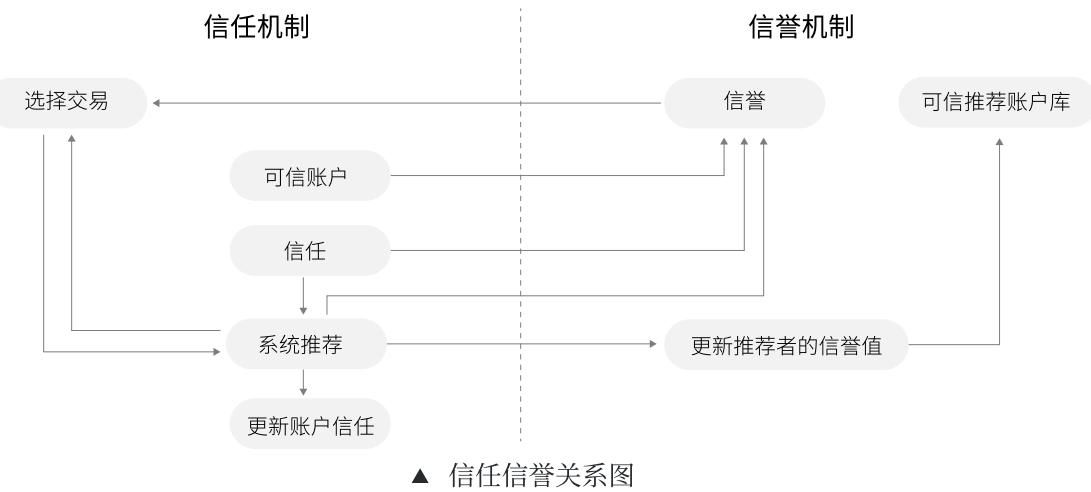
到了账户重要性层面，需要统筹考虑所有因素的叠加影响。首先在CA与AR层面，考虑了所有账户的系统性特点以及真实性程度；其次，在币龄层面，综合考虑账户所有持仓的币龄状态，通过加权累计的方式得到其绝对值，之后通过对数形式做系数换算转化；最后CA为对余额的币龄加权和、 δC 为余额在全部质押当中的相对比例，在总体持仓当中每个账户占据的比例都比较低，做归一化处理，使得整体取值可以得到单量级之内的乘数处理，而避免较多持仓账户与较小持仓账户造成的乘数放大。

公式当中，统筹考虑了BIU和BIUT层面的交易记录特点，即活跃度越高、币龄越高、余额越高，代表函数的值就越大，进而重要度就越高。

7.5 基于交易链的信誉和信任计算

针对某一标的的C2C交易，交易记录计入该交易链。在进行C2C交易应用层，根据Reputation算法，计算该交易的双方Reputation值，根据推荐信任值推荐C2C交易方。

信任值与信誉的计算关系如下图：



7.5.1 直接信任值

直接信任值表示账户提供交易服务的可靠性，账户之间进行交易，账户 i 从账户 j 处购买交易标的，利用 Beta 概率密度函数计算账户 i 对账户 j 直接信任值为：

$$d_{ij} = \begin{cases} 0, & g(i,j) = u(i,j) \\ \frac{\text{Max}(g(i,j) - u(i,j), 0)}{\sum_j \text{Max}(g(i,j) - u(i,j), 0) \times N_p}, & \text{otherwise} \end{cases}$$

其中 g_{ij} 为 i 对 j 成功交易次数， u_{ij} 为 i 对 j 不成功次数， N_p 为惩罚系数（当 $N_p > 1$ ，按照实际值计算；当 ≤ 1 ，按照 1 计算）。账户 i 与账户 j 成功交易次数越多，相互的直接信任值 D_{ij} 越大；通过惩罚系数因子 N_p ，如果恶意账户实施攻击，直接信任值 D_{ij} 将快速下降。

而 N_p 与临近周期的全网不成功交易次数的环比相关，以及重要度 (IR) 有关。初始状态设定为 1，随着系统鲁棒性增大，此参数做自适应调节，但最小值为 1。通过 N_p 使直接信任值的下降速率比上升速率快。

7.5.2 信誉值

利用 AR 活跃度筛选出可信账户集合，使用 Eigen Trust 算法通过直接信任值计算信誉值。假设账户 i 与账户 j 之间经过多次交易后， d_{ij} 为归一化处理后的直接信任值；定义 \vec{t}_i 为包含 t_{ik} 的向量，其中 t_{ik} 为 i 在查询账户的基础上对账户 k 的信任值：

$$t_{ik} = \sum_j d_{ij} d_{jk}$$

矩阵[dij]记为D, 得到:

$$\vec{t}_i = D^T \vec{d}_i$$

而为了获取更为全面的信息, 账户i会向很多有交易关系的账户进行计算, 最后将得到公式: $\vec{t}_i = D^T \vec{d}_i$, 且当n很大的时候, \vec{t}_i 将收敛到每个对等账户的同一个向量, 即 \vec{t} 是该模型中的全局信任向量, 向量中的元素量化了对其他账户的信任程度; $\vec{t} = [t_{i1}, t_{i2}, \dots, t_{im}]^T$, m为全局数据中的可信账户总数。

- 可信账户信誉值计算

根据AR算法值筛选可信账户集合为P, 则账户i的信誉值pi:

$$p_i = \begin{cases} 1/|P|, & V_i \in P \\ 0, & \text{otherwise} \end{cases}$$

- 账号i的信誉值迭代计算公式:

$$\vec{t}_i^{(k+1)} = (1 - a) D^T \vec{t}_i^{(k)} + a P$$

该公式迭代计算直到

$$\left\| \vec{t}_i^{(k+1)} - \vec{t}_i^{(k)} \right\| < \epsilon$$

其中 $\vec{t}_i^{(k)}$ 是第k次迭代后的信誉值; P是该i账户的初始信誉基数矩阵; 上标 T 表示向量转置; ϵ 是一个很小的数; 初始 $\vec{t}_i^{(0)} = \frac{1}{n}$; $a \in [0, 1]$ 为账户对假定信任账户的信任程度。

上述处理方式, 主要在于全局信任来源于所有直接信任的累计, 但在尚未确定账户是否值得信任的情况下, 增加对每个账户的信任赋值, 便于实现算法当中对新加入者的公平性考虑。

7.5.3 推荐信任值

账户的直接信任值,信誉值,与对应的推荐信任成近似正向关系。由于越是临近现在的时间,其行为就更加有意义,因此对公式因子进行时间衰减,使得更加临近的时间更有意义、权重更大。

时间衰减因子 τ_j 表示账户*j*与账户*i*在交易产生的直接信任值和信誉值相比当前时间的衰减比例, s_j 为该次交易的时间。

$$\tau_j = e^{-(s-s_j)}$$

账户*i*推荐信任值TT*i*, 计算公式为:

$$TT_i = \sum_j^k \tau_j \times (1 - \rho^k) d_{ij} + \sum_j^k \tau_j \rho^k t_i^{s_j}$$

$t_i^{s_j}$ 为账户*i*在交易时刻 s_j 的信誉值, 为减少大量的计算, 根据信誉值计算过程, 该值趋于稳定, 则:

$$t_i^{s_j} \approx t_i$$

其中, d_{ij} 为账户*i*和账户*j*的直接信任值; $(1 - \rho^k)$ 是直接信任的权重, 且($0 < \rho < 1$), k 表示账户*i*与其他账户之间成功交易的次数, 两者直接交易次数越多, 则推荐信任越来越倾向直接信任值。

7.6 信任机制算法总结

我们将上述算法以及通过激励模型的信任机制命名为POW TRUST信任机制: 它包含了一套基于PGPOW的激励机制, 分布式记账系统, 分布式信任算法功能。

算法包括: 利用BIU链系统黑洞账户和Leader Rank算法, 以及Trust Rank算法, IR重要度算法, 根据BIU和BIUT交易记录计算活跃度和重要度排名算法; 根据活跃度和重要度算法, 解决了可信账户动态选择, 并利于重要度等因素计算直接信任值、信誉值算法, 推荐信任值三个信任相关的算法。

7.6.1 信任机制算法与恶意交易行为

系统中往往有恶意账户采用各种的策略尝试绕过信任机制, 对系统中的诚实账户进行交易攻击, POW TRUST信任机制中应对常见攻击行为的抵抗方法如下:

- 问题1: 行为不一致, 比如先作为诚实账户交易, 提高信任值后进行违约交易行为;

解决1: POW TRUST单次交易的行为摇摆攻击行为通过抵押解决。在重要度排名和信誉值的算法中, 正向增长是慢的, 但出现欺诈行为, 惩罚因子的存在, 可以使信誉值速下降。

- 问题2: 合作攻击, 如多个恶意账户合作进行虚假交易来刷信誉值;

解决2: POW TRUST解决由于活跃度和重要度算法中, 与交易关系有关, 加上可信账户的存在, 恶意账户通过合作进行虚假交易不能提高信誉值。

- 问题3: 信誉刷单, 恶意同高信誉值账户频繁交易, 比如为获得较好的活跃度和重要度数值, 频繁和大型中心化交易所账户进行转入转出交易;

解决3: POW TRUST解决, 在活跃度排名k, 贡献度排名k算法中, 由于权威账户的出入度都大, 以及可信账户的存在, 因此该行为不能明显增加信誉排名。

- 问题4: 利用多账号进行的攻击, 这类攻击有女巫攻击(sybilattack)和漂白攻击(whitewishing);

解决4: POW TRUST解决, 由于信任账户是BIU链的矿工账户以及和矿工账户, 系统实现普通电脑可以挖矿, 通过PGPOW的矿工账户, 可以有效抵御女巫攻击。而在Activity Rank算法中, 未发生转出行为的账户Activity Rank为零, 而且由于POW TRUST信任机制, 积累信誉缓慢, 违约成本高, 可以有效防止漂白攻击。

- 问题5: 蜕变攻击, 诚实账户丢失或链下行为蜕变为恶意账户, 进行攻击。例如让亲朋好友帮卖家刷单行为, 或购买贿赂大量诚实账户进行攻击行为;

解决5: POW TRUST解决, 对于个别的刷单行为, 从交易记录和行为, 都

很难鉴别是否为攻击行为,通过其他协同过滤算法,但目前POW TRUST信任机制不能有效识别。对于贿赂大量的诚实账户的刷单行为,攻击成本高,攻击收益不确定,因此可以避免。

7.6.2 细粒度信任机制算法

在BIT TURST SYSTEM中目前实施的POW TRUST算法以交易记录为介质,以交易成功为目标形成的交易机制。随着系统的发展,会有更多类型的交易链出现,针对不同的交易标的,会有更适合的交易算法。例如根据评价的信任算法是原来传统电商、社交媒体网站的主流信任算法,虽然有缺陷,但在BIT TURST SYSTEM中会有新的优化方法,新的交易链和信任算法也会对整个系统的账户交易提供更多的数据和维度,更有利于细粒度的信任机制,整个系统信任体系也愈加有效。

8 激励

作为一条用于交易落地的公链,系统激励分为三个层面。DAPP的应用层面,由各DAPP自主设定。信用币(BIUT)固定总量15亿,并通过信用守门人角色和智能合约实现了回收和发行机制。信用令(BIU)固定总量15亿,由PGPOW挖矿产出,前50%量的BIU的产出和每三个月为周期的全网交易数有关,产出BIU跟应用需求有关。网络和生态发展成熟后,后50%量的BIU按照四年减半挖出,确保能激励用户,保证生态的发展和网络的安全。

8.1 BIUT的经济模型

8.1.1 BIUT用途和总量

BIUT用途:应用层支付、挖矿权益证明、交易抵押金、交易市场开通费用、交易纠纷仲裁费用等用途。

BIUT分配方案

类型	Tokens数量	占总量占比	总量占比	说明
信用守门人激励	600,000,000	40%	0.40	随着应用生态，以类似POS机制产出
社区	600,000,000	40%	0.40	
基金会(运营和商务)	150,000,000	10%	0.10	3年成熟期
技术团队	150,000,000	10%	0.10	3年成熟期
	1,500,000,000	100%		总量

8.1.2 挖矿权益证明量

蜂巢节点中BIUT在[0, 上限]区间内, 数量越多, 单次出块挖出的BIU越多。抵押算法写入主网程序。初期BIUT权益证明数量区间为[0, 10万]。

考虑公平性因素, 挖矿的BIUT权益数量上限 $10 \leq \text{权益证明上限} \leq 10\text{万}$, 不锁仓。以365天为一周期, 周期调整。如交易数量有较大幅度增加, 说明系统发展较快, 则下调挖矿的权益证明上限。计算本期权益证明量 PQ_{i+1} :

$$PQ_{i+1} = \begin{cases} PQ_i / (TQ_{i+1} / TQ_i), TQ_{i+1} \leq TQ_i \\ \frac{PQ_i}{\log_{10}(10 + (TQ_{i+1} / TQ_i)) \times \log_{10}(10 \times (TQ_{i+1} / TQ_i))}, TQ_{i+1} > TQ_i, TQ_i \leq 100000 \\ \frac{PQ_i}{\log_{10}(10 + (TQ_{i+1} / TQ_i)) \times \log_{10}(10 \times (TQ_{i+1} / TQ_i)) \times \sqrt{TQ_i / 100000}}, TQ_{i+1} > TQ_i, TQ_i > 100000 \end{cases}$$

其中 PQ_i 为上一期权益证明上限, TQ_i 为上一期全网交易数, TQ_{i+1} 为本期全网交易数。

8.1.3 信用守门人竞选和义务

轮训101人的信用守门人账户, 重要度在一定值以上的账户可以竞选信用守门人, 通过100个账户投票即可参与。信用守门人需要质押BIUT, 质押比例以365天为一周期调整。如交易数量有较大幅度增加, 说明系统发展较快, 则下调挖矿的权益证明上限。初期质押数量为1000BIUT。质押调整与挖矿权益证明上限调整公式类似。

8.1.4 信用守门人收益

- 发起新的交易市场单元,通过市场开通智能合约缴纳市场命名费用。交易市场需要经过信用守门人投票通过才可开通。

市场域名费锁定在智能合约账户中,待该交易市场单元的连续周期活跃交易数量达到1万笔,则市场命名费用按照1.2倍率退回发起人,并由智能合约按照0.8市场命名费用BIUT奖励给信用守门人。如连续720天内,周期活跃交易量未达标准,则交易市场单元的市场命名费回收进入生态账户。

- 交易纠纷初级仲裁,随机指定新人守门人仲裁,交易纠纷的责任方抵押金奖励给信用守门人。
- 算法商店,生态开发者提供关于交易标的的信任算法,应用链的共识算法,数据结构等,交易的千分之一的费用收入放入信用守门人储备账户。

8.2 BIU的经济模型

BIU用于所有网络层消耗,用于全网转账手续费,智能合约执行费用,以及发布新的平行链的网络费用。

8.2.1 BIU总量和产出

以PGPOW共识算法工作量证明挖矿,总量15亿BIU按照两个阶段挖出。前50%的BIU,每个周期根据前一周期的全网交易量数量产出。后50%的BIU,按照四年减半产出。

8.2.1.1 第一个阶段产出

第一个阶段产出根据上两个周期的全网交易数有关,周期为三个月。

计算下一个周期的产出量PQi+1:

$$PQ_{i+1} = \begin{cases} PQ_i / 100, & TQ_{i+1} \geq TQ_i \times 100 \\ \frac{PQ_i}{\sqrt{TQ_{i+1} / TQ_i}}, & TQ_{i+1} < TQ_i \times 100 \end{cases}$$

其中PQi为上一期产出量,TQi为上一个周期全网交易数,TQi-1为上上一个周期全网交易数。

式中:TQi, TQi-1如果为0,则分别加1.

8.2.1.2 第二个阶段产出

剩余50%的BIU根据四年减半规律产出,开始第二阶段后:

第一个四年:3.75亿,第二个四年:1.875亿……

8.2.2 Gas费用

8.2.2.1 转账GAS费用

每一周期的Gas费用基础根据前一周期的全网交易量调整,理论值希望每笔转账手续费维持在合理的价格,周期为365天。

计算下一个周期Gas费基础GQi+1:

$$GQ_{i+1} = \begin{cases} GQ_i / (TQ_i / TQ_{i-1}), & TQ_i \leq TQ_{i-1} \\ \frac{GQ_i}{\log_{10}(10 + (TQ_i / TQ_{i-1})) \times \log_{10}(10 \times TQ_i / TQ_{i-1})}, & TQ_i > TQ_{i-1} \end{cases}$$

其中GQi为上一周期Gas费基础值TQi为上一个周期全网交易数,TQi-1为上上一个周期全网交易数。

8.2.2.2 智能合约以及新增平行链的交易费用

智能合约如果能够得到所用空间,则按照字节空间计算Gas费。

新增交易链的Gas费用,根据相对原交易量的数据结构所占的字节数,相应调节Gas费用。

9 公链商店

区块链应用发展的复杂性和开发者社区发展,我们计划推出公链商店。公链商店提供公链开发的各种模块,DApp开发需要应用组件。

9.1 公链商店的代码框架

Bit Trust System的模块化设计和多链结构,具有良好的扩展性。交易链不仅可以自定义交易标的,也可以自定义主链数据结构,以及共识算法。搭配智能合约和DAPP,有效降低开发区块链应用的开发难度和使用成本。

对代码实现模块化优化,提供公链商店的主要代码框架,实现发行代币,图形

化自定义交易链数据结构、POS、DPOS等自定义共识算法等基础功能。

9.2 组件

提供各种开发组件, 以及公链开发需要的模块。具有完整功能性的算法、功能等看作组件, 而开发者提供的组件, 可以采用MIT开源协议免费开源, 也可以向使用者收费获取奖励。

10 总结

Bit Trust System是第一个建立账户间基于系统信任的区块链网络, 通过区块链交易记录和信任算法来评估账户特定交易标的的信誉值, 并推荐撮合, 采用PGPOW+和POS共识算法建立激励体系, 增加交易用户的违约成本。网络使用JavaScript语言, 模块化设计, 实现多链架构的区块链系统, 从整个网络实现了性能高、安全性高、去中心化程度高, 扩展性强的特点, 并且对开发者友好, 是走向去中心化应用落地的完整系统。

Bit Trust System以去中心化C2C交易应用为基础, 可通过扩展到众筹、众包、社交电商、共享经济和P2P金融等点对点商业中。我们认为Bit Trust System是去中心化信任的基础平台, 帮助互联网用户安全的进行去中心化交易和协作; 也是解决新经济、新金融领域信任重构的关键难题; 是一把帮助开发者、商家和互联网用户打开数字经济大门的钥匙。

Bit Trust System, 无需信任, 自由交易!

11 代码地址

Github

<https://github.com/BIUT-Block>

NPM

<https://www.npmjs.com/org/sec-block>

<https://www.npmjs.com/org/BIUT-Block>

参考

- [1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[J]. 2008.
- [2] 以太坊白皮书.[EB/OL].<https://github.com/ethereum/wiki/wiki/White-paper>
- [3] Kamvar S D, Schlosser M T, Garcia-Molina H. The eigentrust algorithm for reputation management in p2p networks[C]//Proceedings of the 12th international conference on World Wide Web. ACM, 2003: 640-651.
- [4] Page L, Brin S, Motwani R, et al. The PageRank citation ranking: Bringing order to the web[R]. Stanford InfoLab, 1999.
- [5] Li Q, Zhou T, Lü L, et al. Identifying influential spreaders by weighted LeaderRank[J]. Physica A: Statistical Mechanics and its Applications, 2014, 404: 47-55.
- [6] Liu C, Zheng X L, Xu A W, et al. C2C trust evaluation model based on social network and reputation[J]. Computer Engineering, 2010, 2010(24): 41.
- [7] MA Xiaoxue, LIU Yuling, TIAN Junfeng. Trust model based on extended subjective logic for P2P environment. Computer Engineering and Applications, 2011, 47(7) :74-77.
- [8] XU Junming, ZHU Fuxi, LIU Shichao, et al. Identifying opinion leaders by improved algorithm based on LeaderRank. Computer Engineering and Applications, 2015, 51(1):110-114.
- [9] Peng DS, Lin C, Liu WD. A distributed trust mechanism directly evaluating reputation of nodes. Journal of Software, 2008,19(4):946 -955. <http://www.jos.org.cn/1000-9825/19/946.htm>
- [10] LI Fengqi, LI Guangming, YANG Nanhai, et al. TWIT:two-way algorithm for local trust inferring in social networks. Computer Engineering and Applications, 2016, 52(4) :66-73.
- [11] Gyöngyi Z, Garcia-Molina H, Pedersen J. Combating web spam

with trustrank[C]//Proceedings of the Thirtieth international conference on Very large data bases-Volume 30. VLDB Endowment, 2004: 576-587.

- [12] Zhou R, Hwang K. Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing[J]. IEEE Transactions on parallel and distributed systems, 2007, 18(4): 460-473.